System Administration Tasks

HP 9000 Series 800 Computers

HEWLETT PACKARD

HP Part No. B3108-90005
Printed in USA August 1992

First Edition
E0892

Legal Notices

The information in this document is subject to change without notice.

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Warranty. A copy of the specific warranty terms applicable to your Hewlett-Packard product and replacement parts can be obtained from your local Sales and Service Office.

©copyright 1983-92 Hewlett-Packard Company

This document contains information which is protected by copyright. All rights are reserved. Reproduction, adaptation, or translation without prior written permission is prohibited, except as allowed under the copyright laws.

Restricted Rights Legend. Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DOD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

HEWLETT-PACKARD COMPANY
3000 Hanover Street
Palo Alto, California 94304 U.S.A.

Use of this manual and flexible disk(s) or tape cartridge(s) supplied for this pack is restricted to this product only. Additional copies of the programs may be made for security and back-up purposes only. Resale of the programs in their present form or with alterations, is expressly prohibited.

©copyright 1980, 1984, 1986 AT&T Technologies, Inc.
UNIX is a registered trademark of Unix System Laboratories Inc. in the USA and other countries.

©copyright 1979, 1980, 1983, 1985-90 Regents of the University of California
This software is based in part on the Fourth Berkeley Software Distribution under license from the Regents of the University of California.

©copyright 1979 Regents of the University of Colorado, A Body Corporate.
This document has been reproduced and modified with the permission of the regents of the University of Colorado, a body corporate.

©copyright 1986, 1987, 1988 Sun Microsystems, Inc.
©copyright 1986 Digital Equipment Corporation.

©copyright 1985-86, 1988 Massachusetts Institute of Technology.
X Window System is a trademark of the Massachusetts Institute of Technology.

MS-DOS and Microsoft are U.S. registered trademarks of Microsoft Corporation.

OSF/Motif is a trademark of the Open Software Foundation, Inc. in the U.S. and other countries. Certification for conformance with OSF/Motif user environment pending.

All rights reserved.

Printing History

The manual printing date and part number indicate its current edition. The printing date will change when a new edition is printed. Minor changes may be made at reprint without changing the printing date. The manual part number will change when extensive changes are made.

Manual updates may be issued between editions to correct errors or document product changes. To ensure that you receive the updated or new editions, you should subscribe to the appropriate product support service. See your HP sales representative for details.

First Edition: August 1992 (HP-UX Release 9.0)

Contents

1. **Introduction to System Administration**
How to Use this Manual 1-1
Conventions Used in this Manual 1-2
What's in this Manual 1-3
What System Administrators Need to Know 1-6
A System Administrator's Responsibilities 1-7
Where to Find Other Information About System Administration 1-8
Using SAM 1-10
SAM Overview 1-11
A Typical SAM Task 1-13
Overview of Customizing the Object List 1-15
Changing the Object List Type 1-16
Changing the Format of the Object List 1-17
Displaying a Subset of Objects 1-18
Changing the Object List Order 1-20
Using SAM with an X Window System 1-21
Setting Up Your System 1-21
Korn and Bourne Shell 1-22
C Shell 1-22
Entering SAM 1-23
Exiting SAM 1-24
Customizing the SAM Interface 1-25
Using SAM with a Text Terminal 1-26
Entering SAM 1-26
Using Control Buttons 1-29
Using Softkeys 1-30
Exiting SAM 1-32
Entering a Functional Area 1-32
Selecting an Object 1-34
Opening a Menu 1-36

Choosing a Menu Item 1-37
Using Buttons and Checkboxes 1-37
Menu Buttons 1-37
Radio Buttons 1-38
Checkboxes 1-38
Navigating with Keys and Key Combinations 1-39
Changing Windows 1-40
Moving a Window 1-41
Resizing a Window 1-41
Iconifying a Window 1-41
Maximizing a Window 1-41
Opening a Window 1-41
Closing a Window 1-41
Using SAM for Remote System Administration 1-42
Adding and Removing Remote Systems 1-42
Configuring the Remote System 1-43
Logging SAM's Actions 1-44
Turning On SAM Logging 1-44

2. Constructing an HP-UX System
Assessing Users' Needs 2-1
User Survey 2-2
Installing HP-UX 2-4
Updating HP-UX 2-4
Installing and Updating Optional Software 2-5
Adding, Removing, and Moving Peripherals 2-6
After Installing HP-UX 2-7
Setting the System Clock 2-8
Setting the Time Zone (TZ) 2-8
Setting the Time and Date 2-10
Problems You Can Cause by Changing the System Clock . 2-11
Setting Up File Systems 2-11
Setting Up On-Line HP-UX Reference (manpages) 2-12
Setting Up Networking 2-14
Adding Users and Groups 2-14
Setting Up Electronic Mail 2-15
Setting Up the Line-Printer Spooler 2-16
Setting Up news 2-17

Setting Up X Windows 2-18
Setting Up System Accounting 2-19
Customizing the System 2-20
Customizing System Startup 2-23
Editing the /etc/inittab File 2-23
Editing the /etc/rc File 2-24
Customizing the Login Prompt (/etc/issue) 2-25
Posting a Message of the Day (/etc/motd) 2-25
Customizing Users' Login Environments 2-26
Customizing Users' Editing Environments 2-26
Setting Up Non-Standard Terminals 2-27
Creating a New terminfo File 2-29
Backing up the System 2-30
Reconfiguring the Kernel 2-30
When Do You Need to Reconfigure the Kernel? 2-31
Reconfiguring the Kernel Using HP-UX Commands 2-32
An Alternative Series 800 Kernel Configuration Method: regen 2-35
Reconfiguring the Kernel to Add/Remove Device Drivers Using SAM 2-36
Reconfiguring the Kernel to Modify System Parameters Using SAM 2-39
Reconfiguring the Kernel to Modify Special Devices Using SAM 2-41
Booting the Backup Kernel 2-44
Adding and Removing Subsystems 2-45
Adding a Subsystem Using HP-UX Commands 2-45
Removing a Subsystem Using HP-UX Commands 2-47
Adding and Removing Subsystems Using SAM 2-47
Removing System Files 2-50
Important Points 2-50
How to Use rmfn 2-51

3. Starting and Stopping HP-UX
Turning on the Peripherals and the Computer 3-1
Booting the System 3-2
ISL Boot 3-2
If autoboot is Enabled 3-2
Disabling autoboot 3-3
If autoboot is Disabled 3-4

Interacting with ISL to Select a System to Boot 3-5
If AUTO File is Current 3-5
Booting an Alternate Kernel, Same Location 3-6
Booting From Other Locations 3-6
Changing the Primary Boot Path 3-8
Changing the Contents of the AUTO File 3-9
Changing the AUTO file when in ISL 3-9
Changing the AUTO File Using mkboot 3-9
Examples, Creating AUTO File Bootstring 3-10
Checking the File System 3-10
Setting Initial Information 3-11
Completing Bootup 3-12
Shutting Down the System 3-12
An Overview of Shutdown 3-13
Going to the Single-user State for Maintenance 3-14
Shutting Down the System Completely 3-15
Shutting Down the System to Activate a New Kernel 3-16
Designating Shutdown Authorization 3-17
Customizing the Shutdown Process 3-17
Using SAM to Shut Down the System 3-18
Using SAM to Halt or Reboot the System 3-18
Power Failure Considerations 3-19
Power Failure Related Tasks 3-19

4. Controlling Access to Your System
Terms Used in this Chapter 4-2
Overview of Controlling Access Your System 4-4
Controlling User Accounts and Groups 4-5
Primary Groups 4-7
Group Passwords 4-7
Special Groups 4-8
Controlling File Access 4-9
File Access Permissions 4-9
Access Control Lists 4-11
Controlling Run-Levels 4-12
Managing User Accounts and Groups Tasks 4-14
Adding a User Using SAM 4-16
Removing a User Using SAM 4-19

Customizing the SAM "Adding and Removing a User" Capabilities 4-21
Deactivating a User's Account Using SAM 4-24
Reactivating a User's Account Using SAM 4-26
Displaying/Modifying a User's Account Information Using SAM 4-28
Adding a Group Using SAM 4-30
Removing a Group Using SAM 4-31
Adding and Removing Users From Groups Using SAM . . . 4-33
Adding a User Using HP-UX Commands 4-34
Removing a User Using HP-UX Commands 4-47
Deactivating a User's Account Using HP-UX Commands . . . 4-51
Reactivating a User's Account Using HP-UX Commands . . . 4-53
Displaying/Modifying User's Account Information Using HP-UX Commands 4-55
Adding a Group Using HP-UX Commands 4-58
Removing a Group Using HP-UX Commands 4-61
Changing a User's Primary Group Using HP-UX Commands . 4-67
Adding Users to Groups Using HP-UX Commands 4-72
Removing Users From Groups Using HP-UX Commands . . . 4-75
Displaying/Assigning Special Group Privileges Using HP-UX Commands 4-80
Managing Run-Levels 4-84
Creating a New Run-Level Using HP-UX Commands 4-85
Changing System Run-Levels Using HP-UX Commands . . . 4-87
Entering the System Administration Run-Level 4-89
Returning From the System Administration Run-Level . . . 4-90

5. Managing Peripherals
Overview of Adding a Peripheral 5-4
Moving Peripherals Using SAM 5-5
Moving a Local Printer or Plotter 5-5
Moving a Tape Drive 5-8
Moving a Terminal or Modem 5-11
Moving a Disk Drive Used for File Systems and Swap 5-13
Removing Peripherals Using SAM 5-18
Removing a Local Printer or Plotter 5-18
Removing a Tape Drive 5-20

Removing a Terminal or Modem 5-22
Removing a Disk Drive Used for File Systems or Swap 5-24
Creating Device Files Using HP-UX Commands 5-27
Moving Peripherals Using HP-UX Commands 5-27
Moving a Local Printer or Plotter 5-27
Moving a Tape Drive 5-29
Moving a Terminal or Modem 5-31
Moving the System Console 5-32
Moving a Disk Drive Used for File Systems and Swap 5-36
Moving a Disk Drive Containing the Root File System 5-40
Removing Peripherals Using HP-UX Commands 5-47
Removing a Printer or Plotter 5-47
Removing a Tape Drive 5-48
Removing a Terminal and Modem 5-49
Removing a Disk Drive Used for File Systems and Swap . . . 5-50

6. Managing the File System
Terms Used in this Chapter 6-2
Overview of HP-UX File Systems 6-6
What Does "File System" Mean? 6-6
Mountable File Systems 6-7
Listing Mounted File Systems 6-9
Types of File Systems 6-10
Disk Sections 6-11
Logical Volumes 6-11
Disk Arrays 6-12
Creating File Systems 6-13
Why Create New File Systems? 6-13
How Can You Create a File System? 6-13
Creating File Systems Using SAM 6-14
Are You Using LVM? 6-14
How to Use SAM 6-15
Creating a File System in a Logical Volume Using SAM 6-16
Creating a File System in a Disk Section, Using SAM 6-18
Creating File Systems Using HP-UX Commands 6-20
Are You Using LVM? 6-22
Creating a File System in a Logical Volume Using HP-UX Commands 6-23

Variations on this Procedure 6-23
Step 1. Calculate the Size Required for the Logical Volume . 6-24
Step 2. Determine if Disk Space is Available for the Logical Volume 6-25
Do You Plan to Create a Mirrored File System? 6-25
What if There is Not Enough Space In a Volume Group? . 6-25
Step 3. Create the Logical Volume 6-26
Create a Mirrored File System Logical Volume 6-28
Step 4. Find Out the Disk's Product ID 6-29
Step 5. Create the New File System 6-30
Step 6. Create a Mount Point Directory for the File System. . 6-30
Step 7. Mount the New File System. 6-31
Step 8. Confirm the File System is Mounted 6-31
Step 9. Add the File System to /etc/checklist 6-32
After Creating a File System in a Logical Volume 6-32
About the Preceding Procedure 6-32
Creating a File System in a Disk Section Using HP-UX Commands 6-33
Step 1. Select a Disk Section For a File System 6-34
Step 2. Create the New File System 6-37
Step 3. Create a Mount Point Directory for the File System. . 6-38
Step 4. Mount the New File System. 6-38
Step 5. Confirm the File System is Mounted 6-38
Step 6. Add the File System to /etc/checklist 6-39
About the Preceding Procedure 6-39
Adding a Disk 6-40
Run mediainit (if necessary) 6-40
Using a New Disk for LVM 6-41
Using the `newfs` Command. 6-42
Syntax of the `newfs` Command 6-42
Character Device File 6-43
Disk Sections 6-43
Logical Volumes 6-43
Choosing File Name Length 6-44
Long vs. Short File Names 6-44
Overriding the File System Parameters 6-45
Overriding the Default Block and Fragment Values 6-45
Overriding the Default Reserved Disk Space 6-46

Overriding the Default Bytes Per Inode 6-47
Example: Creating a File System with Smaller Block Size . . 6-48
Example: Creating a File System in a Disk Array 6-48
Example: Create a File System in a Logical Volume 6-49
Mounting and Unmounting File Systems 6-51
The Mount Directory 6-51
Mounting and Unmounting Local File Systems Using SAM . . . 6-52
Mounting File Systems Using HP-UX Commands 6-53
Syntax of the `mount` command 6-55
Example: Mounting a File System 6-56
Example: Mounting a File System, Logical Volume 6-56
Example: Mount a File System, Read-Only Permission . . 6-57
Example: CD-ROM file system 6-57
Using `umount` to Unmount a File System 6-58
Syntax of the umount Command 6-58
Examples: Unmounting a File System 6-59
Unmounting More than One File System 6-59
Automatically Unmounting Your File Systems at Shutdown . 6-60
Mounting NFS File Systems 6-61
Mounting and Unmounting NFS File Systems Using SAM . . 6-61
Configuring a Local File System for Export 6-61
Enabling or Disabling the Server File System 6-62
To Make Remote Systems Clients of a Local File System . . 6-62
Mounting Remote File Systems 6-62
Unmounting a Remote File System 6-63
Unmounting an Exported File System on a Server 6-63
Mounting NFS File Systems Using HP-UX Commands . . . 6-64
Preparations for Mounting Remote File Systems 6-64
Example: Mounting an NFS File System 6-64
NFS Mount Problems 6-65
Automatically Mounting Your File Systems at Bootup 6-66
Reading/Modifying the /etc/checklist File 6-66
Making Entries in the Checklist File for Mounting a File System at Boot 6-67
What Happens When the System Boots 6-68
Modifying File Name Length 6-69
Enabling Long File Names Using SAM 6-70
Enabling Long File Names Using convertfs 6-71

Disabling Long File Names 6-73
Checking File System Consistency 6-74
Using the fsck Utility 6-74
Displaying Current Disk Usage Information 6-75
Displaying File System and Disk Usage when in SAM 6-78
Using SAM to Recover Disk Space 6-78
Using Disk Quotas to Limit Users' Disk Space 6-79
Planning for Disk Quotas 6-80
Setting Up Disk Quotas on a File System 6-80
Mount the File System 6-81
Edit /etc/checklist to Add "quota" Option 6-81
Create the "quotas" File in the File System's Root Directory 6-82
Establish Quotas for Users of the File System 6-83
Using edquota -p 6-83
Setting quotas for an Individual User 6-84
Setting Time Limits for a File System's Users 6-84
Turning On Disk Quotas 6-85
Turning On Disk Quotas by Mounting the File System . . 6-85
Turning On Disk Quotas Using the quotaon Command . . 6-86
Turning Off Disk Quotas 6-86
Displaying File System Usage, Soft and Hard Limits 6-87
Reporting File System Usage 6-87
Reporting a Summary of Ownership 6-88
Reporting Individual Usage 6-88
What to Do When Reaching a Hard Limit 6-89
When Not in an Editor 6-89
When In an Editor 6-90
Checking Consistency of File System Usage Data 6-90
Checking Quotas When Starting the System 6-91
Running quotacheck Interactively 6-91
Moving File Systems from One Device to Another 6-92
HP-UX System Files 6-95
The Location of the Root File System 6-99
LVM 6-99
Non-LVM 6-100
Moving the Root File System from Section to Section 6-101
Summary of Steps to Move the Root Disk Section 6-101
Example 6-102

First, build a new kernel 6-102
Then, back up your system completely 6-103
Put the system into single-user mode 6-103
Copy the root file system 6-103
Update two files on the new root 6-104
Bring the new kernel over 6-104
Reboot the system to test the new kernel and root file system 6-104
Create new AUTO file for autoboot 6-105
Make a new /tmp file system over the old root file system . 6-106
Creating a LIF Volume 6-107
Prerequisites and Conditions 6-107
Manual Procedure for Making a LIF Volume 6-108

7. Managing Swap Space
Terms Used in this Chapter 7-2
Types of Swap Space 7-4
Device Swap 7-4
File System Swap 7-4
How Much Swap Space Do You Need? 7-5
How Much Swap Do You Have? 7-6
Estimating Swap Space Needs (A Rule of Thumb) 7-7
Swap Space Defaults 7-8
Allocating Swap Areas: Guidelines 7-9
Distributing Device Swap Areas 7-9
Choosing Device Swap Areas 7-10
Disk Sections 7-10
Logical Volumes 7-10
Guidelines for Setting Up File System Swap Areas 7-10
Guidelines for Assigning Swap Priority 7-12
Adding, Removing, or Changing Device Swap Space 7-13
Changing Primary Swap from Section to Section 7-14
Configuring Primary Swap with SAM 7-14
Example, Changing the Primary Swap Area Using HP-UX Commands 7-14
Changing Primary Swap Logical Volume 7-17
Adding/Removing Device Swap 7-18
Are You Using LVM? 7-18

Adding Device Swap Using SAM 7-19
How to Use SAM 7-19
Adding Device Swap in a Logical Volume Using SAM 7-19
Creating a Logical Volume for Device Swap 7-20
Adding Device Swap in a Logical Volume Using SAM 7-20
Adding Device Swap in a Disk Section Using SAM 7-21
Disabling Device Swap Using SAM 7-23
Adding Device Swap with the /etc/swapon Command 7-24
Determining the Block Device File for the Chosen Section . . 7-24
Determining the Block Device File for a Logical Volume . . . 7-25
Setting the Priority for the Secondary Swap Device 7-25
Example: Enabling Device Swap in a Disk Section 7-25
Example: Enabling Device Swap in a Logical Volume 7-26
Verify that Device Swap is Enabled; the swapinfo Command . 7-26
Enabling Your Swap Device Each Time You Boot the System 7-26
Removing Device Swap Using HP-UX Commands 7-27
Adding, Removing File System Swap 7-28
Adding File System Swap Using SAM 7-29
Changing File System Swap Values Using SAM 7-30
Removing File System Swap You Added With SAM 7-30
Adding File System Swap Using /etc/swapon 7-30
Choosing a File System for Swap 7-31
Determine Mount Point Directory of the File System . . . 7-31
Choosing Values for /etc/swapon Parameters 7-32
Example, Setting Up File System Swap with /etc/swapon . 7-33
Removing File System Swap Using HP-UX Commands . . . 7-35
Enabling Swap Areas on Boot Using the Checklist File . . . 7-36
Making Entries for Device Swap in the Checklist File . . . 7-36
Making Entries for File System Swap in the Checklist File . 7-37
Adding a Disk 7-38
Run mediainit (if necessary) 7-38
Using a New Disk for LVM 7-39
Setting Up Dumps Areas 7-40
How Much Disk Space to Use For Dumps? 7-40
Finding Physical Memory 7-41
Configuring Dumps with SAM 7-41
Configuring Disk Sections as Dumps Devices 7-42
Reconfigure the Kernel 7-42

Configuring Logical Volumes to be Dump Areas 7-43
Updating the BDRA to Specify Logical Volume Dumps Areas 7-44
Updating the BDRA to Remove a Logical Volume as a Dumps Device 7-44
Reboot to Implement Changes to BDRA 7-44
Specifying Logical Volumes Dumps in the S800 File 7-44
Specifying Both Logical Volume and Disk Sections as Dumps Devices 7-45
Reconfigure the Kernel 7-45

8. Managing Logical Volumes
What This Chapter Covers 8-3
Terms Used in this Chapter 8-4
Terms Used in this Chapter (Mirroring Only) 8-7
Are You Running LVM Now? 8-9
Making Logical Volume Manager Available for Use 8-10
If Hewlett-Packard Has Pre-Installed Release 9.0 on Your System 8-10
If You Have Migrated Your Root Disk From Release 8.0 to Release 9.0 8-10
If You Have Updated to Release 9.0 Without Migrating the Root File System 8-11
If You Still Have Release 8.0 Running on Your System 8-12
LVM—An Overview 8-13
Overview of How Logical Volumes are Organized 8-13
Displaying the LVM Organization on Your System 8-15
Physical Extents, Logical Extents 8-17
Mirroring of Data 8-19
Listing What's In the Logical Volumes 8-20
Listing File Systems in Logical Volumes 8-20
Listing Swap Areas in Logical Volumes 8-20
Listing Raw Data in Logical Volumes 8-21
LVM Naming Conventions 8-22
Disk (Physical Volumes) Names 8-22
Volume Groups Names: 8-22
Logical Volume Names 8-22
Physical Volume Group Names 8-23

Guidelines for Using Logical Volumes 8-24
Planning Logical Volumes for File Systems 8-24
What Size Logical Volume Does a File System Require? . . 8-24
Example, Estimating the Size of a Logical Volume for a File System 8-26
Increasing the Size of a File System 8-26
Guidelines for Spanning Disks With File Systems 8-26
Guidelines for Creating Logical Volumes for Swap 8-27
Guidelines for Creating Logical Volumes for Raw Data Storage. 8-27
Sizing a Logical Volume for a Database Partition 8-28
Distributing Logical Volumes Over Disks 8-28
Naming Logical Volumes for Raw Data 8-29
Guidelines for Using Disk I/O Interfaces 8-29
Planning for Bad Block Relocation 8-30
Use SAM or HP-UX Commands to Manage Logical Volumes . . 8-31
Managing LVM Using SAM 8-32
How to Use SAM 8-33
Adding Disks Using SAM 8-34
Some Performance Reminders 8-34
Add a Disk to an Existing Volume Group (Extend a Volume Group) 8-35
Create a New Volume Group with a New Disk. 8-36
Creating Logical Volumes Using SAM 8-37
Creating a Logical Volume in SAM 8-38
Creating a File System in a Logical Volume Using SAM . . . 8-39
Increasing the Size of a File System Using SAM 8-40
Decreasing the Size of a Logical Volume Using SAM 8-41
Creating a Swap Area in a Logical Volume Using SAM . . . 8-42
Managing Logical Volumes Using HP-UX Commands 8-43
LVM Commands - Quick Reference 8-43
LVM Commands - Quick Reference, Mirroring Only 8-44
Creating Volume Groups Using HP-UX Commands 8-46
Points to Remember 8-49
Example: Setting Up a Volume Group with Two Disks . . . 8-50
Adding Disks to an Existing Volume Group 8-51
Example, Adding Two Disks to an Existing Volume Group . 8-51
Displaying Information About Volume Groups, Physical Volumes 8-52

Listing a Specific Volume Group 8-52
Listing All Volume Groups 8-53
Displaying Detailed Volume Group Information 8-53
Creating Logical Volumes 8-55
Creating a Logical Volume, Simple Case 8-55
Creating Logical Volumes, Optional Characteristics 8-56
Options Available When Creating Logical Volumes with `lvcreate` 8-57
More Examples: Creating Logical Volumes 8-58
Displaying Information About Logical Volumes 8-59
Changing the Characteristics of a Logical Volume 8-60
Example, Changing Allocation Policy from Contiguous to Non-Contiguous 8-60
Increasing the Size of a Logical Volume 8-61
Example, Increasing the Size of a Logical Volume 8-61
Example: Increasing the Number of Logical Extents in a Logical Volume. 8-61
Extending a Logical Volume to a Specific Disk 8-62
Extending a Logical Volume and Extending a File System . . 8-63
Reducing the Size of a Logical Volume 8-64
Example, Reducing the Size of a Logical Volume 8-64
Moving Data in Logical Volumes From Disk to Disk 8-65
Example, Moving Logical Volume Data from One Disk to Another 8-65
Example, Moving All Data on One LVM Disk to Another. . 8-66
Removing Logical Volumes 8-66
Managing Mirrors 8-67
Guidelines for Managing Mirrors 8-68
Number of Mirror Copies 8-68
Guidelines for Setting Allocation Policy 8-68
Mirror Write Scheduling Policy 8-71
Configuring Mirrors for Crash Recovery 8-72
Guidelines: Mirroring Logical Volumes with File Systems . 8-74
Guidelines for Mirroring Swap 8-74
Using SAM to Manage Mirrors 8-75
Creating a Mirrored Logical Volume Using SAM 8-76
Splitting and Merging Mirrored Logical Volumes Using SAM . 8-79
Splitting a Logical Volume Using SAM 8-79

Merging Split Logical Volumes 8-81
Modifying Mirrored Logical Volumes Using SAM 8-82
Using HP-UX Commands to Manage Mirrors 8-83
Creating Mirrored Logical Volumes 8-83
Options Available When Creating Mirrored Logical Volumes 8-83
Example: Creating a Mirrored Logical Volume 8-84
Increasing the Number of Mirror Copies 8-85
Examples: Increasing the Number of Mirror Copies 8-85
Example: Creating Mirrored Logical Volumes that Use Specific Disks 8-86
Mirroring Across Separate I/O Channels - PV Groups 8-87
Why Create Physical Volume Groups? 8-87
Setting Up Physical Volume Groups 8-87
Creating Physical Volume Groups with `vgcreate`, `vgextend`, Example: 8-87
Creating Logical Volumes in Physical Volume Groups . . . 8-88
The `/etc/lvmpvg` File 8-89
Splitting and Merging Mirrored Logical Volumes 8-89
Discussion, Splitting and Merging a Mirrored Logical Volume 8-90
Online Backup Using `lvsplit`, `lvmerge`, Example: 8-90
Reducing the Number of Mirror Copies 8-92
Example, Reducing the Number of Mirror Copies 8-92
Example, Reducing a Mirror Copy from a Specific Disk . . 8-92
Moving a Logical Volume Mirrored Copy from One Disk to Another 8-93
Synchronizing a Mirrored Logical Volume 8-94
Automatic Synchronization 8-94
Manual Synchronization 8-94
Example: Synchronizing a Logical Volume 8-95
Synchronizing All Mirrored Logical Volumes in a Volume Group 8-95
Working with Volume Groups 8-96
Setting Volume Group Availability 8-96
Deactivating Volume Groups 8-96
Example: Deactivating a Volume Group 8-96
Activating Volume Groups 8-97
Reactivating an Active Volume Group 8-97

Editing the `lvmrc` File for Automatic Volume Group Activation 8-98
Using Options When Activating Volume Group 8-99
Reconfiguring Disks in a Volume Group 8-100
Moving a Volume Group's Disks 8-100
Exporting and Importing a Volume Group 8-101
Backing Up and Restoring the Volume Group Configuration . . 8-103
When to Use vgcfgbackup 8-103
When You Might Need to Use vgcfgrestore 8-103
Running `vgcfgbackup` 8-103
Running `vgcfgrestore` 8-104
Listing Backed Up Configuration Information 8-104
Restoring Configuration Information, Changed Device File . 8-104
Recovering an lvmtab File - Using vgscan 8-105
Recovering From Disk Failures in a Volume Group 8-106
LVM Quorum 8-106
Recovering Quorum 8-106
The Root Volume Group 8-107
Run lvlnboot When Making Changes to the Root Volume Group 8-109
Running lvlnboot -v to Show Current Root Volume Group . . 8-109
Running lvlnboot When Adding Non-Root Disks to the Root Volume Group 8-110
Creating Boot Disks in the Root Volume Group 8-111
Making Root, Swap, and Dump Logical Volumes 8-111
Creating Root Logical Volumes 8-112
Creating a New Logical Volume for Root 8-112
Mirroring an Existing Root Logical Volume 8-113
Updating Boot Information in the BDRA, Root Logical Volumes 8-114
Primary Swap Logical Volumes 8-115
Creating a New Logical Volume for Primary Swap 8-115
Mirroring an Existing Primary Swap Logical Volume . . . 8-115
Updating Boot Information in the BDRA, Primary Swap Logical Volumes 8-115
Creating Dumps Logical Volumes 8-117
Creating a New Logical Volume for Dumps 8-117

Updating Boot Information in the BDRA, Dumps Logical Volumes 8-117
Using primary swap for dumps 8-118
Example: Mirroring the Root File System Logical Volume . . . 8-119
Moving Root from a Disk Section to a Logical Volume 8-121
Example: Creating a Root Logical Volume on a New Disk . . 8-122
Build a New Kernel, with Root in a Logical Volume 8-126
Reboot and Try the New Configuration 8-127
After Successfully Moving Root 8-128
Change the Primary Boot Path in ISL 8-128
Create Configuration File and Perform Critical Backups . . . 8-129
Create the Volume Group Configuration File 8-129
Back Up the New Root File System 8-129
Transfer Other Data from the Root Disk to Logical Volumes . 8-130
LVM Kernel Configuration Requirements 8-131
LVM-Related Tunable Operating System Parameters 8-131
Configuring Kernel Devices, LVM 8-132
Configuring Root, Swap, and Dump Logical Volumes in Kernel 8-132
Mixed LVM and Non-LVM Kernel Devices 8-133
Non-LVM Root, Swap, and Dumps in Kernel 8-133
Migrating Non-Root Disks From Traditional Sections to Logical Volumes 8-134
Migrating File Systems 8-134
The `LVMMIGRATE` File. 8-135
Using SAM to Migrate File Systems to Logical Volumes . . . 8-135
Migrating Data From DataPair/800 to LVM's MirrorDisk/UX . 8-139
Example: Migrating Two DataPair/800 Disks to LVM 8-139

9. Backing Up and Restoring Your Data
Terms Used in this Chapter 9-2
Determining Which Data to Back Up 9-4
Included Files 9-5
Excluded Files 9-6
Graph Files 9-7
Determining How Often to Back Up Data 9-8
Full Backups vs. Incremental Backups 9-9
Backup Levels 9-9
Recovery Example Using Three Backup Levels 9-11

Recovery Example Using Two Backup Levels 9-12
Choosing the Type of Storage Device to Use 9-13
Automating the Backup Process 9-20
Restoring Your Data 9-21
Backing Up and Restoring Tasks 9-22
Backing Up Your System To a Local Device Using SAM . . . 9-23
Backing Up Your System To a Remote Device Using SAM . . 9-27
Setting Up an Automated Backup Schedule Using SAM . . . 9-31
Viewing Entries 9-33
Adding an Entry 9-34
Removing Entries 9-39
Override the Backup Schedule to Perform the Backup Now . 9-40
Restoring All Files From a Local Device Using SAM 9-41
Restoring Individual Files From a Local Device Using SAM . 9-44
Restoring All Files From a Remote Device Using SAM 9-47
Restoring Individual Files From a Remote Device Using SAM 9-50
Viewing the Index File on the Local Media Using SAM . . . 9-54
Viewing the Index File on a Remote Device Using SAM . . . 9-56
Viewing the Index Files on Your System Using SAM 9-58
Backing Up Your System Using HP-UX Commands 9-59
The `fbackup` Command 9-60
Examples Using the `fbackup` Command 9-63
Backing Up to a Hard Disk Using the `cpio` Command . . . 9-68
Backing Up To a Hard Disk or DDS Format Tape Drive Using the `dd` Command 9-70
Setting Up an Automated Backup Schedule Using HP-UX Commands 9-73
Creating an Automated Backup Schedule File 9-74
Activating Your Automated Backup Schedule 9-76
Displaying Your Automated Backup Schedule 9-77
Deactivating Your Automated Backup Schedule 9-78
Changing Your Automated Backup Schedule 9-78
Restoring Files Using HP-UX Commands 9-78
The `frecover` Command 9-79
Examples Using the `frecover` Command 9-81
Restoring Data From a Hard Disk Using the `cpio` Command . 9-84
Restoring Data From a Hard Disk Using the `dd` Command . . 9-85

Viewing the Index File on the Local Device Using HP-UX Commands 9-87
Other Backup and Restore Utilities 9-88

10. Managing Printers and Printer Output
What is the Line Printer Spooling System? 10-1
Terms Used in this Chapter 10-3
LP Spooler Overview 10-6
The Components of the LP Spooler 10-8
Printer Names 10-8
Printer Classes 10-9
Print Destinations 10-10
System Default Printer (Destination) 10-10
Printer Interfaces 10-11
Printer Models 10-11
Device Files 10-13
Line Printer Scheduler 10-13
Local Printer 10-13
Remote Printer 10-14
Network-Based Printer/Plotter 10-14
Print Request Identification Number 10-14
Remote Spooling 10-15
Priorities of Printers and Print Requests 10-17
Using Plotters with the LP Spooler 10-18
Controlling Data Flow Through the LP Spooler 10-19
Logging and Analyzing Printer Activity 10-20
Initial LP Spooler Set Up 10-21
Line Printer Spooler Tasks 10-22
Viewing Printers and Print Request Status Using SAM . . . 10-23
Adding a Local Printer Using SAM 10-25
Adding a Remote Printer Using SAM 10-29
Adding a Network-Based Printer Using SAM 10-32
Removing a Printer Using SAM 10-34
Starting and Stopping the LP Spooler Using SAM 10-36
Determining the Status of the LP Spooler Using SAM 10-38
Disabling a Printer Using SAM 10-39
Enabling a Printer Using SAM 10-41
Changing A Printer Fence Priority Using SAM 10-43

Setting Up the LP Spooler Using HP-UX Commands 10-44
Ensuring the Printer Driver is in the Kernel Using HP-UX Commands 10-45
Determining if a Device File Exists for your Printer Using HP-UX Commands 10-47
Adding a Local Printer Using HP-UX Commands 10-49
Adding a Remote Printer Using HP-UX Commands 10-53
Adding a Network-Based Printer Using HP-UX Commands . 10-58
Creating a Printer Class Using HP-UX Commands 10-59
Removing a Printer or Printer Class Using HP-UX Commands 10-61
Accepting and Rejecting Print Requests for a Printer Using HP-UX Commands 10-64
Enabling or Disabling a Printer Using HP-UX Commands . . 10-66
Setting a Printer Priority Fence Using HP-UX Commands . . 10-68
Starting and Stopping the LP Spooler Using HP-UX Commands 10-69
Canceling Print Requests Using HP-UX Commands 10-70
Moving *All* Requests Using HP-UX Commands 10-72
Moving Selected Print Requests Using HP-UX Commands . . 10-75
Viewing the Status of Printers and Print Requests Using HP-UX Commands 10-76
Changing the Priority of Print Requests Using HP-UX Commands 10-78
Displaying Statistics about Printer Activity Using HP-UX Commands 10-80

11. Communicating With the Users on Your System
Terms Used in this Chapter 11-2
Displaying Messages to Users Logging into Your System 11-3
`motd` 11-3
`news` 11-3
Communicating With a User Who is Logged into Your System . 11-4
Broadcasting a Message to All Users Logged into Your System . 11-4
Communicating With Users on Other Systems Using UUCP . . 11-5
Using Electronic Mail Systems 11-6

12. System Accounting Concepts and Tasks
What Is in This Chapter? 12-2
Installation and Daily Usage 12-3
How to Install System Accounting 12-3
Update /etc/rc 12-3
Create crontab Entries 12-4
Set PATH for Accounting Commands 12-5
Summary of Daily Operation 12-5
Overview of System Accounting 12-7
Definitions 12-7
prime/non-prime connect time 12-7
process accounting records 12-8
total accounting records 12-8
Introduction to Commands 12-9
Installation 12-9
Disk Space Usage Accounting 12-10
Connect Session Accounting 12-10
Process Accounting 12-11
Charging Fees 12-12
Summarizing and Reporting Accounting Information . . . 12-12
System Data Flow 12-13
Login and Directory Structure 12-15
Logging In 12-15
Directory Structure 12-15
Disk Space Usage Accounting 12-17
Reporting Disk Space Usage 12-17
acctdusg 12-17
diskusg 12-19
Creating Total Accounting Records 12-22
acctdisk 12-22
dodisk 12-22
Connect Session Accounting 12-24
Writing Records to wtmp – acctwtmp 12-24
Displaying Connect Session Records – fwtmp 12-25
Fixing wtmp Errors – wtmpfix 12-27
Creating Total Accounting Records 12-27
acctcon1 12-27
prctmp 12-30

acctcon2 12-30
Process Accounting 12-32
Turning Process Accounting On 12-32
turnacct on 12-33
accton 12-33
Turning Process Accounting Off 12-34
turnacct off 12-34
accton 12-34
Checking the Size of pacct 12-34
ckpacct 12-35
turnacct switch 12-36
Displaying Process Accounting Records – acctcom 12-37
Definitions of Information Produced by acctcom 12-38
Report Format Options 12-42
Record Selection Options 12-44
Sample Reports 12-47
Command Summary Report – acctcms 12-48
Producing a Readable Report – the -a option 12-48
Other Options 12-50
Sample Report 12-51
Creating Total Accounting Records 12-52
acctprc1 12-53
acctprc2 12-54
Charging Fees to Users – chargefee 12-55
Summarizing and Reporting Accounting Information 12-56
Displaying Total Accounting Records – prtacct 12-56
Report Format 12-57
Example 12-59
Merging Total Accounting Files – acctmerg 12-60
Command Options 12-61
Example 12-62
Creating Daily Accounting Information – runacct 12-63
Files Processed by runacct 12-63
The States of runacct 12-64
Recovering from Failure 12-66
Restarting runacct 12-67
Daily Reports 12-68
Displaying runacct Reports – prdaily 12-69

Daily Line Usage Report 12-70
Daily Resource Usage Report 12-71
Daily and Monthly Command Summary 12-71
Last Login 12-71
Creating Monthly Accounting Reports – monacct 12-72
Updating the Holidays File 12-73
Fixing Corrupted Files 12-74
Fixing wtmp Errors 12-74
Fixing tacct Errors 12-75
Sample Accounting Shell Scripts 12-76
grpdusg 12-76
The Shell Script 12-77
acct_bill 12-78
Examples 12-78
Output Produced by acct_bill 12-79
The Shell Script 12-79
System Accounting Files 12-81
Files in the /usr/adm Directory 12-81
Files in the /usr/adm/acct/nite Directory 12-82
Files in the /usr/adm/acct/sum Directory 12-83
Files in the /usr/adm/acct/fiscal Directory 12-83

A. System Parameters
Logical Groups of Parameters A-2
Using SAM to Check or Change HP-UX Kernel Parameters . . A-3
Parameter Descriptions A-6
acctresume A-7
acctsuspend A-8
bufpages A-9
dst A-11
fs_async A-12
iomemsize A-13
maxdsiz A-14
maxfiles A-15
maxfiles_lim A-16
maxssiz A-17
maxtsiz A-18
maxuprc A-19

maxusers A-20
msgmap A-22
msgmax A-24
msgmnb A-25
msgmni A-26
msgseg A-27
msgssz A-28
msgtql A-29
nbuf A-30
ncallout A-32
netisr_priority A-33
nfile A-34
nflocks A-35
ninode A-36
nproc A-37
npty A-38
nstlbe A-39
nswapdev A-41
nswapfs A-42
semaem A-43
semmap A-44
semmni A-46
semmns A-47
semmnu A-48
semume A-49
semvmx A-50
shmmax A-51
shmmni A-52
shmseg A-53
swchunk A-54
timeslice A-55
timezone A-56
unlockable_mem A-57

B. Swap Space Computation
Swap Space Computation B-1
Swap Space Computation Worksheet B-3
Example B-6

C. Federal Information Processing Standard
Restricting the chown(1) Function C-1
Controlling the Group ID of New Files C-2
Truncating Filenames During Pathname Resolution C-3

Index

Figures

1-1. SAM Main Window 1-23
1-2. The Main Window in SAM's Text-terminal Interface 1-27
1-3. Highlighting a New List-item Choice 1-28
1-4. Highlighting a Control Button 1-30
1-5. A Subarea List Window 1-33
1-6. A Functional Area 1-34
1-7. Selecting an Object 1-35
1-8. A List of Menu Items 1-36
2-1. `rmfn`: "Partitions" Screen 2-51
6-1. Typical HFS File System Structure 6-6
6-2. Mounting a File System 6-8
6-3. Activities in Creating a File System 6-21
6-4. 6-54
8-1. Organization of Logical Volumes Within Volume Groups . . . 8-14
8-2. Physical Extents and Logical Extents 8-18
8-3. File System Space 8-25
8-4. Mirror Copy Allocation, Strict and Non-strict 8-70
9-1. An Example Directory Tree 9-5
9-2. Example of Using Incremental Backups 9-11
9-3. Example of Using Incremental Backups Supported by SAM . 9-12
9-4. An Example Directory Tree 9-63
9-5. An Example Directory Tree 9-81
10-1. Line Printer Spooler "Plumbing Diagram." 10-7
12-1. Disk Usage Accounting 12-13
12-2. Connect Session Accounting 12-13
12-3. Process Accounting 12-14
12-4. Charging Fees 12-14
12-5. Summarizing and Reporting 12-14
12-6. System Accounting Directory Structure 12-16

Tables

1-1. What is in Each Chapter of this Manual 1-3
1-2. Other Sources of System Administration Information 1-8
1-3. Function Keys for SAM's Text-Terminal Interface 1-31
1-4. Meanings of Selected Keys 1-39
2-1. Pseudo-Drivers 2-46
4-1. HP-UX Global and Local Shell Initialization Files 4-38
4-2. "Removing a Group" Task Decision Table 4-62
4-3. "Changing a User's Primary Group" Task Decision Table . . 4-68
4-4. "Removing Users From Groups" Task Decision Table 4-76
6-2. File System Parameters 6-47
6-3. Root File System Subdirectories 6-96
6-1. Root File System Subdirectories (Continued) 6-97
6-1. Root File System Subdirectories (Continued) 6-98
7-1. /etc/swapon Parameters 7-32
8-1. Disk Interface Types and LVM Support 8-29
8-2. LVM Commands 8-43
8-3. LVM Commands - Mirroring 8-45
9-1. Determining Which Device to Use for a Backup 9-14
9-2. Disk Drive Storage Capacities 9-15
9-3. Magnetic Tape Storage Capacities 9-16
9-4. HP Format Cartridge Tape Storage Capacities 9-16
9-5. DDS Format (DAT) Tape Storage Capacities 9-17
9-6. Optical Disk Surface Section Storage Capacities 9-18
9-7. QIC-Format Cartridge Tape Storage Capacities 9-19
9-8. fbackup command options 9-61
10-1. LP Spooler Models for Plotters 10-18
10-2. Series 800 Printer Drivers 10-28
10-3. Series 800 Printer Drivers 10-46
10-4. Series 800 Printer Drivers 10-47
10-5. Interpreting `lpana's` Output 10-81

A-1. . . . . A-2

1

Introduction to System Administration

This manual is written for the HP-UX System Administrator. It is written to assist system administrators, with differing levels of expertise, accomplish the various *tasks* associated with maintaining an HP-UX system.

Conceptual material is provided when it directly supports the task being described. For a more thorough coverage of the *concepts* behind administering an HP-UX system, see *How HP-UX Works: Concepts for the System Administrator*, HP part number B2355-90029.

This manual assumes that you are running the 9.0 Release of HP-UX. If you are not yet running Release 9.0, your first task should be to update to 9.0. For details on how to do this, see *Installing and Updating HP-UX*, HP part number B3108-90006.

How to Use this Manual

This manual describes the *tasks* associated with HP-UX system administration. Begin your search for a particular task description in the *Table of Contents*.

The System Administration Manager (SAM) utility can be used to perform many HP-UX system administration tasks. For those tasks, this manual presents both methods (the SAM method and the "HP-UX Commands" method). For information on how to use SAM, see "Using SAM".

Conventions Used in this Manual

This manual uses the following typographic conventions:

Boldface	Words defined for the first time appear in boldface. For example, **non-volatile memory** is not erased when the computer's power is shut off.
`Underlined Computer`	Underlined computer font indicates items you type. For example: `/etc/newfs`
`Computer`	Computer font within text indicates HP-UX commands and utilities. Computer font within text is also used to refer to SAM column titles.
Italics	Manual titles, emphasized words, parameters to an HP-UX command, and entries in the *HP-UX Reference Manual* manual appear in italics.
Return	Words or letters in boxes refer to keys on the keyboard or a control button within SAM.
[ ]	Items enclosed in square brackets are optional.
|	A vertical bar means that you should choose one of the items on either side of the bar, but not both.
`Softkey`	Softkey font indicates menu items within the SAM interface.
"Double Quotes"	Indicate titles of sections within a manual, manuals, SAM windows, and SAM menubar menus.

What's in this Manual

This manual is organized as follows:

Table 1-1. What is in Each Chapter of this Manual

Chapter 1, "Introduction to System Administration" ■ Guidelines for using this manual ■ An Overview of using SAM: □ A typical SAM task description □ Customizing the object list □ X Window System interface □ Text terminal interface
Chapter 2, "Constructing an HP-UX System" Provides a general overview of what to set up after installing or updating from a previous release. It also contains kernel reconfiguration procedures and pointers to useful, related information.
Chapter 3, "Starting and Stopping HP-UX" Discusses how to start up (boot up) your system. It also covers how to properly shut down your system (to avoid file system corruption).
Chapter 4, "Controlling Access to the System" Discusses how to control access to your system and its resources. It contains information about users, groups of users, file access permissions, and run-levels.
Chapter 5, "Managing Peripherals" Provides an overview of: ■ adding a peripheral ■ removing a peripheral ■ moving a peripheral

Table 1-1. What is in Each Chapter of this Manual (continued)

Chapter 6, "Managing the File System" Provides information about: ■ creating new file systems. ■ mounting and unmounting file systems. ■ different types of file systems. This chapter also discusses **disk quotas**; a method of controlling the disk space used by individual users.
Chapter 7, "Managing Swap Space" Describes the tasks associated with managing swap space (part of HP-UX's virtual memory system). It also discusses the various types of swap space.
Chapter 8, "Managing Logical Volumes" Discusses using LVM, the Logical Volume Manager. It describes guidelines and methods for using either SAM or HP-UX commands to: ■ add disks to create or extend volume groups. ■ allocate the disk space in volume groups for logical volumes. ■ use logical volumes for file systems and swap space. Other topics covered include using SAM or HP-UX commands to: ■ mirror data with MirrorDisk/UX, the LVM mirroring subsystem. ■ extend or reduce logical volumes. ■ manage root, primary swap, and dumps in logical volumes. ■ migrate data in non-root disk sections to logical volumes.
Chapter 9, "Backing Up and Restoring Your Data" Describes the tasks of backing up and restoring data as well as guidelines for deciding which device to use to back up your system.
Chapter 10, "Managing Printers and Printer Output" Describes the tasks associated with managing the line printer spooling system. The HP-UX line printer spooling system is used to control output to printers.
Chapter 11, "Communicating With the Users on Your System" Discusses various ways that you can communicate with the users of your system. As a system administrator, you may find it necessary to communicate with the users of your system (for example, to let them know about an upcoming event such as an operating system upgrade).

Table 1-1. What is in Each Chapter of this Manual (continued)

Chapter 12, "System Accounting" Provides information about the System Accounting subsystem which allows you to closely monitor the usage of the resources of your system (and if necessary, bill users for their resource usage).
Appendix A, "System Parameters" Describes numerous operating system parameters that can be used to "tune" the operation of your system.
Appendix B, "Swap Space Computation" Contains some additional (detailed) information (beyond what is in Chapter 7, "Managing Swap Space") about calculating how much swap space you need to allocate for your system. It should be needed only when disk space is very limited.
Appendix C, "Federal Information Processing Standard" Describes how to modify certain areas in HP-UX so that they conform to the Federal Information Processing Standard (FIPS).

What System Administrators Need to Know

This manual is intended to assist system administrators of all skill levels maintain their systems. However, there are basic skills that you should know before you attempt to administer an HP-UX system.

Specifically, you should know how to do the following things:

- Log in and out of your system.
- Move about the HP-UX directory tree (change directories).
- Distinguish between "absolute" and "relative" path names.
- Edit files using one of the HP-UX editors (such as `vi` or `ed`).
- Display the contents of files using `cat` or `more`.
- Search for text in files using `grep`.
- Move, copy, and remove files using `mv`, `cp`, and `rm` respectively.
- Use at least one of the HP-UX shells (such as `sh`, `csh`, `ksh`).
- Display manual reference pages using the `man` command.

If you need to learn more about the above topics, see:

A Beginner's Guide to HP-UX, HP part number B1862-90000

Shells: User's Guide, HP part number B2355-90046

The Ultimate Guide to the `vi` *and* `ex` *Text Editors*, HP part number 97005-90015 (Benjamin/Cummings Publishing Co.)

A System Administrator's Responsibilities

The HP-UX System Administrator is responsible for installing and configuring the HP-UX operating system software, and for maintaining the system and repairing it when something goes wrong. More specifically, the HP-UX system administrator may need to do the following things:

- Install and test the hardware.
- Install the HP-UX operating system software.
- Configure the HP-UX operating system.
- Update the HP-UX operating system software.
- Allow users to access the system (provide user accounts, etc.).
- Add peripheral devices to the system.
- Monitor file system use and growth.
- Back up and restore files.
- Detect and correct file system errors.
- Assist others in using the system.
- Provide a backup system administrator to assist users (when the primary administrator is unavailable).

Where to Find Other Information About System Administration

The following table lists other manuals/books that can help you administer your system:

Table 1-2. Other Sources of System Administration Information

Title	Focus
Installing Peripherals, HP part number B3108-90007	Contains information about adding peripherals to your system and creating device files for those peripherals.
How HP-UX Works: Concepts for the System Administrator, HP part number B2355-90029	Provides detailed information about the *concepts* behind HP-UX system administration tasks.
Installing and Updating HP-UX, HP part number B3108-90006	Explains how to install (or re-install) HP-UX on your system. It also describes how to update HP-UX from a previous release.
HP-UX System Security, HP part number B1862-90009	Explains the special requirements for setting up HP-UX in a secure operating mode. If your system requires above-average security, use the *HP-UX System Security* manual as your primary system administration manual and use *this* manual as a secondary reference.

Table 1-2.
Other Sources of System Administration Information (continued)

Title	Focus
HP-UX Reference Manual, HP part number B2355-90033	Provides a complete reference to HP-UX commands; the printed version of the manual reference pages. Commands in the reference are divided into numbered sections.
Owner's Guide for HP-UX Users	This manual presents much of the initial information you will need to get started with your particular HP 9000 machine. The document title and part number vary according to specific model you ordered.

Using SAM

SAM stands for *S*ystem *A*dministration *M*anager. This tool allows you to perform many system administration tasks without using the underlying HP-UX commands that are associated with the task. SAM can also save you time and keystrokes.

SAM can help you with tasks in the following areas:

- Working with users' accounts
- Working with groups of users
- Maintaining system security
- Working with file systems
- Configuring your swap space
- Adding or removing peripherals
- Working with the line printer spooler
- Backing up and recovering files (automated or manual system backups)
- Configuring network connections (LAN, X.25, FDDI, and TokenRing)
- Configuring UUCP communication
- Administering systems remotely from one location
- Configuring the HP-UX kernel

Note SAM is an optionally loadable part of HP-UX. If you have not loaded SAM onto your system, you will not be able to use it. You can use the `update` program to add SAM to your system if you did not originally load it and currently want to use it. For details about how to do this, see the *Installing and Updating HP-UX* manual.

SAM Overview

SAM has two user interfaces, an X Window System interface and a text terminal interface. The two interfaces differ in the screen appearance and keyboard/mouse interactions. For text terminal systems, refer to "Using SAM with a Text Terminal".

SAM provides an on line help system to assist you when you need additional information. A procedure for accessing the SAM help system is provided later in this section. Refer to the SAM help system for information about navigating within SAM and the following primary components of SAM:

- SAM main window
- functional areas
- step menus
- message boxes
- checkboxes
- control buttons:

  [OK]

  [Apply]

  [Cancel]

  [Help]

  [Add]

  [Modify]

  [Remove]
- menu buttons
- grey areas
- "View" menu
- "Options" menu
- "Actions" menu
- "Help" menu

To get into the SAM help system:

1. Ensure that you have superuser capabilities.
2. Enter SAM; type:

   `/usr/bin/sam`

3. Activate the (Help) control button. SAM displays the help window. If necessary, resize the window to read the text. The help text reformats automatically.
4. Scroll the text until you see the "See Also" section.
5. Enter the "Using SAM (The System Administration Management Tool)" area.
6. Scroll the text until you see a list of additional SAM topics.
7. Enter the "SAM Overview" area.
8. Scroll the text until you see the following two sections:

   "Primary Components of the SAM User Interface"

   "Terms Used in Describing the SAM User Interface"

9. Enter and read each of these areas. The "Terms Used in Describing the SAM User Interface" area describes:

   - Highlighting objects
   - Choosing menu items
   - Turning on and off checkboxes
   - Activating control buttons

There are two additional methods to get information from the SAM help system:

1. You can choose one of the following items from the "Help" menu within a functional area:

   `Overview...` to get information about the current functional area.

   `Keyboard...` to get information about keyboard navigation within SAM.

   `Using Help ...` to get information about using the SAM Help system.

   `Product Information...` to display the version of SAM you are running.

2. You can press the (f1) *keyboard key* from within a dialog or message box. Pressing the (f1) key gives you context-sensitive information for the object field at the location of the cursor. For text terminals, refer to "Using SAM with a Text Terminal" to for keyboard key mappings.

A Typical SAM Task

A typical task using SAM consists of the following steps:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

3. Enter a functional subarea by highlighting an item from the list in the SAM main window and activating the (Open) button. If you are given another list, repeat this procedure. For example, highlight **Peripheral Devices** and activate the (Open) button and then highlight **Tape Drives** and activate the (Open) button.
4. (optional) Customize the contents or format of an object list within a functional area. To customize the object list, see "Overview of Customizing the Object List".

5. (optional) Update the objects displayed by choosing `Refresh List` from the "Options" menu. This updates the data displayed.

   For example, if you are monitoring your print request in the print request queue, you would frequently update the displayed data by choosing `Refresh List` to reflect the current state of the print request queue.

6. Highlight an object or objects from the object list and then choose a task from the "Action" menu. If you are adding to the object list, you need not highlight an object before choosing a task from the "Actions" menu.

   For example, if you are performing user account tasks, you can choose from the tasks listed from the "Action" menu (`add`, `remove`, `modify`, `deactivate`, `reactivate`, and `task customization`).

7. Fill in dialog boxes and then activate the [Apply], [OK], or [Cancel] control button.

   Activating the [Apply] control button performs the task, but does not return you to the object list.

   Activating the [OK] control button performs the task and returns you to the object list.

   Activating the [Cancel] button returns you to the object list without applying changes or performing a task.

8. If SAM requires you to read a message before continuing, read the message and activate the [OK] button to proceed with the task

9. Exit SAM:
   - From the SAM main menu, activate the (Exit SAM) control button.
   - From the object list, choose **Exit** from the "List" menu and then activate the (Exit SAM) control button from the SAM main menu.

   If you have performed tasks that require you to generate a new kernel, SAM will prompt you to create the kernel and reboot the system.

   In the case of complex tasks, SAM displays a step menu. Each task in the step menu is required for the overall complex task. If there is a required order to the step menu, SAM allows you access to the step to be performed first and greys out the remaining tasks until you have completed the prerequisite task. If there is no particular order to the step menu, SAM allows you to access them in any order. When a step is completed, SAM summarizes the result of the tasks next to the step menu item.

Overview of Customizing the Object List

Customizing the object list allows you to:

- select the type of data you want to view. See "Changing the Object List Type".
- alter the format of the data displayed. See "Changing the Format of the Object List".
- conditionally display a subset of the total available data. See "Displaying a Subset of Objects"
- alter the order of the data displayed. See "Changing the Object List Order".

Once you customize the object list, you can preserve this view of the data for the next time you use SAM. If you do not explicitly preserve your changes to the object list, SAM returns to the default format the next time you use SAM.

Changing the Object List Type

To change the objects displayed within a functional area, choose an alternative object type from the "List" menu.

Note Changing the object list type from the "List" menu changes the associated tasks listed in the "Action" menu.

For example, within the "Users and Groups" functional area the "Actions" menu items for the users object list are:

`Add...`

`Remove...`

`Modify`,

`Deactivate..`

`Reactivate...`

`Task Customization`

In comparison, the "Actions" menu items for the groups object list are:

`Add...`

`Remove...`

`Modify`

Changing the Format of the Object List

To change the format of the object list:

1. Enter a functional area by highlighting an area from the list and activating the [Open] control button until an object list is displayed.
2. Choose `Columns...` from the "View" menu.

   For each column (attribute) in the object list, you can define:

   - a numeric column position (1 to the number of columns displayed) using a menu button.
   - justification (left *or* right) using a menu button.
   - width (in characters) by filling in the field.
3. Activate one of the following control buttons:

   [OK] to view the newly defined format and exit the "Columns" window.

   [Apply] to implement the newly defined format, but remain in the "Columns" window.

   [Cancel] to cancel the most recent change and return to the object list.

To preserve this view of the objects for the next time you enter this functional area of SAM:

1. Exit the "Columns" window by activating the [OK] or [Cancel] control button.
2. Choose `Save View as Default` from the "View" menu.

To return to the original listing of the objects:

1. Choose `Columns...` from the "View" menu.
2. Activate the [System Defaults] button and activate the [OK] control button.

Displaying a Subset of Objects

Selectively displaying elements is helpful for viewing the relative subset of total objects for your task. For example, within the "Users and Groups" functional area, you can display only those users with a user identification number below 20 belonging to user group *laboratory*. You define the criteria based on an operator (matches, not, greater than, or less than) and a field value. You can define your subset based on multiple attribute criteria. Multiple attribute criteria use the AND logical operation.

To selectively view objects that met particular criteria:

1. Choose `Filter...` from the "View" menu.
2. Choose one of the following operators for the field attribute(s):

   `matches`

   `not`

   `greater than`

   `less than`

3. For each field you defined an operator for, fill in the associated value to base the conditional operator.
4. Activate one of the following control buttons:

   (OK) to view the newly defined object list subset and exit the "Filter" window.

   (Apply) to implement the newly defined object list subset, but remain in the "Filter" window. The number of filtered items is displayed in the "Filter" window. For example:

   ```
   Current filter: showing 8 of 57 total items
   ```

   (Cancel) to cancel the most recent change and return to the object list.

To preserve this view of the objects for the next time you enter this functional area of SAM:

1. Exit the "Filter" window by activating the [OK] or [Cancel] control button.
2. Choose `Save View as Default` from the "View" menu.

To return to the original listing of the objects:

1. Choose `Filter...` from the "View" menu.
2. Activate the [System Defaults] button and activate the [OK] control button.

Changing the Object List Order

You can change the object list order (ascending or descending) based on the value of one or multiple attributes.

To change the object list order:

1. Choose `Sort...` from the "View" menu.
2. Based on the attribute on which you want to base the order of the object list, choose the priority from the attribute's `Priority` menu button.
3. For each priority defined, choose `Ascending` or `Descending` from the attribute's `Direction` menu button.
4. Activate one of the following control buttons:

   [OK] to view the newly defined object list order and exit the "Sort" window.

   [Apply] to implement the newly defined object list order, but remain in the "Sort" window.

   [Cancel] to cancel the most recent change and return to the object list.

To preserve this view of the objects for the next time you enter this functional area of SAM:

1. Exit the "Sort" window by activating the [OK] or [Cancel] control button.
2. Choose `Save View as Default` from the "View" menu.

To return to the original listing of the objects:

1. Choose `Sort...` from the "View" menu.
2. Activate the [System Defaults] button and activate the [OK] control button.

Using SAM with an X Window System

This section describes the following tasks for using SAM with and X Window System:

- Setting up your system
- Running SAM
- Exiting SAM
- Customizing the SAM interface

If you are using a text terminal, this section of the chapter does *not* apply to your terminal; refer to "Using SAM with a Text Terminal".

Setting Up Your System

There are three system requirements for using SAM with an X Window System:

1. The `SAM` fileset must be part of your HP-UX system. You can use the `update` program to add the SAM fileset to your system, see *Installing and Updating HP-UX*.
2. The `X11-RUN` fileset must be part of your HP-UX system. You can use the `update` program to add the `X11-RUN` fileset to your system, see *Installing and Updating HP-UX*.

Note The *SAM* and *X11-RUN* filesets must also be part of any system intended to be administered locally or remotely by SAM.

---

3. The `DISPLAY` environment variable must be set to reflect the display on which you want SAM to appear.

To view the current environment variable values for all shells, type `env`.

To view the current local shell variable values for all shells, type `set`.

Korn and Bourne Shell. To set the `DISPLAY` environment variable on the local system in the Korn and Bourne shell, type:

```
DISPLAY=hostname:0.0
export DISPLAY
```

hostname is the name returned from typing the `/bin/hostname` command on your local system.

Note In Korn and Bourne shells, subsequent changes to a shell variable are globally reflected after you have exported a variable with the `export` command.

Typically, the DISPLAY environment variable for the local system is set and exported in the `.profile` file. If you are running HP VUE, the DISPLAY environment variable for the local system is set in the `.vueprofile` file.

Note If the DISPLAY environment variable is not set, SAM will attempt to execute the text terminal interface.

C Shell. To set the `DISPLAY` environment variable on the local system in the C shell, type:

```
% setenv DISPLAY hostname:0.0
```

hostname is the name returned from typing the `hostname` command on the local system.

Typically the DISPLAY environment variable for the local system is set in the `.login` file. If you are running HP VUE, the DISPLAY environment variable for the local system is set in the `.vueprofile` file.

Note If the DISPLAY environment variable is not set, SAM will attempt to execute the text terminal interface.

Entering SAM

To start up the System Administration Manager (SAM):

1. Ensure that you have superuser capabilities.
2. Enter SAM; type:

   `/usr/bin/sam`

To perform system administration tasks on a remote system using SAM, refer to "Using SAM for Remote System Administration".

Note SAM is an optionally loadable part of HP-UX. If you have not loaded SAM onto your system, you will not be able to use it. You can use the `update` program to add SAM to your system if you did not originally load it and currently want to use it. For details about how to do this, see the *Installing and Updating HP-UX* manual.

SAM presents you with the main window which looks similar to this:

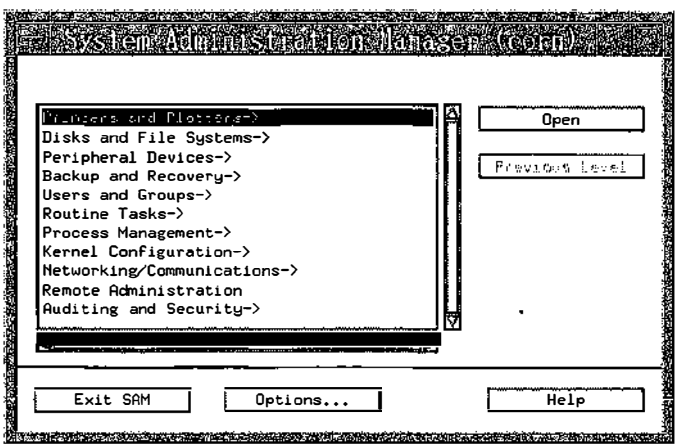


Figure 1-1. SAM Main Window

Exiting SAM

To exit SAM from the SAM main window (see Figure 1-1 in "Entering SAM") or in a functional subarea menu, activate the [Exit SAM] control button.

To exit SAM from a functional area with an object list displayed:

1. Choose `Exit` from the "List" menu.
2. Activate the [Exit SAM] button from the SAM main window.

To exit SAM from a dialog box:

1. Activate the [Cancel] button.
2. Choose `Exit` from the "List" menu.
3. Activate the [Exit SAM] button from the SAM main window.

You can also close the SAM main window with the window manager.

Customizing the SAM Interface

You can customize the following two features of the SAM interface:

- colors
- fonts

SAM colors and fonts are determined by the environment in which SAM is running. SAM requires only foreground and background color. However, if SAM is running under VUE, SAM will use VUE colors (for example, different colors for the menubar, dialog boxes, etc.). The VUE definitions for resources *supersede* the X11 resource definitions in your **$HOME/.X11defaults** file. To customize the resources for your VUE environment refer to the VUE documentation.

The **.X11defaults** file entries for customizing SAM colors are:

```
sam*foreground
sam*background
```

SAM uses two fonts, a proportional, sans-serif "system" font, and a monospace "user" font. The "system" and "user" fonts are implemented for the following primary components of the SAM interface:

"system"
- menubar
- menu items
- most labels and titles

"user"
- user input text
- control buttons
- static message text
- scrollable message text
- list boxes (object list and selection list items)
- list box column titles (object list and selection list column headings)

The entries for customizing SAM fonts are:

```
sam*userFont
sam*systemFont
```

Using SAM with a Text Terminal

A **text terminal** is a combination video display/keyboard. The video display of a text terminal is composed of **characters**—letters, numbers, and symbols. It may be able to use **graphics characters** to draw lines and other elements of simple images. Such a terminal is not capable of rendering the detailed **bit-mapped graphics** that are used by the X Window System.

SAM has a special interface for use on text terminals. Instead of using a mouse to navigate through the SAM screens, you must use particular keys (or combinations of keys) to move from one part of a screen to another and to move among screens.

The purpose of this section is to help you understand how to control SAM from the text-terminal interface.

Entering SAM

Note SAM is an optionally loadable part of HP-UX. If you have not loaded SAM onto your system, you will obviously not be able to use it. You can use the `update` program to add SAM to your system if you did not originally load it and currently want to use it. For details about how to do this, see the *Installing and Updating HP-UX* manual.

To start SAM, type:

```
/usr/bin/sam
```

In a few moments, SAM's main window (see Figure 1-2) will appear.

To perform system administration tasks on a remote system using SAM, see "Using SAM for Remote System Administration".

Figure 1-2 shows the appearance of the main window in SAM.

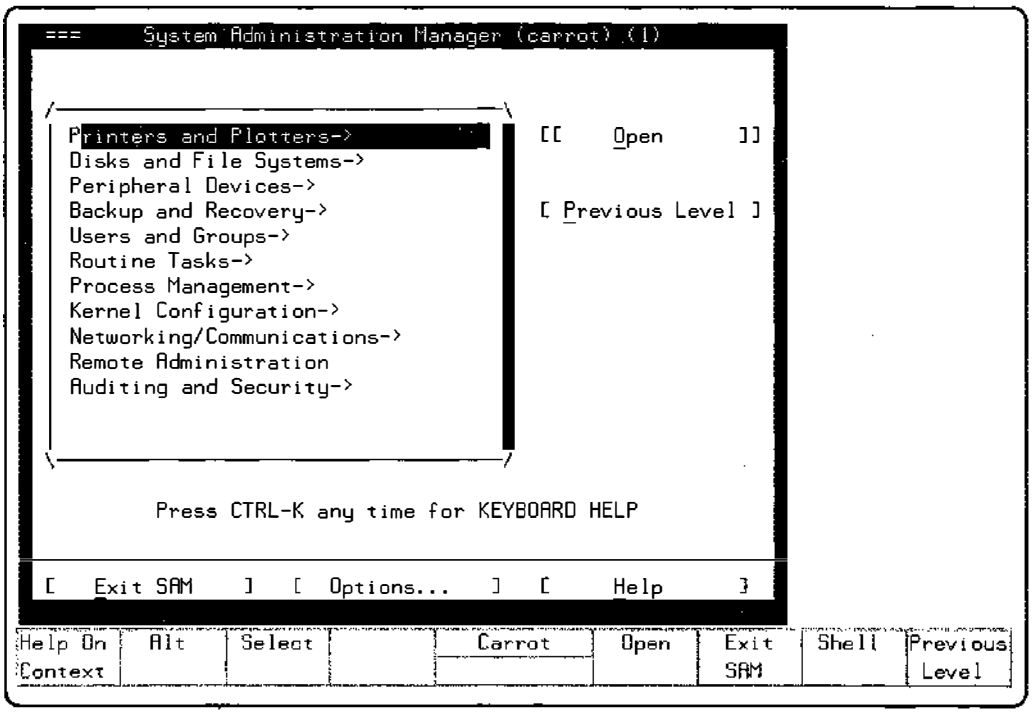


Figure 1-2. The Main Window in SAM's Text-terminal Interface

Note that there is a list of items within a box inside the window. `Printers and Plotters->` is **highlighted** (displayed in inverse video). A highlighted item can be thought of as having been designated for attention.

To choose another entry in this list, press the ▲ or ▼ arrow keys until the highlight is on your choice. For example, if you want to work with peripherals, move the cursor to the position shown in Figure 1-3:

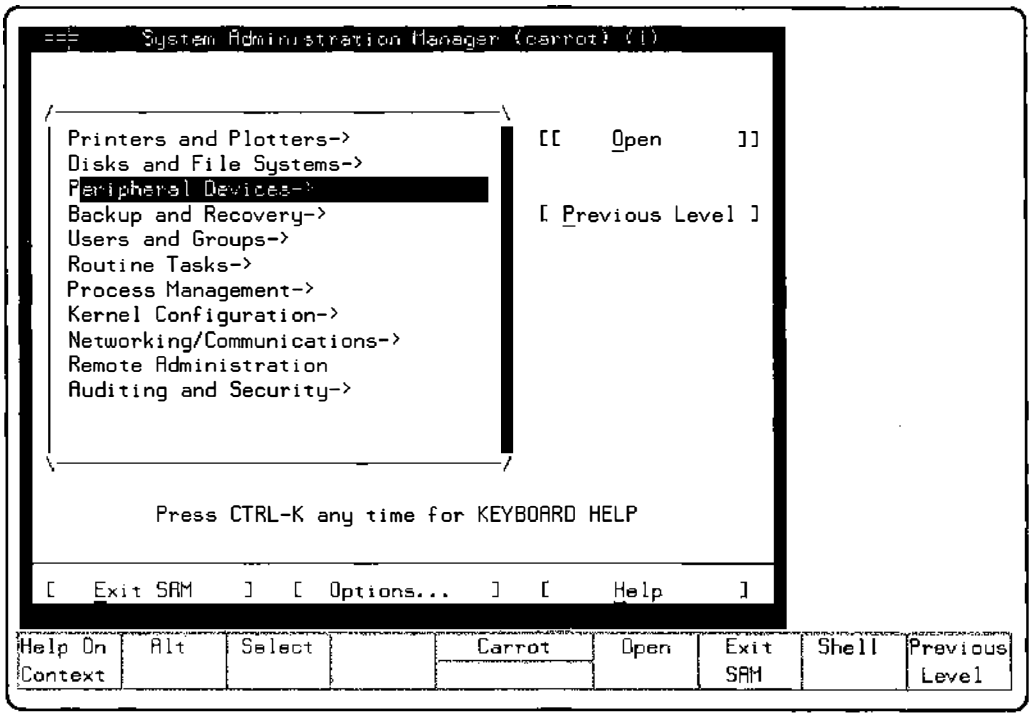


Figure 1-3. Highlighting a New List-item Choice

The highlight is now on the `Peripheral Devices->` list item.

Using Control Buttons

In the window are **control buttons** with these labels:

```
[[ Open      ]]

[ Exit       ]

[ Options... ]

[ Help       ]
```

Activate these buttons to make SAM carry out different actions.

Use the (Tab) key to return to "cycle through" all of the control buttons. To cycle through the control buttons in reverse order, hold down the (Shift) key while you press (Tab). Eventually you will return to the list of functional areas.

To activate a control button, do one of the following:

- **Highlight** the button by pressing the (Tab) key one or more times. When a button is highlighted, that indicates that is ready for activation.

  Activate the highlighted control button by pressing the (Return) on the keyboard.

- **Activate** a control button immediately by pressing a *mnemonic* key. For example, notice in Figure 1-3 that the letter O on the (Open) control button is underlined. Press the (O) key on the keyboard, and the (Open) control button is highlighted and activated immediately.

Figure 1-4 shows the appearance of the SAM main window when the Help control button is highlighted.

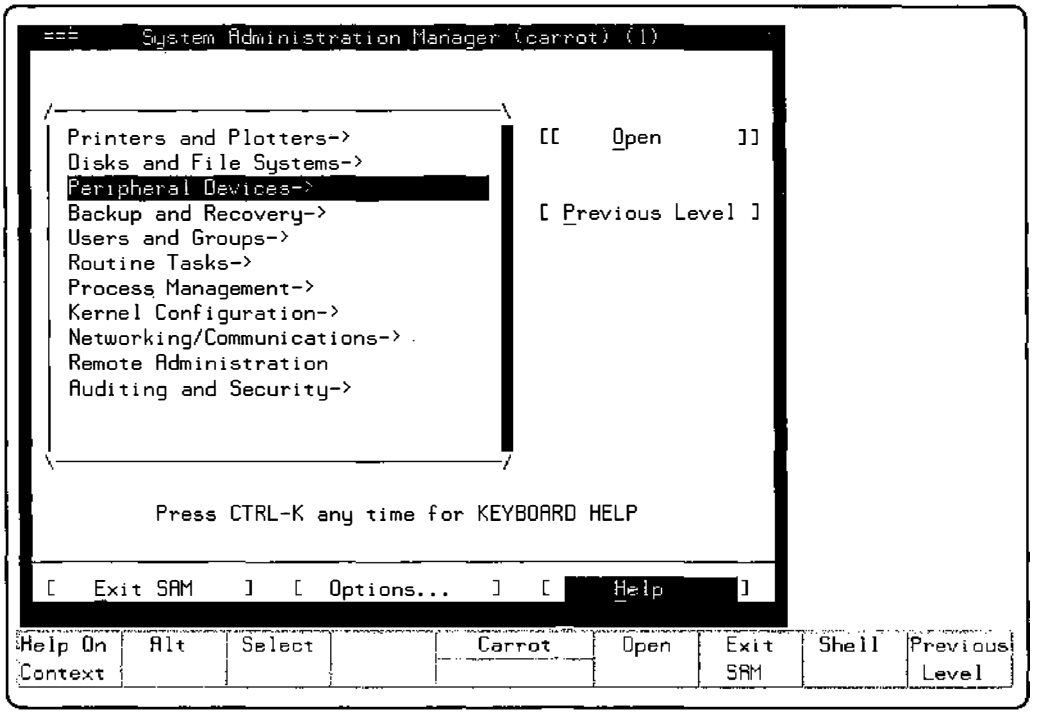


Figure 1-4. Highlighting a Control Button

Using Softkeys

Hewlett-Packard terminals (and some others) display eight **softkey labels** below the window area. The keyboard keys to which these labels correspond are in a row across the top of the terminal's keyboard, and they are usually labeled f1 through f8.

The labels may change when a new window appears. Table 1-3 lists the labels which you will see most often.

Note VT-100 (and other ANSI-standard) terminals will not display the function-key labels.

The last column of Table 1-3 lists the keys or key combinations that give the equivalent result for these terminals.

Table 1-3. Function Keys for SAM's Text-Terminal Interface

Label	Meaning	Key(s)[1]	
		HP or Wyse	VT-100 or ANSI
Help on Context	Get help in understanding an element displayed on the screen	f1	Help or PF 2
Alt	Type alternate character	f2	PF 1
Select	Highlight an item or open a menu	f3 or Spacebar	Spacebar
Menubar on/off	Move cursor to menubar	f4	PF 1, Spacebar or PF 1, =
Open	Open the highlighted functional area or subarea	f5	Return
Previous Level	Return to the previous level of SAM	f8	(none)
Shell	"Escape" (temporarily) to a shell	f7	(none)
Exit	Exit the current window	f6	(none)
Exit SAM	Exit SAM entirely	f8	(none)

1 Keys are specified by the symbols which appear on their keycaps. Presence of a comma (",") between two keycaps means that the keys should be pressed in sequence.

Exiting SAM

To exit SAM:

- Activate any control button labeled (Exit SAM), or
- Press the softkey labeled `Exit SAM`.

The main window (and any other windows that may be open) will close, and the shell prompt will return.

Entering a Functional Area

From SAM's main window, highlight `Printers and Plotters->`.

Note If a functional area item ends in "->", you will be routed to a subarea list window. From there you will choose a functional subarea.

If you choose a functional area list item that does *not* end in "->", you will be presented with a functional area window immediately.

Press (Return). A subarea list window (like Figure 1-5) appears on your screen.

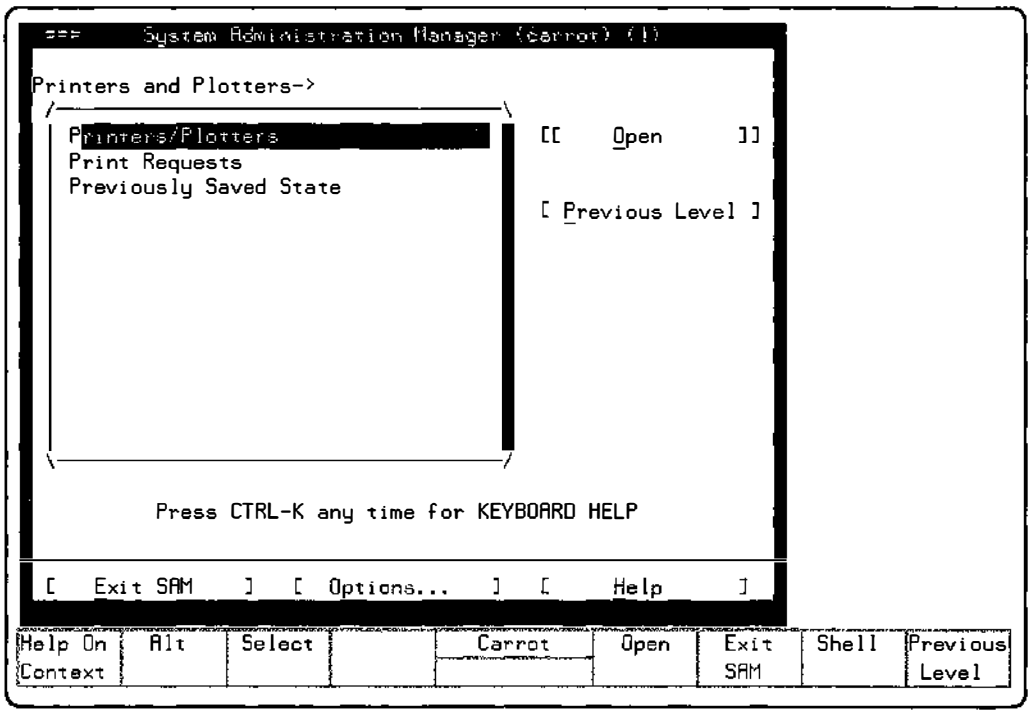


Figure 1-5. A Subarea List Window

Highlight the `Printers and Plotters` subarea. Press (Return). The "Printer/Plotter Manager" functional area appears. See Figure 1-6.

Selecting an Object. In the functional area illustrated by Figure 1-6, the cursor is on the first **object** in the **object list**.

Note Occasionally you may find that an object list is empty. You can use items from the "Actions" menu to add objects to the list in any functional area. See Figure 1-8 for an example of the items in an "Actions" menu.

```
===                    Printer/Plotter Manager (carrot) (1)
List View Options Actions                                          Help
Printers/Plotters                                        0 of 4 selected
Print Spooler: RUNNING              Default Printer: sam

                              Fence
  Name      Status            Priority   Type     Location
/-------------------------------------------------------------------\
  joe       enabled, idle     2          local    /dev/lp_joe
  lj2000    enabled, idle     0          remote   lj2000 on hpfcrz
  sadie     enabled, idle     0          remote   sadie on hpaland
  sam       enabled, idle     0          remote   sam on hpfcrp
\-------------------------------------------------------------------/

Help On | Alt | Select | Menubar | Carrot | | | Shell | Exit
Context |     |        |  on/off |        |
```

Figure 1-6. A Functional Area

Move the cursor to any item you wish SAM to act upon by using the ▲ and ▼ keys. Then highlight the item by pressing Spacebar. Figure 1-7 shows one highlighted item in an object list.

```
===                    Printer/Plotter Manager (carrot) (1)
List View Options Actions                                          Help
Printers/Plotters                                        1 of 4 selected
Print Spooler: RUNNING                 Default Printer: sam

                                 Fence
  Name      Status               Priority   Type      Location

  joe       enabled, idle        2          local     /dev/lp_joe
  lj2000    enabled, idle        0          remote    lj2000 on hpfcrz
  sadie     enabled, idle        0          remote    sadie on hpaland
  sam       enabled, idle        0          remote    sam on hpfcrp

Help On  | Alt | Select | Menubar | Carrot | | | Shell | Exit
Context  |     |        | on/off  |        | | |       |
```

Figure 1-7. Selecting an Object

SAM can now work with the highlighted object. You can highlight more than one object at a time. For more information on selecting objects, see Table 1-4.

Opening a Menu. There is a **menubar** near the upper-left corner of the screen. It contains the titles "List", "View", "Options", and "Actions".

To move the cursor to the menubar:

- Press (f4) on HP or Wyse terminals
- Press (PF 1), then (Spacebar) on VT100 or ANSI terminals.

Each of the titles on the menubar has one or more **menu items** associated with it. To see the menu items associated with a particular title, use the (◀) and (▶) keys to highlight the title whose menu you wish to see, then press (Spacebar). See Figure 1-8 to see what an "Actions" menu looks like.

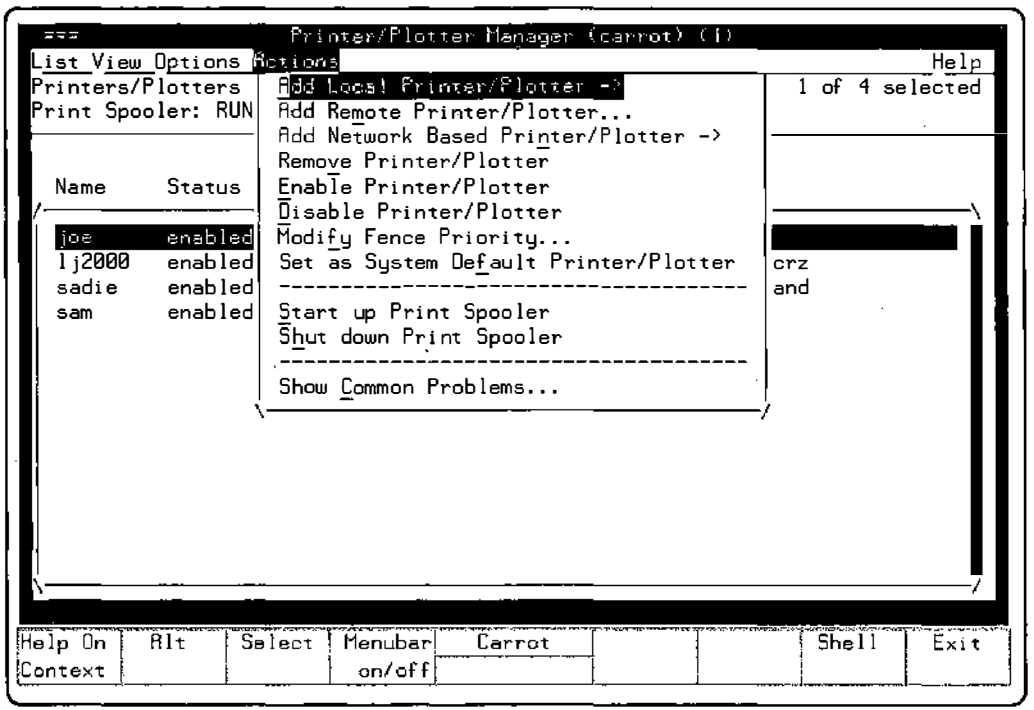


Figure 1-8. A List of Menu Items

Choosing a Menu Item. You can **choose** a menu item in either of two ways:

- **Highlight** the menu item of your choice by using the [▲] and [▼] keys. Notice that if you press either of the arrow keys many times, the highlight "cycles through" the menu over and over again.

  Choose the highlighted menu item by pressing the [Return] on the keyboard.

- If the menu item contains an underlined character, you can **choose** it immediately by pressing a *mnemonic* key. For example, notice in Figure 1-8 that the letter "A" in the `Add Local Printer/Plotter` menu item is underlined. Press the [E] key on the keyboard, and the `Add Local Printer/Plotter` menu item is highlighted and activated immediately.

Note You may not be able to highlight and choose some of the menu items. For example, an "Actions" menu item for removing a device cannot be chosen unless one of the objects in the object list is highlighted. These items may be presented at low intensity or in an alternate color, or they may be filled in with tildes ("~").

Using Buttons and Checkboxes

There are several features in SAM that have a special appearance and which exhibit special behavior.

Menu Buttons. Some screen buttons present a range of settings from which you must choose. These **menu buttons** differ in appearance from ordinary buttons by the presence of -> after the button label. Menu buttons look like this:

[*label* ->]

To "open " a menu button, highlight it and press [Spacebar]. The menu button expands into a small menu.

To choose an item from the menu, press the [▲] and [▼] keys to highlight your choice, then press [Spacebar]. The choice is displayed on the menu button.

Radio Buttons. Within the "List" menu, you may find two or more views of a functional area. Access to these alternate views is controlled by a **radio button** which may be turned "on" or "off". Radio buttons differ in appearance from ordinary buttons by the presence of a diamond shape to the left of the button label. A radio button looks like this when it is turned "on":

<*> *view_name*

A radio button looks like this when it is turned "off":

< > *view_name*

To turn a radio button on or off, , highlight the radio button with the [▲] or [▼] arrow keys and press [Spacebar]. The screen will change to another view of the functional area.

Radio buttons are *mutually exclusive*: within a "List" menu, only one button at a time may be turned on.

Checkboxes. A **checkbox** is an object which can be turned "on" or "off." Checkboxes differ in appearance from ordinary buttons by the presence of a square shape to the left of the button label. A checkbox looks like this when it is turned "on":

[x] *label*

A checkbox looks like this when it is turned "off":

[ ] *label*

To turn a checkbox "on" or "off", use the [Tab] key to move the highlight over the checkbox, then press [Spacebar]. If it was "on," the **x** in the checkbox disappears. If it was "off," an **x** appears in the checkbox.

Checkboxes are *not* mutually exclusive. You may turn "on" or "off" as many as you need.

Navigating with Keys and Key Combinations

You must use particular keys and combinations of keys to navigate and perform particular tasks in SAM. Table 1-4 lists the special meanings of the keys you must use to navigate within the windows in SAM's text-terminal interface.

Table 1-4. Meanings of Selected Keys

Action	Key(s)[1]	
	HP or Wyse	VT-100 or ANSI
Move the cursor one space to the right	[▶]	[▶]
Move the cursor one space to the left	[◀]	[◀]
Move the cursor up one line	[▲]	[▲]
Move the cursor down one line	[▼]	[▼]
Move the cursor to the next field	[Tab]	[Tab]
Move the cursor to the menubar	[f4]	[PF 3]
Scroll a list up one page	[Shift]-[▲]	(none)
Scroll a list down one page	[Shift]-[▼]	(none)
Scroll a list up one line	[f2], [▲]	[PF1], [▲]
Scroll a list down one line	[f2], [▼]	[PF1], [▼]
Scroll a list left one page	[Prev]	(none)
Scroll a list right one page	[Next]	(none)
Scroll a list left one character	[f2], [◀]	(none)
Scroll a list right one character	[f2], [▶]	(none)

1 Keys are specified by the symbols which appear on their keycaps. Presence of a comma (",") between two keycaps means that the keys should be pressed in sequence; presence of a hyphen ("-") between two keycaps indicates that the keys should be pressed simultaneously.

Table 1-4. Meanings of Selected Keys (continued)

Action	Key(s)[1]	
	HP or Wyse	VT-100 or ANSI
Highlight one item	f3 or Spacebar	Spacebar
Highlight all items in a list	f2, /	PF 1, /
Highlight a range of items	1. f2, f3 on first item 2. Move cursor 3. f2, f3 on last item	1. PF 1, . on first item 2. Move cursor 3. PF 1, . on last item or 1. Find on first item 2. Move cursor 3. Find on last item
Dehighlight one item	f3 or Spacebar	Spacebar
Dehighlight all items in a list	f2, \	PF 1, \
Open a menu on the menubar by using a **mnemonic** (first letter of menu)	f2, *mnemonic* key	PF 1, *mnemonic* key
Close a menu	f4 or Spacebar	PF 3 or Spacebar

Changing Windows

As you interact with SAM's text-terminal interface, you may have more than one window open at the same time. The interface allows you to perform various actions with windows: open, close, move, resize, and **iconify** (reduce the window to a small symbol on the screen).

These actions may be chosen from the **window control menu**. To open the window control menu:

- on a Hewlett-Packard or Wyse terminal, press f2, = or f2, Spacebar.
- on a VT-100/ANSI terminal, press PF 1, Spacebar or PF 1, =.

Moving a Window. Open the window control menu, highlight `move` and press [Spacebar]. Notice that the four corners of the window have begun to "blink." Press any of the [▲], [▼], [◀], and [▶] to move the window to the desired location, then press [Spacebar].

Resizing a Window. Open the window control menu, highlight `resize` and press [Spacebar]. Notice that the corners of the window have begun to "blink." (If one of the corners is not blinking, it indicates that the window cannot be resized in that direction.) Press any of the [▲], [▼], [◀], and [▶] to move the window corners to the locations that indicate the new size you require, then press [Spacebar].

Iconifying a Window. Open the window control menu, highlight `iconify` and press [Spacebar]. The window disappears, but you can see that it is still available by opening the **window list menu** (see "Opening a Window"). If there is an uniconified window available, the cursor will move to that window.

Maximizing a Window. Open the window control menu, highlight `maximize` and press [Spacebar]. The window is resized to fill the entire screen.

Opening a Window. To "raise" (open up) an iconified window:

- on a Hewlett-Packard or Wyse terminal, press [f2], [0].
- on a VT-100/ANSI terminal, press [PF 1], [0].

This opens the **window list menu**. This menu lists the titles of all visible or iconified windows. To choose the title of the window you wish to open, use the [▲] and [▼] cursor keys to highlight the desired window title, then press [Spacebar].

Closing a Window. Open the window control menu, highlight `close`, then press [Spacebar].

The window disappears, but it is added to the window list menu, and it may be opened as described in "Opening a Window".

Using SAM for Remote System Administration

SAM allows single-point administration of remote systems by executing SAM on the remote system while displaying on the local machine. Single-point administration refers to administering multiple remote systems from one location. There are no hardware related restrictions. You can use any HP 9000 Series 300/400/700/800 to remotely administer any other HP 9000 Series 300/400/700/800.

Adding and Removing Remote Systems

To add remote systems using SAM:

1. Run SAM; type:

   `/usr/bin/sam`

2. Highlight `Remote Administration` and activate the [Open] button.

3. Choose `Add System ...` from the "Actions" menu.

4. Fill in the dialog box information and activate the [Ok] button.

The local machine's `/etc/hosts` file must have an entry for the remote system.

To remove remote systems using SAM:

1. Run SAM; type:

   `/user/bin/sam`

2. Highlight `Remote Administration` and activate the (Open) button.
3. Highlight the system(s) in the object list to be removed.
4. Choose `Remove System(s) ...` from the "Actions" menu.

Additional Task Information. Choosing an item from the "Help" menu gives you information about:

- the current functional area.
- keyboard navigation within SAM.
- using the SAM help system.
- displaying the version of SAM you are currently running.

Activating the (Help) button from a dialog or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

Pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Configuring the Remote System

The remote system must meet the following requirements:

- The SAM fileset is installed.
- The local machine name is in the remote system's `/etc/rhosts` file.

Logging SAM's Actions

SAM provides a log file mechanism for tracking the actions SAM performed on your system. You can activate this tracking mechanism, define the log file name, level of detail, and log file size by activating the [Options...] control button from the SAM main window.

Turning On SAM Logging

To activate the logging mechanism:

1. Activate the [Options...] control button. The "Options" window appears.
2. Choose the level of log file detail (`Summary`, `Detail`, `Verbose`) from the `Logging level` menu button.
3. Fill in the log file name.
4. Choose the log file length (in bytes) (`20000`, `50000`, `500000`, or `2000000`) from the `Maximum logfile size` menu button.
5. Turn on the `Save settings` check box if you want to preserve these values for the next time you enter SAM. If you do not turn on the `Save settings` check box, the default values are reset the next time you enter SAM.
6. Activate the [OK] control button.

Summary Level Log. The following listing illustrates the summary level log file entries for adding a new user to the system:

```
S:****** Adding user(s):
S:****** Successfully added user "newuser".
```

Detail Level Log. The following listing illustrates the detail level log file entries for adding a new user to the system:

```
S:****** Adding user(s):
D:          Executing (pid = 1152):
              addusr -u newuserA -g users -d /users/newuserA -i ",,," -s /bin/sh
            -p "" -v 203
D:          addusr:  Adding user "newuserA" to group "users".
D:          addusr:  Modifying "/etc/passwd":  login_name="newuserA" password=""
            aging="" uid="203" gid="20" gecos="" shell="/bin/sh"
            home_directory="/users/newuserA" audit_id="-1" audit_flag="0".
D:          addusr:  Creating home directory "/users/newuserA" for user
            "newuserA".
D:          addusr:  Calling "chown(2)":  chown(/users/newuserA,203,20)
D:          addusr:  Copying default files to directory "/users/newuserA".
D:          Child (pid = 1152) terminated with status 0x0
S:****** Successfully added user "newuserA".
```

Verbose Level Log. The following listing illustrates the verbose level log file entries for adding a new user to the system:

```
V:          Processing internal message UI_GENERIC_EXEC (0)
V:          Executing (pid = 1085):
              /usr/sam/bin/users_init
V:          Child (pid = 1085) terminated with status 0x0
V:          Processing internal message UG_GET_DEFAULT_UID (0)
V:          Processing internal message UG_GET_DEFAULT_UID (0)
V:          Processing internal message UG_VALIDATE_NEWUSER (0)
S:****** Adding user(s):
V:          Processing internal message UG_ADD_USER (0)
D:          Executing (pid = 1166):
              addusr -u newuserB -g users -d /users/newuserB -i ",,," -s /bin/sh
            -p "" -v 204
D:          addusr:  Adding user "newuserB" to group "users".
D:          addusr:  Modifying "/etc/passwd":  login_name="newuserB" password=""
            aging="" uid="204" gid="20" gecos="" shell="/bin/sh"
            home_directory="/users/newuserB" audit_id="-1" audit_flag="0".
D:          addusr:  Creating home directory "/users/newuserB" for user
            "newuserB".
D:          addusr:  Calling "chown(2)":  chown(/users/newuserB,204,20)
D:          addusr:  Copying default files to directory "/users/newuserB".
D:          Child (pid = 1166) terminated with status 0x0
S:****** Successfully added user "newuserB".

V:          Processing internal message UG_GET_DEFAULT_UID (0)
V:          Processing internal message UI_GENERIC_EXEC (0)
V:          Executing (pid = 1178):
              /usr/sam/bin/users_init
V:          Child (pid = 1178) terminated with status 0x0
V:          Processing internal message UG_GET_DEFAULT_UID (0)
```

2

Constructing an HP-UX System

This chapter explains how to build a new HP-UX system. It outlines what you need to do to create a functioning system, and tells you where you can find more detailed information if you need it. You should start on the first task, which is to assess your users' needs, before you assemble the components of the system.

Assessing Users' Needs

Most users think in terms of the job they have to do, not "I need FORTRAN, Graphics, a plotter, and 500,000 bytes of data storage." The sample survey that follows will help you to find out what your users need and to translate their needs into the hardware and software that comprise a functioning HP-UX system.

You may need to change the form to fit your particular situation.

User Survey

Name _ _ _ _ _ _ _ _ _ _ _ _ _ _ Location _ _ _ _ _ _ _ _

Phone _ _ _ _ _ _ _ _ _ _ _ _ _ _

Location where you will use the system: _ _ _ _ _ _ _ _ _ _ _ _

Which of the following best describes your job? (Please check all that apply):

_ Engineer/ Manager	Run existing application programs; enter data, create models.
_ Technical Data Entry Operator	Run existing application programs; enter data or automatically read data from instrumentation.
_ Secretary - Word Processing Operator	Run existing application programs; enter data/text.
_ Applications Programmer	Develop application programs.
_ Systems Programmer Support Person	Develop programs for improving computer system performance or for use by other programmers.

Describe your application (the program(s) you use or create):

_ _

_ _

What programming language(s) will you use?_ _ _ _ _ _ _ _ _ _ _ _

_ _

What applications software (such as graphics) will you use? _ _ _

_ _

_ _

What computer hardware or peripherals do you need?

_ Inkjet printer _ Plotter

_ Impact printer

_ Graphics terminal

_ Laser printer

_ Removable mass storage devices

_ Other _ _ _ _ _ _ _ _ _ _ _ _ _

_ _ _ _ _ _ _ _ _ _ _ _ _

Are there other users with whom you want to share programs or data?

If so, list these users _

_ _

Will you generate or use large amounts of data? _ _ _ _ _ _ _ _

If so, how much must be "on-line" (accessible at all times)?

_ _

_ _

What long-term data storage does your application require?_ _ _ _

_ _

How many programs/processes will you be running at one time?_ _ _

_ _

Which programs are interactive, and which will you run in
background mode? _

_ _

_ _

Can any programs be run overnight?_ _

_ _

Installing HP-UX

As the system administrator, you might be responsible for installing HP-UX. (On some systems, the operating system may be installed for you by a Hewlett-Packard Customer Engineer; and some systems come from the factory with the operating system installed.)

The manual *Installing and Updating HP-UX* provides instructions for installing HP-UX.

Updating HP-UX

As the system administrator, you will normally be responsible for updating your system from one version of HP-UX to another (for example from Release 9.0 to a subsequent release), although on larger systems you may decide to have your Hewlett-Packard Customer Engineer or Software Engineer do it for you.

Use the *update*(1M) utility to update the operating system to a new release. Full instructions are in the manual *Installing and Updating HP-UX*.

Note If you are running a pre-Release 9.0 system, you can use `update` to add Release 9.0 software, but if you want the root file system on Logical Volumes (highly recommended) you must migrate your system to Release 9.0. Refer to chapter 3, "Migrating the Root Disk to Logical Volumes," in the *Installing and Updating HP-UX* manual, part number B3108-90006.

Installing and Updating Optional Software

You can use the *update*(1M) utility to add optional HP software (such as networking) to your system, or to update such software to a new release. `Update` is also the installation and update tool for some third-party software.

When adding and updating software, be sure to follow the supplier's directions. If the directions tell you to use `update`, use the manual *Installing and Updating HP-UX* for more information.

Adding, Removing, and Moving Peripherals

By "peripherals", we mean hardware that you can add to your system, including terminals, printers, disk drives, tape drives, other storage and communication devices, and the interface cards they attach to.

On some systems, you might choose to have your Hewlett-Packard Customer Engineer (CE) or Software Engineer (SE) install a given peripheral, but often you will do it yourself.

To install a peripheral, you need the following manuals:

- The installation manual that came with the peripheral.
- The *Installing Peripherals* manual. Look up the particular peripheral you are installing in *Installing Peripherals*. This material contains the software driver and the device special file information for communication with the peripheral.

Depending on the peripheral, and the system you are installing it on, you might need to rebuild your system's **kernel** to make it aware of the new peripheral—for example, you might need to add a new **driver**.

The **kernel** is the piece of software that controls the computer—it is often referred to more loosely as the operating system. You can reconfigure the kernel by means of HP-UX commands or by the menu-driven System Administration Manager (SAM) program.

A **driver** is a program that enables the kernel to communicate with a given type of peripheral.

If you do need to rebuild the kernel for the device you are adding, the *Installing Peripherals* will show you which device drivers need to be in your kernel and the device special files needed to communicate with your peripheral. Refer to the "Reconfiguring the Kernel Using HP-UX Commands" or "Reconfiguring the Kernel to Add/Remove Device Drivers Using SAM" sections later in this chapter for a general procedure for rebuilding an HP-UX kernel.

Refer to Chapter 5, "Managing Peripherals" of this manual for moving and removing a peripheral.

After Installing HP-UX

After you have installed HP-UX and added peripherals such as disk drives, tape drives, printers and terminals, you still have some work to do before your users can sit down at their terminals and workstations and do their jobs productively.

The sections that follow outline the most important tasks, and where necessary point you to sources of further information.

Setting the System Clock

Only the **superuser** can change the system clock. (The superuser is the person who can log in as `root`. This should be you, and one other person you nominate as a backup.)

Before you can set the clock, you must set the time zone environment variable (`TZ`).

Setting the Time Zone (`TZ`)

You can set the time zone with a declaration in any one of three files, `/etc/csh.login`, `/etc/rc`, or `/etc/profile`. (An application program can also set `TZ` by means of the `tzset` library routine.)

The `TZ` declaration takes the following forms in the files:

`/etc/rc`:

```
TZ=xxxhyyy
```

`/etc/profile`:

```
TZ=xxxhyyy
export TZ
```

`/etc/csh.login`:

```
setenv TZ xxxhyyy
```

where:

xxx is an alphabetic abbreviation of the standard time zone, usually three letters.

For example, `MST` for Mountain Standard Time.

h is the difference between standard local time and Greenwich Mean Time (GMT), in hours. Measured to the West, `h` is a positive number. Measured East, `h` is a negative number. This field can include a designation for minutes.

Some examples are:

3:30 for Newfoundland

7 for Mountain Standard Time

-9:30 for South Australia

yyy is an alphabetic abbreviation of the daylight time zone for your area, usually three letters (for example, `MDT` for Mountain Daylight Time).

Do not use this field if Daylight Savings Time is not observed in your geographic area.

The following U.S. examples should give you the idea:

- In the Eastern time zone, use `TZ=EST5EDT`
- In the Central time zone, use `TZ=CST6CDT`
- In Arizona, where Daylight Savings Time is not observed, use `TZ=MST7`

For more information on setting the time zone environment variable, refer to `TZ` under the *environ*(5) entry in the *HP-UX Reference Manual.*

Note `CST6CDT` has two meanings: "Central Standard Time" and "Canadian Standard Time."

This matters because the United States changes to daylight saving time on the first Sunday in April and Canada makes the same transition on the last Sunday in April. Canadian systems must use `CST6CDT#Canada` to distinguish between the two.

Setting the Time and Date

After you have set the time zone variable, terminate the `cron` process (if it is running) and enter the `date` command. Do this as follows:

1. Log in as superuser.
2. Kill the `cron` process.

   To terminate `cron`, first find the `cron` process information by entering:

   ```
   ps -ef | grep cron
   ```

   This gives the process ID (PID) for `cron`.

   Now terminate `cron` by entering:

   `kill` *pid*

   where *pid* is the Process ID associated with `cron` (for example, `16442`).
3. Set the time and date by entering:

   ```
   date mmddhhmm{yy}
   ```

   where:

`mm`	is a two-digit integer representing the month. For example, `03` represents March.
`dd`	is a two-digit integer representing the day of the month. For example, `02` represents the second day of the month.
`hh`	is a two-digit integer specifying the hour on a twenty-four hour clock. For example, `03` specifies 3:00 am and `14` specifies 2:00 pm.
`mm`	is a two-digit integer specifying the number of minutes past the hour. For example, `04` specifies four minutes past the hour.
`{yy}`	is an optional two-digit integer specifying the last two digits of the year. For example, `91` specifies 1991.

When `date` executes, it shows the time and date on your screen.

4. Restart `cron` if you terminated it in step 1.

   Enter:

   ```
   /etc/cron
   ```

Problems You Can Cause by Changing the System Clock

make: The `make` program is sensitive to a file's time and date information and to the current value of the system clock.

Setting the clock forward will not affect `make`, but setting the clock backward by even a small amount may cause `make` to behave strangely.

Backups: Incremental backups depend heavily on the date being correct because they rely on a dated file.

cron: Altering the system clock can cause unexpected results for jobs scheduled by `cron`.

- If you set the time back, `cron` does not run any jobs until the clock catches up to the point from which it was set back.

  For example, if you set the clock back from 8:00 to 7:30, `cron` will not run any jobs until the clock again reaches 8:00.

- If you set the clock ahead, `cron` attempts to catch up by immediately starting all jobs scheduled to run between the old time and the new.

  For example, if you set the clock ahead from 9:00 to 10:00, `cron` immediately starts all jobs scheduled to run between 9:00 and 10:00.

Setting Up File Systems

When you install HP-UX, the `install` program that builds your system comes with the file systems HP-UX needs, but you will need to create additional file systems for your users. Chapter 6, "Managing the File System", in this manual, explains how to do this.

Setting Up On-Line HP-UX Reference (manpages)

Every HP-UX command and system file is documented in the *HP-UX Reference Manual*.

This manual, often referred to simply as the "manpages", comes in two forms: as a book, shipped with the core documentation shipped with your system, and as a set of files that allow you and your users to look up the entries on-line and to print them out individually. The on-line version requires some set-up. You have three methods to choose from:

1. Create a formatted version of all the manpages.

   The advantage of doing this is that users will get quick response when they call up a manpage on-line. The disadvantage is that the formatted versions take up a considerable amount of disk space (about as much again as the `nroff` originals from which they are created), which you may not have. However, once the pages have been formatted, you can recover disk space by getting rid of the `nroff` originals.

   This is a good method if you have enough disk space to hold both versions of the manpages for as long as it takes to finish formatting them.

   If you decide to do this, enter:

   `/etc/catman`

   The process of formatting all the manpages can take as long as five or six hours, so you might want to run it at a lower priority, in the background, or at night.

2. Format only certain sections of the manpages.

   This could give you the advantage of quicker access to heavily used sections without incurring the cost in disk space of formatting all sections.

   If you decide to do this, enter:

   `/etc/catman` *sections*

   (where *sections* is one or more logical sections in the *HP-UX Reference Manual* such as 1).

3. Do not execute `/etc/catman` at all.

   Use this method if you can spare some disk space but do not want to use any more than is necessary.

   If you don't run `catman`, HP-UX formats each manpage the first time a user calls it up via the `man` command. The formatted version is added to the appropriate `cat` directory and used in subsequent accesses.

   If you decide to use this method, you must make directories to hold the formatted manpages. The following script creates these directories:

```
cd /usr/man
for num in 1 1m 2 3 4 5 7 8 9
do
 mkdir cat$num
done
```

When all the manpages have been formatted, you can remove the `nroff` source files.

Setting Up Networking

If your computer is to be part of a **network** (that is, if it will be connected to other computers via Local Area Network (LAN), Wide Area Network (WAN), or some other kind of link), you now need to establish the connection. A network link consists of both hardware (for example, a LAN card) and software (for example, the LAN/9000 and ARPA Services packages).

What you need to do at this point depends on the type of network you're using, and whether you're connecting the computer to an existing network or setting up a new one.

Hewlett-Packard documentation you may need includes:

- *Networking Overview*
- *Installing and Administering LAN/9000*
- *Installing and Administering FDDI/9000 Software*
- *Installing and Administering Token Ring/9000 Software*
- *Installing and Administering NFS Services*
- *Installing and Administering Network Services*
- *Installing and Administering ARPA Services*

If you have not administered a network before, start with *Networking Overview*.

Adding Users and Groups

It is now time for you to add **users** and **groups** to the system. These are the names of the data structures by which an HP-UX system recognizes a given person or class of people who use it.

There are SAM screens for setting up users and groups, or you can do it by means of HP-UX commands. Full directions are in Chapter 4, "Controlling Access to the System".

Setting Up Electronic Mail

Electronic mail (or "email" as it is often called) can be run by any of these three utilities: `elm`, `mailx`, or `mail`. Any user can use any one of these.

- If your users will be exchanging messages only with each other, and will not need to send mail to users on other systems in a network, then you need not do any set-up.

  The mailer will do the initialization needed for each user when the user first invokes the mailer. However, you may want to supply each `mailx` and `elm` user with a customization file, setting up useful defaults. Depending on the mailer, the customization file should be:

  for `mail`: (none)

  for `mailx`: `$HOME/.mailrc` (That is, a file named `.mailrc` in the user's home directory.)

  (In addition, `mailx` uses a system-wide defaults file `/usr/lib/mailx/mailx.rc`.)

  for `elm`: `$HOME/.elm/elmrc`

- If your users will be sending and receiving mail over a network, you need to set up routing either through UUCP or ARPA Services.
  - To configure UUCP, follow directions in the manual *Remote Access*.
  - To configure ARPA Services, follow directions in *Installing and Administering ARPA Services*.
    - You will also need to install the ARPA Services `sendmail` utility. Chapter 6 of *Installing and Administering ARPA Services* contains the directions you'll need.

Setting Up the Line-Printer Spooler

You share printers among users via the line-printer spooler, which intercepts print requests, organizes them into a queue, and feeds them to the printer one by one. A printer that has been configured into the line-printer spooler is referred to as a **spooled** printer. Any printer Hewlett-Packard supports can be spooled.

If your system will have more than one user at any one time, you should spool the printer(s); if you don't, any listing sent to the printer while another listing is printing will be interleaved with it, garbling both listings.

Even if this will be a single-user system, you may still want to spool the printer, since spooling allows you to batch up print requests so you don't have to wait for one to complete before you can send the next.

If your system is part of a network, the line-printer spooler also lets you send print requests to, or receive them from, other computers in the network, allowing you to make the most efficient use of your printers.

Setting up the line-printer spooler is one of the more complicated tasks you need to do at this stage. Chapter 10, "Managing Printers and Printer Output" contains full explanations and directions. If you have not administered an HP-UX system before, or have not set up a spooled printer before, read the chapter carefully before you attempt the task.

Setting Up news

`news` is a utility that allows you to post messages for users to read (see also "Posting a Message of the Day (/etc/motd)" later in this chapter).

- To create a `news` item, create a file with your text editor and place it in the directory `/usr/news`.
- To make sure users know about `news` items they haven't read yet, do the following:
  - For Korn and Bourne shell users, edit `/etc/profile` to include the following statement:

```
if [ -f /usr/bin/news ]
then news -n    #notify if news.
fi
```

  - For C shell users, edit `/etc/csh.login` to include the following:

```
if ( -f /usr/bin/news ) then
   news -n    #notify if new news.
endif
```

When they log in, if there are news items they haven't read, users will see a message like this:

`news:` *news_filename*

where *news_filename* is the name you gave the file in `/user/news`.

Users can enter `news` and the item or items will print on the screen. For more information, consult *news*(1) in the *HP-UX Reference Manual.*

Setting Up X Windows

X Windows is a package that allows the owner of a graphics workstation to interact with different applications on the same screen at the same time, on the analogy of a desktop on which a person is working concurrently on several different tasks.

X Windows is installed along with the operating system, and the only additional set-up you need to do is to allow users who want it to have X Windows as their default environment (so that the windows they specify in their `.x11start` file come up when they log in).

The easiest way to do this is to add the users in SAM, and respond `y` to the prompt `Login with X11 windows?`

If you respond `y`, SAM adds the following lines to the `.profile` file in a Bourne or Korn shell user's home directory:

```
# Add windows at login:
if [ "`who am i | grep console`" != "" ]
then
     exec /usr/bin/x11start
fi
echo "Press <Shift> <Ctrl> <Reset> simultaneously to exit all windows."
```

In the case of a C shell user, SAM adds the following lines to the `.login` file in the home directory:

```
# Add windows at login:
if ( "`who am i`" =~ *console* ) then
    exec /usr/bin/x11start
endif
echo "Press <Shift> <Ctrl> <Reset> simultaneously to exit all windows."
```

If you have already added the user, or if you choose not to use SAM, you can add the lines to the appropriate file yourself, or tell your users how to do it.

Setting Up System Accounting

System accounting, which is primarily intended for multi-user systems, allows you to:

- Monitor individual users' disk space usage.
- Record logins and logouts.
- Collect data about individual processes, such as memory usage and execution time.
- Charge fees for usage.
- Generate summaries and reports that you can use to analyze system performance and bill users for resource consumption.

If you need to set up system accounting, you should do so now. Details are in Chapter 12, "System Accounting".

Customizing the System

Customizing the system usually means editing a file, either to change the way the system behaves in general, or to modify the way a particular user interacts with it.

The most important files you can customize are:

`/etc/inittab`:	Contains information about system run levels and also has an entry for each terminal. See "Editing the /etc/inittab File" later in this chapter.
`/etc/rc`:	Defines actions taken during startup. See "Editing the /etc/rc File" later in this chapter.
`/etc/passwd`:	Determines who can log into your system. You can add, delete and modify entries either by editing the file, or by means of SAM screens under the `Users` menu. Chapter 4, "Controlling Access to the System", contains more information.
`/etc/group`:	Identifies the users that form a group, associates group IDs (GIDs) with group names, lists users, and associates those users with a group name and a group ID. There's more information in Chapter 4, "Controlling Access to the System".

`/etc/ttytype`: Used by the `tset` command as a database of terminal types on your system.

Edit this file when you add a new type of terminal or modem to your HP-UX system. For example:

```
300h console
2397 tty00
2397 tty01
```

`.exrc`: Maps terminal characteristics and sets up key definitions for the `ex` family of HP-UX editors (`vi`, `ex`, and so on).

See "Customizing Users' Login Environments" later in this chapter.

`/etc/issue`: Determines what the user will see before the login prompt.

See "Customizing the Login Prompt (/etc/issue)" later in this chapter.

`/etc/motd`: Contains the message of the day.

See "Posting a Message of the Day (/etc/motd)" later in this chapter.

`/etc/profile`, `/etc/csh.login`: Executes automatically during the login process.

The `/etc/profile` file executes for Bourne, Korn, and restricted shell users. The `/etc/csh.login` file executes for C shell users.

See "Customizing Users' Login Environments" later in this chapter.

$HOME files: (Files in the user's home directory).

.profile: Executes each time the user successfully logs in using the Bourne shell, Korn shell, or restricted shell.

.kshrc: Korn shell script that supplements actions taken by the .profile file.

Executes whenever a new Korn shell is spawned, if specified by the following statements in the user's .profile:

```
ENV=$HOME/.kshrc
export ENV
```

The name .kshrc is merely a convention: whatever file you specify will execute.

.cshrc: executes when a new C shell starts.

.login: executes when a C shell user logs in, after .cshrc.

.environ: executes when a user logs in using PAM.

(PAM is a user interface to HP-UX and MSDOS, suitable for novices and infrequent computer users.)

See "Customizing Users' Login Environments" later in this chapter.

Customizing System Startup

When the system boots, it executes a series of programs and shell scripts. (Details are in Chapter 3, "Starting and Stopping HP-UX".) Of the files involved, you can customize `/etc/rc` and `/etc/inittab`.

Editing the /etc/inittab File

`/etc/inittab` is input to `/etc/init`, the first program HP-UX runs after obtaining control from the boot ROM. Use `/etc/inittab` to set system run-levels (see Chapter 4, "Controlling Access to the System").

You need to edit `/etc/inittab` whenever you add a new terminal to your system. For more information on `/etc/inittab`, see *inittab*(4) in the *HP-UX Reference Manual*. An entry for a terminal whose device file name is `/dev/tty04` would look like this:

```
04:2:respawn:/etc/getty tty04 H #comment to identify user
```

When you start up the system, this terminal will receive a `login:` prompt, and the prompt will be redisplayed after the user logs out.

If you use SAM to add the terminal, SAM adds the `getty` line to `/etc/inittab`.

For instructions on adding a terminal, see *Installing Peripherals*, Chapter 4, "Installing Terminals and Modems". in the *Installing Peripherals* manual. If you decide to use HP-UX commands instead of SAM, additionally refer to Chapter 8, "Setting Up Devices Using HP-UX Commands" in the *Installing Peripherals* manual.

Editing the /etc/rc File

The `/etc/rc` script is executed by the `/etc/init` program during system startup.

`/etc/rc` performs a number of functions, including setting the timezone and the date, and initializing system processes such as the **syncer daemon** and the **line-printer spooler**.

Edit `/etc/rc` to do any processing you might want this particular system to do when it boots.

For example, if you have Network Services, you might want to start the proxy server here.

You should put this processing in a separate script, such as `/etc/rc.local`, and call the script from `/etc/rc`. This way, it is easier to recreate your customization if `/etc/rc` is overwritten when you update HP-UX to a new release.

Customizing the Login Prompt (/etc/issue)

`/etc/issue` contains text users will see immediately before the `login:` prompt. Normally it identifies the system (by the **host name** from `/etc/hosts` if this is a networked system, and its "friendly" alias, if any), the release of HP-UX, and includes any other information you want to put there.

For example:

```
Folly [HP-UX Release A.09.0  9000/870]
```

Posting a Message of the Day (/etc/motd)

The message of the day appears each time a user logs in if the user's personal customization file (`/etc/profile` for Bourne and Korn shell users or `/etc/csh.login` file for C shell users) has the following line:

```
cat /etc/motd        # message of the day
```

Edit `/etc/motd` to display topical messages. For example:

```
Monthend reports due this week.
Going away party for Leslie on Wednesday.
Sign your time-sheets by Friday.
```

Customizing Users' Login Environments

`/etc/profile` and the `.profile` file in the user's home directory execute when a Bourne or Korn shell user logs in. When a C shell user logs in, `/etc/csh.login` executes, and so do the `.cshrc` and `.login` files in the user's home directory.

`/etc/profile` and `/etc/csh.login` should contain the defaults for variables such as the timezone setting, the terminal type, search path, and mail and news notification. These can be overridden if necessary in individual users' `.profile` or `.login` files.

`.cshrc` in the user's home directory performs additional set-up such as setting **aliases** (user-defined commands). `.kshrc` performs similar tasks for a Korn shell user if it is declared in the `ENV` variable.

When you add a new user, you may want to place default versions of these files in the user's home group. (If you use SAM to add a user, SAM puts the appropriate files in the home group for you.) You can use the sample files in the `/etc/` directory (such as `/etc/d.profile`) as templates, editing them as you wish.

A Beginner's Guide to Using Shells has examples of all these files, with explanations of each entry in the default files.

Customizing Users' Editing Environments

This means editing the `.exrc` file in the user's home group to enable keyboard features such as the cursor arrow keys, and to set other options in the `ex` family of editors, including `vi`.

The `.exrc` file functions only if the `EXINIT` variable is *not* defined in the `/etc/profile` or `$HOME/.profile` files.

`/etc/d.exrc` is a sample `.exrc` file. You may want to customize the file and provide it to new users as a default. There's a commented version of the file in Chapter 11 of *The Ultimate Guide to the vi and ex Text Editors*, a Hewlett-Packard publication that you can order separately.

Setting Up Non-Standard Terminals

Files in directories under `/usr/lib/terminfo` enable you to use a wide variety of terminals, including terminals not manufactured by Hewlett-Packard.

To set a user up with a non-HP terminal, do the following:

1. Make sure the fileset `NONHPTERM` has been loaded:

   ```
   ls /etc/filesets/NONHPTERM
   ```

   If the fileset is not there, you can get it from your latest set of `update` media (the media on which you got the current release of HP-UX).

   Run the `update` program and select the `NOHPTERM` fileset. Chapter 7, "Updating HP-UX" in the manual *Installing and Updating HP-UX* shows how to load an individual fileset.

2. Find the file that corresponds to the terminal you want to set up, if the file exists.

   Suppose you want to set someone up with a Wyse (TM) 100 terminal. All supported terminals whose names begin with "w" are listed under `/usr/lib/terminfo/w`.

   Enter

   ```
   ls /usr/lib/terminfo/w
   ```

   and you'll see an entry for `wy100`. This is the `terminfo` file for the Wyse 100.

   If there is no `terminfo` file for the terminal you want to add, you can create one. See "Creating a New terminfo File" later in this chapter.

3. Find the terminal name in the file.

   For example,

   ```
   more /usr/lib/terminfo/w/wy100
   ```

   This will produce a screenful of special characters, but near the beginning you'll see `wy100|100|wyse 100`. This means you can refer to the Wyse 100 by any of the names `wy100`, `100` or `wyse 100`.

4. Set the user's `TERM` variable in the appropriate login script in their home directory: `.profile` for a Korn or Bourne shell user, or `.login` for a C shell user.

   For example (Bourne or Korn shell):

   ```
   TERM=wy100
   export TERM
   ```

   (C shell):

   ```
   set TERM wy100
   ```

   The default versions of these scripts prompt the user for the terminal type when he or she logs in, so rather than editing the script, you could simply tell the user to respond with the terminal name, for example:

   ```
   TERM = (hp) wy100
   ```

Creating a New terminfo File

If there is no `terminfo` file for the terminal you want to set up, you can create one. The *terminfo*(4) entry in the *HP-UX Reference Manual* explains the rules for constructing a `terminfo` file.

You may want to copy an existing `terminfo` file. In this case, get into the directory containing the file you want to copy and create a ASCII version of the file.

For example, to make a copy the file `/usr/lib/terminfo/w/wy100`, do the following:

1. Log in as superuser.
2. Change directories:

   ```
   cd /usr/lib/terminfo/w
   ```

3. Make an ASCII version of the file:

   ```
   untic wy100>filename
   ```

   where *filename* is whatever you want to call the new file. Make it similar to the terminal's product name and model if you can.

4. Edit the file to reflect the capabilities of the new terminal.

   Make sure you change the name(s) of the terminal in the first line. See *terminfo*(4) for rules for entries.

5. Compile the new `terminfo` file:

   ```
   tic filename
   ```

For more information on using the `terminfo` compiler, refer to *tic*(1M) in *HP-UX Reference Manual*.

Backing up the System

Now that you have built the system, you should do a full back up. This will allow you to reconstruct your system—kernel, system files, file system structure, user structures, and your customized files—should you ever need to. (If you often have to build a new system, the backup will also give you a basis from which to recreate the next system.)

SAM provides backup capability in addition to the several HP-UX tools available for back up. Each is explained in Chapter 9, "Backing Up and Restoring Your Data".

Reconfiguring the Kernel

This section describes:

- When you need to configure the kernel.
- Reconfiguring the kernel using HP-UX commands.
- Reconfiguring the kernel using SAM when:
  - adding/removing device drivers
  - modifying system parameters
  - adding/removing/modifying swap, dumps, and console devices

When Do You Need to Reconfigure the Kernel?

You need to reconfigure the kernel in the following situations:

- When you add a new peripheral to your system that requires a device driver that is not already configured in the kernel, you need to add that driver. By "peripherals", we mean hardware that you can add to your system, including terminals, printers, disk drives, tape drives, other storage and communication devices, and the interface cards they attach to. For adding a peripheral, refer to the *Installing Peripherals* manual.

  You may also want to remove a driver from your kernel if your system no longer uses any peripherals of that type. (This is not an absolute requirement, but it's desirable, since a smaller kernel is more efficient.)

- When you change kernel parameters, you need to reconfigure the kernel.

  When you first install your system, the configurable kernel parameters are set to default values. These values are correct for most systems, but under some circumstances you might decide to change one or more parameters—to accommodate a specialized application, for example, or an exceptionally large number of users.

  Refer to Appendix A, "System Parameters", which contains descriptions of all tunable parameters, their ranges, and how they interact with other parameters. Appendix A, "System Parameters" also contains a description of how to use SAM to change the tunable kernel parameters.

- If you add certain Hewlett-Packard software, such as LAN (Local Area Network) or NS (Network Services), you might need to reconfigure the kernel. In almost all cases, however, you should use *update*(1M) (the `/etc/update` program) to add a subsystem, rather than the method described in this chapter. For example, the *uxgen*(1M) command will add the subsystem "device drivers" to your kernel, but will not get the software application and its files onto your system or do any other set-up needed.

  The *update*(1M) utility, on the other hand, helps you load the software and does the necessary configuration. Consult the manual that came with the software for installation instructions. *Installing and Updating HP-UX* contains more information on `update`.

- When you change the locations of the following kernel devices:
  - System console.
  - `root` (`/`) file system.
  - Primary swap.
  - Dumps.

Reconfiguring the Kernel Using HP-UX Commands

This section summarizes the steps you need to take to modify HP-UX.

1. Log in as super-user.
2. At the HP-UX prompt, issue the command:

   ```
   cd /etc/conf/gen
   ```

3. Copy the existing kernel configuration file (`S800`) to a different name, so that you will have a backup copy in case of problems. For example:

   ```
   cp S800 S800.BCKUP
   ```

   We highly recommend the use of `S800` for the kernel configuration file and that it be kept upto date with the currently executing kernel `/hp-ux`.

4. Edit the `S800` file to reflect changes to drivers, system parameters, I/O addresses for non-autoconfigurable I/O cards and devices.

   For example:

   ```
   vi S800
   ```

   (If you named the file something other than `S800`, use that name here instead. However, we strongly recommend sticking with `S800` because many system programs, such as SAM and `update`, expect to find it named `S800`.)

5. After making changes, regenerate the kernel with **uxgen**, using the edited S800 file as input:

   ```
   /etc/uxgen S800
   ```

   The **uxgen** program generates a new kernel, calling it **hp-ux**, and puts it in the directory **/etc/conf/S800**. The **uxgen** program also creates the source files to create the new kernel (**conf.c**, **config.h**, **libs_file**, and **Makefile**).

   If you make a syntax error in the S800 file (for example if you spelled a driver's name incorrectly), the **uxgen** program issues an error message. Re-edit the S800 file so that it is correct, and run **uxgen** again.

6. Copy the old kernel (**/hp-ux**) in the root (**/**) directory. Write down the name of this file in case the new kernel does not boot. For example:

   ```
   cp /hp-ux /SYSBCKUP
   ```

   Do not perform this step if your system is booted from the **/SYSBCKUP** backup kernel. If you do, you could overwrite the only bootable kernel for your system.

7. Change the working directory:

   ```
   cd /etc/conf/S800
   ```

8. Move **hp-ux** to **/hp-ux** by entering:

   ```
   mv hp-ux /hp-ux
   ```

9. Write down the information you need in case the new kernel doesn't boot:

   The name of the device driver for the system disk (`disc1` for HP-IB, `disc2` for HP-FL, `disc4` or HP-FL on NIO computers, `disc3` for SCSI).
   The hardware address of the system disk (Primary Boot Path).
   The section number of the root file system.
   The name of the backup kernel (typically `/SYSBCKUP`).

   To determine the device driver and hardware address of the system disk and to find the section number for the root file system on the system disk, you can use the `/etc/devnm` and `/etc/lssf` commands as follows:

   Enter the following command to get the device file name for the root file system by entering:

   ```
   /etc/devnm /
   ```

   and you will get something like:

   ```
   /dev/dsk/c0d0s13 /
   ```

   Then, by entering:

   ```
   /etc/lssf /dev/dsk/c0d0s13
   ```

   you will get output to the screen like the following that will provide the information you need:

   ```
   disc1 lu 0 unit 0 section 13 address 4.0.0 /dev/dsk/c0d0s13
   ```

   The driver (`disc1`), the hardware address (4.0.0), and the disk section number (`s13`) are in the output.

10. Shut down the system:

Caution Before shutting down the system, be sure to write down the information required for booting from the old kernel. Otherwise it may not be easy to get HP-UX running if the new kernel fails to boot.

```
cd /
shutdown -h 0
```

Wait until the console displays a message like:

```
Halting (in tight loop) -- OK to Hit Reset Button
```

11. Turn off system power.
12. Now, if you are installing peripherals, install any cards or devices. Follow the instructions in *Installing Peripherals* and be sure to follow all ESD precautions.

Warning **Be sure to follow the ESD precautions when handling cards and devices. ESD precautions are described in the hardware** *Installation and Configuration Guides*.

13. Turn on the devices and then turn on the system to reboot. The kernel with the new configuration should load.

An Alternative Series 800 Kernel Configuration Method: regen

Instead of `uxgen`, you may want to use the `regen(1M)` utility, which is friendlier and does more for you than `uxgen`; however, `regen` allows you less flexibility.

There is more information about `regen` in the manual *HP-UX Reference Manual*.

Reconfiguring the Kernel to Add/Remove Device Drivers Using SAM

To reconfigure the HP-UX kernel using SAM:

1. Login as superuser.
2. Run SAM; type:

   `/usr/bin/sam`

3. Highlight `Kernel Configuration` and activate the [Open] control button.
4. Highlight `Drivers` and activate the [OK] control button.

   A list of drivers, their current status, and their pending status appear. Current status refers to a driver being in or *not* in (out of) the currently executing kernel. Pending status gives the status of the driver when you next regenerate the kernel (in or out).
5. Perform one of the two following tasks:
   - To *add* a driver (or drivers) to the kernel, highlight the diver (or drivers) you want to add and choose `Add Driver to Kernel` from the "Actions" menu.

     The "Pending State" column entry for the driver(s) changes to "In". You must recreate the kernel and install it to implement the change.
   - To *remove* a driver (or drivers) from the kernel, highlight the diver (or drivers) you want to remove and choose `Remove Driver to Kernel` from the "Actions" menu.

     The "Pending State" column entry for the driver(s) changes to the value "Out." You must recreate the kernel and install it to implement the change.

6. Choose `Create a New Kernel...` from the "Actions" menu.
7. Activate the (Yes) control button to confirm that you want to reconfigure the kernel now.
8. Choose one of the following:

   - **`Create a new kernel now`**

     This requires a reboot of your system. SAM prompts you to continue. SAM creates directory `/etc/conf/S800.SAM` to hold the files created during the kernel creation process (`conf.c`, `config.h`, `libs_file`, and `Makefile`). This directory contains the same files as in `/etc/conf/S800` created by the HP-UX command `uxgen`; see "Reconfiguring the Kernel Using HP-UX Commands".

   - **`Defer kernel creation until later`**

     SAM preserves the request to reconfigure the kernel. If you attempt to exit the Kernel Configuration area of SAM before you have reconfigured the kernel, SAM prompts you to reconfigure the kernel or cancel your reconfiguration request.

   - **`Cancel all kernel modifications`**

9. Follow the prompts to regenerate and reinstall the new kernel. After SAM generates a new kernel, choose one of the following actions and activate the [OK] control button:
   - `Move the kernel into place and reboot the system now`
   - `Move the kernel into place but do not reboot the system`
   - `Exit without moving the kernel into place`

   There is an option to enable or disable overwriting the kernel configuration file, `/etc/conf/gen/S800`.

   If you *enable* overwriting the kernel configuration file, SAM moves `/etc/conf/gen/S800.SAM` to `/etc/conf/gen/S800`, overwriting any comments you have added to the configuration file.

   If you *disable* overwriting the kernel configuration file, `/etc/conf/gen/S800` will not represent your current kernel (`/hp-ux`) when you reboot your system. `/etc/conf/gen/S800.SAM` represents your current kernel configuration after you reboot your system.

   We highly recommend the use of `/etc/conf/gen/S800` for the kernel configuration file so that it remains up to date with the executing kernel, `/hp-ux`. Some system software depends on `/etc/conf/gen/S800` representing the currently executing kernel.

   If you do not want SAM to overwrite `/etc/conf/gen/S800`, because of comments you want to retain:
   a. Choose the SAM option to disable overwriting the kernel configuration file.
   b. Move the kernel into place (optionally rebooting the system).
   c. Copy your comments from `/etc/conf/gen/S800` to `/etc/conf/gen/S800.SAM`. Be careful to add only your comments to the file. At this stage, you want `/etc/conf/gen/S800.SAM` to reflect your current kernel configuration.
   d. Copy or save `/etc/conf/gen/S800.SAM` to `/etc/conf/gen/S800`.

   The kernel configuration file `/etc/conf/gen/S800` now represents the current `/hp-ux` kernel.

Reconfiguring the Kernel to Modify System Parameters Using SAM

See Appendix A, "System Parameters" for reference and discussion of specific tunable system parameters.

To reconfigure the HP-UX kernel using SAM:

1. Login as superuser.
2. Run SAM; type:

   `/usr/bin/sam`

3. Highlight `Kernel Configuration` and activate the (Open) control button.
4. Highlight `Configurable Parameters` and activate the (OK) control button.
5. Highlight the parameter you wish to update and choose `Modify Configurable Parameter....`
6. Choose one of the following:
   - `Specify new formula`
   - `Specify new value`
   - `Select formula/value from source file`
7. Fill in the new information and activate the (OK) control button.
8. Choose `Create a New Kernel...` from the "Actions" menu.
9. Activate the (Yes) control button to confirm that you want to reconfigure the kernel now.
10. Choose one of the following:
    - `Create a new kernel now`

      This requires a reboot of your system. SAM prompts you to continue. SAM creates directory `/etc/conf/S800.SAM` to hold the files created during the kernel creation process (`conf.c`, `config.h`, `libs_file`, and `Makefile`). This directory contains the same files as in `/etc/conf/S800` created by the HP-UX command `uxgen`, see "Reconfiguring the Kernel Using HP-UX Commands".

- `Defer kernel creation until later`

  SAM preserves the request to reconfigure the kernel. If you attempt to exit SAM before you have reconfigured the kernel, SAM prompts you to reconfigure the kernel or cancel your reconfiguration request.

- `Cancel all kernel modifications`

11. Follow the prompts to regenerate and reinstall the new kernel. After SAM generates a new kernel, choose one of the following actions and activate the [OK] control button:

    - `Move the kernel into place and reboot the system now`
    - `Move the kernel into place but do not reboot the system`
    - `Exit without moving the kernel into place`

    There is an option to enable or disable overwriting the kernel configuration file, `/etc/conf/gen/S800`.

    If you *enable* overwriting the kernel configuration file, SAM moves `/etc/conf/gen/S800.SAM` to `/etc/conf/gen/S800`, overwriting any comments you have added to the configuration file.

    If you *disable* overwriting the kernel configuration file, `/etc/conf/gen/S800` will not represent your current kernel (`/hp-ux`) when you reboot your system. `/etc/conf/gen/S800.SAM` represents your current kernel configuration after you reboot your system.

    We highly recommend the use of `/etc/conf/gen/S800` for the kernel configuration file so that it remains up to date with the executing kernel, `/hp-ux`. Some system software depends on `/etc/conf/gen/S800` representing the currently executing kernel.

If you do not want SAM to overwrite `/etc/conf/gen/S800`, because of comments you want to retain:

a. Choose the SAM option to disable overwriting the kernel configuration file.
b. Move the kernel into place (optionally rebooting the system).
c. Copy your comments from `/etc/conf/gen/S800` to `/etc/conf/gen/S800.SAM`. Be careful to add only your comments to the file. At this stage, you want `/etc/conf/gen/S800.SAM` to reflect your current kernel configuration.
d. Copy or save `/etc/conf/gen/S800.SAM` to `/etc/conf/gen/S800`.

The kernel configuration file `/etc/conf/gen/S800` now represents the current `/hp-ux` kernel.

Reconfiguring the Kernel to Modify Special Devices Using SAM

Note

- SAM supports removing dumps devices.
- SAM supports modifying:
  - The location of primary swap.
  - The location of the system console.
  - Adding dumps devices.
- SAM does not support adding, removing, or modifying the root device.

See Chapter 7, "Managing Swap Space" for guidelines about configuring primary swap and dumps devices.

To modify special devices using SAM:

1. Login as superuser.
2. Run SAM; type:

   `/usr/bin/sam`

3. Highlight `Kernel Configuration` and activate the [Open] control button.
4. Highlight `Special Devices` and activate the [OK] control button.
5. Highlight the device you want to modify and choose `Modify Device Configuration` from the "Actions" menu.
6. Highlight and fill in the modifications you want to make and activate the [OK] control button. Refer to Chapter 7, "Managing Swap Space" for additional information to modify your primary swap device.
7. Choose `Create a New Kernel...` from the "Actions" menu.
8. Activate the [Yes] control button to confirm that you want to reconfigure the kernel now.
9. Choose one of the following:

   - `Create a new kernel now`

     This requires a reboot of your system. SAM prompts you to continue. SAM creates directory `/etc/conf/S800.SAM` to hold the files created during the kernel creation process (`conf.c`, `config.h`, `libs_file`, and `Makefile`). This directory contains the same files as in `/etc/conf/S800` created by the HP-UX command `uxgen`, see "Reconfiguring the Kernel Using HP-UX Commands".

   - `Defer kernel creation until later`

     SAM preserves the request to reconfigure the kernel. If you attempt to exit SAM before you have reconfigured the kernel, SAM prompts you to reconfigure the kernel or cancel your reconfiguration request.

   - `Cancel all kernel modifications`

10. Follow the prompts to regenerate and reinstall the new kernel. After SAM generates a new kernel, choose one of the following actions and activate the [OK] control button:

    - `Move the kernel into place and reboot the system now`
    - `Move the kernel into place but do not reboot the system`
    - `Exit without moving the kernel into place`

    There is an option to enable or disable overwriting the kernel configuration file, `/etc/conf/gen/S800`.

    If you *enable* overwriting the kernel configuration file, SAM moves `/etc/conf/gen/S800.SAM` to `/etc/conf/gen/S800`, overwriting any comments you have added to the configuration file.

    If you *disable* overwriting the kernel configuration file, `/etc/conf/gen/S800` will not represent your current kernel (`/hp-ux`) when you reboot your system. `/etc/conf/gen/S800.SAM` represents your current kernel configuration after you reboot your system.

    We highly recommend the use of `/etc/conf/gen/S800` for the kernel configuration file so that it remains up to date with the executing kernel, `/hp-ux`. Some system software depends on `/etc/conf/gen/S800` representing the currently executing kernel.

    If you do not want SAM to overwrite `/etc/conf/gen/S800`, because of comments you want to retain:

    a. Choose the SAM option to disable overwriting the kernel configuration file.

    b. Move the kernel into place (optionally rebooting the system).

    c. Copy your comments from `/etc/conf/gen/S800` to `/etc/conf/gen/S800.SAM`. Be careful to add only your comments to the file. At this stage, you want `/etc/conf/gen/S800.SAM` to reflect your current kernel configuration.

    d. Copy or save `/etc/conf/gen/S800.SAM` to `/etc/conf/gen/S800`.

    The kernel configuration file `/etc/conf/gen/S800` now represents the current `/hp-ux` kernel.

Booting the Backup Kernel

If the new kernel doesn't boot properly, reboot with the old kernel instead:

1. Turn the system's power off, and then on again.
2. Interrupt the autoboot process by pressing any key.
3. Respond Y to the prompt: `Boot from primary path?`.
4. Respond Y to the prompt: `Interact with IPL?`.
5. Using the information you acquired before shutting the system down (see previous section), formulate the ISL command using the syntax described as follows:

   ```
   ISL> hpux disc1(x.y.z;s)/SYSBCKUP
   ```

   Where `x.y.z` is the hardware path of the system disk and `s` is the section number of the root file system. If the system disk is HP-FL, then use **disc2** instead of **disc1**. If you named the backup kernel something other than `/SYSBCKUP`, use that name instead.

   Use the hardware address you determined in step 9 of "Reconfiguring the Kernel Using HP-UX Commands", otherwise refer to Chapter 10, "System Architectures" in the *How HP-UX Works: Concepts for the System Administrator* for determining the hardware path of your disk drive based on your model of computer.

Once you have the backup kernel running, you can re-edit the `S800` file and try the process again.

Adding and Removing Subsystems

In order to run certain Hewlett-Packard subsystems, you must not only install the software, but also reconfigure the kernel to make it aware of the subsystem. In general, you don't need to worry about this, because the `update` utility, which loads the software, also makes the necessary modifications to the kernel and to any other files in which the subsystem may require specific entries.

If you are about to install a new Hewlett-Packard subsystem, do not continue with this section. Instead, follow directions in the documentation that came with the product. Usually these documents will tell you to run `/etc/update`, and will contain all the information you need to install the software.

If you need more information on `/etc/update`, you'll find it in the manual *Installing and Updating HP-UX*.

Adding a Subsystem Using HP-UX Commands

Adding a subsystem is necessary only when you already have all the files a given Hewlett-Packard subsystem comprises on your system, and need to add the corresponding "pseudo-driver" to the kernel. A pseudo-driver is a piece of software that enables the kernel to communicate with the subsystem, which by comparison with a device driver controls a hardware device such as a disk drive).

This case could occur if you had disabled a subsystem by removing the pseudo-driver from the kernel, but had not removed the software. To enable the subsystem again, you could simply add the pseudo-driver back into the kernel, and then regenerate the kernel and reboot.

The configurable pseudo-drivers and the subsystems they support are as follows:

Table 2-1. Pseudo-Drivers

Driver Name	Used For:
nfs	Support for NFS networking
lla, lan0, lan1, lan01	Support for NS-ARPA networking (formerly the ieee802 and ethernet drivers)
lan0, lan1, uipc, inet, netdiag1, netman, lla, lan01	Support for LAN
nipc	Support for NS networking
x25ip, x25pa	Support for X.25 networking

Table 2-1 shows which pseudo-drivers are used by which subsystems.

Caution If you have installed a new version of HP-UX since you removed the subsystem pseudo-driver form the kernel, you must not simply add the pseudo-driver back by editing the kernel generation file and creating a new kernel. The older subsystem might be incompatible with the new kernel, in which case the kernel build will fail.

Unless you are certain you have not updated HP-UX since you removed the pseudo-driver from the kernel, use `/etc/update` to recover the entire subsystem from the latest set of `update` tapes, or from the latest product tape if you got the product after the latest HP-UX release. We recommend this as the safest method in all cases.

Removing a Subsystem Using HP-UX Commands

If you are removing a subsystem (all the programs and data files) from the system, it's a good idea to remove the subsystem's pseudo-driver from the kernel as well. This will make the kernel smaller and it will run more efficiently.

Table 2-1 shows which pseudo-drivers go with which subsystems.

Reconfiguring the kernel using `uxgen` removes only the pseudo-driver, not the software. To remove the software filesets, use the `rmfn` tool described in the later section, "Removing System Files".

Adding and Removing Subsystems Using SAM

To reconfigure the HP-UX kernel using SAM:

1. Login as superuser.
2. Run SAM; type:

   `/usr/bin/sam`

3. Highlight `Kernel Configuration` and activate the [Open] control button.
4. Highlight `Subsystems` and activate the [OK] control button.
5. To add a subsystem:
   a. The subsystem file set must be loaded on the system; refer to *Installing and Updating HP-UX*.
   b. Highlight the subsystem you want to add and choose `Add Subsystem to Kernel` from the "Actions" menu. The "Pending State" column entry for the subsystem(s) should have the value "In."
6. To remove a subsystem, highlight the subsystem you want to remove and choose `Remove Subsystem from Kernel` from the "Actions" menu. The "Pending State" column entry for the subsystem(s) should have the value "Out."
7. Choose `Create a New Kernel` from the "Actions" menu.

8. Activate the (Yes) control button to confirm that you want to reconfigure the kernel now.
9. Choose one of the following:
   - `Create a new kernel now`

     This requires a reboot of your system. SAM prompts you to continue. SAM creates directory `/etc/conf/S800.SAM` to hold the files created during the kernel creation process (`conf.c`, `config.h`, `libs_file`, and `Makefile`). This directory contains the same files as in `/etc/conf/S800` created by the HP-UX command `uxgen`, see "Reconfiguring the Kernel Using HP-UX Commands".
   - `Defer kernel creation until later`

     SAM preserves the request to reconfigure the kernel. If you attempt to exit SAM before you have reconfigured the kernel, SAM prompts you to reconfigure the kernel or cancel your reconfiguration request.
   - `Cancel all kernel modifications`

10. Follow the prompts to regenerate and reinstall the new kernel. After SAM generates a new kernel, choose one of the following actions and activate the [OK] control button:

    - `Move the kernel into place and reboot the system now`
    - `Move the kernel into place but do not reboot the system`
    - `Exit without moving the kernel into place`

    There is an option to enable or disable overwriting the kernel configuration file, `/etc/conf/gen/S800`.

    If you *enable* overwriting the kernel configuration file, SAM moves `/etc/conf/gen/S800.SAM` to `/etc/conf/gen/S800`, overwriting any comments you have added to the configuration file.

    If you *disable* overwriting the kernel configuration file, `/etc/conf/gen/S800` will not represent your current kernel (`/hp-ux`) when you reboot your system. `/etc/conf/gen/S800.SAM` represents your current kernel configuration after you reboot your system.

    We highly recommend the use of `/etc/conf/gen/S800` for the kernel configuration file so that it remains up to date with the executing kernel, `/hp-ux`. Some system software depends on `/etc/conf/gen/S800` representing the currently executing kernel.

    If you do not want SAM to overwrite `/etc/conf/gen/S800`, because of comments you want to retain:

    a. Choose the SAM option to disable overwriting the kernel configuration file.

    b. Move the kernel into place (optionally rebooting the system).

    c. Copy your comments from `/etc/conf/gen/S800` to `/etc/conf/gen/S800.SAM`. Be careful to add only your comments to the file. At this stage, you want `/etc/conf/gen/S800.SAM` to reflect your current kernel configuration.

    d. Copy or save `/etc/conf/gen/S800.SAM` to `/etc/conf/gen/S800`.

    The kernel configuration file `/etc/conf/gen/S800` now represents the current `/hp-ux` kernel.

Removing System Files

You can use the `rmfn` (remove functionality) utility to remove system software you don't need. The interactive screens of `rmfn` allow you to remove software in groupings at the level of **filesets** and **partitions**. (A fileset is a logical group of files that make up a piece of software functionality. A partition is a logical group of filesets.)

Before it removes filesets, `rmfn` checks fileset dependencies. If a fileset or partition you select for removal is required by other filesets, `rmfn` asks you if you want to remove these dependent filesets as well.

The `rmfn` command keeps a log of its actions in `/tmp/rmfn.log`.

Caution Although `rmfn` checks dependencies to prevent you from inadvertently removing functionality, you must still be cautious because `rmfn` removes major pieces of software quickly.

Important Points

- The filesets and partitions that `rmfn` displays depend on the contents of the directories `/etc/filesets` and `/system`. Do not change the contents of these directories or `rmfn` will display an inaccurate list of filesets.
- `rmfn` checks that removing the selected filesets will not harm the integrity of your system. For example:
  - `rmfn` will not allow you to remove a minimum set of filesets needed by the system. For instance, you cannot remove `UX-CORE`.
  - If you have a mirrored system, and select a mirrored fileset while the mirror is on, `rmfn` will not remove the fileset.
- The `rmfn` command will not remove files on a remote mounted system (NFS).
- As `rmfn` removes a symbolic link contained in a fileset, it does not remove a symbolic link's target file. A target file remains in tact until `rmfn` removes the fileset that contains it.

How to Use rmfn

To use the **rmfn** command, log in as superuser, and enter:

```
/etc/rmfn
```

The example below shows a typical **rmfn** menu screen.

To remove an entire partition, mark it with a **y**. To remove individual filesets within a partition, press `Select Filesets` and then mark the filesets you want removed with a **y**. (After you select individual filesets within a partition for removal, the partition is automatically marked with a **p** for partially selected.)

To prevent a fileset or partition from being removed (to keep it on your system), mark it **n**; **n** is the default choice for all filesets on the screen.

```
rmfn                                Partitions

Press "y" to select an entire partition for deletion.  Press "n" to undo a
selection.  Press the "Select Filesets" key to view the filesets within a
partition.  Press the "Start Removing" key when selection is complete.

Mark  Partition Name    Arch.   Partition Description            Size in Kbytes

 p    DIAGNOSTICS       800     Hardware Diagnostic Programs       37663
 n    NETWORKING        800     Networking Products                 8919
 n    NLS               800     Native Language Support              472
 n    OS-ADMIN          800     Recommended Administration Cmds     2292
 n    OS-CORE           800     Recommended System Core             5517
 n    OS-FEATURES       800     Selectable OS Features              8176
 n    PROG-LANGUAGES    800     Programming Languages               8542
 n    REFERENCE-DOC     800     Reference Manual Pages               348
 n    SHARED-LIBS       800     Runtime Shared Libraries            2757
 n    WINDOWS           800     Windowing Products                   102

   Help        Shell   Start                 View    Select        Exit
                       Removing              Selected Filesets     rmfn
```

Figure 2-1. `rmfn`**: "Partitions" Screen**

Help	Explains how to the use the **rmfn** tool.
Shell	Lets you escape to the shell. Type `exit` to return to the **rmfn** screen.
Start Removing	Removes the selected filesets and partitions from your system.
View Selected	Lists the names and sizes of the partitions and filesets selected for removal.
Select Filesets	Allows you to individually select the filesets within a partition for removal.
Exit rmfn	Exits **rmfn**.

3

Starting and Stopping HP-UX

This chapter describes:

- Turning on the Peripherals and the Computer
- Booting the System
- Checking the File System
- Setting Initial Information
- Stopping the System

Refer to chapter 2, System Startup, in *How HP-UX Works: Concepts for the System Administrator* for more information on system startup.

The following sections describe how to start HP-UX if you are starting with a system that has everything connected but not turned on.

Turning on the Peripherals and the Computer

Before turning on the computer, turn on the console, terminals, any other peripherals and peripheral busses that are attached to the computer so that the computer will recognize them when it comes up. Then, turn on the computer.

Note

All peripheral busses must be turned on before the computer is turned on. If not, neither the bus nor its peripherals will be configured into the system.

If you have an HP-PB bus converter, it must be connected to the Mid-Bus and turned on when you attempt to boot the system. If the HP-PB bus converter is turned off or disconnected, the system will not boot.

Booting the System

You must reset the system to get it to boot. Reset the system by turning the reset key, pressing the reset button, or cycling the main power, depending the model of your computer.

3

ISL Boot

"ISL" refers to the "initial system loader."

The following is an example of what you will see on the console of a Series 800 computer when it is booted. Note that the display may vary depending on the model of your computer.

```
Processor Dependent Code (PDC) Revision 4

Console path = 8.1.0.0
Primary boot path = 8.0.0.0
Alternate boot path = 8.2.3.0
```

Now, depending on whether the autoboot feature is enabled or not, the boot will follow one of two courses.

If autoboot is Enabled

If `autoboot` is enabled, the boot process continues. The following message is displayed:

```
Autoboot from primary boot path enabled.
To override, press any key within 10 seconds.
```

If you do not interrupt the boot process by pressing a key, the boot process will continue automatically.

As the boot process completes, you will see messages that convey the following kinds of information:

```
real mem = amount of physical memory on your system
lockable mem = amount of memory that may be locked via plock(2) or
               shmctl(2)
avail mem = amount of real memory available to user processes
Console information
Processor information
Interface cards
Subsystem information
File system information
Buffer information
Root Device information
Copyright information
```


Disabling autoboot

Your HP-UX system is supplied with the autoboot feature enabled. You can disable the autoboot by entering:

```
autoboot off
```

at the ISL prompt. If the autoboot feature had been disabled, you could have entered autoboot on to enable it.

If autoboot is Disabled

If autoboot is disabled, the boot process pauses, asking you this question on the console:

3

```
Boot from primary boot path (Y or N)?>
```

- If you answer Y, the system proceeds to boot the system specified in the boot area of the disk at the primary path hardware address.
- If you type N, the loader prompts for alternate boot path.

```
Boot from alternate boot path (Y or N)?>
```

- If you answer Y, the system proceeds to start up the system specified in the boot area of the disk at the alternate path hardware address.
- If you type N, the loader prompts for the boot path.

```
Enter boot path, command, or ?>
```

When you specify a valid boot path, which is the hardware path to a disk with boot utilities, the system displays messages similar to the following:

```
Interact with IPL (Y or N)?> Y

Booting.

Console IO Dependent Code (IODC) Revision 17
Boot    IO Dependent Code (IODC) Revision 17

 HARD Booted

ISL Revision 2805 - January, 1991

ISL>
```

Interacting with ISL to Select a System to Boot

After specifying the boot path, you must tell the computer what to boot. Usually, you would specify `hp-ux`, the kernel or operating system that resides in the `root` file system. You can boot from an alternate kernel (see the next section).

For reference, in addition to the *How HP-UX Works: Concepts for the System Administrator*, you can obtain detailed information from the following:

- *ISL*(1M)
- *hpux_800*(1M)

For standalone or root server systems, the disk (usually) on which your bootable system resides has a boot area and a root file system located in either a root logical volume (in the case the root disk is an LVM disk) or a disk section (the root section is 13 by default).

The boot area contains the bootstrap program and other files needed for bringing up the system.

If AUTO File is Current

The AUTO file, which is a LIF file that resides in the boot area of the boot disk, contains an `hpux` boot command string that ISL uses to automatically boot the system.

If the AUTO file correctly identifies the location of the kernel, you can boot in multiuser mode by entering:

```
ISL> hpux
```

Or, to boot to single user mode,

```
ISL> hpux -is
```

The two commands shown above boot the kernel (`hp-ux`) in the root file system specified in the AUTO file.

Booting an Alternate Kernel, Same Location

To boot an alternate kernel (for example, one with a different name, such as ALTKERNEL) you can enter:

3

```
ISL> hpux -is ALTKERNEL
```

The system boots using the kernel, ALTKERNEL, in single-user mode. The above command works if the AUTO file specifies the correct location of the root file system.

Booting From Other Locations

You can boot a kernel from another disk (as long as it is connected to the same interface card used by the disk at the boot path you selected earlier). This procedure is not recommended.

For example, to boot from the kernel in the root file system on another disk, you would enter:

The `hpux` syntax for booting a kernel at a specific location is:

`hpux` *disc#(path; [section])kernel*

Where:

disc# is `disc1`, `disc2` or `disc3` depending on the type of driver required for the disk that contains root.

path specifies the disk drive from which to boot and has the following format:

```
bus_converter/channel_address.cio_slot.bus_address
```

or

```
bus_converter/hp-pb_slot.bus_address
```

section is the number of the section containing the root file system.

In the case of LVM, the correct number is section 2 (the whole disk).

kernel is the name of the kernel you want to boot (the default is `hp-ux`, but the kernel could be called `SYSBCKUP` or some other name for a customized kernel).

For example, if the kernel is `hp-ux` and resides in the root file system located in section 13 of an HP-IB disk, you can boot in single-user mode by entering:

```
hpux -is disc1(8.0.0;13)hp-ux
```

Note There are variations on how to enter this command. Review *hpux_800*(1M) in *HP-UX Reference Manual* for descriptions of the capabilities of the `hpux` command.

For example, to boot from another section on the *same* disk:

```
hpux -is (;4)hp-ux
```

As the system boots, it reports its activities and provides you key information including the status of the system's hardware, its software configuration, and the file system.

Changing the Primary Boot Path

To reboot automatically from a new root file system at its new hardware address, change the PRIMPATH variable in the initial system loader ISL (see *isl*(1M) in *HP-UX Reference Manual*).

3

1. Reboot.

```
/etc/reboot
```

2. Interact with ISL by pressing a key when you see:

```
Autoboot from primary path enabled.
To override, press any key within 10 seconds.
```

3. Answer "n" to the questions concerning booting from the primary boot path and alternate boot path, and enter the hardware path to the new root disk at the next prompt:

```
Enter boot path, command, or ?> 52.3.0
```

4. At the prompt, interact with IPL, specify the new PRIMPATH variable, and boot:

```
Interact with IPL?> y

ISL> primpath 52.3.0
ISL> hpux (;13)/hp-ux
```

The above sequence specifies your new primary boot path, but the change does not become implemented until you reboot. The system boots using the boot path entered in answer to the question `Enter boot path, command, or ?>`.

Changing the Contents of the AUTO File

You can change the contents of the AUTO File. The AUTO file, a LIF file that resides in the boot area of the boot disk, contains the **hpux** command that ISL uses to automatically boot the system.

You might want to change the contents of the AUTO file, for example, when you move the root disk to another location or decide to boot routinely from a different disk.

Changing the AUTO file when in ISL

From ISL, you can display the boot string in the AUTO file in section 6 of a disk at hardware address 52.3.0, for example, as follows:

```
hpux show autofile (52.3.0;6)
```

You can change the boot string in the AUTO file from ISL. Suppose you wanted to change the AUTO file to reflect that you want to boot from the kernel in section 7 on disk LU 4:

```
hpux set autofile  "hpux (;13)/hp-ux"
```

Changing the AUTO File Using mkboot

You can use the *mkboot*(1M) command to do this.

Examples, Creating AUTO File Bootstring

- For root in a disk section, you could enter:

```
mkboot -a "hpux (52.3.0;13)/hp-ux" /dev/rdsk/c4d0s6
```

3

This command copies a command to boot the kernel `hp-ux` from section 13 of the disk at hardware address "52.3.0.". The `mkboot` command places the AUTO file in section 6 of the disk with LU 4; notice that you must specify the *character* device file.

- For root in a logical volume, you could enter:

```
mkboot -a "hpux (52.3.0;2)/hp-ux" /dev/rdsk/c4d0s2
```

This command is just like the one used for disk sections, except that the section number for both the kernel location and the boot area is 2. This is because the boot area on the root disk in the root volume group is always at a specific location and stores the root file system location.

Checking the File System

If necessary, `fsck` will check the file system. You will see the status message:

```
Checking file systems
```

`fsck` will correct certain inconsistencies in the file systems, without your intervention, and without removing data. For more information on executing `fsck` and interpreting the messages it generates, refer to the manual *Solving HP-UX Problems*.

If there are errors that `fsck` corrects, you will receive the message:

```
FILE SYSTEM WAS MODIFIED
```

If the root file system was changed, the system will reboot.

Setting Initial Information

The first time you boot HP-UX, the operating system will ask you to provide the following information:

- System Name.
- Internet Protocol (IP) Address (If networking is installed).
- Time Zone.
- Date.
- Time.

Be prepared by having the following information available:

- Your unique system name. This is the host name and must be less than nine characters long, contain only alphabetic characters, numbers, underscores, or dashes, and must start with an alphabetic character.
- Your internet protocol (IP) address. This address has four numeric components, each of which is separated by a period, and each number must be between 0 and 256. An example of an IP address is: 255.32.3.10. If you do not have networking installed, you will not be prompted for the IP address.
- Your time zone. This is the time zone where your system is located. For example: Pacific Standard Time.
- The current date and time.

If you do not know any of this information, you can use the default values provided, and then change the information later. However, this initial information will not be requested at later system boots, and you will have to enter the information manually following the instructions in chapter 2 of this manual.

Completing Bootup

The operating system will continue to boot up and will display additional information about your system. After the bootup process has completed, you will see the login prompt. You are now ready to use your HP-UX system.

Shutting Down the System

Typically, you shut down the system down to:

1. Put it in single-user state so you can update the system, reconfigure the kernel, or check the file systems. See "Going to the Single-user State for Maintenance".
2. Turn it off totally so you can perform a task such as installing a new interface card. See "Shutting Down the System Completely".
3. Activate a new kernel. See "Shutting Down the System to Activate a New Kernel".

Caution Stopping the system improperly can corrupt (damage) the file systems or any data that is targeted for a block device. Never stop the system by turning it off.

An Overview of Shutdown

- Only the system administrator or a designated superuser should shut down the system.
- You can use SAM to shut down the system.
- The `/etc/shutdown` command warns users of impending shutdown, halts daemons, kills unauthorized processes, unmounts file systems, puts the system in single user mode, and writes the contents of the I/O buffers to a disk.
- The `shutdown` command warns all users to log off the system, using a grace period you can specify.
- Some system administration tasks will do the rebooting for you.
- If you use a network service, do not run `shutdown` from a remote system via `rlogin`. The shutdown process logs you out prematurely and returns control to the system console.
- See the *shutdown*(1M) entry in the *HP-UX Reference* manual for information about options and features.
- The *How HP-UX Works: Concepts for the System Administrator* manual has information on system shutdown concepts.

Going to the Single-user State for Maintenance

You must be a the system administrator with superuser capabilities or a designated user with superuser capabilities to shut down the system.

3 1. Change to the root directory.

```
cd /
```

2. Shut down the system.

   The `shutdown` command warns all users to log off the system, using a grace period you can specify.

   If you do not specify a grace period, users get 60 seconds to log off. You should notify active users as to when the system will be shut down. Give them enough time to finish their work and log off. You can do this physically or use the `/etc/wall` or `/etc/cwall` commands. The chapter named "Constructing an HP-UX System" has information.

   The following example show alternatives for shutting down to the single-user state:

   `shutdown` *Shuts down to single-user state, allowing the default 60 second grace period*

   `shutdown 0` *Shuts down the system with no grace period*

   `shutdown 30` *Begins the shutdown to the single-user state after a 30-second grace period*

   As always, watch the messages to see that everything is happening correctly.

3. While the system is in the single-user state, perform the necessary system administration tasks. When you finish, you can start up the system without turning off anything by executing:

```
reboot
```

Shutting Down the System Completely

You must be a the system administrator with superuser capabilities or a designated user with superuser capabilities to shut down the system.

Caution Stopping the system improperly can corrupt (damage) the file systems. Never stop the system by turning it off.

1. Change to the root directory.

   `cd /`

2. Shut down the system. It is best to shut down the system in two steps:

   a. Execute:

      `shutdown 20` *This gets the system into the single-user state, allowing a 20 second grace period*

      The shutdown process asks if you want to send a message. If you elect to broadcast a message, respond with y and then type the message. When you finish, press [Return] (or [Enter]), and then [CTRL]-[d].

   b. Execute:

      `reboot -h` *This brings the system to a complete stop*

      You see several messages during the process. You should watch them to note actions and possible problems.

      You know the system is shut down completely when the system displays `halted` and pressing a key takes no action.

3. When the system is halted, turn the system off as follows:

   a. If you have only a computer (no expander), turn the computer off. Then, turn the devices off as required.

   b. If you have a computer and an expander, turn the computer off, turn the expander off, and then turn the devices off as required.

4. When you want to start up the system again, see the earlier procedure for starting up HP-UX.

Note From the multi-user state, you can also shut down the system completely by executing:

```
shutdown -h
```

This process is more harsh and sudden than the two-step procedure described above.

Shutting Down the System to Activate a New Kernel

You may want to shut down the system only to activate a new kernel. To do this, execute:

```
shutdown -r 0
```

The `-r` option causes the system to reboot immediately after the system gets into the single-user state.

Do not execute `shutdown -r` from run-level s. You must reboot using the `reboot` command.

Designating Shutdown Authorization

You can designate which users are authorized to run `shutdown` by listing these users in the file `/etc/shutdown.allow`. If this file is empty, only the superuser has shutdown authority, but if this file is not empty, and the superuser login (usually **root**) is not listed in the file, then the superuser will not be permitted to shut down the system. In an non-empty `shutdown.allow` file, only those users listed will have shutdown authority.

The following wildcards are allowed in the `shutdown.allow` file:

- `+` specifies that any host/user has shutdown authority.

For example:

- `systemA user1` allows user1 to shut down systemA.
- `+ root` allows root to shut down the system.
- `systemC +` allows any user to shut down systemC.

Customizing the Shutdown Process

You can customize the shutdown process by placing Bourne shell scripts in the file `/etc/shutdown.d`. These scripts will be executed in an ASCII (machine-collated) order. These scripts are optional, and are not required to run `shutdown`.

Using SAM to Shut Down the System

You can use SAM to shut down the system. You can access SAM's shut down capability:

3

1. Highlight `Routine Tasks->` and activate the [OK] control button.
2. Highlight `System Shutdown->` and activate the [OK] control button.

In SAM's `System Shutdown->` window you can choose to:

- Halt the system
- Reboot (restart) the system
- Go to single user state

Using SAM to Halt or Reboot the System

The key things to consider when you reboot or shut down the system are:

- Choosing a grace period after issuing a warning message.
- Broadcasting a message to the other users on your system to give them time to end their activities and log off.

1. Activate the control button that corresponds to your action in the "Shutdown Type" control box.
2. Choose and activate an appropriate grace period in the "Time Before Shutdown" control box.
3. Activate the [OK] control button.
4. Note SAM's confirmation note and activate the [Yes] control button to proceed.
5. Type in the message to issue to the users when SAM prompts you and proceed to shut down. You will have one more opportunity to discontinue the shutdown process.

Power Failure Considerations

A local power failure means a power failure that halts the computer by affecting its central bus. The HP-UX operating system (Series 800 computers only) provides a mechanism for recovery from a local power failure such that whatever was running on the system at the time of failure can resume executing when power to the bus is restored.

Remote power failures (affecting a remote bus) or device power failures (affecting a device) do not necessarily affect the system as a whole, unless the remote devices provide critical resources for the system.

Power Failure Related Tasks

If you know power is going out soon: Shut down the computer and turn off power.

If local power failure occurs: Do not turn off the computer or peripherals that contains vital system resources, but TURN OFF all other equipment that is not protected by power failure recovery software. An electrical surge as power is coming back on could seriously damage hardware that has been left turned on. When power is fully restored, turn on other equipment.

4

Controlling Access to Your System

It is rare to find a computer installation where everyone has access to *all* of the computer's files, commands and hardware resources. It is therefore likely that you will want to control who has access to your system, its data and its commands.

Authorized users gain access to the system by supplying a valid user name (login name) and password.

For additional information about the login process and the `/etc/passwd` file, refer to *login(1)* and *passwd(4)* in the *HP-UX Reference Manual* manual and Chapter 4, "Login" of *How HP-UX Works: Concepts for the System Administrator*, HP part number B2355-90029.

Terms Used in this Chapter

access permissions Values associated with each file that control who has permission to read, write (modify) or execute the file.

effective group If a user changes their default or primary group with the `newgrp` command, the new current group is the effective group (see **group** and **primary group** below).

group ownership The secondary ownership associated with each file, associating the file with a group (see **group**, below).

group Users on an HP-UX system can be grouped. If a group has access to a file, then any user who is defined as a member of that group will have access to the file. Users can be members of more than one group.

group_ID Also known as GID, is a unique number associated with each group (see **group**, above) that identifies the group to HP-UX. These group_ID numbers are defined in the `/etc/group` file.

log in Process used to gain access to the computer by supplying a user name and (if required) a password.

multi-user mode An HP-UX mode of operation that allows multiple users to access the system simultaneously. This is the "normal" mode of operation for HP-UX systems. See also **single-user mode**.

ownership Each file on the system has an owner. The owner controls access to the file by setting its access permissions. The owner is typically (but not always) the user who created the file.

primary group or default group A user can be a member of multiple groups, but only one of those groups is considered to be the user's primary or default group. In addition to being listed as a member of groups in the file `/etc/group`, an entry exists in the `/etc/passwd` file that indicates the user's primary group. When users first log into the system, they are affiliated with their primary group. Refer to the "Controlling User Accounts and Groups" and "Managing User Accounts and Groups Tasks" sections of this chapter for details.

run-level s See **single-user** mode.

run-level An HP-UX mode of operation. Modes of operation are defined in the file, `/etc/inittab`. The `/etc/inittab` file defines which terminals and processes are active at each run-level.

single-user mode A special HP-UX mode of operation that restricts user input to the system console. It is usually used by the system administrator to prevent others from accessing the system during special system administration activities when it is not advisable to have other system activity (for example, when updating the operating system to a new revision).

user_ID Also known as UID, is a unique number that HP-UX uses to identify a particular user. The user_ID number zero ("0") is used to identify the superuser. User_ID numbers between 1 and 99 are used by HP-UX subsystems. User_ID numbers above 99 are used for "normal" users.

user account The environment created on the system to allow the user access. Creating a user account involves updating the system to recognize the user's login name and password. You also need to give the user access to files, system resources, and applications.

Overview of Controlling Access Your System

Securing your data against deliberate, unauthorized access is only one reason for controlling access to your system. There are three levels of access control to your system. The following list of levels also includes reasons why you would want to control access at a particular level.

1. Controlling User Accounts and Groups

   By controlling who can log in to your system, you can prevent unauthorized users from running programs that consume valuable system resources (making them unavailable for the authorized users of your system). By creating and controlling groups of users, you can create unique group environments. Most systems are used for multiple purposes, and user groups allow you to customize according to multiple and varying group needs.

2. Controlling File Access

   By setting the appropriate access permissions for files and directories on your system, you can prevent them from being *accidentally* deleted or overwritten.

   By setting the appropriate ownership and group ownership for files (in addition to the file permissions) on your system, you can limit their use to specific users (or groups of users).

3. Controlling Run-Levels

   By configuring appropriate run-levels, you can activate different groups of terminals (and processes) for different situations (such as different work shifts).

Controlling User Accounts and Groups

Each user is defined by an entry in the file `/etc/passwd`. The `/usr/bin/vipw` command is the recommended editor for modifying the `/etc/passwd` file. The `vipw` command guarantees exclusive access to the `/etc/passwd` file. The `/etc/ptmp` file is created by the `vipw`, `chsh`, `chfn`, and `passwd` commands when access to the `/etc/passwd` file is granted. The `/etc/vipw` command requires the `EDITOR` environment variable set to `vi`.

If you are in single-user mode and the `vipw` command denies you access to the `/etc/passwd` file with an error message, "password file busy", delete the `/etc/ptmp` file and try the `vipw` command again. It is possible that the process that created this file terminated without removing this file. Refer to *vipw*(1M), *chsh*(1), *chfn*(1), *passwd*(1), and *passwd*(4) in the *HP-UX Reference Manual* manual for additional information.

Users on your system can be divided into various working groups, so that files owned by members of a given group can be shared and yet protected from access by users who are not members of the group. A user can be a member of *more than one group*. A group can have a maximum of 200 members.

If you prefer not to divide the users of your system into separate working groups, it is customary to set up one group (usually called "users") and assign all users of your system to that group.

Users may change their current group by using the `newgrp` command. The new group is referred to as the **effective group** for the user. Changing to an effective group does not alter the user's primary group entry in the `/etc/passwd` file. The user can return to their primary group by specifying no parameters or options to the `newgrp` command.

Group information is defined in `/etc/group` and `/etc/logingroup`, which are ASCII files that you can edit with a text editor such as `vi`.

`/etc/group` defines for each group:

- group name
- encrypted password (optional)
- numerical group identifier (*group_ID*)
- comma-separated list of group members by user login name

For example:

```
root:*:0:
other:*:1:
bin:*:2:
sys:*:3:
adm:*:4:
daemon:*:5:
mail:*:6:
lp:*:7:
users:*:20:john,naomil,patrickd,kerschen,michelem,dennism,pvallis
pub:*:24:patrickd,naomil,dennism
```

Note that a blank line in the `/etc/group` file is not allowed. If a blank line appears in the `/etc/group` file, all entries after the blank line are ignored.

`/etc/logingroup` contains the identical information, but the group name and encrypted password fields are not used. It is common practice to link the `/etc/group` and `/etc/logingroup` files together using the `link` command (refer to *link*(1M) in the *HP-UX Reference Manual*).

`/etc/group` is used by the `newgrp` command to check access privileges. If the user's login name appears in the access list of the group for which access is being requested, the access is granted thus changing the user's current group to the requested group. `/etc/logingroup` in contrast to `/etc/group` allows users listed in more than one group access to files belonging to other groups without changing their primary or effective groups.

For additional information about group related tasks, refer to the "Managing User Accounts and Groups Tasks" section of this chapter. For more details on the `/etc/group` and `/etc/logingroup` files, refer to the "Adding a Group Using HP-UX Commands" section of this chapter and *group*(4) in the *HP-UX Reference Manual* manual.

Primary Groups

A user can be a member of multiple groups, but only one of those groups is considered to be the user's primary or default group. In addition to being listed as a member of the group in the file `/etc/group`, an entry exists in the file `/etc/passwd`, indicating which group is the user's primary group. When users first log into the system, they are affiliated with their primary group.

To change the *primary* (default) group that your user is a member of, you will need to change user's entry in the `/etc/passwd` file to reflect a new *group_ID* value. The *group_ID* uniquely identifies an entry in `/etc/group` and `/etc/logingroup`. For instructions, see the "Displaying/Modifying a User's Account Information Using SAM" or "Changing a User's Primary Group Using HP-UX Commands" section of this chapter.

Group Passwords

When a user first logs into your system, their default or primary group affiliation is the one pointed to by the *group_ID* entry (fourth field in `/etc/passwd`). A user may be a member of more than one group. To change which group a user is affiliated with, a user can use the `newgrp` command. `newgrp` will require a password if the *group has a password* and the *user does not*, or if the *group has a password* and the *user is not listed as being a member of that group* (in the file `/etc/group`). If the user only needs to access files in another group, a entry in the `/etc/logingroup` would permit access to other group's files without changing the user's effective group. See *group*(4) in the *HP-UX Reference Manual* manual.

Note The use of group passwords is *not* encouraged, and they are rarely used. They encourage poor security practices.

Special Groups

Commands that permit access to all of the system's resources (and files) are usually restricted for use by superusers only. Although it is possible to have more than one superuser defined for your system (see "Adding a User Using SAM" or "Adding a User Using HP-UX Commands"), you may prefer to have only a subset of the superuser's capabilities available to a group of users. There are five types of **special privileges** that you can assign to a group of users using the `setprivgrp` command:

4

RTPRIO	controls group access to the `rtprio` command and system call that allow the setting of real-time priorities.
MLOCK	controls group access to the `plock` system call that allow processes to be locked in memory and allow the use of the `shmctl` system call SHM_LOCK parameter.
CHOWN	controls group access to the `chown` command and system call that allow changing the ownership of files on the system.
LOCKRDONLY	controls group access to the `lockf` system call that sets locks on files that are open for reading only.
SETRUGID	controls group access to the `setuid` and `setgid` system calls. The `setuid` system call changes the real user ID of a process and the `setgid` system call changes the real group ID of a process.

For additional information refer to *rtprio*(1), *rtprio*(2), *plock*(2), *shmctl*(2), *chown*(1), *chown*(2), *lockf*(2), *setuid*(2), *setgid*(2), *setprivgrp*(2) and *setprivgrp*(1M) in the *HP-UX Reference Manual* manual.

Any user whose current group_ID matches the group_ID of a privileged group will have access to the special capabilities assigned to that group. A group can have any one or a combination of the special privilege capabilities. Refer to the "Displaying/Assigning Special Group Privileges Using HP-UX Commands" section of this chapter for specific instructions.

Controlling File Access

All of the files on an HP-UX system have access permissions, ownership, and group ownership associated with them. Together the permissions and ownerships determine who can access the file.

File Access Permissions

There are three types (modes) of file access:

read — Determines who can view the file's contents. For directories, read access allows access to the directory with the `cd` command.

write — Determines who can alter the file's contents. For directories, write access allows modify and remove privileges.

execute — If the file is an *executable program*, the execute permissions determine who can run the program. For directories, execute access allows listing the directory contents.

There are three sources of file access:

Owner — The owner is usually the person who created the file (unless ownership has since been changed using the `chown` command).

Group — Members of the group that the file belongs to.

Other — All other users on your system.

There are three commands that change file access privileges:

- `chmod`

  The `chmod` command changes the type of access (read, write, and execute privileges) for every access source (owner, group, or other). For example, you can give the owner of the file read, write, and execute privileges, restrict group members to read and execute, and give only execute privileges to all other users on the system. Only the owner of a file (or the superuser) can change its read, write, and execute privileges.

- `chown`

  The `chown` command changes file ownership. In order to change the owner, you must own the file or have superuser privileges. Special group privileges determines a group's ability to use the `chown` command on files not owned by the user. The `setprivgrp` command controls special group privileges. To use the `setprivgrp` command, refer to the "Displaying/Assigning Special Group Privileges Using HP-UX Commands" section of this chapter.

- `chgrp`

  The `chgrp` command changes file group ownership. In order to change the group, you must own the file or have superuser privileges.

Refer to *A Beginner's Guide to HP-UX* or *chmod*(1), *chown*(1), and *chgrp*(1) in the *HP-UX Reference Manual* manual. Default file permissions are assigned by the system whenever you create a new file or directory, and these are governed by your `umask` setting. Unless set up otherwise by you or your system administrator, your default `umask` setting will be 0, which means that new *files* you create will have read/write permission for everyone (666 or -rw-rw-rw-) and new *directories* you create will have read/write/search permission for everyone (777 or `drwxrwxrwx`).

For additional information on file protection refer to Chapter 8, "HFS File System", in *How HP-UX Works: Concepts for the System Administrator*, HP part number B2355-90029. See also *ll*(1), *setprivgrp*(1M), and *umask*(1) in the *HP-UX Reference Manual* manual.

Access Control Lists

Access control lists (ACLs) offer a finer degree of file protection than traditional file-mode protection. With ACLs, you can allow or restrict file access to individual users, unrelated to what group the users belong to, with the `chacl` command. Only the owner of a file (or the superuser) can create ACLs with the `chacl` command.

Since you can use both the `chmod` and the `chacl` commands to change access permissions, you need to be aware of how the two commands interact.

- The `chacl` command is a superset of the `chmod` command. Any specific permissions you assign with the `chacl` command are added to the more general permissions assigned with the `chmod` command. For example, suppose you use the `chmod` command to allow only yourself write permission to `myfile`. You can use the `chacl` command to make an exception and allow your manager write permission to `myfile` also. Users other than yourself and your manager will still be denied write permission as previously specified by the `chmod` command.
- Use `chmod` with the `-A` option when working with files that have additional permissions assigned with the `chacl` command. The additional permissions will be deleted if you fail to use the `-A` option with `chmod`.

For additional ACL information see *lsacl*(1), *chacl*(1), and *acl*(5) in the *HP-UX Reference Manual* manual and Chapter 4, "Controlling User Access to Directories and Files", of *HP-UX System Security*, HP part number B1862-90009.

Controlling Run-Levels

A **run-level** is an HP-UX state of operation in which a specific set of processes (and their offspring) are permitted to run. This set of processes is defined for each run-level in a file called `/etc/inittab`.

Run-level 2 is the normal operating mode (often called **multi-user mode**). Users must log in to the system in order to gain access. Special processes called **getty**s run in this mode and post the "login" prompt on your system's terminals. When users log in to your system, the gettys initiate other processes (usually HP-UX shells) which in turn allow still other processes to be executed as users enter commands.

A special run-level called run-level "s" is also defined. Run-level "s" is a special system administration mode, called **single-user mode**. It is used for performing special tasks where it is desirable to have no one else on the system.

Run levels `s` and `2` are predefined. You can create new run-levels or change which processes can run at these predefined run-levels, if your needs require. You can define up to six run-levels (1-6). Most systems do not need to define additional run-levels, and modifications to the predefined run-level `2` are usually done to allow `getty` processes to run on terminals being added to a system.

To create a new run-level, make (or change) entries in the `/etc/inittab` file that define how you want the system to operate when the system is in that run-level. For information on the `/etc/inittab` file, refer to *inittab*(4) in the *HP-UX Reference Manual* manual.

Note When you use SAM to add terminals to your system, SAM makes the entries in the file `/etc/inittab` for you.

Only the superuser can use the `init` command, which changes the system from one run level to another, but anyone having write permission to the file `/etc/inittab` can create new run-levels or redefine existing run-levels.

To protect your system from tampering, ensure that the permissions (and ownership) for the files `/etc/init` and `/etc/inittab` are:

```
-r-xr-xr-x    root    other          /etc/init
-r--r--r--    root    root           /etc/inittab
```

See the "Creating a New Run-Level Using HP-UX Commands", "Changing System Run-Levels Using HP-UX Commands", "Entering the System Administration Run-Level", and "Returning From the System Administration Run-Level" sections of this chapter for specific instructions. For additional information refer to Chapter 6, "Run-Levels", in *How HP-UX Works: Concepts for the System Administrator*, HP part number B2355-90029.

Managing User Accounts and Groups Tasks

There are two ways to perform user account and user group tasks on your system:

- Using the System Administration Manager (SAM)
- Using HP-UX Commands (editing a series of files and creating user directories)

Generally you should use the SAM method because it is simpler and faster than performing the task with commands. For information about running SAM and navigating within SAM, refer to Chapter 1, "Introduction to System Administration".

SAM allows you to control access to your system through its menu-selection and data-entry screens. By combining multiple "manual commands" into single tasks, SAM can save you time and keystrokes. SAM also eliminates the need to know command names and options.

Although HP-UX commands require you to learn more details than SAM does, you might need or prefer to use HP-UX commands, for the following reasons:

- HP-UX commands give you a greater degree of control.
- SAM might not be configured into your system. You *have* to use HP-UX commands.
- You might be more comfortable using HP-UX commands.

The following are tasks covered in this section:

Adding a User

Removing a User

Customizing the SAM "Adding and Removing a User" Capabilities

Deactivating a User's Account

Reactivating a User's Account

Displaying/Modifying a User's Account Information

Adding a Group

Removing a Group

Changing a User's Primary Group

Adding Users to Groups

Removing Users From Groups

Displaying/Assigning Special Group Privileges

Each task has an ordered list of instructions, an area for additional information if necessary, and specific examples. In some of the HP-UX commands method examples the `xargs` command is used with the `find` command. Output from `find` is piped to `xargs` instead of using the `-exec` *primary* option to the `find` command. This is because when a large number of files or directories are to be processed by a single command, the `-exec` *primary* spawns a separate process for each file or directory, whereas `xargs` collects filenames or directory names into multiple arguments to a single `chgrp` or `chown` command, resulting in fewer processes and greater system efficiency. Specify the full pathname to the command following `xargs` to guarantee expected command behavior. Refer to *find*(1) and *xargs*(1) in the *HP-UX Reference Manual* manual for additional information.

Adding a User Using SAM

To add a user to your system:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

4

3. Highlight `Users and Groups->` and activate the [Open] control button.
4. Highlight `Users` and activate the [Open] control button.
5. Choose `Add ...` from the "Actions" menu.
6. Fill in the "Add a User Account" window fields and activate the [Apply] control button.
7. After reading the messages, activate the [OK] control button.

To return to the functional area list or functional subarea, choose `Exit` from the "List" menu.

Additional Task Information

SAM provides an on-line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Refer to "Customizing the SAM 'Adding and Removing a User' Capabilities" in this chapter for specific SAM customization instructions.

Note

Even though the user should be unique, there are a few circumstances where it is useful to have several `/etc/passwd` entries with the same user_ID number. For example, consider the following three `/etc/passwd` entries:

```
root:9WsbljlTvWbbw:0:3:Root User Account:/:/bin/sh
croot:NPt3HW.jBpVz2:0:3:Root User Account (C-Shell):/:/bin/csh
kroot:dGJbw/DBeDLdo:0:3:Root User Account (K-Shell):/:/bin/ksh
```

On the system with these entries in the `passwd` file, there are three separate accounts (`root`, `croot`, `kroot`) which have superuser capability. Depending on which one is used, the superuser will start up in either the Bourne Shell, the C Shell or the Korn shell.

Because all three accounts have the user_ID "0", the system views all three as the same user. When the system checks to see which user "owns" a file, it compares the "user_ID" associated with the file against the user_ID entries in the `/etc/passwd` file. When it does so, it scans the `/etc/passwd` file from beginning until it finds a user_ID match. This is why files created by the users `croot` and `kroot` (in the above example) will be listed (in the output of the `ll` command) as being owned by the user `root`. .

Removing a User Using SAM

To remove a user from your system using SAM:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.
3. Highlight `Users and Groups->` and activate the (Open) control button.
4. Highlight `Users` and activate the (Open) control button.
5. Choose `Remove ...` from the "Actions" menu.
6. Turn on the check box associated with the action regarding the user's files and activate the (OK) control button.
7. After reading the messages, activate the (OK) control button.

To return to the functional area list or functional subarea, choose `Exit` from the "List" menu.

Additional Task Information

SAM provides an on-line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Note

- SAM views a user as a specific *user_ID* (as opposed to a specific login name) and will not remove any user with the same *user_ID* as the superuser (*user_ID* 0). You should never remove the user called "root" from your system. If you have other superusers (users with the *user_ID* of zero) on your system you can remove them by simply removing their entry from the `/etc/passwd` file.
- If SAM detects that another user has the same UID (user_ID), SAM does not remove the user's files.
- SAM will *not* remove system directories, even if they are owned by a given user. And, SAM will *not* remove files across NFS mounts.
- SAM updates the `/etc/passwd` and `/etc/group` files, but does not update the `/etc/logingroup` file. If you use `/etc/logingroup`, edit the file to remove the user from all group entries.

Refer to "Customizing the SAM 'Adding and Removing a User' Capabilities" in this chapter for specific SAM customization instructions.

Customizing the SAM "Adding and Removing a User" Capabilities

To customize the procedure for adding and/or removing a user:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.
3. Highlight `Users and Groups->` and activate the [Open] control button.
4. Highlight `Users` and activate the [Open] control button.
5. Choose `Task Customization ...` from the "Actions" menu.
6. Fill in the script file name to be executed in one or a combination of the before/after adding/removing a user fields.
7. Activate the [OK] control button.
8. After reading the messages, activate the [OK] control button.

Additional Task Information

SAM provides an on-line help system to assist you when you need additional information.

Activating the [Help] button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the [f1] key gives you context-sensitive information for the object at the location of the cursor.

There are often additional steps (specific to your operations) that you may want to perform whenever you add or remove a user to your system. SAM allows you to set up shell scripts or executable programs which it will run for you before adding or removing the user, after adding or removing the user, or both before *and* after adding or removing the user.

There are strict permission and ownership requirements that must be followed for your custom script/program:

1. The file must be owned by root (group ownership not checked).

   Acceptable:

```
-r-xr--r--   1 root     bin          994 May  3 07:44 ct_addnode.ex
-r-xr--r--   1 root     other        994 May  3 07:44 ct_addnode.ex
```

   Unacceptable (areas that are highlighted):

```
-r-xr--r--   1 bin      bin          994 May  3 07:44 ct_addnode.ex
-r-xr--r--   1 joe      bin          994 May  3 07:44 ct_addnode.ex
```

2. The file must be writable and executable only by root (note that the file does not have to be writable, but if it is, it can only be writable by root).

   Acceptable:

```
-rwxr--r--   1 root     bin          994 May  3 07:44 ct_addnode.ex
-r-xr--r--   1 root     bin          994 May  3 07:44 ct_addnode.ex
-rwxr-----   1 root     bin          994 May  3 07:44 ct_addnode.ex
-r-x---r--   1 root     bin          994 May  3 07:44 ct_addnode.ex
-r-x------   1 root     bin          994 May  3 07:44 ct_addnode.ex
```

   Unacceptable (areas that are highlighted):

```
-rwxrw-rw-   1 root     bin          994 May  3 07:44 ct_addnode.ex
-rwxrw-r--   1 root     bin          994 May  3 07:44 ct_addnode.ex
-r-xr-xr-x   1 root     bin          994 May  3 07:44 ct_addnode.ex
-rwxr-xr--   1 root     bin          994 May  3 07:44 ct_addnode.ex
```

3. The file must reside in a directory path where all directories (that is, each directory in the directory path) are writable only by owner.

Suppose the custom command lies in directory `/usr/local/bin`. To successfully pass the validation, the permissions on `/usr`, `/usr/local`, and `/usr/local/bin` must all be "drwxr-xr-x". The permissions cannot be "drwxrwxr-x" or "drwxrwxrwx" (must be writable only by owner). This is typically a problem because `/usr/local` and `/usr/local/bin` are installed with permissions "drwxrwxrwx".

This means that the system administrators must take care in locating a directory (path) that meets the above requirements (`/usr`, `/usr/bin`, `/usr/sam`, `/usr/sam/bin`, `/usr/sam/config` are just a few examples of directories that at least meet the criteria when installed), or make one of their own that meets the requirements.

These restrictions are only applied at the time of SAM field validation of the command. Once SAM has registered a custom command to be used, the restrictions above are no longer checked by SAM for that command unless the user alters the custom script/program with SAM.

Deactivating a User's Account Using SAM

To deactivate a user's account:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

4

3. Highlight `Users and Groups->` and activate the [Open] control button.
4. Highlight `Users` and activate the [Open] control button.
5. Highlight the user entry in the object list you wish to deactivate.
6. Choose `Deactivate...` from the "Actions" menu.
7. Turn on the check box for the action regarding the user's files and activate the [OK] control button.
8. After reading the messages, activate the [OK] control button.

To return to the functional area list or functional subarea, choose `Exit` from the "List" menu.

Additional Task Information

SAM provides an on-line help system to assist you when you need additional information.

Activating the [Help] button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the [f1] key gives you context-sensitive information for the object at the location of the cursor.

Sometimes it is useful to temporarily suspend a user's ability to log into your system (such as when the user will be away for an extended period of time). The user's files remain on the system and intact, ready for the user when they return and you reactivate their account.

Deactivating a user's account means to make it so that *no login password is valid*.

Reactivating a User's Account Using SAM

To reactivate a user's account:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

4

3. Highlight `Users and Groups->` and activate the (Open) control button.
4. Highlight `Users` and activate the (Open) control button.
5. Highlight the user entry in the object list you wish to reactivate.
6. Choose `Reactivate...` from the "Actions" menu.
7. Optionally fill in a password for the user and activate the (OK) control button.
8. After reading the messages, activate the (OK) control button.

To return to the functional area list or functional subarea, choose `Exit` from the "List" menu.

Additional Task Information

SAM provides an on-line help system to assist you when you need additional information.

Activating the Help button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the f1 key gives you context-sensitive information for the object at the location of the cursor.

Displaying/Modifying a User's Account Information Using SAM

To display or modify a user's account information:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

3. Highlight `Users and Groups->` and activate the [Open] control button.
4. Highlight `Users` and activate the [Open] control button.
5. Highlight the user in the object list.
6. Choose `Modify...` from the "Actions" menu.
7. To view the user's information, activate the [Cancel] control button after gathering the information you need.

   To modify the user's information, fill in the new information in the "Modify a User" window and activate the [OK] control button. After reading the messages, activate the [OK] control button. The following user information can be modified:

   a. login name (*user_name*)
   b. password
   c. user identification number (*user_ID*)
   d. primary group identification number (*group_ID*)
   e. comment
   f. login directory
   g. start up program

To return to the functional area list or functional subarea, choose `Exit` from the "List" menu.

Additional Task Information

SAM provides an on-line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Adding a Group Using SAM

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.
3. Highlight `Users and Groups->` and activate the (Open) control button.
4. Highlight `Groups` and activate the (Open) control button.
5. Choose `Add...` from the "Actions" menu.
6. Fill in the new group name and optionally highlight the users to be members of the newly created group.
7. Activate the (OK) control button if this is the only group you are adding. Otherwise, activate the (Apply) and subsequent (OK) control buttons to return to the "Add a Group" window.

To return to the functional area list or functional subarea, choose `Exit` from the "List" menu.

Additional Task Information

SAM provides an on-line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Removing a Group Using SAM

To remove a group:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.
3. Highlight `Users and Groups->` and activate the Open control button.
4. Highlight `Groups` and activate the Open control button.
5. Choose `Remove ...` from the "Actions" menu.

   You can assign the group's files or another group if desired. Otherwise, SAM will reassign the group's files to the primary group of each of the file's owner.
6. After reading the messages, activate the OK control button.

To return to the functional area list or functional subarea, choose `Exit` from the "List" menu.

Additional Task Information

If the group SAM is removing is a user's primary group, SAM displays an error message and does not remove the group.

SAM provides an on-line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Refer to "Customizing the SAM 'Adding and Removing a User' Capabilities" in this chapter for specific SAM customization instructions.

Adding and Removing Users From Groups Using SAM

To modify a group's membership list:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.
3. Highlight `Users and Groups->` and activate the [Open] control button.
4. Highlight `Groups` and activate the [Open] control button.
5. Choose `Modify...` from the "Actions" menu.
6. Follow the instructions displayed in the "Modify a Group" window and to add and remove members to and from a group.
7. Activate the [OK] control button.

To return to the functional area list or functional subarea, choose `Exit` from the "List" menu.

Additional Task Information

SAM provides an on-line help system to assist you when you need additional information.

Activating the [Help] button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the [f1] key gives you context-sensitive information for the object at the location of the cursor.

Adding a User Using HP-UX Commands

Generally you should use the SAM method because it is simpler and faster than performing the task with commands. For information about running SAM and navigating within SAM, refer to Chapter 1, "Introduction to System Administration".

To add a user to your system using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Make a copy of the `/etc/passwd` file so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/passwd /etc/passwd.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the file using the command:

   ```
   cp /etc/passwd.old /etc/passwd
   ```

3. Create an entry in the file `/etc/passwd` for the new user using the text editor of your choice. HP recommends that you use the `vipw` command to ensure that you have exclusive access to the `/etc/passwd` file. The `/etc/vipw` command requires the `EDITOR` environment variable set to `vi`. Each user attribute must be colon-separated with no leading or trailing spaces. The one line entry must be in the form:

   user_name:*password*:*UID*:*GID*:*comment*:*login_directory*:*start_up_program*

   where:

   user-name — This is the user's login name (the one that they will enter at the `login:` prompt). The login name must have the following characteristics:

   - It must begin with an alphabetic character.
   - It can include up to eight alphanumeric characters.
   - It cannot contain blank spaces.
   - It cannot already exist on the system.

password

To ensure the user sets a password when they log in for the first time, place the characters ",.." in this field. For example:

```
bhewlett:,..:567:40:Bill Hewlett:/users/bhewlett:/bin/csh
```

Note

Putting an unencrypted password in the *password* field will *not* work. For example, if you want to assign a user the password "secret", the following entry will *not* allow the user to log in using the password "secret":

```
bhewlett:secret:567:40:Bill Hewlett:/users/bhewlett:/bin/csh
```

To set the password, use the `/bin/passwd` command.

To leave the new user's account without a password, do not put any characters between the colon (":") separators. For example:

```
dpackard::123:40:David Packard:/users/dpackard:/bin/csh
```

Caution

Leaving an account unprotected (without a password), even for a short period of time, is a security risk. If you entered ",.." in the *password* field, have the user log in as soon as possible to set a password for the account.

The `passwd` command is used to set or change the user's password.

UID

The *UID* (*user_ID*) is a *unique* integer value that the system uses to identify the user. If the *user_ID* is 0 (zero), then that user has superuser capabilities. When the system was shipped to you, the *user_ID* "0" was associated with the user `root`. By convention, the values 1 through 99 are reserved for system use. Therefore, when you are adding a new user to your system, pick for them any unused number greater than 99 (but less than 60000) for this field. *user_IDs* greater than 59999 are invalid.

GID

This value is the user's primary group *GID* (*group_ID*) as defined in the third field of the user's primary group entry in the `/etc/group` file. The *group_ID* is an integer value shared by all members of the same group. Refer to the "Adding a Group Using HP-UX Commands" section of this chapter for a description of the `/etc/group` and `/etc/logingroup` file formats.

comment

The *comment* field is used to log information about the identity of the user (or to identify this entry). Although this field is "free-format", using the following comma-separated subfield format is recommended:

User's Full Name, Office Location, Office Phone, Home Phone

login_directory

This is the absolute pathname to the directory that the user will be placed in when they first log in to the system. The directory need not exist when the entry to `/etc/passwd` is made. However, the directory *must* exist before the user can log in. This field can be no longer than 63 characters.

start_up_program

This field contains the name of a single command to be executed for the user when they log in; it should be an absolute pathname (for the command). This field can be no longer than 44 characters. Typically this is the name of a shell (`/bin/sh`, `/bin/csh`, `/bin/ksh`, etc.). However, the name can be that of any executable program or command. The command can be either a compiled program or a shell script, but no arguments to the command or script should be supplied. If the command field is left blank, `/bin/sh` is executed by default.

When the user logs in, the command listed in this field is executed and control is passed to that program. Once the program terminates, the user is logged out.

4. Create a login ("home") directory for the user with the `mkdir` command:

   `/bin/mkdir` [`-p`] [`-m` *mode*] *directory*

   where:

`-p`	specifies intermediate directories are created as necessary. Otherwise, the full path prefix of *directory* must already exist. The `mkdir` command requires write permission in the parent directory.
`-m` *mode*	specifies creating the directory as specified by *directory* with the file permissions are set to *mode*, which is a symbolic mode string.
directory	specifies the user login directory.

   The **login directory** is the directory that the users are *first* placed in when they log in to the system.

   The login directory that is defined for a user in the `/etc/passwd` file *must* exist when the user logs into the system or the login attempt will fail.

   It is not necessary for each user to have a separate login directory, but this is how systems are usually set up. It is easier to keep the files of the various users separated if each has their own login directory. This, in turn, makes it easier to work with a given user's files (for example when doing backups, determining how much disk space a given user is using, etc.).

5. Use the `pwck` command to verify that the `/etc/passwd` file has valid entries:

   `/etc/pwck`

   The `pwck` command validates the number of fields, login name, user ID, group ID, and whether the login directory and optional program name exist. Refer to *pwck*(1M) in the *HP-UX Reference Manual* manual for additional information.

6. Create login (shell initialization) files for the user.

   In most cases, the *start_up_program* for a user will be one of the HP-UX shells (`/bin/sh`, `/bin/csh`, `/bin/ksh`, `/bin/rsh`, etc.). Each of these shells has a set of initialization files that they read when they begin executing as a user is logging in to the system. Table 4-1 lists the initialization file executed for each shell in the order they are executed.

Table 4-1. HP-UX Global and Local Shell Initialization Files

Shell	Initialization Files Executed at Login
/bin/sh	`/etc/profile` `$HOME/.profile`

/bin/csh	`/etc/csh.login`
	`$HOME/.cshrc`
	`$HOME/.login`

/bin/ksh	`/etc/profile` `$HOME/.profile`
/bin/keysh	`/etc/profile` `$HOME/.profile`
/bin/rsh	`/etc/profile` `$HOME/.profile`
/bin/rksh	`/etc/profile` `$HOME/.profile`

The `/etc/profile` and `/etc/csh.login` files contain global instructions/commands that you want executed for *every* user who logs into the system. These files are in the `/etc` directory so that they are accessible to all users. You should *not* copy them to each user's directory. The local initialization files are located in the user's login directory ($HOME). These local files typically contain shell commands and environment variable definitions that customize the user's environment and/or automatically run one or more programs for the user.

If the local initialization files exist in a user's directory, the shell attempts to execute the commands in the local files after completing the global files, but before the user receives the first shell prompt.

Examples of the local initialization files are located in the `/etc` directory. Their names begin with the characters "`d.`" (for example `d.profile`). You may copy these files to a user's login directory and customize them. When you copy these file to the user's login directory, rename the files without the `d` prefix.

7. Create or customize other initialization files for the user.

   Other programs such as the editor, `vi` and the various mail programs (`mail`, `elm`, `mailx`) have initialization files which you may also want to set up for the user.

   You may need to change the access permissions of particular files within the user's login directory. Refer to *A Beginner's Guide to HP-UX*. For additional information on file protection refer to Chapter 8, "HFS File System", in *How HP-UX Works: Concepts for the System Administrator*, HP part number B2355-90029. See also *ll*(1), *chmod*(1), *chown*(1), *chgrp*(1), and *umask*(1) in the *HP-UX Reference Manual* manual.

8. Change the *file ownership* and *group ownership* of the new user's home directory and the files to the user's login name and primary group with the `chown` and `chgrp` commands respectively. You must change the ownership and group ownership of these new files to those of your new user so that *the new user* can access them.

   Change the *file ownership* of a file or directory with the `chown` command:

   `/bin/chown` [`-R`] *new_owner login_dir*

   where:

   `-R` specifies to recursively change the file ownership to *new_owner*. For each *login_dir*, the owner of the directory and all files and subdirectories in the file hierarchy below it are changed to *new_owner*.

   new_owner specifies the login name of the new user.

   login_dir specifies the login directory of the new user.

   Change the *group ownership* of a file or directory with the `chgrp` command:

   `/bin/chgrp` [`-R`] *new_group login_dir*

   where:

   `-R` specifies to recursively change the group ownership to *new_group*. For each *login_dir*, the group of the directory and all files and subdirectories in the file hierarchy below it are changed to *new_group*.

   new_group specifies the primary group of the new user.

   login_dir specifies the login directory of the new user.

9. Edit the `/etc/group` file, and optionally, the `/etc/logingroup` file to add the user's *user_name* to the names in the comma-separated list of members for the group(s). If you want the user to be able to access files belonging to another group other than their primary group without using the `chgrp` command, edit the `/etc/logingroup` file to add the user to all necessary groups.

Note

- A blank line in the `/etc/group` or `/etc/logingroup` file is not allowed. If a blank line appears in the files, all entries after the blank line are ignored. See "Adding Users to Groups Using HP-UX Commands" for specific instructions on editing the `/etc/group` and `/etc/logingroup` files.
- A group can have a maximum limit of 200 users.

---

10. Use the `grpck` command to check for inconsistencies and verify all entries in the `/etc/group` and `/etc/logingroup` files. This verification includes a check of the number of fields, group name, group ID, and whether all login names appear in the password file. The `grpck` command has the following format:

    `/etc/grpck` [*group_file*]

    where:

    group_file is the name of the group file to be checked. The default group file is `/etc/group`.

11. Have the new user log into the system so that you can verify that you have set up their environment correctly.

Additional Task Information

Refer to "Controlling User Accounts and Groups" for more information about the `vipw` command.

If the *comment* field information in the `/etc/passwd` file is entered in a comma-separated sub-field format, you can use the `/usr/bin/finger` command to display this information. If you need to modify the user's *comment* field information, you can modify the `/etc/passwd` file directly with the `vipw` command or you can use the `/usr/bin/chfn` command. The information in the *comment* field is referred to as "gecos" information. Refer to the "Displaying/Modifying User's Account Information Using HP-UX Commands" section of this chapter and/or *finger*(1) and *chfn*(1) in the *HP-UX Reference Manual* manual for additional information.

Note

Even though the user should be unique, there are a few circumstances where it is useful to have several `/etc/passwd` entries with the same user_ID number. For example, consider the following three `/etc/passwd` entries:

```
root:9WsbljlTvWbbw:0:3:Root User Account:/:/bin/sh
croot:NPt3HW.jBpVz2:0:3:Root User Account (C-Shell):/:/bin/csh
kroot:dGJbw/DBeDLdo:0:3:Root User Account (K-Shell):/:/bin/ksh
```

On the system with these entries in the `passwd` file, there are three separate accounts (`root`, `croot`, `kroot`) which have superuser capability. Depending on which one is used, the superuser will start up in either the Bourne Shell, the C Shell or the Korn shell.

Because all three accounts have the user_ID "0", the system views all three as the same user. When the system checks to see which user "owns" a file, it compares the "user_ID" associated with the file against the user_ID entries in the `/etc/passwd` file. When it does so, it scans the `/etc/passwd` file from beginning until it finds a user_ID match. Files created by the users `croot` and `kroot` (in the above example) will be listed (in the output of the `ll` command) as being owned by the user `root` because `root` is the first of the three entries.

If you have several users sharing a login directory, the ownerships and permissions of the shell local initialization files may need to be adjusted so that all users sharing that login directory have read access to the local initialization files. There are several ways to do this. One way is to assign one of the users sharing the login directory to be the "owner" of the files, have all of the users sharing the directory be members of the same group, and give the group members read access to the files. For information about setting file permissions and ownership, see *A Beginner's Guide to HP-UX*. For additional information on file protection refer to Chapter 8, "HFS File System", in *How HP-UX Works: Concepts for the System Administrator*, HP part number B2355-90029. See also *ll*(1), *chmod*(1), *chown*(1), *chgrp*(1), *setprivgrp*(1M), and *umask*(1) in the *HP-UX Reference Manual* manual.

Another way is to create ACLs (Access Control Lists) for the startup files. ACLs allow access control at the user level versus the group level. For additional ACL information see *lsacl*(1), *chacl*(1), and *acl*(5) in the *HP-UX Reference Manual* manual and Chapter 4, "Controlling User Access to Directories and Files", of *HP-UX System Security*, HP part number B1862-90009.

Examples

To add new user accounts for *john*, *patrickd*, and *naomil* to the system:

1. Login as `root`.
2. Make a backup copy of the `/etc/passwd` file.

   ```
   cp /etc/passwd /etc/passwd.old
   ```

3. Update the `/etc/passwd` file to include the users' entries. HP recommends using the `/etc/vipw` command. The `/etc/vipw` command requires the `EDITOR` environment variable set to `vi`.

   To set the Korn and Bourne shell `EDITOR` environment variable, type:

   ```
   export EDITOR=vi
   ```

   To set the C shell `EDITOR` environment variable, type:

   ```
   setenv EDITOR vi
   ```

   Add the following entries to the `/etc/passwd` file:

   ```
   john:,..:342:20:John Smith, 125 Elm Street, 555-2324:/users/john:/bin/ksh
   naomil:,..:1667:20:Naomi Adams,540 Market Ave, 555-9078:/users/naomil:/bin/ksh
   patrickd:,..:24:Patrick Daly,421 Orange Road, 555-6140:/users/patrickd:/bin/ksh
   ```

   Note the primary group for *john* and *naomil* is `users` while the primary group for *patrickd* is pub.

4. Create a login directory for each of the new users:

   ```
   mkdir -p /users/john
   mkdir -p /users/patrickd
   mkdir -p /users/naomil
   ```

5. Check the `/etc/passwd` file format:

   ```
   pwck
   ```

6. Copy local initialization files to each user's login directory:

   ```
   cp /etc/d.profile /users/john/.profile
   cp /etc/d.profile /users/patrickd/.profile
   cp /etc/d.profile /users/naomil/.profile
   ```

7. Create or customize initialization files for the users.

8. Change the ownership and permissions of all of the files and subdirectories created in the login directories for *john*, *patrickd*, and *naomil* using the `chown` and `chgrp` commands with the `-R` option:

```
chown -R john /users/john
chown -R patrickd /users/patrickd
chown -R naomil /users/naomil
chgrp -R users /users/john
chgrp -R pub /users/patrickd
chgrp -R users /users/naomil
```

Check that ownership and permissions are correct in the new login directories use the `ll` command:

```
ll /users
drwxr-xr-x  17 john      users       2048 Feb  6 11:18 john
drwxr-xr-x   7 naomil    users       3072 Feb  5 15:57 naomil
drwxr-xr-x   9 patrickd  users       5152 Feb  5 18:07 patrick
```

If you need to globally change the default owner, group, or other permissions on the files and subdirectories created for the new users, use the `chmod` command with the `-R` option. For example:

```
chmod -R u=rwx,g=x,o= /users/john/
```

or

```
chmod -R 710 /users/john/
```

For additional information about the `chown`, `chgrp`, and `chmod` commands, refer to *A Beginner's Guide to HP-UX*. See also *chown*(1), *chgrp*(1), and *chmod*(1) in the *HP-UX Reference Manual* manual.

9. Update the `/etc/group` file, and optionally, the `/etc/logingroup` file to add the three users' login names to each users' primary group member list. Additionally, add *naomil* to the `pub` group and *patrickd* to the `users` group. For example:

```
root:*:0:
other:*:1:
bin:*:2:
sys:*:3:
adm:*:4:
daemon:*:5:
mail:*:6:
lp:*:7:
users:*:20:john,naomil,patrickd,kerschen,michelem,dennism,pvallis
pub:*:24:patrickd,naomil,dennism
```

   Note that a blank line in the `/etc/group` file is not allowed. If a blank line appears in the `/etc/group` file, all entries after the blank line are ignored.

   Users *patrickd* and *naomil* may access files belonging to both groups without changing their current group if the `/etc/logingroup` file has entries for both users in both groups. Otherwise, the `chgrp` command will be necessary for *naomil* and *patrickd* to access files belonging to another group.

10. Check the `/etc/group` file format:

```
/etc/grpck
```

11. Instruct users *patrickd*, *naomil*, and *john* to log in.

Removing a User Using HP-UX Commands

Generally you should use the SAM method because it is simpler and faster than performing the task with commands. For information about running SAM and navigating within SAM, refer to Chapter 1, "Introduction to System Administration".

To remove a user from your system using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Decide the future of the user's files and directories. Your choices are:

   a. Remove the user's files and directories from the system
   b. Assign the user's files and directories to another user
   c. A combination of the above.

   Use the `find` command to get a list of the files on your system which are owned by the user you are removing:

   `/bin/find / -fsonly hfs -user` *user_name* `-print`

   where:

   user_name is the user's login name as defined in the user's `/etc/passwd` file entry.

   Files, especially those representing executable programs, can be shared among users in a group or among all of the users of the system. If you decide to remove the user's files, be sure that they will not be needed by other users of your system. Refer to *find*(1) in the *HP-UX Reference Manual* manual for additional information.

3. Find and remove all ACL (Access Control List) entries for the user. To find and remove all of the ACL entries for the user `naomil` in the `users` group type:

```
/bin/find / -fsonly hfs -acl naomil.users -depth -print | xargs chacl -d naomil.users
```

   For additional ACL information see *lsacl*(1), *chacl*(1), and *acl*(5) in the *HP-UX Reference Manual* manual and Chapter 4, "Controlling User Access to Directories and Files", of *HP-UX System Security*, HP part number B1862-90009.

4. Search for and remove the user's login name from all entries in the `/etc/group` file. Optionally remove the user's login name from all entries in the `/etc/logingroup` file if it exists.

   Use the command `grep` (or the search command in your text editor) to find the entries in `/etc/group` that contain the *user_name* (login name) belonging to the user you are removing.

   Using a text editor, delete the *user_name* from those entries.

5. Use the `grpck` command to check for inconsistencies and verify all entries in the `/etc/group` and `/etc/logingroup` files. This verification includes a check of the number of fields, group name, group ID, and whether all login names appear in the password file. The `grpck` command has the following format:

   `/etc/grpck` [*group_file*]

   where:

   group_file is the name of the group file to be checked. The default group file is `/etc/group`.

6. Make a copy of the `/etc/passwd` file so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/passwd /etc/passwd.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the file using the command:

   ```
   cp /etc/passwd.old /etc/passwd
   ```

7. Remove the user's entry in the `/etc/passwd` file using the `vipw` command.

8. Use the `pwck` command to verify that the `/etc/passwd` file has valid entries:

   ```
   /etc/pwck
   ```

   The `pwck` command validates the number of fields, login name, user ID, group ID, and whether the login directory and optional program name exist. Refer to *pwck*(1M) in the *HP-UX Reference Manual* manual for additional information.

Additional Task Information

Note You should never remove the user called "root" from your system.

Refer to "Controlling User Accounts and Groups" for more information about the `vipw` command.

Examples

To remove user `michelem` from the system:

1. Login as `root`.
2. Change file ownership of all files in the login directory for `michelem` to `rykl` and remove *all* of the files on the rest of the system owned by user `michelem`.

   To change file ownership of *all* files and directories in the `/users/michelem` login directory to `rykl`:

   ```
   /bin/find /users/michelem -fsonly hfs -user michelem -depth -print | xargs chown -R rykl
   ```

   To remove files owned by `michelem` from the system:

   ```
   /bin/find / -fsonly hfs -user michelem !-type d -depth -print | xargs rm
   ```

   To remove *all* of the empty directories owned by user `michelem`:

   ```
   /bin/find / -fsonly hfs -user michelem -depth -print | xargs rmdir
   ```

3. Remove `michelem` from any Access Control List entries (ACLs):

   ```
   /bin/find / -fsonly hfs -acl michelem.users -depth -print | xargs chacl -d michelem.users
   ```

4. Locate and remove the login name for the user `michelem` from all entries in the `/etc/group` file:

   ```
   grep michelem /etc/group
   ```

   The output might look like this:

   ```
   photo:*:23:dennisp,janetn,michelem,stevens
   therapy:*:23:kimz,michelem,bsmith
   database:*:23:michelem,lynnf,rykl
   ```

   After removing `michelem` from the group member lists, the updated `/etc/group` file entries should look like this:

   ```
   photo:*:23:dennisp,janetn,stevens
   therapy:*:23:kimz,bsmith
   database:*:23:lynnf,earlg
   ```

5. Check the `/etc/group` file format:

   ```
   /etc/grpck
   ```

6. Make a backup copy of the `/etc/passwd` file.

   ```
   cp /etc/passwd /etc/passwd.old
   ```

7. Update the `/etc/passwd` file to delete the line containing the information for user `michelem`. HP recommends using the `/etc/vipw` command. The `/etc/vipw` command requires the `EDITOR` environment variable set to `vi`.

   To set the Korn and Bourne shell `EDITOR` environment variable, type:

   ```
   export EDITOR=vi
   ```

   To set the C shell `EDITOR` environment variable, type:

   ```
   setenv EDITOR vi
   ```

8. Check the `/etc/passwd` file format:

   ```
   pwck
   ```

Deactivating a User's Account Using HP-UX Commands

To deactivate a user's account:

1. Ensure that you have superuser capabilities.
2. Make a copy of the `/etc/passwd` file so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/passwd /etc/passwd.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the file using the command:

   ```
   cp /etc/passwd.old /etc/passwd
   ```

3. Edit the `/etc/passwd` file using the `vipw` command:
   a. Locate the entry in the `/etc/passwd` file that corresponds to the user's account that you are planning to deactivate.
   b. Replace the encrypted password in the second field with an asterisk "*" (fields are separated by colons ":").
   c. Save the `/etc/passwd` file and exit the editor.
4. Use the `pwck` command to verify that the `/etc/passwd` file has valid entries:

   ```
   /etc/pwck
   ```

   The `pwck` command validates the number of fields, login name, user ID, group ID, and whether the login directory and optional program name exist. Refer to *pwck*(1M) in the *HP-UX Reference Manual* manual for additional information.

Additional Task Information

Note You should never deactivate the user called "root".

Deactivating a user's account means to make it so that *no login password is valid*.

Sometimes it is useful to temporarily suspend a user's ability to log into your system (such as when the user will be away for an extended period of time). The user's files remain on the system and intact, ready for the user when they return and you reactivate their account.

Refer to "Controlling User Accounts and Groups" for more information about the `vipw` command.

Examples

To deactivate the user account for *paul*, edit the `/etc/passwd` file. HP recommends using the `/etc/vipw` command. The `/etc/vipw` command requires the `EDITOR` environment variable set to `vi`.

To set the Korn and Bourne shell `EDITOR` environment variable, type:

```
export EDITOR=vi
```

To set the C shell `EDITOR` environment variable, type:

```
setenv EDITOR vi
```

The `/etc/passwd` file entry before deactivating user *paul*:

```
paul:sIgNXHLuFptVE:209:20:Paul Avonette,Mailstop F13,555-7086,:/users/paul:/bin/ksh
```

The `/etc/passwd` file entry after deactivating user *paul*:

```
paul:*:209:20:Paul Avonette,Mailstop F13,555-7086,:/users/paul:/bin/ksh
```

Check the `/etc/passwd` file format:

```
pwck
```

Reactivating a User's Account Using HP-UX Commands

To reactivate a user's account:

1. Ensure that you have superuser capabilities.
2. Make a copy of the `/etc/passwd` file so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/passwd /etc/passwd.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the file using the command:

   ```
   cp /etc/passwd.old /etc/passwd
   ```

3. Edit the `/etc/passwd` file using the `vipw` command:
   a. Locate the entry in the `/etc/passwd` file that corresponds to the user's account that you are planning to reactivate.
   b. Replace the asterisk "*" in the second field of the file with the string ",.." (comma-dot-dot) which forces the user to set a new password for their account the next time they log in.
   c. Save the `/etc/passwd` file and exit the editor.
4. Use the `pwck` command to verify that the `/etc/passwd` file has valid entries:

   ```
   /etc/pwck
   ```

   The `pwck` command validates the number of fields, login name, user ID, group ID, and whether the login directory and optional program name exist. Refer to *pwck*(1M) in the *HP-UX Reference Manual* manual for additional information.

Additional Task Information

Refer to "Controlling User Accounts and Groups" for more information about the `vipw` command.

Examples

To reactivate the user account for *paul*, edit the `/etc/passwd` file. HP recommends using the `/etc/vipw` command. The `/etc/vipw` command requires the `EDITOR` environment variable set to `vi`.

To set the Korn and Bourne shell `EDITOR` environment variable, type:

```
export EDITOR=vi
```

To set the C shell `EDITOR` environment variable, type:

```
setenv EDITOR vi
```

The `/etc/passwd` file entry before reactivating user *paul*:

```
paul:*:209:20:Paul Avonette,Mailstop F13,555-7086,:/users/paul:/bin/ksh
```

The `/etc/passwd` file entry after reactivating user *paul*:

```
paul:,,,:209:20:Paul Avonette,Mailstop F13,555-7086,:/users/paul:/bin/ksh
```

Check the `/etc/passwd` file format:

```
pwck
```

Displaying/Modifying User's Account Information Using HP-UX Commands

To display a user's account information:

- Use the `finger` command to display the user's `/etc/passwd` file information:

  `/usr/bin/finger` *user_name*

  where:

  user_name is the user's login name as defined in the user's `/etc/passwd` file entry.

  or

- Use the `grep` command to display the user's `/etc/passwd` file information:

  `/bin/grep` *user_name* `/etc/passwd`

  where:

  user_name is the user's login name as defined in the user's `/etc/passwd` file entry.

To modify a user's account information:

1. Ensure that you have superuser capabilities.
2. Make a copy of the `/etc/passwd` file so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/passwd /etc/passwd.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the file using the command:

   ```
   cp /etc/passwd.old /etc/passwd
   ```

3. Edit the `/etc/passwd` file using the `vipw` or `chfn` commands to update the following user information:

   - login name (*user_name*)
   - password
   - user identification number (*user_ID*)
   - primary group identification number (*group_ID*)
   - comment
   - login directory
   - start up program

   For a description of these fields, refer to the "Adding a User Using HP-UX Commands" section of this chapter.

   The `chfn` command has the following syntax:

   `/usr/bin/chfn` [*user_name*]

   where:

   user_name is the user's login name as defined in the user's `/etc/passwd` file entry.

   You must have superuser capabilities to use the `chfn` command to update other users' account information, but you can change your own account information without superuser capabilities.

4. Use the `pwck` command to verify that the `/etc/passwd` file has valid entries:

   `/etc/pwck`

   The `pwck` command validates the number of fields, login name, user ID, group ID, and whether the login directory and optional program name exist. Refer to *pwck*(1M) in the *HP-UX Reference Manual* manual for additional information.

Additional Task Information

If you do not have superuser capabilities and you attempt the use the `chfn` command to update another user's account information, the error message "You are not allowed to change another person's finger entry." is displayed.

Refer to "Controlling User Accounts and Groups" for more information about the `vipw` command.

Examples

To display account information for user `jdoe`:

```
finger jdoe

Login name:  jdoe                          In real life:  John Doe
Bldg: Building 5
Directory: /users/jdoe                     Shell: /bin/ksh
On since Feb 10 11:17:04 on pty/ttys5 from mountian.net.ca
2 minutes 25 seconds Idle Time
No Plan.

grep jdoe /etc/passwd
jdoe:QAJZL4Xjg/BMM:1667:20:John Doe,Building 5,555-1234:/users/jdoe:/bin/ksh
```

To update the telephone number for user `jdoe` with the `chfn` command:

```
chfn jdoe
Default values are printed inside of of '[]'.
To accept the default, type <return>.
To have a blank entry, type the word 'none'.

Name [John Doe]:
Location (Ex: 42U-J4) [Building 5]
Office Phone (Ex: 1632) [1234]: 2233
Home Phone (Ex: 5555678) []:
```

Run `pwck` to check your `/etc/passwd` file format.

Adding a Group Using HP-UX Commands

To add a new group to your system using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Make a copy of the `/etc/group` file, and optionally, the `/etc/logingroup` file so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/group /etc/group.old
   cp /etc/logingroup /etc/logingroup.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the files using the commands:

   ```
   cp /etc/group.old /etc/group
   cp /etc/logingroup.old /etc/logingroup
   ```

3. Create an entry for the new group in the `/etc/group` file with the editor of your choice. Optionally create an entry in the `/etc/logingroup` file to allow access to files belonging to other groups without changing the users' effective group. The `/etc/group` and `/etc/logingroup` one-line entries must have the following format:

   group_name:*group_password*:*group_ID*:*user1[,user2][,user3]...*

   where:

group_name	This is the name of your new group. It must begin with an alphabetic character and can include up to 16 alphanumeric characters.
group_password	It is recommended that you put an asterisk "*" in this field, which indicates that you will not be using group passwords.
group_ID	Like the *user_ID* field in the file `/etc/passwd`, the *group_ID* is a *unique* integer, which is used by HP-UX to identify the group.
user1,...	This is a list of comma-separated *user_names* (from the first field of the entries in the `/etc/passwd` file).

4. Use the `grpck` command to check for inconsistencies and verify all entries in the `/etc/group` and `/etc/logingroup` files. This verification includes a check of the number of fields, group name, group ID, and whether all login names appear in the password file. The `grpck` command has the following format:

   `/etc/grpck` [*group_file*]

   where:

   group_file is the name of the group file to be checked. The default group file is `/etc/group`.

Additional Task Information

The `/etc/group` file is used by the `newgrp` command to check access privileges when a user is attempting to change their effective group. If the user's login name appears in the access list of the group for which access is being requested, the access is granted, thus changing the user's current group to the requested group. The `/etc/logingroup` file, in contrast to `/etc/group`, allows users listed in more than one group access to files belonging to other groups that they are members of without changing their primary or effective groups.

Note

- A blank line in the `/etc/group` or `/etc/logingroup` file is not allowed. If a blank line appears in the files, all entries after the blank line are ignored. Refer to the "Adding a Group Using HP-UX Commands" section of this chapter for a description of the `/etc/group` and `/etc/logingroup` files.
- A group can have a maximum limit of 200 users.

Optionally, add user entries to Access Control Lists (ACL). For additional ACL information see *lsacl*(1), *chacl*(1), and *acl*(5) in the *HP-UX Reference Manual* manual and Chapter 4, "Controlling User Access to Directories and Files", of *HP-UX System Security*, HP part number B1862-90009.

Examples

The following listing is a sample /etc/group file:

```
therapy:*:20:dennisp,janetn,jdoe,stevens
photo:*:30:kimz,jdoe,bsmith
leader:*:59:blink,pgomez,stevens
manager:*:67:obones,jab,mlee,fjones
```

To add user group "users":

4

1. Login as root.
2. Make a backup copy of /etc/group and /etc/logingroup:

   ```
   cp /etc/group /etc/group.old
   cp /etc/logingroup /etc/logingroup.old
   ```

3. Create an entry in the /etc/group file for user group "users". For example:

   ```
   therapy:*:20:dennisp,janetn,jdoe,stevens
   users:*:23:michelem,rykl,karens,starsky,stevens
   photo:*:30:kimz,jdoe,bsmith
   leader:*:59:blink,pgomez,stevens
   manager:*:67:obones,jab,mlee,fjones
   ```

4. Check the /etc/group file format:

   ```
   /etc/grpck
   ```

5. Edit the /etc/logingroup file to enable user stevens to access files belonging to groups therapy leader, and users without changing effective groups:

   ```
   more /etc/logingroup
   therapy:*:20:stevens
   users:*:23:stevens
   leader:*:59:stevens
   ```

6. Check the /etc/logingroup file format:

   ```
   grpck /etc/logingroup
   ```

Removing a Group Using HP-UX Commands

To remove a group from your system:

1. Ensure that you have superuser capabilities.
2. Make a copy of the `/etc/group` file, and optionally the `/etc/logingroup` file, so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/group /etc/group.old
   cp /etc/logingroup /etc/logingroup.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the files using the commands:

   ```
   cp /etc/group.old /etc/group
   cp /etc/logingroup.old /etc/logingroup
   ```

3. Make a copy of the `/etc/passwd` file so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/passwd /etc/passwd.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the file using the command:

   ```
   cp /etc/passwd.old /etc/passwd
   ```

4. Use Table 4-2 to determine the next sequence of steps based on the status of the group members and their files.

   For each user that is a member of the group being removed, perform the steps that correspond to the matching combination of actions taken regarding the user account (table row) and files (table column).

Table 4-2. "Removing a Group" Task Decision Table

	Delete User Files	Keep User Files
Delete User(s) From System	• For files matching the user and group ownership, remove all files and directories from the system using `find`, `rm`, and `rmdir`. • For the remaining files and directories owned by each user, remove them or reassign file ownership. Remove the files and directories using `find`, `rm`, and `rmdir`. Reassign file ownership using `find` and `chown`. • Edit `/etc/passwd` to remove user entries.	• For all files on the system owned by the user being removed with the group, change file and group ownership using `find`, `chown`, and `chgrp`. • Edit `/etc/passwd` to remove user entries.
Keep User(s) on System	• If the group to be removed is a user's primary group, edit the `/etc/passwd` file to reassign user(s) to a new primary group. • For each user assigned a new primary group, change the group ownership of the login directories using the `chown` command. • For each user assigned a new primary group, if there are files within the login directory that are to be preserved, change their group ownership to match the user's new primary group using `chgrp`. • For each user that is a member of the group being removed, delete the files owned by the user and belonging to the group being removed using `find`, `rm`, and `rmdir`.	• If removed group is a primary group for a user, edit the `/etc/passwd` file to reassign user(s) to a new primary group. • For each user assigned a new primary group, change the group ownership of the login directories using the `chown` command. • If the user *is* to retain access to the files belonging to the group being removed, change the group ownership of files owned by the user to a group that the user remains a member of using `chgrp`. • If the user is *not* to retain access to the files belonging to the group being removed, change the file ownership to a user that is *not* a member of the original owner's group using `chown`. For the files changing owner, change the group ownership to new owner's group using `chgrp`.

Use the `find` command to globally search the system for files with particular file and group ownership:

```
/bin/find / -fsonly hfs -user user_name -group group_name -depth -print
```

If you are globally reassigning file or group ownership of all files and directories, use the `find`, `xargs`, and the command to be executed globally (`rm`, `rmdir`, `chown`, `chgrp`, or `chacl`). For example:

```
/bin/find / -fsonly hfs -user user_name -group group_name -depth -print | xargs chgrp new_group
```

If you are not globally removing or changing file access permissions, use the `find` separately from the `rm`, `rmdir`, `chown`, `chgrp`, or `chacl` command.

5. Edit the `/etc/passwd` file to remove user entries if you are removing users from the system.
6. Remove Access Control List (ACL) entries for the group being removed using the `find` and `chacl` commands:

   ```
   /bin/find / -fsonly hfs -acl %.group_name -depth -print | chacl -d %.group_name
   ```

   where:

   group_name is the name of the group being removed.

7. Use the `pwck` command to verify that the `/etc/passwd` file has valid entries:

   ```
   /etc/pwck
   ```

   The `pwck` command validates the number of fields, login name, user ID, group ID, and whether the login directory and optional program name exist. Refer to *pwck*(1M) in the *HP-UX Reference Manual* manual for additional information.

8. Edit the `/etc/group` file, and optionally the `/etc/logingroup` file, to remove the group entry for the group you are removing. If users are being removed from the system, remove the user from any other groups.

9. Use the `grpck` command to check for inconsistencies and verify all entries in the `/etc/group` and `/etc/logingroup` files. This verification includes a check of the number of fields, group name, group ID, and whether all login names appear in the `/etc/passwd` file. The `grpck` command has the following format:

   `/etc/grpck` [*group_file*]

Additional Task Information

A blank line in the `/etc/group` or `/etc/logingroup` file is not allowed. If a blank line appears in the files, all entries after the blank line are ignored. Refer to "Adding a Group Using HP-UX Commands" for a description of the `/etc/group` and `/etc/logingroup` files. See also *group*(4) in the *HP-UX Reference Manual* manual.

Refer to "Controlling User Accounts and Groups" for more information about the `vipw` command.

For additional ACL information see *lsacl*(1), *chacl*(1), and *acl*(5) in the *HP-UX Reference Manual* manual and Chapter 4, "Controlling User Access to Directories and Files", of *HP-UX System Security*, HP part number B1862-90009.

Examples

Here is a sample `/etc/group` file:

```
therapy:*:20:dennisp,janetn,jdoe,stevens,kimz
users:*:23:michelem,rykl,karens,starsky,stevens
photo:*:30:kimz,jdoe,bsmith
leader:*:59:blink,pgomez,stevens
lab:*:67:obones,jab,mlee,fjones,bsmith
```

To remove the group "photo", the group members, and their files:

1. login as `root`.
2. Make a copy of `/etc/group` and `/etc/logingroup`:

   ```
   cp /etc/group /etc/group.old
   cp /etc/logingroup /etc/logingroup.old
   ```

3. Make a copy of the `/etc/passwd` file:

```
cp /etc/passwd /etc/passwd.old
```

4. Remove all of the user files and directories with group ownership corresponding to the group being removed.

   To remove files:

```
/bin/find / -fsonly hfs -user kimz -group photo ! -type d -depth -print | xargs rm
/bin/find / -fsonly hfs -user jdoe -group photo ! -type d -depth -print | xargs rm
/bin/find / -fsonly hfs -user bsmith -group photo ! -type d -depth -print | xargs rm
```

   To remove directories:

```
/bin/find / -fsonly hfs -user kimz -group photo -type d -depth -print | xargs rmdir
/bin/find / -fsonly hfs -user jdoe -group photo -type d -depth -print | xargs rmdir
/bin/find / -fsonly hfs -user bsmith -group photo -type d -depth -print | xargs rmdir
```

   To list remaining files and directories owned by `kimz`, `jdoe`, and `bsmith`:

```
/bin/find / -fsonly hfs -user kimz -depth -print
...
/bin/find / -fsonly hfs -user jdoe -depth -print
...
/bin/find / -fsonly hfs -user bsmith -depth -print
...
```

   To reassign the remaining file to other owners:

```
/bin/find / -fsonly hfs -user kimz -depth -print | xargs chown stevens
/bin/find / -fsonly hfs -user jdoe -depth -print | xargs chown janetn
/bin/find / -fsonly hfs -user bsmith -depth -print | xargs chown mlee
```

5. Edit `/etc/passwd` to remove user entries for users `kimz`, `jdoe`, and `bsmith`. HP recommends using the `/etc/vipw` command. The `/etc/vipw` command requires the `EDITOR` environment variable set to `vi`.

   To set the Korn and Bourne shell `EDITOR` environment variable, type:

```
export EDITOR=vi
```

   To set the C shell `EDITOR` environment variable, type:

```
setenv EDITOR vi
```

6. Remove `kimz`, `jdoe`, and `bsmith` from all ACL entries:

```
/bin/find / -fsonly hfs -acl kimz.photo -depth -print | xargs chacl -d %.photo
```

or

```
/bin/find / -fsonly hfs -acl kimz.photo -depth -print | xargs chacl -d kimz.photo
/bin/find / -fsonly hfs -acl jdoe.photo -depth -print | xargs chacl -d jdoe.photo
/bin/find / -fsonly hfs -acl bsmith.photo -depth -print | xargs chacl -d bsmith.photo
```

7. Check the `/etc/passwd` file format:


```
pwck
```

8. Edit `/etc/group` to remove the "photo" group entry and the user entries in other groups for users `kimz`, `jdoe`, and `bsmith`.

9. Run `grpck`:

```
/etc/grpck
```

Changing a User's Primary Group Using HP-UX Commands

To change a user's primary group:

1. Ensure that you have superuser capabilities.
2. Make a copy of the `/etc/passwd` file so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/passwd /etc/passwd.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the file using the command:

   ```
   cp /etc/passwd.old /etc/passwd
   ```

3. Determine the *group_ID* number of your user's *new* primary group by looking at the *new* primary group's entry in the `/etc/group` file. The third field of the group entry in the `/etc/group` file (fields are separated by colons ":") contains the *group_ID*.
4. Edit the `/etc/passwd` file using the `vipw` command to replace the primary group ID, *pri_group_ID* (fourth field), of your user's entry with the *new* primary group ID.
5. Use the `pwck` command to verify that the `/etc/passwd` file has valid entries:

   ```
   /etc/pwck
   ```

   The `pwck` command validates the number of fields, login name, user ID, group ID, and whether the login directory and optional program name exist. Refer to *pwck*(1M) in the *HP-UX Reference Manual* manual for additional information.

6. Make a copy of the `/etc/group` file, and optionally the `/etc/logingroup` file, so that it is easy to undo any mistakes that you might make:

```
cp /etc/group /etc/group.old
cp /etc/logingroup /etc/logingroup.old
```

If you need to undo a mistake later, you can restore the "old" contents of the files using the commands:

```
cp /etc/group.old /etc/group
cp /etc/logingroup.old /etc/logingroup
```


7. Evaluate the access privileges needed on the files and directories owned by the user. Use Table 4-4 to determine the next sequence of steps based on the access need of the owner (rows) and the new and old primary groups (columns).

Table 4-3. "Changing a User's Primary Group" Task Decision Table

	New Primary Group Access	Old Primary Group Access
Retain Owner Access	• Change group ownership to the new primary group using **chgrp**.	No change. • Optionally, update the ACL entries to reflect the new primary group using **chacl -r**.
Deny Owner Access	• Change group ownership to the new primary group using **chown**. • Change file ownership to a user in the new primary group using **chown**. • Set ACLs to deny access using **chacl**.	• Change file ownership to a member of user's old primary group using **chown**. • Set ACLS to deny access using **chacl**, or globally deny other group access privileges using **chmod**.

Use the `find` command to globally search the system for files with particular file and group ownership:

```
/bin/find / -fsonly hfs -user user_name -group group_name -depth -print
```

If you are globally reassigning file or group ownership of all files and directories, use the `find`, `xargs`, and the command to be executed globally (`chown`, `chgrp`, or `chacl`). For example:

```
/bin/find / -fsonly hfs -user user_name -group group_name -depth -print | xargs chgrp new_group
```

If you are not globally changing file access permissions, use the `find` separately from the `chown`, `chgrp`, or `chacl` command.

8. Edit the `/etc/group` file to add the user's login name to the entry which corresponds to their *new* primary group. If you do *not* want the user to continue to be a member of their *previous* primary group, you will also need to remove the user's login name from the list of user members of their previous primary group.

Note

- A blank line in the `/etc/group` or `/etc/logingroup` file is not allowed. If a blank line appears in the files, all entries after the blank line are ignored. Refer to the "Adding a Group Using HP-UX Commands" section of this chapter for a description of the `/etc/group` and `/etc/logingroup` files.
- A group can have a maximum limit of 200 users.

9. Use the `grpck` command to check for inconsistencies and verify all entries in the `/etc/group` and `/etc/logingroup` files. This verification includes a check of the number of fields, group name, group ID, and whether all login names appear in the `/etc/passwd` file. The `grpck` command has the following format:

   `/etc/grpck` [*group_file*]

   where:

   group_file is the name of the group file to be checked. The default group file is `/etc/group`.

Additional Task Information

Optionally, update the Access Control List (ACL) entries to replace, add, or delete the necessary file access permissions. For additional ACL information see *lsacl*(1), *chacl*(1), and *acl*(5) in the *HP-UX Reference Manual* manual and Chapter 4, "Controlling User Access to Directories and Files", of *HP-UX System Security*, HP part number B1862-90009.

Refer to "Controlling User Accounts and Groups" for more information about the `vipw` command.

Examples

Here is a sample of the `/etc/group` file:

```
therapy:*:20:dennisp,janetn,jdoe,stevens
users:*:23:rykl,karens,starsky,stevens
photo:*:30:kimz,jdoe,bsmith
leader:*:59:blink,pgomez,stevens,jdoe
lab:*:67:obones,jab,mlee,fjones, michelem
```

Her is a sample of an `/etc/passwd` entry:

```
michelem:uNx.Q3QPMEqjs,Q1sF:342:67:M.Mansfield, 47LG, (415) 555-1101:/users/mmm:/bin/ksh
```

To change the primary group for user `michelem` from the "lab" group to the "users" group:

1. login as `root`.
2. Make a copy of the `/etc/passwd` file:

   ```
   cp /etc/passwd /etc/passwd.old
   ```

3. After looking at the `/etc/group` file, the new primary group (`users`) ID is 23.

4. Edit /etc/passwd to replace the existing *pri_group_ID* with the new primary group *pri_group_ID*, 23. HP recommends using the /etc/vipw command. The /etc/vipw command requires the EDITOR environment variable set to vi.

   To set the Korn and Bourne shell EDITOR environment variable, type:

   ```
   export EDITOR=vi
   ```

   To set the C shell EDITOR environment variable, type:

   ```
   setenv EDITOR vi
   ```

   The new /etc/passwd entry should be:

   ```
   michelem:uNx.Q3QPMEqjs,Q1sF:342:23:M.Mansfield, 47LG, (415) 325-1101:/users/mmm:/bin/ksh
   ```

5. Check the /etc/passwd file format:

   ```
   pwck
   ```

6. Make a copy of the /etc/group file:

   ```
   cp /etc/group /etc/group.old
   ```

   Optionally, create a copy of the /etc/logingroup file.

7. Change file access privileges such that user michelem and the members of the new primary group have access to all of the user's files. Change the group ownership of all files owned by user michelem to the "users" group:

   ```
   /bin/find / -fsonly hfs -user michelem -group lab -depth -print | xargs chgrp users
   ```

   Change ACLs to replace user and old primary group entries with user and new primary group entries:

   ```
   /bin/find / -fsonly hfs -acl michelem.lab -depth -print | xargs chacl -r michelem.users
   ```

8. Edit /etc/group to add user michelem to the new primary group "users." For this example, user michelem can remain a member of the previous primary group "lab".

   Optionally, edit the /etc/logingroup file.

9. Check the /etc/group file format:

   ```
   /etc/grpck
   ```

   Optionally, run grpck on the /etc/logingroup file.

Adding Users to Groups Using HP-UX Commands

To add users to a group (without changing the users' primary group):

1. Ensure that you have superuser capabilities.
2. Make a copy of the `/etc/group` file, and optionally the `/etc/logingroup` file, so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/group /etc/group.old
   cp /etc/logingroup /etc/logingroup.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the files using the commands:

   ```
   cp /etc/group.old /etc/group
   cp /etc/logingroup.old /etc/logingroup
   ```

3. Edit the `/etc/group` file, and optionally the `/etc/logingroup` file, to add the user's *user_name* to the names in the comma-separated list of members for the group(s) for which they are to be members. If you want the user to be able to access files belonging to another group other than their primary group without using the `chgrp` command, edit the `/etc/logingroup` file to add the user to all necessary groups.
4. Use the `grpck` command to check for inconsistencies and verify all entries in the `/etc/group` and `/etc/logingroup` files. This verification includes a check of the number of fields, group name, group ID, and whether all login names appear in the `/etc/passwd` file. The `grpck` command has the following format:

   `/etc/grpck` [*group_file*]

   where:

   group_file is the name of the group file to be checked. The default group file is `/etc/group`.

Additional Task Information

Note

- A blank line in the `/etc/group` or `/etc/logingroup` file is not allowed. If a blank line appears in the files, all entries after the blank line are ignored. Refer to the "Adding a Group Using HP-UX Commands" section of this chapter for a description of the `/etc/group` and `/etc/logingroup` file formats.
- A group can have a maximum limit of 200 users.

Optionally, update the Access Control List (ACL) entries to replace, add, or delete the necessary file access permissions. For additional ACL information see *lsacl*(1), *chacl*(1), and *acl*(5) in the *HP-UX Reference Manual* manual and Chapter 4, "Controlling User Access to Directories and Files", of *HP-UX System Security*.

Examples

Here is a sample /etc/group file:

```
cats:*:15:donna,woody
naomil:*:20:mj,michelem,kerschen
```

To add users pixie and pepper to groups "cats" and "naomil":

1. login as root.
2. Make a copy of the /etc/group and /etc/logingroup files:

   ```
   cp /etc/group /etc/group.old
   ```

   Optionally, make a copy of the /etc/logingroup file.
3. Edit the /etc/group file to add the users pixie and pepper to the group member lists. For example:

   ```
   cats:*:15:donna,woody,pixie,pepper
   naomil:*:20:mj,michelem,kerschen,pixie,pepper
   ```

   Optionally edit the /etc/logingroup file.
4. Check the /etc/group file format:

   ```
   /etc/grpck
   ```

Optionally run grpck on the /etc/logingroup file.

Removing Users From Groups Using HP-UX Commands

Note If you are removing users from the system, see the "Removing a User Using HP-UX Commands" section of this chapter. If you are changing the user's primary group, see the "Changing a User's Primary Group Using HP-UX Commands" section of this chapter. Otherwise, this procedure assumes that you are *not* removing users from their primary groups.

To remove users from groups:

1. Ensure that you have superuser capabilities
2. Make a copy of the `/etc/group` file, and optionally the `/etc/logingroup` file, so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/group /etc/group.old
   cp /etc/logingroup /etc/logingroup.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the files using the commands:

   ```
   cp /etc/group.old /etc/group
   cp /etc/logingroup.old /etc/logingroup
   ```

3. List the files and directories owned by the user using the `find` command:

   `/bin/find / -fsonly hfs -user` *user_name* `-group` *group_name* `-depth -print`

   where:

   user_name is the login name of the user as defined in the `/etc/passwd` file.

   group_name is the group from which the user is being removed.

   If the list of files is long you can redirect the output to a file or redirect the output to the `more` command. For example:

   `/bin/find / -fsonly hfs -user` *user_name* `-group` *group_name* `-depth -print | more`

4. Use Table 4-4 to determine the next sequence of steps based on the status of the group members and their files.

Removing Users From Groups Using HP-UX Commands

For each user that is being removed, perform the steps that correspond to the matching combination of actions taken regarding the user account (table row) and files (table column).

Table 4-4. "Removing Users From Groups" Task Decision Table

	Delete User Files	Keep User Files
Delete User(s) From System	• For files matching the user and group ownership, remove all files and directories from the system using `find`, `rm`, and `rmdir`. • For the remaining files and directories owned by each user, remove them or reassign file ownership. Remove the files and directories using `find`, `rm`, and `rmdir`. Reassign file ownership using `find` and `chown`. • Edit `/etc/passwd` to remove user entries.	• For all files on the system owned by the user being removed with the group, change file and group ownership using `find`, `chown`, and `chgrp`. • Edit `/etc/passwd` to remove user entries.
Keep User(s) on System	• For each user, delete the files owned by the user and group combination using `find`, `rm`, and `rmdir`.	• If the user *is* to retain access to the files, change the group ownership of files owned by the user to a group that the user remains a member of using `chgrp`. • If the user is *not* to retain access to the files, change the file ownership to a user that is *not* a member of the current owner's group using `chown`. For the files changing owner, change the group ownership to new owner's group using `chgrp`.

Use the `find` command to globally search the system for files with particular file and group ownership:

```
/bin/find / -fsonly hfs -user user_name -group group_name -depth -print
```

If you are globally removing files or reassigning file permissions, use the `find`, `xargs`, and the command to be executed globally (`rm`, `rmdir`, `chown`, `chgrp`, or `chacl`). For example:

```
/bin/find / -fsonly hfs -user user_name -group group_name -depth -print | xargs chgrp new_group
```

If you are not globally removing files or changing file access permissions, use the `find` separately from the `rm`, `rmdir`, `chown`, `chgrp`, or `chacl` command.

5. Remove Access Control List (ACL) entries for all users being removed from the group using the `find` and `chacl` commands:

```
/bin/find / -fsonly hfs -acl user_name.group_name -depth -print | xargs chacl -d user_name.group_name
```

6. Edit the `/etc/group` file, and optionally the `/etc/logingroup` file, to remove the user's *user_name* from the list of members for a group.

Note

- Do *not* remove the user's name from their primary group defined in the `/etc/passwd` file unless you are removing the user from the system. If you want to change the user's primary group, see the "Changing a User's Primary Group Using HP-UX Commands" section of this chapter.
- A blank line in the `/etc/group` or `/etc/logingroup` file is not allowed. If a blank line appears in the files, all entries after the blank line are ignored. Refer to the "Adding a Group Using HP-UX Commands" section of this chapter for a description of the `/etc/group` and `/etc/logingroup` file formats.

7. Use the `grpck` command to check for inconsistencies and verify all entries in the `/etc/group` and `/etc/logingroup` files. This verification includes a check of the number of fields, group name, group ID, and whether all login names appear in the password file. The `grpck` command has the following format:

   `/etc/grpck` [*group_file*]

   where:

   group_file is the name of the group file to be checked. The default group file is `/etc/group`.

Additional Task Information

For additional ACL information see *lsacl*(1), *chacl*(1), and *acl*(5) in the *HP-UX Reference Manual* manual and Chapter 4, "Controlling User Access to Directories and Files", of *HP-UX System Security*, HP part number B1862-90009.

Examples

The following listing is a sample group file:

```
therapy:*:20:dennisp,janetn,jdoe,stevens
photo:*:30:kimz,jdoe,bsmith
leader:*:59:blink,pgomez,stevens
lab:*:67:obones,jab,mlee,fjones
```

To remove user `obones` from the `lab` group (which is not the user's primary group):

1. Login in as `root`.
2. Make a copy of the `/etc/group` file:

   ```
   cp /etc/group /etc/group.old
   ```

   Optionally, make a copy of the `/etc/logingroup` file.

3. List the files owned by the user obones and belonging to group lab:

```
/bin/find / -fsonly hfs -user obones -group lab -depth -print
...
...
```

4. Remove user obones from access privileges to the /users/development directory and its files, change the ownership of these files to a user belonging to the group from which you removed user obones. For example, change the file ownership of the project related files to user fjones:

```
chown /users/development blink
/bin/find /users/development -fsonly hfs -user obones -group lab -depth -print | xargs chown fjones
```

Enable only user obones to access his personal files by transferring the group ownership of personal files to another group in which obones is a member. If obones personal file's group ownership is not changed, any member of the group (for which obones is no longer a member) may gain access to the files depending on how the group permissions are set. To ensure privacy change the group ownership:

```
/bin/find /users/obones -fsonly hfs -user obones -group lab -depth -print | xargs chgrp photo
```

5. Remove obones.lab from any ACL entries:

```
/bin/find / -fsonly hfs -acl obones.lab -depth -print | xargs chacl -d obones.lab
```

6. Edit the /etc/group file to remove obones from the "lab" group entry. The following listing is how the file would appear after the edit:

```
therapy:*:20:dennisp,janetn,jdoe,stevens
photo:*:30:kimz,jdoe,bsmith
leader:*:59:blink,pgomez,stevens
lab:*:67:jab,mlee,fjones
```

7. Run grpck:

```
/etc/grpck
```

Optionally edit and run grpck on the /etc/logingroup file.

Displaying/Assigning Special Group Privileges Using HP-UX Commands

To display special group privileges use the `getprivgrp` command:

`/usr/bin/getprivgrp` [`-g` | *group_name*]

where:

`-g` lists access privileges that have been granted to all groups. Otherwise, access privileges are listed for all privileged groups to which the requestor belongs.

group_name is the name of the group as specified in the `/etc/group` file.

If *group_name* is supplied, access privileges are listed for that group only. If the requestor is not a member of the *group_name* specified, no information is displayed. The superuser is a member of all groups.

To assign special group privileges:

1. Ensure that you have superuser capabilities.
2. Assign special privileges using the `setprivgrp` command. There are three formats for the `setprivgrp` command.

Displaying/Assigning Special Group Privileges Using HP-UX Commands

One format of the `setprivgrp` command is:

`/etc/setprivgrp` *group_name* [*privilege*]

where:

group_name is the name of the group as specified in the `/etc/group` file.

privilege is one or more of the following privileges:

RTPRIO	gives access to the `rtprio` command and system call, which allows the setting of real-time priorities.
MLOCK	gives access to the `plock` system call, which allows processes to be locked in memory and allows use of the SHM_LOCK command (used with the `shmctl` system call).
CHOWN	gives access to the `chown` command and system call, which allows members of the group to change the ownership of files on the system.
LOCKRDONLY	gives access to the `lockf` system call to set locks on files that are open for reading only.
SETRUGID	gives access to the `setuid` and `setgid` system calls to change, respectively, the real user ID or real group ID of a process.

Displaying/Assigning Special Group Privileges Using HP-UX Commands

The other two formats of the `setprivgrp` command are:

```
/etc/setprivgrp -g [privilege]

/etc/setprivgrp -n [privilege]
```

where:

`-g`	specifies that *all* groups have access to the specified *privilege*(s).
`-n`	specifies that *no* groups have access to the specified *privilege*(s).
privilege	is one or a combination of the special privileges (RTPRIO, MLOCK, CHOWN, LOCKRDONLY, or SETRUGID).

For additional information refer to *rtprio*(1), *rtprio*(2), *plock*(2), *shmctl*(2), *chown*(1), *chown*(2), *lockf*(2), *setuid*(2), *setgid*(2), *setprivgrp*(2), and *setprivgrp*(1M) in the *HP-UX Reference Manual* manual. Capabilities set by this command are not added to existing capabilities for the same group. If you want to add a capability for a particular group, you must respecify all capabilities that were already set for that group as well as the new capability.

Note Specifying no privileges removes all special privileges for the group.

Additional Task Information

Any user whose current *group_ID* matches the *group_ID* of a privileged group will have access to the special capabilities assigned to that group. A group can have any one, two or all five of the capabilities associated with it.

Examples

- To set real-time priorities and enable user processes to lock process text and data into memory for the `development` group:

```
setprivgrp development RTPRIO MLOCK
getprivgrp
global privileges: CHOWN
development: RTPRIO MLOCK
```

- To set real-time priorities and enable user processes to lock process text and data into memory for all groups:

```
setprivgrp -g RTPRIO MLOCK
```

- To deny real-time priorities and disable user processes to lock process text and data into memory for all groups:

```
setprivgrp -n RTPRIO MLOCK
```

For additional information, refer to *setprivgrp*(1M) and *getprivgrp*(1) in the *HP-UX Reference Manual* manual.

Managing Run-Levels

A **run-level** is an HP-UX state of operation in which a specific set of processes (and their offspring) are permitted to run. This set of processes is defined, for each run-level, in a file called `/etc/inittab`.

The following list contains tasks covered in this section:

- Creating a New Run-Level Using HP-UX Commands
- Changing System Run-Levels Using HP-UX Commands
- Entering the System Administration Run-Level
- Returning From the System Administration Run-Level

Creating a New Run-Level Using HP-UX Commands

To create new run-levels:

1. Ensure that you have superuser capabilities.
2. Make a copy of the `/etc/inittab` file so that it is easy to undo any mistakes that you might make:

   ```
   cp /etc/inittab /etc/inittab.old
   ```

   If you need to undo a mistake later, you can restore the "old" contents of the file using the command:

   ```
   cp /etc/inittab.old /etc/inittab
   ```

3. Edit the `/etc/inittab` file with the editor of your choice to create a one-line entry in `/etc/inittab` with the following format:

   id:*rstate*:*action*:*process*

   where:

id	is a unique four-character identifier, used to identify an entry.
rstate	is a list of run-levels to which each entry applies.
action	is a action to be performed, such as `respawn`.
process	is the command that will be executed when that run-level is entered.

   Refer to *init*(1M) and *inittab*(4) in the *HP-UX Reference Manual* manual for a detailed description of entries in the `/etc/inittab` file.

4. Edit the `/etc/inittab` file to change the `initdefault` entry in your test version to "s". By changing the `initdefault` entry to "s", you will come up in run-level "s" when you boot. You can *change* to run-level 2 after booting by executing `init 2`. If your new run-level 2 does not work, you can still reboot.

   Note that "s" is not a normal run-level. If you create this test version, you should replace the "s" with "2" after testing is complete. Run-level "s" is for system maintenance only.

   If you do not have a working state for the `initdefault` state, you may not be able to boot your system. After thoroughly testing your changes, *restore the original initdefault value.*

Examples

The following is an example `/etc/inittab` for a system that contains a system console and six terminals. The initdefault run-level is run-level 2. Run-level 2 is a multi-user run-level, with a `getty` on every terminal.

```
init:2:initdefault:
stty::sysinit:stty 9600 clocal icanon echo opost onlcr ienqak ixon icrnl ignpar </dev/systty
brc1::bootwait:/etc/bcheckrc </dev/console >/dev/console 2>&1 # fsck, etc.
slib::bootwait:/etc/recoversl </dev/console >/dev/console 2>&1 #shared libs
brc2::bootwait:/etc/brc >/dev/console 2>&1          # boottime commands
link::wait:/bin/sh -c "rm -f /dev/syscon; \
    ln /dev/systty /dev/syscon" >/dev/console 2>&1
cwrt::bootwait:cat /etc/copyright >/dev/syscon      # legal requirements
rc  ::wait:/etc/rc </dev/console >/dev/console 2>&1 # system initialization
powf::powerwait:/etc/powerfail >/dev/console 2>&1   # power fail routines
lp  ::off:nohup sleep 999999999 </dev/lp & stty 9600 </dev/lp
cons:012456:respawn:/etc/getty -h console console           # system console
vue :34:respawn:/etc/vuerc                      # VUE validation and invocation
t1:2:respawn:/etc/getty tty01 H
t2:2:respawn:/etc/getty tty02 H
t3:2:respawn:/etc/getty tty03 H
t4:2:respawn:/etc/getty tty04 H
t5:2:respawn:/etc/getty tty05 H
t6:2:respawn:/etc/getty tty06 H
```

Changing System Run-Levels Using HP-UX Commands

The following is a general procedure for changing the system from one run-level to another. You must be logged in at the system console as the superuser to change the system's run-level.

1. Warn all users who are currently logged in before you change run-levels.

   Changing to another run-level while users are logged on will kill (terminate) their processes if the run-level you are moving to does not contain `rstate` entries in `/etc/inittab` for their terminal. You can use the `write` or `wall` commands to communicate with the users. Note that the `wall` (write all) command immediately sends your message to the terminal of each user on the system.

   If each `getty` (terminal) entry has the new run-level in its `rstate` field, or if the `rstate` field is empty (implies all numbered run-levels), you don't need to ask them to log off; their processes will not be killed (unless your new run-level is run-level "s").

2. To change to a run-level other than run-level "s", use the command:

   `/etc/init` *new_run-level*

   where *new_run-level* is the number of the run-level you want to enter.

   To change to run-level "s", use the command:

   `/etc/shutdown`

Caution

- You should not change to run-level "s" without using the `shutdown` command (that is, do not execute `init s`). The `shutdown` command provides safeguards to "cleanly" bring your system to single-user mode (run-level "s").
- Run-level 0 is a special run-level reserved for system installation. Do not use run-level 0.

Entering the System Administration Run-Level

Many of the system maintenance tasks you perform as system administrator require the system to be in single-user mode (run-level "s") so that you can ensure that no one else is on the system while you're performing those tasks. In this run-level, the only access to the system is through the system console by the user `root`, and the only processes running on the system will be the shell on the system console, background daemon processes started by `/etc/rc`, and processes that you invoke. Commands requiring an inactive system (such as `fsck`) should be run in run-level "s".

Use the `shutdown` command, instead of `init s` when changing your system's run-level from any numbered run-level (run-levels 1 through 6) to run-level "s". The `shutdown` command kills all non-essential processes and brings the system *safely* to run-level "s" (without leaving system resources in an unusable state).

The `shutdown` command also allows you to specify a **grace period** to allow time for your users to terminate their work before the system goes down. The grace period is given (in number of seconds to wait) immediately following the command name. For example:

To enter run-level "s" with a grace period of 30 seconds, type in:

```
/etc/shutdown 30
```

This will automatically warn all users that they have 30 seconds to log off, kill all processes, and safely bring the system to run-level "s".

For details on how to use the `shutdown` command, see Chapter 3, "Starting and Stopping HP-UX" and *shutdown*(1M) in the *HP-UX Reference Manual* manual.

Returning From the System Administration Run-Level

When you want to change your system's run-level *from* run-level "s" (single-user mode) to one of the other run-levels, it is best to do so by rebooting your system. You can use the `reboot` command to do this. You can also use the `init` command as described earlier in this chapter (See "Changing System Run-Levels Using HP-UX Commands") to change to the new run-level.

5

Managing Peripherals

Managing peripherals on your system is a very important system administration task. This chapter provides general guidelines and procedures for managing your peripherals. The purpose of this chapter is to give an overview of adding a peripheral and detailed instructions on moving and removing peripherals. There are references to other chapters of this book and other books in the documentation set that provide specific instructions and details.

The task guidelines covered in this chapter are:

- Overview of adding a peripheral
- Moving peripherals using SAM
- Removing peripherals using SAM
- Creating device files using HP-UX commands
- Moving peripherals using HP-UX commands
- Removing peripherals using HP-UX commands

This chapter describes special cases, for example, moving your system console or root disk to a different hardware address.

The *Installing Peripherals* manual has details on adding a peripheral to your system.

Note If you are manipulating disk drives that are part of LVM (Logical Volume Manager), or if you are adding disks to LVM, refer to Chapter 8, "Managing Logical Volumes".

You can manage peripherals in either of two ways:

1. The System Administration Manager (SAM)
2. HP-UX commands

SAM configures the majority of all peripherals supported by HP-UX. For those devices that SAM supports, SAM creates the device files and adds the device driver to the kernel, if necessary.

When adding local printers and plotters with SAM, the device is automatically configured into the lp spooler.

When adding disk drives with SAM, you can create and mount file systems. Additionally, you can add your disk to the Logical Volume Manager. Logical volumes allow flexible disk partitioning, disk mirroring, and file systems to span several disk drives. For additional information, refer to the "Managing Logical Volumes" chapter.

5

The peripherals that are not supported by SAM are:

- Graphics interface cards and graphics displays
- HP C2753A Model F100 Printer
- Flexible disk drives
- Scanjets/digitizers
- Instruments

For the devices that SAM does not support, SAM does not create device files, but you can use SAM to add the device drivers to the kernel.

Although SAM supports adding the single optical disks and optical disk library systems (autochangers), SAM only supports using section 2 (the entire disk surface) when configuring disk surfaces (2 surfaces per disk) for your system. SAM does not support using sections 0, 1, 3, 4, 5, 6, 7, 8, 9, 12, 13, 14, or 15 when configuring optical disk surfaces. If you need to configure your disk surfaces to use disk sections other than section 2, use the HP-UX commands method of adding an optical disk drive or optical disk library to the system.

SAM provides an on line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Refer to Chapter 1, "Introduction to System Administration", for additional SAM information.

Overview of Adding a Peripheral

There are two steps to adding a peripheral to your system:

1. You must physically connect the peripheral to your system and ensure that the peripheral's communication protocol is configured properly. Refer to the documentation shipped with the peripheral to locate the interface port, set and configure the communication protocol, and perform a self test (if applicable).

2. Your operating system must be configured to recognize the peripheral. Configuring your system includes:

   a. ensuring that the device driver for the peripheral is configured into the HP-UX kernel (typically `/hp-ux`). There can be more than one driver required in the kernel for the device.

   b. creating a device file or multiple device files.

   c. configuring HP-UX utilities to use the peripheral.

   SAM can perform all of these tasks for you.

   If your device driver is part of the kernel and the device driver is supported by `autoconfig`, your device files are created automatically when you boot.

Refer to the *Installing Peripherals* manual to add a particular peripheral to your system. The *Installing Peripherals* manual provides device driver and device file information.

Moving Peripherals Using SAM

Moving peripherals using SAM is different depending on the type of peripheral. Instructions for each of the following types of peripherals are included:

- local printer or plotter
- tape drive
- terminal or modem
- disk drive containing file systems or swap space

If users will be affected, notify the users on the system of the device location change, see the "Communicating With the Users on Your System" chapter of this manual.

Moving a Local Printer or Plotter

Moving a local printer on your system requires that you remove the printer from the lp spooler and then add the printer using the new hardware location encoded in the device file. When SAM removes a printer, SAM also removes the model script used to communicate with the printer located in the `/usr/spool/lp/interface` directory. Prior to removing the printer, you should preserve this file under another name and use it when adding the printer back to the lp spooler.

To move a local printer or plotter using SAM:

1. Ensure that you have superuser capabilities.
2. If users will be affected, notify the users on the system about the device being moved. See the "Communicating With the Users on Your System" chapter of this manual.
3. Copy `/usr/spool/lp/interface/`*printer_name* to a temporary location.

```
cp /usr/spool/lp/interface/printer_name /usr/spool/lp/interface/printer_name.old
```

4. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on line help system to assist you when you need additional information.

   Activating the [Help] button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the [f1] key gives you context-sensitive information for the object at the location of the cursor.

   Refer to Chapter 1, "Introduction to System Administration", for additional SAM information.

5. Highlight `Printer and Plotters` and activate the (Open) control button.
6. Highlight `Printer/Plotters` and activate the (Open) control button.
7. Highlight the printer that you wish to move and choose `Remove Printer/Plotter`. SAM removes the device file if the device file is `/dev/lp_`*printer_name*; otherwise, the device file is not removed.
8. Exit SAM.
9. Turn off, unplug, and disconnect the printer or plotter.
10. Add the local printer to your system at the new hardware location; refer to the *Installing Peripherals* manual for specific instructions. When using SAM to add a printer, SAM also adds the printer to the lp spooler. If the printer is an HP-IB printer, ensure that the printer has a unique bus address.

5

Note Specify the temporary file (`/usr/spool/lp/interface/`*printer_name*`.old`) you created in step 3 for the model script.

---

11. Remove the temporary file (`/usr/spool/lp/interface/`*printer_name*`.old`) you created in step 3.
12. Update any software application configurations that use the relocated local printer. Refer to your software application documentation for specific instructions.

Refer to the "Managing Printers and Printer Output" chapter of this manual for additional information about adding and removing a local printer or plotter using SAM.

Note To return your system to the original configuration, repeat this procedure.

Moving a Tape Drive

Moving a tape drive on your system requires that you remove the tape drive and then add the tape drive using the new hardware location.

To move a tape drive using SAM:

1. Ensure that you have superuser capabilities.
2. Notify the users on the system about the system shutdown as a result of moving the tape drive. See the "Communicating With the Users on Your System" chapter of this manual.

Note It is recommended that you shut down and power off your computer while moving devices on the SCSI or HP-IB bus.

5

3. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   Refer to Chapter 1, "Introduction to System Administration", for additional SAM information.

4. Highlight `Peripheral Devices` and activate the (OK) control button.
5. Highlight `Tape Drives` and activate the (OK) control button.
6. Choose `Remove...` from the "Actions" menu. SAM does not remove the tape drive device file.
7. Exit SAM.
8. Remove any media currently loaded in the tape drive.
9. Shut down your system:

   ```
   cd /
   shutdown grace_period_in_seconds
   ```

   For additional information, see Chapter 3, "Starting and Stopping HP-UX" of this manual.

10. Halt the system:

    ```
    reboot -h
    ```

11. Turn off the computer.
12. Turn off, unplug, and disconnect the tape drive.
13. Physically move the tape drive. Refer to the documentation shipped with your tape drive if you intend to change the HP-IB or SCSI bus address. Ensure that each device on the SCSI or HP-IB bus has a unique bus address.

14. Turn on the tape drive.

Note For SCSI devices, it is important to power up your device before turning on your system.

---

15. Turn on the computer.
16. Log in and run SAM to add the tape drive to your system at the new hardware address. SAM will create a new device file for the tape drive and will not remove the old device file, unless you specify the same device file name.
17. Update any automated backup processes you have scheduled. See the "Backing Up and Restoring Your Data" chapter of this book for specific instructions.
18. Reload media into the tape drive.
19. Update any software application configurations that use the relocated tape drive. Refer to your software application documentation for specific instructions.

Moving a Terminal or Modem

Moving a terminal or modem on your system requires that you remove the terminal or modem and then add the terminal or modem using the new hardware location.

To move a terminal or modem using SAM:

1. Ensure that you have superuser capabilities.
2. If users will be affected, notify the users on the system about the device being moved. See the "Communicating With the Users on Your System" chapter of this manual. The terminal or modem to be moved cannot be in use, it must be inactive.
3. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   Refer to Chapter 1, "Introduction to System Administration", for additional SAM information.

4. Highlight `Peripheral Devices` and activate the (OK) control button.
5. Highlight `Terminals and Modems` and activate the (OK) control button.
6. Highlight the terminal or modem you intend to move and choose `Remove Device...` from the "Actions" menu. SAM removes the associated terminal and modem device file(s). SAM also updates the `/etc/inittab` file by removing the `getty` entry for the terminal or modem.
7. Exit SAM.
8. Turn off, unplug, and disconnect the terminal or modem.
9. Add the terminal or modem to the system at the new hardware location; refer to the *Installing Peripherals* manual. If you use SAM, SAM will create the necessary device files and entries in the `/etc/inittab` file.
10. Update any software application configurations that use the relocated terminal or modem. Refer to your software application documentation for specific instructions.

Moving a Disk Drive Used for File Systems and Swap

Note

- You can only change the hardware address of a disk used for a file system or swap using SAM.
- SAM does not support:
  - changing the hardware address of a disk drive containing the root file system. See "Moving a Disk Drive Containing the Root File System".
  - changing the hardware address of a disk that is labeled "unused". See "Moving a Disk Drive Used for File Systems and Swap".
  - changing the hardware address of a disk array.
  - changing the hardware address of disk that is part of the logical volume manager.
  - moving a logical volume.

5

To move a disk drive from one hardware address to another using SAM:

1. Ensure that you have superuser capabilities.
2. Back up the data on the disk. See the "Backing Up and Restoring Your Data" chapter of this manual for specific instructions.
3. Notify the users on the system about the system shutdown as a result of moving the disk drive. See the "Communicating With the Users on Your System" chapter of this manual.

Note It is recommended that you shut down and power off your computer while moving devices on the SCSI or HP-IB bus.

4. If your system is an NFS server and file systems on the disk you are moving are exported, unmount the file systems from the NFS client. If you do not unmount the file systems from the client, the client will receive NFS error messages when accessing the files on the disk.

   To find the NFS clients, log in to the NFS server and look at the `/etc/exports` file. Refer to *exports*(4) in the *HP-UX Reference Manual*.

   Notify the users on the NFS client systems that data on the disk to be relocated will be temporarily inaccessible. See "Communicating With the Users on Your System".

   There are three methods to unmount the NFS client file systems:

   a. Enter the `Remote Administration` area of SAM on the NFS server and unmount the file systems remotely.

   b. Log in directly to each NFS client and run SAM to unmount the file systems.

   c. Log in directly to each NFS client and unmount the file systems using HP-UX commands.

   Refer to Chapter 6, "Managing the File System" for specific instructions on unmounting file systems.

5. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   Refer to Chapter 1, "Introduction to System Administration", for additional SAM information.

6. Highlight `Disks and File Systems` and activate the (OK) control button.
7. Highlight `CD-ROM and Hard Disks` and activate the (OK) control button.
8. Highlight the disk you intend to move.
9. Choose `Change a Disk Address...` from the "Actions" menu.
10. Highlight the interface to which the relocated disk will be attached. Fill in the hardware address, interface slot, and bus address, then activate the (OK) control button.

    SAM creates a new device file for the relocated disk and updates the `/etc/checklist`. SAM does not remove the old device file.

Note Your `/etc/checklist` file now reflects the future configuration of your system, *not* the current configuration of your system.

---

11. Exit SAM. SAM copied your original `/etc/checklist` file to `/etc/checklist.old`.
12. Shut down your system:

    ```
    cd /
    shutdown grace_period_in_seconds
    ```

    For additional information, see Chapter 3, "Starting and Stopping HP-UX" of this manual.
13. Halt the system:

    ```
    reboot -h
    ```
14. Turn off the computer.
15. Turn off, unplug, and disconnect the disk drive.
16. Physically move the disk drive. Refer to the documentation shipped with your disk drive if you intend to change the HP-FL, HP-IB, or SCSI bus address. Ensure that each device on the SCSI or HP-IB bus has a unique bus address.

17. Turn on the disk drive.

Note For SCSI devices, it is important to power up your device before turning on your system.

18. Turn on the computer.
19. Update any software application configurations that use the relocated disk drive. Refer to your software application documentation for specific instructions.
20. If your system is an NFS server, remount the NFS client's file system(s) that were temporarily inaccessible. As superuser on the NFS client system, type:

    ```
    mount -a
    ```

21. Update any software application configurations that used the relocated disk drive. Refer to your software application documentation for specific instructions.

Removing Peripherals Using SAM

When SAM removes peripherals from the system, SAM removes the associated device files for

- printers and plotters with device filenames `lp_`*printer_name*.
- terminals and modems.

When SAM removes disk or tape drives, SAM does not remove the associated devices files.

SAM does not remove device drivers from the kernel when removing peripherals from the system.

If users will be affected, notify the users on the system about the device being removed. See the "Communicating With the Users on Your System" chapter of this manual.

Removing a Local Printer or Plotter

To remove a local printer or plotter using SAM:

1. Ensure that you have superuser capabilities.
2. If users will be affected, notify the users on the system about the device being moved. See the "Communicating With the Users on Your System" chapter of this manual.

3. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   Refer to Chapter 1, "Introduction to System Administration", for additional SAM information.

4. Highlight `Printer and Plotters` and activate the (Open) control button.

5. Highlight `Printer/Plotters` and activate the (Open) control button.

6. Highlight the printer that you wish to remove and choose `Remove Printer/Plotter`. SAM removes the device file if the device file is `/dev/lp_`*printer_name*, otherwise the device file is not removed.

7. Exit SAM.

8. Turn off, unplug, and disconnect the printer or plotter.

9. Update any software application configurations that used the removed local printer. Refer to your software application documentation for specific instructions.

Removing a Tape Drive

To remove a tape drive using SAM:

1. Ensure that you have superuser capabilities.
2. Notify the users on the system about the system shutdown as a result of removing the tape drive. See the "Communicating With the Users on Your System" chapter of this manual.

Note It is recommended that you shut down and power off your computer while removing devices from the SCSI or HP-IB bus.

---

3. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on line help system to assist you when you need additional information.

   Activating the [Help] button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the [f1] key gives you context-sensitive information for the object at the location of the cursor.

   Refer to Chapter 1, "Introduction to System Administration", for additional SAM information.

4. Highlight `Peripheral Devices` and activate the (OK) control button.
5. Highlight `Tape Drives` and activate the (OK) control button.
6. Highlight the tape drive you wish to remove from the object list.
7. Choose `Remove...` from the "Actions" menu. SAM does not remove the tape drive device file.
8. Exit SAM.
9. Remove any media currently loaded in the tape drive.
10. Shut down your system:

    ```
    cd /
    shutdown grace_period_in_seconds
    ```

    For additional information, see Chapter 3, "Starting and Stopping HP-UX" of this manual.

11. Halt the system:

    ```
    reboot -h
    ```

12. Turn off the computer.
13. Turn off, unplug, and disconnect the tape drive.
14. Turn on the computer.
15. Modify any automated backup processes scheduled by the `cron` utility that expect the tape drive to be present. See the "Backing Up and Restoring Your Data" chapter of this manual and *cron*(1M) in the *HP-UX Reference Manual*.
16. Update any software application configurations that used the removed tape drive. Refer to your software application documentation for specific instructions.

Removing a Terminal or Modem

To remove a terminal or modem using SAM:

1. Ensure that you have superuser capabilities.
2. If users will be affected, notify the users on the system about the terminal or modem being removed. See the "Communicating With the Users on Your System" chapter of this manual. The terminal or modem to be removed cannot be in use, it must be inactive.
3. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   Refer to Chapter 1, "Introduction to System Administration", for additional SAM information.

4. Highlight `Peripheral Devices` and activate the [OK] control button.
5. Highlight `Terminals and Modems` and activate the [OK] control button.
6. Highlight the terminal or modem you intend to move and choose `Remove Device...` from the "Actions" menu. SAM removes the associated terminal and modem device file(s). SAM also updates the `/etc/inittab` file by removing the `getty` entry for the terminal or modem.
7. Exit SAM.
8. Turn off, unplug, and disconnect the terminal or modem.
9. Update any software application configurations that used the removed terminal or modem. Refer to your software application documentation for specific instructions.

Removing a Disk Drive Used for File Systems or Swap

Note SAM does not support removing the disk containing the root file system. See "Moving a Disk Drive Containing the Root File System".

To remove a disk drive using SAM:

1. Ensure that you have superuser capabilities.
2. Optionally, back up the data on the disk drive; refer to the "Backing Up and Restoring Your Data" chapter of this manual for specific instructions.
3. Notify the users on the system about system shutdown as a result of removing the disk drive. See the "Communicating With the Users on Your System" chapter of this manual.

5

Note It is recommended that you shut down and power off your computer while removing devices from the SCSI or HP-IB bus.

4. If your system is an NFS server and file systems on the disk you are moving are exported, unmount the file systems from the NFS client. If you do not unmount the file systems from the client, the client will receive NFS error messages when accessing the files on the disk.

   To find the NFS clients, log in to the NFS server and look at the `/etc/exports` file. Refer to *exports*(4) in the *HP-UX Reference Manual*.

   Notify the users on the NFS client systems that data on the disk to be removed will be permanently inaccessible. See "Communicating With the Users on Your System".

There are three methods to unmount the NFS client file systems:

a. Enter the `Remote Administration` area of SAM on the NFS server and unmount the file systems remotely.

b. Log in directly to each NFS client and run SAM to unmount the file systems.

c. Log in directly to each NFS client and unmount the file systems using HP-UX commands.

Update the NFS client's `/etc/checklist` to remove any mount entries for file systems that are resident on the disk drive being removed. Refer to Chapter 6, "Managing the File System" for specific instructions to remove entries from the `/etc/checklist` file.

5. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on line help system to assist you when you need additional information.

   Activating the [Help] button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the [f1] key gives you context-sensitive information for the object at the location of the cursor.

   Refer to Chapter 1, "Introduction to System Administration", for additional SAM information.

6. Highlight `Disks and File Systems` and activate the [OK] control button.
7. Highlight `CD-ROM and Hard Disks` and activate the [OK] control button.
8. Highlight the disk you intend to remove and choose `Remove a Hard Disk Drive` from the "Actions" menu. SAM updates the `/etc/checklist` file to remove the mount entry. SAM does not remove the associated disk drive device file(s).
9. Exit SAM.
10. Shut down and halt your system:

    ```
    cd /
    shutdown -h time_in_seconds
    ```

11. Turn off your computer.
12. Turn off the disk drive.
13. Physically disconnect the disk drive.
14. Turn on the computer.
15. Log in. Ensure that you have superuser capabilities.
16. Update any software application configurations that used the removed disk drive. Refer to your software application documentation for specific instructions.

Creating Device Files Using HP-UX Commands

Refer to Chapter 11, "System Configuration" of *How HP-UX Works: Concepts for the System Administrator*, the *Installing Peripherals* manual, and the *mknod*(1M), *insf*(1M), or *mksf*(1M) commands in the *HP-UX Reference Manual.*

Moving Peripherals Using HP-UX Commands

Moving peripherals using HP-UX commands is different depending on the type of peripheral. Instructions for each of the following types of peripherals are included:

- printer or plotter
- tape drive
- terminal or modem
- system console
- disk drive containing file systems or swap
- disk drive containing the root file system

If users will be affected, notify the users on the system of the device location change. See the "Communicating With the Users on Your System" chapter of this manual.

Moving a Local Printer or Plotter

Moving a local printer on your system requires that you remove the printer from the lp spooler and then add the printer using the new hardware location encoded in the device file. When a printer is removed, SAM also removes the model script used to communicate with the printer located in the `/usr/spool/lp/interface` directory. Prior to removing the printer, you should preserve this file under another name and use it when adding the printer back to the lp spooler.

To move a printer using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. If users will be affected, notify the users on the system about the device being moved. You will need to notify users, if you will be changing the printer name as well as the hardware address of the printer. See the "Communicating With the Users on Your System" chapter of this manual.
3. Copy `/usr/spool/lp/interface/`*printer_name* to a temporary location:

   ```
   cp /usr/spool/lp/interface/printer_name /usr/spool/lp/interface/printer_name.old
   ```

4. Remove the printer from the lp spooler; see the "Managing Printers and Printer Output" chapter of this manual. This procedure includes shutting down your lp spooler and disabling the printer.
5. Turn off, unplug, and disconnect the printer or plotter.
6. Add the newly located printer to the system at the new hardware address; see the *Installing Peripherals* manual or the "Managing Printers and Printer Output" chapter of this manual for additional information. if the printer is an HP-IB printer, ensure that the printer has a unique bus address.

Note Specify the temporary file (`/usr/spool/lp/interface/`*printer_name*`.old`) you created in step 3 for the model script when adding the printer to the system at the new hardware address.

---

7. Remove the temporary model script file (`/usr/spool/lp/interface/`*printer_name*`.old`) created in step 3.
8. If you changed the printer name, update any software application configurations that used the local printer. You may also need to update your software application if you refer to the hardware address of the printers on the system. Refer to your software application documentation for specific instructions.

Moving a Tape Drive

Moving a tape drive on your system requires that you remove the tape drive and then add the tape drive using the new hardware location.

Note It is recommended that you shut down and power off your computer while moving devices on the SCSI or HP-IB bus. For SCSI devices, it is important to power up your device before turning on your system.

To move a tape drive using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Notify the users on the system about system shutdown as a result of moving the tape drive. See the "Communicating With the Users on Your System" chapter of this manual.

Note It is recommended that you shut down and power off your computer while moving devices on the SCSI or HP-IB bus.

---

3. Remove any media currently loaded in the tape drive.
4. Shut down your system:

   ```
   cd /
   shutdown grace_period_in_seconds
   ```

   For additional information, see Chapter 3, "Starting and Stopping HP-UX" of this manual.

5. Halt the system:

   ```
   reboot -h
   ```

6. Turn off the computer.
7. Turn off, unplug, and disconnect the tape drive.
8. Physically move the tape drive. Refer to the documentation shipped with your disk drive if you intend to change the HP-IB or SCSI bus address. Ensure that each device on the SCSI or HP-IB bus has a unique bus address.
9. Turn on the tape drive.

Note For SCSI devices, it is important to power up your device before turning on your system.

---

10. Turn on the computer.
11. Log in. Ensure that you have superuser capabilities.
12. Add the tape drive to the system at the new hardware location; refer to the *Installing Peripherals* manual.
13. Modify any automated backup processes scheduled by the `cron` utility to reflect an updated device file for the tape drive. See the "Backing Up and Restoring Your Data" chapter of this manual and *cron*(1M) in the *HP-UX Reference Manual.*
14. Reload media into the tape drive.
15. Update any software application configurations that used the moved tape drive. Refer to your software application documentation for specific instructions.

Moving a Terminal or Modem

Moving a terminal or modem on your system requires that you remove the terminal or modem and then add the terminal or modem using the new hardware location.

To move a terminal or modem using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. If users will be affected, notify the users on the system about the device being moved. See the "Communicating With the Users on Your System" chapter of this manual. The terminal to be moved cannot be in use, it must be inactive.
3. Create a backup copy of the `/etc/inittab` file:

   ```
   cp /etc/inittab /etc/inittab.old
   ```

4. Turn off, unplug, and disconnect the terminal or modem.
5. Add the terminal or modem to the system at the new hardware location; refer to the *Installing Peripherals* manual.
6. Modify the `/etc/inittab` file to reflect the new device file for the terminal or modem.
7. Activate the updated `/etc/inittab` file; type:

   ```
   /etc/telinit q
   ```

8. Update any software application configurations that used the relocated terminal or modem. Refer to your software application documentation for specific instructions.

Moving the System Console

Moving the system console on your system requires that you update the console information in stable non-volatile memory and reconfigure the kernel to define a new hardware path to the system console.

Caution Making a mistake during this procedure could make your system inaccessible from the system console. HP recommends testing the new hardware location for the system console with a terminal before actually changing ISL to reflect the new hardware path to the console.

To move the system console using HP-UX commands:

5

1. Ensure that you have superuser capabilities.
2. This procedure requires you to reboot your system. Notify the users on the system about the system being unavailable for a short time. See the "Communicating With the Users on Your System" chapter of this manual. You will need to regenerate a new kernel and reboot to update the system.
3. Determine the hardware path for the new location of the system console. Refer to Chapter 10, "System Architectures", of *How HP-UX Works: Concepts for the System Administrator*. You can use the `ioscan -f -C tty` command to help you locate an address.

4. Update the kernel configuration file, typically `/etc/conf/gen/S800`, to reflect the new console hardware path. Look for the entry that looks similar to:

   ```
   console on  cio_ca0.mux0  at 4.1;
   ```

   or

   ```
   console on   default;
   ```

   Your entry will vary based on the model of your Series 800 machine.

   `default` defers the hardware path for the console specified by ISL.

5. Regenerate the kernel, halt the system, and turn off the power. Halt the system using the `-h` option to the `shutdown` command. Refer to Chapter 2, "Constructing an HP-UX System".

   Your kernel and ISL now refer to different hardware addresses for the system console.

6. To test the new hardware address before moving your system console, install a terminal at the new hardware address.

7. Turn on your computer and allow the new kernel to boot. The initial message to override the autoboot sequence should appear on the sytem console. Do not override the autoboot sequence. The terminal installed at the new system console hardware address should display the boot sequence of the kernel.

Note If the kernel boot sequence does not appear on the terminal installed at the new system console hardware address:

- Ensure that the terminal is connected to port zero (0) if the terminal is connected to a multiplexer (ADP panel).
- Check the hardware connections. The terminal cable and interface card connections should be properly seated.
- Check that the terminal is turned on and that the terminal configurations are set correctly; baud rate, parity, etc.. Refer to the *Installing Peripherals* for additional information.

If after checking the hardware connections the terminal does not respond, do not continue with step 8 of this procedure; Instead:

5

a. Cycle power on your computer.

b. Boot your system, override the autoboot sequence, boot from your primary boot path, and enter "y" to interacting with IPL.

c. Boot your backup kernel, typically `/SYSBCKUP`. Type:

```
hpux /SYSBCKUP
```

d. Check the console hardware address defined in the kernel configuration file (`/etc/conf/gen/S800`).

e. Either regenerate your kernel and try again, or return your system to it's original condition by coping the backup kernel to `/hp-ux` and rebooting. See Chapter 2, "Constructing an HP-UX System".

f. DO NOT overwrite your backup kernel. You will destroy the only bootable kernel on your system with the correct hardware address defined for the system console.

8. log in. Ensure that you have superuser capabilities.

9. Reboot your system:

```
cd /
/etc/shutdown -r time_period_in_seconds
```

10. Override the autoboot sequence, boot from your primary boot path, and anwer "y" to interacting with IPL . You need to enter ISL to perform system console hardware path configuration.

11. Reconfigure the hardware path for your system console. Refer to your system's owner's guide to change the system console's hardware path.

12. Turn off the the system console and your computer.

13. Replace the terminal used to test the new hardware address with the system console:

    - Disconnect the terminal.
    - Disconnect and reconnect the system console.
    - Ensure that the hardware connections are properly seated.

    If you are moving your system console to a multiplexer (ADP panel), only port zero (0) can be used for the system console.

14. Turn on your system console and your computer. Your boot sequence should appear on your newly relocated system console.

Moving a Disk Drive Used for File Systems and Swap

Note

Moving the root disk is a special case and the instructions given in this section do not apply. Refer to "Moving a Disk Drive Containing the Root File System" in this chapter.

If you are moving the root section on your disk, refer to the "Moving the Root File System from Section to Section" in the Chapter 6, "Managing the File System" chapter of this manual.

If you are using the Logical Volume Manager (LVM), refer to the "Moving the Root File System from a Disk Section to a Logical Volume" in the "Managing Logical Volumes" chapter of this manual to reinstall the root file system.

5

To move a disk drive used for file systems or swap using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Back up the disk drive to be moved; see the "Backing Up and Restoring Your Data" chapter of this manual.
3. Notify the users on the system about system shutdown as a result of moving the disk drive. See the "Communicating With the Users on Your System" chapter of this manual.
4. If your system is an NFS server and file systems on the disk you are moving are exported, unmount the file systems from the NFS client. If you do not unmount the file systems from the client, the client will receive NFS error messages when accessing the files on the disk.

   To find the NFS clients, log in to the NFS server and look at the `/etc/exports` file. Refer to *exports*(4) in the *HP-UX Reference Manual*.

   Notify the users on the NFS client systems that data on the disk to be relocated will be temporarily inaccessible. See "Communicating With the Users on Your System".

   Log in directly to each NFS client and unmount the file systems using HP-UX commands. Refer to Chapter 6, "Managing the File System" for specific instructions on unmounting file systems.

5. Determine the hardware address for the new location. See Chapter 10, "System Architectures" in the *How HP-UX Works: Concepts for the System Administrator* manual.

   You can use the `ioscan` command to help you. For example:

```
/etc/ioscan -f

Class        LU  H/W Path Driver                  H/W Status        S/W Status
==========================================================================
processor    -   0        processor               ok(0x401)         ok
cio          -   4        cio_ca0                 ok(0x1000)        ok
hpib         -   4.0      cio_ca0.hpib0           ok(0x2)           ok
disk         1   4.0.1    cio_ca0.hpib0.disc1     ok(0x22b)         ok
disk         2   4.0.2    cio_ca0.hpib0.disc1     ok(0x260)         ok
tape_drive   0   4.0.3    cio_ca0.hpib0.tape1     ok(0x178)         ok
tty          1   4.1      cio_ca0.mux0            ok(0x7)           ok
lan          1   4.4      cio_ca0.lan0            ok(0x6)           ok
memory       -   12       memory                  ok(0x800)         ok
```

   Make note of the LU number for your disk and the driver path. You will need to use the LU number for creating your new disk's device files. You will need the driver path for updating the kernel configuration file.

6. Ensure that a driver is not bound to the new hardware address for the disk. Use the `-k` option to the `ioscan` command to list the current driver path and hardware address bindings.

   If there is a driver path bound to the new hardware address for the disk:

   - Ensure that there is not a device connected at the new hardware address.
   - Unbind the driver using the `rmsf` command.
   - Reboot the system.

   For example,

```
cd /dev
rmsf -H new_hardware_path
cd /
shutdown -r time_in_seconds
```

   Log in. Ensure that you have superuser capabilities.

7. Decide whether or not to keep the same LU number for the disk.

Note By preserving your LU number, you do not need to update the `/etc/checklist` file.

Perform the following three steps to preserve your disk's LU number or skip to step 8.

a. Bind the device driver to the hardware path:

```
/etc/ioscan -M driver_path -H new_hardware_path
```

b. Unbind the old hardware paths from the LU:

```
cd /dev
/etc/rmsf -H old_hardware_path
```

```
rmsf: Warning - Cannot remove driver_name at old_hardware_path
```

Ignore the error message.

c. Bind the old LU number to the new hardware path:

```
/etc/insf -H new_hardware_path -l LU_number
```

Note The `insf` command creates device files relative to the current directory. Ensure that you are located in the `/dev` directory to create device files.

8. Shut down and halt your system:

```
cd /
shutdown -h time_in_seconds
```

9. Turn off you system and turn off the disk drive.
10. Physically move the disk drive to the new hardware location.
11. Turn on the disk drive.

Note For SCSI devices, it is important to power up your device before turning on your system.

12. Turn on the computer.
13. Log in. Ensure that you have superuser capabilities.
14. If you decided to let `insf` reassign a new LU number to the newly located disk:

    a. Locate the device files corresponding to the newly created LU number:

    ```
    ls /dev/dsk/cLU*
    ```

    where *LU* is the new LU number. For example:

    ```
    ls /dev/dsk/c6*

    /dev/dsk/c6d0s0    /dev/dsk/c6d0s12   /dev/dsk/c6d0s2    /dev/dsk/c6d0s6
    /dev/dsk/c6d0s1    /dev/dsk/c6d0s13   /dev/dsk/c6d0s3    /dev/dsk/c6d0s7
    /dev/dsk/c6d0s10   /dev/dsk/c6d0s14   /dev/dsk/c6d0s4    /dev/dsk/c6d0s8
    /dev/dsk/c6d0s11   /dev/dsk/c6d0s15   /dev/dsk/c6d0s5    /dev/dsk/c6d0s9
    ```

    b. Create a backup copy of the `/etc/checklist` file:

    ```
    cp /etc/checklist /etc/checklist.old
    ```

    c. Edit `/etc/checklist` to replace the file system device files with device files that reflect the new LU number. Choose the device file matching the same section of the disk as the old device file. For example, replace `/dev/dsk/c3d0s13` with `/dev/dsk/c6d0s13` to retain section 13.

    d. Mount the file system on the newly relocated disk you specified in the local `/etc/checklist` file:

    ```
    /etc/mount -a
    ```

15. If your system is an NFS server, remount file system(s) on any NFS client systems. As superuser on the NFS client system(s), type:

    ```
    /etc/mount -a
    ```

16. Update any software application configurations that use the relocated disk drive. Refer to your software application documentation for specific instructions.

Moving a Disk Drive Containing the Root File System

moving using HP-UX commands| moving using HP-UX commands| Moving a disk drive containing the root file system requires regenerating your kernel, updating several system files, and booting in attended mode to select the new root file system to boot from.

Note

- If you are moving the root file system to another section on the disk, refer to the "Moving the Root File System from Section to Section" in the Chapter 6, "Managing the File System" chapter of this manual.
- If you are using the Logical Volume Manager (LVM), refer to the "Moving the Root File System from a Disk Section to a Logical Volume" in the "Managing Logical Volumes" chapter of this manual to reinstall the root file system.
- This procedure assumes that your root disk is accessible and functioning properly. This procedure does not describe repairing the boot area or data on your root disk.

5

To move the root disk drive using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Back up the disk drive to be moved; see the "Backing Up and Restoring Your Data" chapter of this manual.
3. Notify the users on the system about the system shutdown to move the root disk. See the "Communicating With the Users on Your System" chapter of this manual.

Note It is recommended that you shut down and power off your computer while physically moving devices on the SCSI or HP-IB bus.

---

4. If your system is an NFS server and file systems on the disk you are moving are exported, unmount the file systems from the NFS client. If you do not unmount the file systems from the client, the client will receive NFS error messages when accessing the files on the disk.

   To find the NFS clients, log in to the NFS server and look at the `/etc/exports` file. Refer to *exports*(4) in the *HP-UX Reference Manual.*

   Notify the users on the NFS client systems that data on the disk to be relocated will be temporarily inaccessible. See "Communicating With the Users on Your System".

   Log in directly to each NFS client and unmount the file systems using HP-UX commands. Refer to Chapter 6, "Managing the File System" for specific instructions on unmounting file systems.

5. Determine the hardware address for the new location. See Chapter 10, "System Architectures" in the *How HP-UX Works: Concepts for the System Administrator* manual.

   You can use the `ioscan` command to help you. For example:

```
/etc/ioscan -f

Class        LU  H/W Path Driver                 H/W Status        S/W Status
==============================================================================
processor    -   0        processor              ok(0x401)         ok
cio          -   4        cio_ca0                ok(0x1000)        ok
hpib         -   4.0      cio_ca0.hpib0          ok(0x2)           ok
disk         1   4.0.1    cio_ca0.hpib0.disc1    ok(0x22b)         ok
disk         2   4.0.2    cio_ca0.hpib0.disc1    ok(0x260)         ok
tape_drive   0   4.0.3    cio_ca0.hpib0.tape1    ok(0x178)         ok
tty          1   4.1      cio_ca0.mux0           ok(0x7)           ok
lan          1   4.4      cio_ca0.lan0           ok(0x6)           ok
memory       -   12       memory                 ok(0x800)         ok
```

   Make note of the LU number for your root disk and the driver path. You will need to use the LU number for creating your new root disk's device files. You will need the driver path for updating the kernel configuration file.

6. Ensure that a driver is not bound to the new hardware address for the root disk. Use the `-k` option to the `ioscan` command to list the current driver path and hardware address bindings.

If there is a driver path bound to the new hardware address for the root disk, ensure that there is not a device connected at the new hardware address and unbind the driver using the `rmsf` command and rebooting the system. For example,

```
cd /dev
rmsf -H new_hardware_path
cd /
shutdown -r time_in_seconds
```

Log in. Ensure that you have superuser capabilities.

7. Decide whether or not to keep the same LU number for the root disk.

Note By preserving your LU number, you do not need to update the `/etc/checklist` file.

5

Perform the following three steps to preserve your root disk's LU number or skip to step 8.

a. Bind the device driver to the hardware path:

```
/etc/ioscan -M driver_path -H new_hardware_path
```

b. Unbind the old hardware paths from the LU:

```
cd /dev
/etc/rmsf -H old_hardware_path
```

```
rmsf: Warning - Cannot remove driver_name at old_hardware_path
```

Ignore the error message.

c. Bind the old LU number to the new hardware path:

```
/etc/insf -H new_hardware_path -l LU_number
```

Note The `insf` command creates device files relative to the current directory. Ensure that you are located in the `/dev` directory to create device files.

---

8. Update the autoexecute (auto) file on the root disk boot area to reflect the new hardware address with the `mkboot` command, for example:

   /etc/mkboot -a "hpux (*new_hardware_path*;*section_num***)/hp-ux"** *root_disk_dev_file*

   where:

new_hardware_path	is the new hardware address for the root disk. If omitted, the hardware address defaults to the primary boot path.
section_num	is the section number of the root partition.
root_disk_dev_file	is the device file root disk's LIF area.

Caution The `mkboot` command overwrites the entire contents of the autoboot string. To view the current contents of the autoexecute file, type:

```
/usr/bin/lifcp root_device_file:AUTO -
```

The `-` copies the autoexecute file to standard out. You can specify a file name.

5

9. Update the kernel configuration file, typically `/etc/conf/gen/S800`, to reflect the new root disk hardware path. Also update the `swap` and `dumps` entries if they are located on the same disk as the root file system. Look for the entries that looks similar to:

```
root on   cio_ca0.hpib0.disc1 at 4.0.1 section 13;
swap on   cio_ca0.hpib0.disc1 at 4.0.1 section 15;
dumps on   cio_ca0.hpib0.disc1 at 4.0.1 section 15;
```

or

```
root on    hpib1.disc1 at 4.0  section 13;
swap on    hpib1.disc1 at 4.0  section 15;
dumps on    hpib1.disc1 at 4.0  section 15;
```

or

```
root   on      default section 13;
swap   on      default section 15;
dumps   on      default section 15;
```

Your entry will vary based on the model of your Series 800 machine.

`default` defers the root file system location to the disk specified by ISL; primary boot path, alternate boot path, or the interactively given hardware address.

10. Regenerate the kernel. Refer to Chapter 2, "Constructing an HP-UX System". After you have installed your new kernel and your system is halted, turn off the root disk drive, physically move it to it's new hardware location, and turn it on.

Note It is recommended that you shut down and power off your computer while physically moving devices on the SCSI or HP-IB bus.

For SCSI devices, it is important to power up your device before turning on your system.

---

11. Boot your system and override the autoboot. Refer to "Booting the System" in Chapter 3, "Starting and Stopping HP-UX". Do not boot from the primary or alternate boot path.

12. Enter the hardware address of you newly located root disk and boot your system.
13. Answer "y" to interacting with IPL.
14. Set the primary boot path to the new hardware path with the **primpath** command, type:

    `primpath`

    The system will prompt you to enter the primary boot path.
15. Verify the contents of your auto-execute file on the newly located root disk. To list your auto-execute file containing the **hpux** command string, type:

    `lsautofl`

    ```
    Auto-execute file contains:
     hpux (4.0.1;13)/hp-ux
    ```

    If the autoexecute file contains the correct string, skip to step 16. If you forgot to update the autoexecute file in step 8 or the autoexecute file is incorrect, reset the autoexecute file.

    To set the auto-execute file to default to the primary boot path, type:

    `hpux set autofile "hpux (;13)/hp-ux"`

    To set the auto-execute file to particular harware path, type:

    `hpux set autofile "hpux (`*hardware_boot_path*`;13)/hp-ux"`
16. Boot your system with the command line specified in the auto-execute file Type in the contents of the auto-execute file

---

Note If you did not preserve your LU number, a new LU is assigned to the newly located root disk and the associated device files are created.

---

17. Log in. Ensure that you have superuser capabilities.

5

18. If you decided to let `insf` reassign a new LU number to the newly located root disk:

    a. Locate the device files corresponding to the newly created LU number:

    ```
    ls /dev/dsk/cLU*
    ```

    where *LU* is the new LU number. For example:

    ```
    ls /dev/dsk/c6*

    /dev/dsk/c6d0s0    /dev/dsk/c6d0s12   /dev/dsk/c6d0s2    /dev/dsk/c6d0s6
    /dev/dsk/c6d0s1    /dev/dsk/c6d0s13   /dev/dsk/c6d0s3    /dev/dsk/c6d0s7
    /dev/dsk/c6d0s10   /dev/dsk/c6d0s14   /dev/dsk/c6d0s4    /dev/dsk/c6d0s8
    /dev/dsk/c6d0s11   /dev/dsk/c6d0s15   /dev/dsk/c6d0s5    /dev/dsk/c6d0s9
    ```

    b. Create a backup copy of the `/etc/checklist` file:

    ```
    cp /etc/checklist /etc/checklist.old
    ```

    c. Edit `/etc/checklist` to replace the root and other file system device files with device files that reflect the new LU number. Choose the device file matching the same section of the disk as the old device file. For example, replace `/dev/dsk/c3d0s13` with `/dev/dsk/c6d0s13` to retain section 13.

    d. Mount the file system on the newly relocated disk you specified in the local `/etc/checklist` file:

    ```
    /etc/mount -a
    ```

19. If your system is an NFS server, remount file system(s) on any NFS client systems. As superuser on the NFS client system(s), type:

    ```
    /etc/mount -a
    ```

20. Update any software application configurations that use the relocated root disk. Refer to your software application documentation for specific instructions.

Removing Peripherals Using HP-UX Commands

If users will be affected, notify the users on the system of the device location change. See the "Communicating With the Users on Your System" chapter of this manual.

To remove peripherals:

1. Ensure that the device does not contain information critical to the operation of the system or users. For removing mass storage devices, copy critical data to another device.
2. Remove the device from your application software configuration. Refer to your software application documentation for specific instructions.
3. (optional) Remove the device from the HP-UX configuration:
   a. If there is no future need for a device of this type at this hardware address, remove the device file(s) associated with the device.
   b. If this is the only device of it's interface type and there is no future need for this device interface type on your system, reconfigure the kernel to remove the device driver. Removing the device driver can also reduce the physical size of the kernel (`/hp-ux`). See Chapter 2, "Constructing an HP-UX System".
4. Update any software application configurations that use the relocated local printer. Refer to your software application documentation for specific instructions.

Removing a Printer or Plotter

If you are removing a printer or plotter using HP-UX commands, see the "Managing Printers and Printer Output" chapter of this manual.

Update any software application configurations that use the removed printer. Refer to your software application documentation for specific instructions.

Removing a Tape Drive

To remove a tape drive using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Notify the users on the system about system shutdown as a result of removing a tape drive. See the "Communicating With the Users on Your System" chapter of this manual.

Note It is recommended that you shut down and power off your computer while physically removing devices from the SCSI or HP-IB bus.

---

3. Shut down your system:

   ```
   cd /
   shutdown grace_period_in_seconds
   ```

   For additional information, see Chapter 3, "Starting and Stopping HP-UX" of this manual.

4. Halt the system:

   ```
   reboot -h
   ```

5. Turn off the computer.
6. Remove any media currently loaded in the tape drive.
7. Turn off, unplug, and disconnect the tape drive.
8. Turn on the computer.
9. Log in. Ensure that you have superuser capabilities.
10. Remove/Modify any backup process scheduled by the `cron` utility that expect the tape drive to be present. See the "Backing Up and Restoring Your Data" chapter of this manual and *cron*(1M) in the *HP-UX Reference Manual*.
11. Update any software application configurations that use the removed tape drive. Refer to your software application documentation for specific instructions.

Removing a Terminal and Modem

Note Removing the system console is not supported.

To remove a terminal or modem using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. If users will be affected, notify the users on the system about the device being moved. See the "Communicating With the Users on Your System" chapter of this manual. The terminal or modem to be moved cannot be in use, it must be inactive.
3. Edit the `/etc/inittab` file to remove any `getty` entries for the terminal. Refer to *inittab*(4) in the *HP-UX Reference Manual*.
4. Activate the updated `/etc/inittab` file; type:

   ```
   /etc/telinit q
   ```

5. Unplug and disconnect the terminal or modem.
6. Update any software application configurations that use the removed terminal or modem. Refer to your software application documentation for specific instructions.

Removing a Disk Drive Used for File Systems and Swap

Note Removing the disk containing the root file system is not possible. If you are moving the disk drive, see the "Moving a Disk Drive Used for File Systems or Swap" section of this chapter for further instruction.

To remove a disk drive using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Notify the users on the system about system shutdown as a result of removing a disk drive. See the "Communicating With the Users on Your System" chapter of this manual.

5

Note It is recommended that you shut down and power off your computer while adding to or removing devices from the SCSI or HP-IB bus.

3. If your system is an NFS server and file systems on the disk you are moving are exported, unmount the file systems from the NFS client. If you do not unmount the file systems from the client, the client will receive NFS error messages when accessing the files on the disk.

   To find the NFS clients, log in to the NFS server and look at the `/etc/exports` file. Refer to *exports*(4) in the *HP-UX Reference Manual*.

   Notify the users on the NFS client systems that data on the disk to be relocated will be temporarily inaccessible. See "Communicating With the Users on Your System".

   Log in directly to each NFS client and unmount the file systems using HP-UX commands. Refer to Chapter 6, "Managing the File System" for specific instructions on unmounting file systems.
4. Update the `/etc/checklist` on all NFS client systems to remove the mount entries for file systems that are on the disk drive being removed.
5. Create a backup copy of the `/etc/checklist` file:

```
cp /etc/checklist /etc/checklist.old
```

6. Shut down your system; type:

   ```
   cd /
   shutdown grace_period_in_seconds
   ```

   For additional information, see Chapter 3, "Starting and Stopping HP-UX" of this manual.

7. Edit the `/etc/checklist` file to remove any mount entries for the disk being removed.

8. If you are removing the disk drive your kernel uses for primary swap, reconfigure the kernel to reassign your primary swap device. Refer to Chapter 6, "Managing the File System".

9. Halt the system:

   ```
   reboot -h
   ```

10. Turn off the computer.

11. Turn off, unplug, and disconnect the disk drive.

12. Turn on your system.

13. Log in. Ensure that you have superuser capabilities.

14. Update any software application configurations that use the removed disk drive. Refer to your software application documentation for specific instructions.

6

Managing the File System

As a system administrator responsible for managing your HP-UX file system, you will often be performing the following major tasks:

- Creating new file systems.

  If you have LVM on your system, you will create a file system in a logical volume.

  If you don't use LVM, you will create a file system in a disk section.

- Adding (mounting) and removing (unmounting) file systems from the file system tree.
- Monitoring and controlling the space consumed by users' files.

In most cases, this chapter describes how to accomplish these tasks using both SAM and HP-UX commands. SAM is recommended because SAM enables you to perform many of these tasks easily and with less chance of errors.

SAM will not be able to perform any activities related to disk quotas.

Terms Used in this Chapter

The following terms appear frequently in discussions in this chapter. The list of terms appears here so it can be located easily. You can scan the list now and refer to it later.

block (Also known as a **sector**.) The smallest unit of relocatable space on a disk, with a size of 1024-bytes.

block device A hardware device that transmits and receives data in multiple-byte blocks (rather than by streams of individual bytes) or does block-buffered input/output.

block special file A special file associated with a mass storage device (such as a hard disk or tape cartridge drive) that transfers data in multiple-byte blocks, rather than by series of individual bytes. See "device file."

CD-ROM file system A Read Only Memory file system on Compact Disk. You can read data from a CD-ROM file system, but you cannot write to one.

6

character special file A special file associated with I/O devices that transfer data byte-by-byte. Typical byte-mode I/O devices include printers, nine-track magnetic tape drives, and disk drives when accessed in "raw" mode. Disk drive access via character devices is typically faster than via block devices. Character device files are sometimes called "raw" device files.

cylinder On disk drives consisting of several disks, the arrangement of disk tracks under read/write heads that are in the same relative position.

device file Used to identify and communicate with a device (such as a disk) or pseudo device (such as a logical volume). There are block device files (used for transmitting data in multiple-byte blocks) and character device files (used for transmitting data byte-by-byte). Device files are typically stored in the `/dev` directory.

The full path name of block device files follows a convention: for disk sections, `/dev/dsk/c`*n*`d0s`*n*, and for logical volumes, `/dev/vg`*nn*`/lvol`*n*.

The full path name of character device files follows a convention: for disk sections, `/dev/rdsk/c`*n*`d0s`*n*, and for logical volumes, `/dev/vg`*nn*`/rlvol`*n*.

device swap space A disk or disk section reserved exclusively as swap space.

disk quotas Disk usage limits that a system administrator can assign to users of a file system.

disk arrays A group of disks that can function in one of several possible factory-configured modes. In **independent** mode, each disk has a unique address, but all disks use a single power supply. In **striped mode** (2-way or 4-way), disks are synchronized to appear as a single logical unit, and the equivalent section on each of the disks is logically combined to function as one section.

disk section On Series 800/600 computers, a logical division of space on a hard disk. To HP-UX, each section appears as an independent file system or swap area: that is, a separate entity with its own device file. (Disks used on Series 300/400 computers do not have multiple sections, they contain only one section.)

file system The organization of files on storage devices. The term "file system" can refer either to the entire file system tree or to a subsection of that file system (contained within either a disk section or a logical volume) that can be mounted or unmounted from the tree.

fragment A part of a block. The end of a file that is not a whole block is typically stored as a fragment. The size of a fragment can be specified; small fragments adversely affect performance but leave less wasted space.

HFS file system A file system in which the files are arranged into hierarchical directories on a disk.

independent mode In disk arrays, each disk module in the array functions as a single logical device.

inode A data structure containing information about a file, such as file type, pointers to data, owner, group, and protection information.

kernel The actual operating system program that executes and runs, controlling the processes, hardware, file system, and so on.

logical volume A *logical* (rather than physical) construction that is a map of data stored on physical volumes (disks). A logical volume can be conceptualized as a storage device of flexible size. The data in a logical volume can map to one or more physical volumes.

long file names File names using more than 14 (but not exceeding 255) characters. Long file names are incompatible with file systems configured for file names with 14 character limit.

mount To add an auxiliary (removable) file system to an active existing file system.

6

mount directory A directory in an existing file system that serves as the root directory (the mount point) of a mounted file system.

NFS client A machine that mounts, via the network, a file system that is located on a remote **NFS server**.

NFS file system An HFS-type file system that is accessible over a network via the NFS Services product.

NFS server A computer with file systems that are being accessed by remote computers, or **NFS clients**.

root directory The highest level directory in a file system. In any mountable file system (any file system other than the `root` file system) the root directory is the mount directory.

The `/` directory, also known as the `root` directory, is the highest level in the HP-UX file system overall; the `/` file system cannot be unmounted because it contains the running operating system.

root file system Or `root` (`/`); the file system that contains the kernel and other operating system files.

root volume group The volume group that contains `root`, the file system that contains the kernel and operating system files.

short file names File names consisting of 14 or fewer characters. Short file names are compatible with file systems configured for long file names.

single-user mode When a computer system is accessible to only one user, usually the system administrator.

striped-mode In disk arrays, two or four disks are factory-configured to function as a single logical unit. The equivalent section on each of the disks is logically combined to function as a single section. Data transfers to the disks in the array occur in parallel.

swap space Space on a disk used for temporarily storing the process image.

unmount To remove an auxiliary file system from an active existing file system.

volume group A group of disks whose pooled space contains logical volumes; logical volumes can contain file systems, swap space, or raw data.

Overview of HP-UX File Systems

This section briefly describes the characteristics of the file systems you will work with on your computer system. The manual *How HP-UX Works: Concepts for the System Administrator* discusses file systems conceptually and in detail.

What Does "File System" Mean?

In one sense, the word "file system" refers to the entire HP-UX file system tree, the organization of all files on the system.

The HP-UX file system is a hierarchical, upside down tree-like structure in which the files—like leaves—are at the bottom or the ends of a branching structure that leads upward through subdirectories to a single **root** (written "/") directory. A diagram of how a typical HP-UX file system structure appears to a user is shown in Figure 6-1.

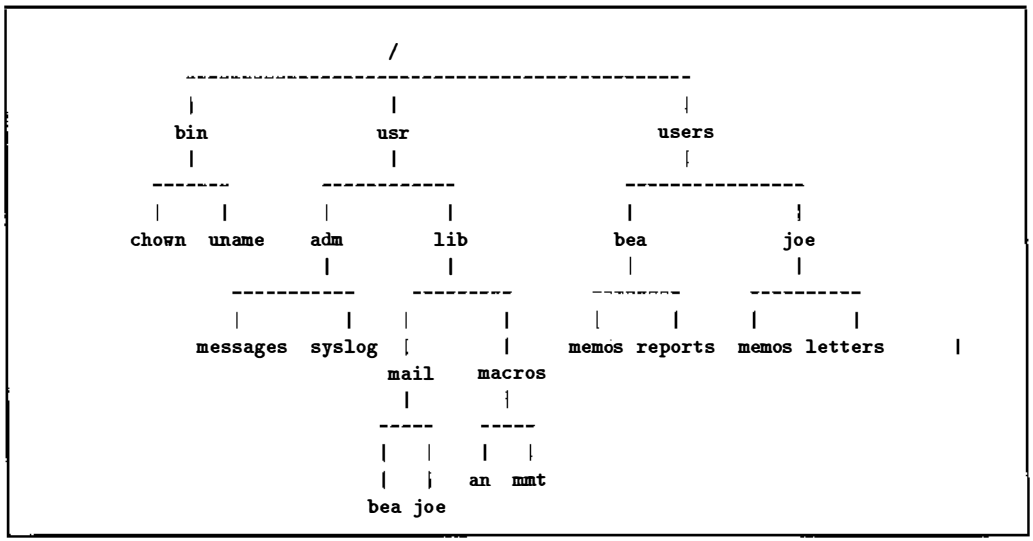


Figure 6-1. Typical HFS File System Structure

Mountable File Systems

In another sense, the word "file system" refers to the specific collection of files on a storage device such as a disk section or a logical volume.

You can create a new file system (in a disk section or logical volume) that consists only of a file system structure, no files, using the *newfs*(1M) command. Once created, the file system, even though it is empty, encompasses the disk section or logical volume.

To use or access the file system on the disk section or logical volume, you need to **mount** that file system to the existing file system tree. Use the *mount*(1M) command, described later. Except for the `root` (/) file system on the system disk, you can mount and unmount all file systems on disks or logical volumes to and from the existing HP-UX file system tree.

You refer to an auxiliary file system by the name of the device file associated with the disk section or logical volume that contains the file system. You can **mount** the auxiliary file system by attaching it to a directory (the mount directory) in the root file system.

Figure 6-2 shows how you can mount a file system on one disk or disk section to the existing file system. In the example, the files on the disk section, `/dev/dsk/c5d0s8` join the hierarchy of the existing file system at the mount directory `/users`.

```
BEFORE MOUNTING
   ---------------------          ---------------------
   |        /          |          |        /          |
   |        |          |          |        |          |
   | -------------     |          | -------------     |
   | |     |     |     |          | |     |     |     |
   | bin   usr  users  |          | Al    Bea   Bob   |
   ---------------------          ---------------------

      ROOT FILE SYSTEM            AUXILIARY FILE SYSTEM
      /dev/dsk/c0d0s13               /dev/dsk/c5d0s8

AFTER MOUNTING, using the command: mount /dev/dsk/c5d0s8 /users
            ---------------------
            |         /         |
            |         |         |
            |   -------------   |
            |   |     |     |   |   /dev/dsk/c0d0s13
            |  bin   usr  users |
            |               |   |
            |               |   |
            ----------------|----
                            |
                   ---------|-----------
                   | -------------     |
                   | |     |     |     |
                   | Al    Bea   Bob   |   /dev/dsk/c5d0s8
                   |                   |
                   -----------------------
```

Figure 6-2. Mounting a File System

You can unmount a file system, too, and then mount it again at a different mount point.

When users traverse the file system, they can move from the `/` directory to the files in `/users/Bob` as easily as they can move from `/` to the files in `/usr` even though `/usr` and `/users/Bob` are on different disks. As a user moves from one part of the HP-UX file system to another, it isn't apparent that the file system actually consists of separate file systems on different devices.

Listing Mounted File Systems

To see a list of the file systems mounted on your system, use the *bdf*(1M) command. For example:

```
bdf
Filesystem          kbytes    used   avail capacity Mounted on
/dev/dsk/c0d0s13    484960  239992  196472    55%   /
/dev/dsk/c5d0s8     237810   47943  166086    22%   /users
.
.
```

In the above example listing, the file system on device (the devices are disk sections, in this case) **/dev/dsk/c5d0s8** is mounted at the mount directory **/users** on the root (**/**) file system on **/dev/dsk/c0s0s13**. You can see this in Figure 6-2.

6

Types of File Systems

The principal types of file systems used by HP-UX are the **HFS**—or high performance file system, the **NFS**, or network file system, and the **CDFS**, or CD-ROM file system.

HFS File Systems. HFS is an acronym for High-performance File System. HFS file systems physically reside on mass storage devices, usually hard disk drives.

NFS File Systems. NFS is an acronym for Network File Services. NFS file systems are remote HFS file systems that are accessible over a network that can be used in a local file system.

CD-ROM File Systems. CD-ROM is an acronym for Compact Disk Read-Only Memory. The information on the CD is virtually permanent; you can read data from a CD, but you cannot write to one. Data on a CD is prepared and mastered using a specialized publishing process. A CD-ROM file system (CDFS) allows easy retrieval of large amounts of information that requires no modification.

The arrangement of files in a CD-ROM file system is tree-like as in HFS file systems. You can use HP-UX commands to list, print, or copy files in the CD-ROM file system. However, some commands, such as `fsck` or `mkfs` for example, are not supported because of the read-only nature of a CD-ROM file system.

Disk Sections

On Series 800 computers, a disk can be divided into fixed-sized logical **sections**. A hard disk can contain one large section, but, usually, it is divided into several sections. To HP-UX, each disk section appears as if it were a separate disk drive.

If you look at the file `/etc/disktab` (type: `more /etc/disktab`), you can see the sixteen possible sections (available to most disks) and their sizes. (See the later section "Referring to the /etc/disktab File" for an illustration and examples.)

Usually, you will use a disk section for a file system, although you can use a section for swap space, for raw data, or as a boot area. See the chapter "Managing Swap Space " for more discussion about swap space.

Logical Volumes

The Logical Volume Manager (LVM) enables you to create groups of disks (**physical volumes**) called **volume groups**. You can then subdivide the space within a volume group into **logical volumes**.

A logical volume is like a disk section in that it can contain a file system or swap space. But a logical volume is a "soft" partition of disk space; you can define the size of a logical volume (disk sections have fixed sizes) and you can increase the size of an existing logical volume (you cannot change the size of a disk section).

You can extend a logical volume, if necessary, to a size greater than one disk (that is, a logical volume can span disks). You can mirror a logical volume if you have the separately orderable mirroring product, MirrorDisk/UX.

This chapter discusses creating logical volumes for file systems. However, if you want to add a disk to volume group, you need to refer to chapter 8, "Managing Logical Volumes." There, you will find more detailed discussions about setting up and managing volume groups and logical volumes.

Disk Arrays

A **disk array** consists of two or four data disks, and possibly another disk for data backup purposes.

Striped Mode. The disks in the array can be configured to function logically as a single disk, so that the equivalent disk sections on each disk function together as a single larger disk section. When data is written to the array, the array controller writes in parallel to each of the disks in the array, providing a higher data transfer rate than a standalone disk.

The disk sections in a disk array can be used the same as traditional disk sections on a standalone disk. The entire disk array, that is, section 2 of the array, can be used as a physical volume in an LVM volume group.

You cannot use SAM to add disk arrays to your system, but SAM will display disk arrays and allow you to modify the swap and file systems

---

Note LVM does not support the use of 4-way striped disk arrays (HP 2254B or HP2254HA) on either CIO computers or HP-PB machines using the CIO-type adapter for fiber optic peripheral interface card (part number A1749A).

LVM does support 4-way striped disk arrays on HP-PB computers that use HP-PB type fiber optic peripheral interface cards.

Independent Mode. In this mode, each disk in an array can function as an independent disk with a unique logical unit number. While disks in an independent mode array share the same power supply and controller, they can perform concurrent I/O transactions.

Creating File Systems

You can expand your file system's overall capacity by creating new file systems and attaching (mounting) them to the existing file system tree.

You can create a new file system in either a logical volume or a disk section.

Why Create New File Systems?

Reasons to create new file systems might include:

- Growth. You should keep the `root` file system as small as possible; let file system growth occur in other file systems.

  The `root` file system contains the HP-UX operating system files, and it should be kept as small as possible for the sake of performance and efficiency. Keeping `/` separate from general use also makes it more secure.

  You can find more information about the `root` file system and HP-UX system files toward the end of this chapter

- You should separate file systems that have different uses and different groups of users. You can handle and maintain them as distinct units.
- You want to create a file system with unique characteristics of ownership or of mirroring.
- All current file systems are getting full.

How Can You Create a File System?

You can use either SAM or a sequence of HP-UX commands to create a file system. SAM is the easier method.

Creating File Systems Using SAM

We recommend you use SAM when possible because SAM can usually accomplish tasks more easily and with less chance of error.

Are You Using LVM?

If you are not sure whether you are running logical volumes or not, before you run SAM, type:

```
/etc/vgdisplay
```

- You are *not* using LVM if you get the message,

  ```
  /etc/vgdisplay: Command not found.
  ```

  You need to create a file system in a disk section. Go to the section "Creating a File System in a Disk Section Using SAM."

- You do use LVM if you get any other message, usually a message that discusses LVM features. Proceed to the next section, "Creating a File System in a Logical Volume Using SAM."

  (See chapter 8, "Managing Logical Volumes," for a detailed description of LVM.)

How to Use SAM

To use SAM,

- Ensure that you have superuser capabilities.
- Type:

  `/usr/bin/sam`

  Activating the (Help) button from a dialog or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

  Choosing an item from the "Help" menu within a functional area gives you information about:

  - the current functional area
  - keyboard navigation within SAM
  - using the SAM help system
  - displaying the version of SAM you are currently running

Pressing the (f1) key gives you context-sensitive information for the object field at the location of the cursor.

See Chapter 1, "Introduction to System Administration" for additional information about using SAM. If you aren't familiar with SAM, take a moment to look at that overview material. There is a help system in SAM to assist you.

Also, the section in Chapter 8, "Managing Logical Volumes", "Guidelines for Using Logical Volumes," can help you make optimal decisions.

Creating a File System in a Logical Volume Using SAM

This procedure describes creating a file system in a logical volume. It describes:

- creating a logical volume in available space within a volume group,
- creating a file system in the logical volume.

Remember, chapter 8, "Managing Logical Volumes," discusses LVM in detail.

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate the [Open] control button.
3. Highlight `Logical Volumes` and activate the [Open] control button.
4. From "Actions" on the menu bar, choose `Create...`.
5. Highlight a volume group in which to create the logical volume.

   Note that at this time, you can see how much space is available on the existing volume groups.

6. Enter a name in the "Logical Volume Name:" field, or leave the field blank and let LVM assign the logical volume name.
7. Enter the size of your logical volume in the `Logical Volume Size (MB)` field. Note that values are rounded up to the nearest multiple of the physical extent size.

   If you want to mirror the logical volume, refer to the mirroring discussion in Chapter 8, "Managing Logical Volumes."

8. Activate the [OK] control button to create the logical volume.

   Note that once SAM creates the logical volume, you can proceed to the subarea in SAM that enables you to build the file systems.

9. From "List" on the menu bar, choose `Local File Systems`.

10. Without highlighting a file system on the `Local File Systems` object list, choose `Add` and then choose `Using the Logical Volume Manager...` from "Actions" on the menu bar. (If no logical volume exists, you must create it in the `Logical Volumes` subarea.)

11. Enter a `Mount Directory`, such as `/projects`. The mount directory can be new or an existing empty directory. This is where the new file system will join the existing file system hierarchy. If the directory you specify does not exist, SAM will create it for you.

12. Highlight the logical volume from the object list.

13. Turn on the "Create new file system" checkbox; accept the checkbox setting for file name length unless you want to specify otherwise.

14. Use defaults for `Set user ID`, `When to Mount:`, and `Access Permissions:` unless you want to change them.

15. Activate [OK] to create the file system.

    SAM creates the file system in the logical volume you created, mounts it, and adds it to the `/etc/checklist` file.

    The file system is now ready to use. You can increase the size of a file system in a logical volume at a later time, if you need to.

Creating a File System in a Disk Section, Using SAM

To create a file system in a disk section, you must have a disk section available. If no disk sections are available, then you must add a disk.

Refer to the *Installing Peripherals* manual for instructions on adding your disk. Then, refer to the section "Adding a Disk" later in this chapter.

When you have added the disk and restarted your computer, you can create a file system using SAM as follows:

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate the (Open) control button.
3. Highlight `Local file Systems` and activate the (Open) control button.
4. When the list of local file systems is displayed, from "Actions" on the menu bar, choose `Add` and then choose `Not Using the Logical Volume Manager...`. A list containing the disk (or disks) you added will appear on the "Add a Local File System" screen.

   If the disk you are adding is not on the list of unused disks:

   - Activate the `Device Missing...` control box. SAM will now direct you through a diagnostic process that enables you to configure your disk.

     Some of the things SAM will ask you to check are:

     - Whether the device is turned on.
     - Whether the device was on when SAM started.
     - The connections.
     - Whether the driver for the disk is configured in the kernel.

   When you have successfully added your disk, you can resume.

5. Highlight the disk you want to add. Also, choose the section you want to use for the file system from the box labled "Select a section:".

   Note that sections are numbered and their sizes are noted. You can refer to the earlier discussion, "Disk Sections", for hints about choosing disk sections. Use the file, `/etc/disktab`, for reference.

6. Enter the name of a new or an existing empty directory in the field: `Mount Directory:`. This is where the new file system will join the existing file system hierarchy. If the directory you specify does not exist, SAM will create it for you.

7. Skip the **Modify Default Options.....** control button, unless you want to change **When to Mount**, the **Access Permissions**, or the **Set user ID** default settings.

   If you want to modify the default options, SAM's (Help) can provide information.

8. Activate the **Create new file system** control button.

9. Activate the (OK) control button to have SAM create the new file system.

   If there is any data in the disk section, SAM will ask you to confirm that it is right to proceed with creating the file system.

   As SAM creates the new file system, it creates new device files for the section, creates the file system, mounts the file system, and adds the file system to `/etc/checklist` (so you can have the file system mounted each time you boot the system).

   Activate (OK) to acknowledge SAM's actions and wait for SAM to add the new file system to the **Local File System** list. The file system is ready for use.

Creating File Systems Using HP-UX Commands

The steps to create a file system with HP-UX commands have a different sequence depending on whether you plan to create it in a logical volume or in a disk section.

Look at Figure 6-3 for an overview of activities involved in creating a file system using HP-UX commands.

```
                    Decide to Create a File System
                                   |
                         Are you using LVM?
                   (Use "/etc/vgdisplay" to find out)
                                   |
                                   |
           -----------YES---------------NO------
           |                                   |
           |                                   |
   Is space available in a           Is disk a section
     volume group?                 of right size available?
           |                                   |
   ---YES------NO----                ---YES------NO----
   |                |                |                |
   |           Add a disk            |           Add a disk
   |       (Use Installing           |       (Use Installing
   |       Peripherals Manual)       |       Peripherals Manual)
   |                |                |                |
   |         Make it a               |     Select a Disk Section
   |        physical volume          |                |
   |                |                |                |
   |           Add it to a           |<----------------
   |         volume group            |
   |                |            Create FS in the disk section
   |<----------------                with "newfs"
   |                                 |
Create logical volume            Add FS with "mount"
   |
Create FS in the logical
  volume with "newfs"
   |
Add FS with "mount"
```

Figure 6-3. Activities in Creating a File System

6

Are You Using LVM?

To tell if you using logical volumes, type:

```
/etc/vgdisplay
```

- You are *not* using LVM if you get the message,

  ```
  /etc/vgdisplay: Command not found.
  ```

  You need to create a file system in a disk section. Go to the section "Creating a File System in a Disk Section Using HP-UX Commands."

- You do use LVM if you get any other message, usually a message that discusses LVM features. Proceed to the next section, "Creating a File System in a Logical Volume Using HP-UX Commands."

  (See chapter 8, "Managing Logical Volumes," for a detailed description of LVM.)

Creating a File System in a Logical Volume Using HP-UX Commands

If you are using LVM (Logical Volume Manager), you can create a file system in a logical volume.

Look at the following checklist of steps.

```
[ ] Calculate a size for the file system's logical volume.
    (How much space does your file system require? Calculate)

[ ] Tell if space is available in the volume group.
    (Command: vgdisplay vol_group_name )

[ ] Create a logical volume.
    (Command: lvcreate - l size_of_logical_vol  vol_group_name )

[ ] Identify the disk's product ID (if you don't know).
    (Command: diskinfo disk's_character_dev_file )

[ ] Create a file system in the logical volume.
    (Command: newfs logical_vol's_character_dev_file  disk_product_ID) )

[ ] Make a mount point directory for the file system.
    (Command: mkdir new_directory )

[ ] Mount the file system for use in the file system tree.
    (Command: mount logical_vol's_block_dev_file  mount_directory) )
```

If you need to add a disk to get the space for a file system, see the later section "Add a Disk." If you don't know if you need to add a disk, that section provides methods to help you find out.

Variations on this Procedure

This section covers a straightforward method of creating a file system in a logical volume. Many of the commands described provide options. LVM related commands are described in Chapter 8, "Managing Logical Volumes." See the later sections in this chapter for more information about *newfs*(1M), *mount*(1M), and *umount*(1M).

Step 1. Calculate the Size Required for the Logical Volume

The logical volume for a file system should be big enough to hold:

1. User data; this is your estimate of what users need for data space.
2. Additional data space reserved for system administrator's use; known as "minfree," this is typically 10% of the data space.
3. Overhead structures of the file system; approximately five percent of the overall space.

These are approximate values. The important point is: not all of an entire logical volume is available for user data.

Example

If you estimate that users require 20 Mbytes, then add 2 MB (ten percent of 20 MB), and increase that subtotal by five percent (approximately 1.1 MB, which you can round up to 2 Mbytes). So make your logical volume large enough to hold 24 Mbytes.

Step 2. Determine if Disk Space is Available for the Logical Volume

Use the */etc/vgdisplay*(1M) command to see if there is room in the volume group.

```
/etc/vgdisplay
```

You will see information for each volume group on the system. If you had only one volume group, the output might resemble the following partial example:

```
--- Volume groups ---
VG Name                 /dev/vg00
VG Status               available
.
PE Size (MB)            4
.
.
Free PE                 58
.
.
```

You can see the physical extent size (`PE Size` is 4 megabytes), and the number of available physical extents (`Free PE`, is 58).

That means that, for this example, there are 232 megabytes available—plenty of room for a 24 megabyte logical volume.

Do You Plan to Create a Mirrored File System?

If you plan to mirror the logical volume that will hold your file system, then you will need to account for required additional disk space.

Double the size you calculated in step 1 (for a single mirror copy) or triple it (for two mirror copies).

For example, a file system that needs a 24 megabyte logical volume will require 48 megabytes of disk space if you want the file system singly mirrored.

What if There is Not Enough Space In a Volume Group?

If you lack space in one volume group, you can check the space available in another volume group.

However, if no volume group has sufficient space, you will need to add a disk to a volume group.

1. To add your disk, you need to refer to *Installing Peripherals* and then "Adding a Disk" later in this chapter.
2. Then, to include that disk in a volume group, go to "Adding Physical Volumes to an Existing Volume Group" in Chapter 8, "Managing Logical Volumes".
3. Return here to continue creating a file system.

Step 3. Create the Logical Volume

Use the *lvcreate*(1M) command to create a logical volume; the command has the form:

```
lvcreate -L N /dev/vgnn
```

where:

- *N* is the size in megabytes you want to make the logical volume.

  If you provide a value for *N* that is not an even multiple of the logical extent size, LVM will round the value you supply upward to the nearest whole extent. (You will receive a message to this effect when you issue the command.)
- *nn* is the volume group number.

When LVM creates the logical volume, it assigns the logical volume a name, such as `lvol`*n*, where *n* is the next number in sequence used by the existing logical volumes.

LVM also creates both the block and character device files for the logical volume.

- Example.

  Create a logical volume of 6 logical extents (24 Mbytes) in volume group `/dev/vg00`:

  ```
  lvcreate -L 24 /dev/vg00
  ```

  LVM reports that it created the logical volume. You can verify it exists by entering:

  ```
  vgdisplay -v /dev/vg00
  ```

  You will find the logical volume listed.

  Also, you can verify that the new device files exist. Enter:

  ```
  ll /dev/vg00
  ```

  You will see two new device files listed; the date and time of creation will help you identify them. For example, if LVM created `lvol4`, you will see the device files:

```
total 0
crw-rw-rw-   1 root     other     64 0x000000 Feb  7 12:53 group
brw-r-----   1 root     other     64 0x000001 Feb  7 17:58 lvol1
brw-r-----   1 root     other     64 0x000002 Feb  9 12:00 lvol2
brw-r-----   1 root     other     64 0x000003 Feb  8 18:47 lvol3
brw-r-----   1 root     other     64 0x000004 Feb 10 08:21 lvol4
crw-r-----   1 root     other     64 0x000001 Feb  7 17:58 rlvol1
crw-r-----   1 root     other     64 0x000002 Feb  9 12:00 rlvol2
crw-r-----   1 root     other     64 0x000003 Feb 10 14:03 rlvol3
crw-r-----   1 root     other     64 0x000004 Feb 10 08:21 rlvol4
```

  This example of the use of *lvcreate*(1M) covers a straightforward case. The *lvcreate*(1M) command has other options available to enable you to create logical volumes with special characteristics. For example, you can give a logical volume a custom name or create it with special permissions. Chapter 8, "Managing Logical Volumes," contains discussions for using these options.

Create a Mirrored File System Logical Volume

Refer to the previous example of creating a logical volume for the file system. When you create a mirrored logical volume, you can use the same *lvcreate*(1M) command, but with the `-m` flag.

For example, the command:

```
lvcreate -L 24 -m 1 /dev/vg00
```

creates a logical volume of 24 extents with one mirror copy. There will be a total of 48 physical extents of disk space used in the volume group `/dev/vg00`, 24 extents on each of two separate disks.

Note Mirroring of logical volumes requires the separately orderable MirrorDisk/UX, B2491A.

Caution For best file system performance, the mirrored disks in a volume group should be the same type. Chapter 8, "Managing Logical Volumes," discusses LVM mirroring in detail.

6

Step 4. Find Out the Disk's Product ID

LVM has allocated physical extents for the logical volume on one of the disks in the volume group. You need to know the disk type (the disk's product ID) to create a file system on it.

If you know the disk's product ID, then you can skip to step 5.

You can find out the disk's product ID in two steps:

1. Type `lvdisplay -v /dev/vg`*nn*`/lvol`*n*

   In the long listing of extents that prints on the screen, column two contains the block device file (listed repeatedly) for the disk.

   Use the *block* device file to derive the *character* device file. For example, a disk with the block device file `/dev/dsk/c2d0s2` has a character device file `/dev/rdsk/c2d0s2`.

2. Now, use the *diskinfo*(1M) command in conjunction with the disk's character device file to determine the disk's product ID. Type:

```
diskinfo /dev/rdsk/c2d0s2
SCSI describe of /dev/rdsk/c2d0s2:
             vendor: HP
         product id: 2213A
               type: direct access
               size: 648192 Kbytes
   bytes per sector: 512
```

   The disk's product ID is 2213A.

   Remember, the `diskinfo` command requires the character (raw) device file, which has an "r" before "dsk."

6

Step 5. Create the New File System

In the new logical volume, create the file system using the *newfs*(1M) command, which has the form:

```
newfs lv_char_dev_file disk_product_ID
```

- Example.

  Suppose you want to create a file system in the newly created logical volume, `/dev/vg00/lvol4`, that uses the disk (physical volume) with product ID of 2213A. Then you would type:

```
newfs /dev/vg00/rlvol4 2213A
```

Note Remember, the `newfs` command requires the character (raw) device file, which has an "r" before "lvol."

As `newfs` successfully creates your new file system, it reports its actions in a form that resembles:

6

```
Warning: 256 sector(s) in last cylinder unallocated
Unknown driver type lv, using default rotational delay
/dev/vg00/rlvol5:       5120 sectors in 12 cylinders of 16 tracks, 28 sectors
5.2Mb in 1 cyl groups (16 c/g, 7.34Mb/g, 2048 i/g)
super-block backups (for fsck -b#) at:
        16,
```

Step 6. Create a Mount Point Directory for the File System.

Make an empty directory to serve as the mount point for your new file system. You can create a mount directory anywhere in your existing file system tree. (Don't select a non-empty directory to mount the file system; the files there will not be accessible until you unmount the file system. Don't select a directory that is already a mount point for another file system.)

```
mkdir /projects
```

This creates the empty directory `projects` in the `/` directory. Supply the full path name to the desired directory.

Step 7. Mount the New File System.

Now the logical volume contains an empty file system that consists of structures for the files and directories that you will create. In order to use the file system, you need to mount it.

Use the *mount*(1M) command, which has the form:

```
mount lv_blk_dev_file  mount_point_directory
```

- Example.

  To mount the file system in the logical volume `/dev/vg00/lvol4` to an empty directory `/projects` in the existing file system, you would enter:

  ```
  mount /dev/vg00/lvol4 /projects
  ```

  When you succeed in mounting the file system, you will get no immediate confirmation.

Step 8. Confirm the File System is Mounted

You can use *bdf*(1M) to verify your mounted file system. For example, enter:

```
bdf
```

Look for the file system you mounted in the list:

```
Filesystem          kbytes    used   avail capacity Mounted on
.
.
/dev/vg00/lvol4      23347       9   21003     0%   /projects
.
```

The file system is ready to use.

Step 9. Add the File System to /etc/checklist

Edit the `/etc/checklist` file and add an entry for the new file system. The entry in the `/etc/checklist` file will enable the file system to be automatically mounted each time you reboot the system.

The entry should resemble (each entry might be different, depending on the specific conditions):

```
/dev/vg00/lvol4  /projects hfs defaults 0 5 # for_projects
```

For more information about `/etc/checklist` entries, see "Making Entries in Checklist File for Mounting a File System at Boot."

After Creating a File System in a Logical Volume

After creating and using a file system in a logical volume, you might want to do the following:

- Increase the size of the file system at some later time.

  If you run short of room in the file system after using it awhile, you can extend the size of the file system. This procedure is described in Chapter 8, "Managing Logical Volumes".

- Mirror the file system, if you have MirrorDisk/UX configured on your system.

These activities are described in Chapter 8, "Managing Logical Volumes".

About the Preceding Procedure

The preceding procedure shows use of commands that have many options. Sections later in this chapter discuss the commands *newfs*(1M) and *mount*(1M) in greater detail.

Also, see chapter 8, "Managing Logical Volumes," which discusses LVM related commands in detail.

Creating a File System in a Disk Section Using HP-UX Commands

If you do not use logical volumes, then you need to create a file system in an unused disk section.

Look at the following checklist of steps.

```
[ ] Look in /etc/disktab and select a section of the disk to use.
    (Command: more /etc/disktab

[ ] Create a file system in the disk section.
    (Command: newfs disk_section's_character_dev_file disk_type)

[ ] Make a mount point directory for the file system.
    (Command: mkdir mount_directory )

[ ] Mount the file system for use in the file system tree.
    (Command: mount disk_section's_block_dev_file mount_directory)
```

If you need to add a disk to get the space for a file system, see the later section "Add a Disk." If you don't know if you need to add a disk, that section provides methods to help you find out.

Step 1. Select a Disk Section For a File System

Refer also to the discussion about "Disk Sections" earlier in this chapter. The manual *How HP-UX Works* provides a detailed discussion of disk sections.

The `/etc/disktab` file describes the characteristics of all of the disks supported by HP-UX.

For example, suppose you have added an HP 7959 disk. Look in the file `/etc/disktab` by typing:

```
more /etc/disktab
```

Near the beginning of the file, you will find a diagram that looks similar to the following one:

```
#
#       Section layout:
#
#       ----------------------------------------
#       6                       ^         ^
#       ----------------------  |   ---   |
#       0            15         | 7  ^    |
#                ------------   |    |    |
#       ---------          ^    |    |    |
#       1            14    |    v    |    |
#       ------------------ | -------- |    | 2
#       10                 |    ^    |    |
#       ------------------ |    |    |    |
#       3             ^    | 13 |    |    |
#       -----------   |    |    | 11 | 12 |
#       4      ^      | 8  |    |    |    |
#       ------ | 9    |    |    |    |    |
#       5      v      v    v    v    v    v
#       ----------------------------------------
#
#
```

The chart shows the logical arrangement of the various possible disk sections.

Although you can build a file system in any section on the disk, by looking carefully at the chart describing the possible sections, you can select a set of sections that will efficiently use the entire disk and not overlap.

You could select section 2 and devote the entire disk to one section. This is desirable if you want to use the entire disk for a single file system or swap space. Or, you could select sections 11 and 7. In this case, the entire disk is logically divided into two sections.

You could also select only a portion of the disk, section 13, for example, leaving sections 15 and 6 blank for future use.

You *cannot* choose sections that overlap. For example, you cannot select the combination of sections 13 and 7, nor can you select a combination that includes 1 and 15.

Caution On 2-way or 4-way striped disk arrays (HP 2252B/HA and HP 2254B/HA), *do not* use section two for a file system. Section two exceeds two gigabytes, which is the maximum size for an HP-UX file system. Using section two will waste disk space and could result in data loss. The `/etc/disktab` file lists disk arrays.

Further on in the file `/etc/disktab` you can find a listing that corresponds to your HP 7959 disk. That listing looks like the following:

```
.
hp7959|hp7959B|hp79591|hp7963|hp7963B|hp9263B|hp2280A:\
 :ty=winchester:ns#15:nt#12:nc#1572:rm#3350:\
 :s0#24280:b0#8192:f0#1024:\
 :s1#16384:b1#8192:f1#1024:\
 :s2#297108:b2#8192:f2#1024:\
 :s3#19532:b3#8192:f3#1024:\
 :s4#36864:b4#8192:f4#1024:\
 :s5#197504:b5#8192:f5#1024:\
 :s6#1998:b6#8192:f6#1024:\
 :s7#42902:b7#8192:f7#1024:\
 :s8#254204:b8#8192:f8#1024:\
 :s9#234548:b9#8192:f9#1024:\
 :s12#295028:b12#8192:f12#1024:\
 :s13#278586:b13#8192:f13#1024:\
 :s14#24280:b14#8192:f14#1024:\
 :s15#16384:b15#8192:f15#1024:
.
.
```

6

This listing describes the sizes of each of the disk sections. It also shows the sizes of the blocks and the sizes of the fragments in a section. After the first line, each line describes a section; for example, section 15's information is in the line starting `:s15`.

Caution The diagram contained in the `/etc/disktab` file provides a general idea of how sections are arranged on various disks. Some disks vary from this general description. Look closely at the entry in `/etc/disktab` for your specific disk. Some disks, for example, do not have all of the sections shown in the diagram.

For some disk types, sections 0 and 13 overlap, in which case they cannot be used together. Sections 0 and 13 do not overlap—and therefore can be used together—if the sum of their sizes does not exceed the size of section 2.

Step 2. Create the New File System

In the disk section you select, create the file system using the *newfs*(1M) command.

```
newfs char_dev_file disk_product_ID
```

- Example.

  Suppose you want to create a file system in disk section 7, which has character device file **/dev/dsk/c1d0s7**. Also suppose you are using disk type 7959. Then you would type:

```
newfs /dev/rdsk/c1d0s7 hp7959
```

Note Remember, use the character (raw) device file (signified by the *r* in *r*dsk).

If you don't know the disk's product ID, you can use the *diskinfo*(1M) command along with a character device file of any of the disk's sections. For example,

```
diskinfo /dev/rdsk/c1d0s7

CS80 describe of /dev/rdsk/c1d0s7:
        product id: 7959 (Option 1)
              type: fixed disc
              size: 304238592 bytes
   bytes per sector: 256
```

When **newfs** successfully creates your new file system, it reports its actions.

6

Step 3. Create a Mount Point Directory for the File System.

Make an empty directory to serve as the mount point for your new file system. You can create your mount point directory anywhere in your existing file system tree.

Note Do not select a non-empty directory to mount the file system; otherwise the files there will not be accessible until you unmount the file system. Do not select a directory that is already a mount point for another file system.

For example,

```
mkdir /projects
```

This creates the empty directory `projects` in the `/` directory.

Step 4. Mount the New File System.

Now, mount the new file system to the mount point directory.

6

Use the *mount*(1M) command, which has the form:

```
mount blk_dev_file  mount_point_directory
```

- Example

  To mount the file system in the disk section `/dev/dsk/c1d0s7` to an empty directory `/projects` in the existing file system, you would enter:

```
mount /dev/dsk/c1d0s7 /projects
```

When you succeed in mounting the file system, you will get no immediate confirmation.

Step 5. Confirm the File System is Mounted

However, you can run *bdf*(1M) to verify you have mounted the file system.

```
bdf
Filesystem          kbytes    used   avail capacity Mounted on
.
```

```
/dev/dsk/c1d0s7        40757        1    36681      0%    /projects
```

You can now use the empty file system.

Step 6. Add the File System to /etc/checklist

Edit the `/etc/checklist` file and add an entry for the new file system. The entry in the `/etc/checklist` file will enable the file system to be automatically mounted each time you reboot the system.

The entry should resemble this (each entry might be different, depending on the specific conditions):

```
/dev/dsk/c1d0s7 /projects hfs defaults 0 5 # for_projects
```

For more information about `/etc/checklist` entries, see "Making Entries in Checklist File for Mounting a File System at Boot."

About the Preceding Procedure

The preceding procedure shows use of commands that have many options. Sections later in this chapter discuss the commands *newfs*(1M) and *mount*(1M) in greater detail.

Adding a Disk

When you add your disk to the computer, refer to the *Installing Peripherals* manual.

The manual *Installing Peripherals* describes making the physical connections, determining the physical address of your disk, and adding the disk to your systems I/O configuration.

When you reboot your system after connecting and turning on your disk, the system automatically configures the disk into its I/O configuration, assigns the disk a logical unit number, and makes all its necessary device files. Be sure to remember the disk's model number because you will need to refer to it later.

Run mediainit (if necessary)

This step might not be necessary in the case of new Hewlett-Packard disks, which have been initialized and formatted, or disks that you have been using already without difficulty.

However, you can use the `mediainit` command to format a disk if you have doubts about its integrity or condition. It takes a while (possibly an hour, maybe more) to initialize a disk.

Refer to `mediainit(1)` in the *HP-UX Reference Manual.* Before initializing a disk, refer to the manual supplied with the particular disk for any interleaving instructions.

When issuing the `mediainit` command to initialize a disk, use the device file that specifies in the minor number that the diagnostic bit is set. These files reside in the `/dev/diag/dsk` directory. For example, if you want to initialize the disk that has been assigned logical unit number 3, you would specify the device file `/dev/diag/dsk/c3d0`.

Caution *Do not* initialize a disk that contains valuable data that you have not backed up. Initializing will destroy all data on a disk.

Using a New Disk for LVM

Before you can use the new disk you add as an LVM disk, you must add the disk to an LVM volume group. That is, you must include the disk in a group of disks that compose a volume group.

Use SAM to add the disk to a volume group, create a logical volume, and then create a file system. Chapter 8, "Managing Logical Volumes," shows how.

Or, you can use HP-UX commands. See chapter 8, "Managing Logical Volumes," for the procedure to add a disk to a volume group using the *pvcreate*(1M) and *vgextend*(1M) commands. You can return to this chapter for discussion of using `newfs` to create a file system.

Using the `newfs` Command.

The previous sections describe creating file systems in logical volumes or disk sections, using simplified configurations.

This section describes the use of the *newfs*(1M) command in greater detail. Whether you create a file system in a logical volume or in a disk section, the command works very much the same, with some exceptions where noted.

Syntax of the `newfs` Command

The **newfs** utility has the following syntax:

newfs [**-L** | **-S**] [**-v**] [*mkfs-options*] *device_file disk_type*

where:

-L — Creates a file system with long file names (up to 255 characters).

-S — Creates a file system with short file names, 14 characters maximum.

6

-v — Specifies verbose mode, which lists **newfs** actions.

mkfs-options — These options override default **mkfs** parameters (the **newfs** utility calls **mkfs(1M)**). The **/etc/disktab** file defines some of these options.

-b *blksize*	defined in **/etc/disktab**
-f *fragsize*	defined in **/etc/disktab**
-m *minfree*	default is 10%
-i *nbpi*	default is 2048 bytes per inode

device_file — The *character* device file name for the disk section or logical volume in which you are creating the file system.

disk_type — The type of disk, as defined in **/etc/disktab**. The disk type is required even if you are creating a file system in a logical volume.

Note If you plan to create a file system in a logical volume that spans two disks, HP recommends that the logical volume span *identical* disks; you will experience unpredictable performance otherwise.

Character Device File

You need to supply the character (raw) device file for the disk section or the logical volume in which you plan to create the new file system.

Disk Sections

Disk sections have character device files that have the following form:

`/dev/rdsk/c`*N*`d0s`*nn*

where:

- the **r** in "rdsk" signifies the character (or raw) device file
- *N* is the logical unit number of the disk
- *nn* is the disk section number to contain the file system

Note On 2-way or 4-way striped disk arrays (HP 2252B/HA and HP 2254B/HA), *do not* use section two for a file system. Section two exceeds two gigabytes, which is the maximum size for an HP-UX file system. Using section two will waste disk space and could result in data loss.

The character device files for disk sections reside in the `/dev/rdsk` directory.

Logical Volumes

Logical Volumes have character device files that have the following form:

`/dev/vg`*N*`/rlvol`*nn*

where:

- *N* is the volume group number
- **r** in "rlvol*n*" signifies the character (or raw) device file
- *nn* is the logical volume number

A logical volume can have a custom name, but the character device file still requires the "r" in the name. For example, a logical volume created with the name "sales" with have the character device "rsales."

Character device files for logical volumes are created when the logical volume is created and reside in the directory, `/dev/vg`*N*.

Choosing File Name Length

By default, `newfs` creates a file system of the same type as the `root` (`/`) file system. However, you can explicitly specify the type of file system with either `-L`, which indicates a file system that allows long file names, or `-S`, which indicates a file system that permits only short file names.

Long vs. Short File Names

HP-UX files systems can be configured to support either long or short file names. Short file names are 14 characters maximum. Optionally, this limit can be increased to 255 characters. You are prompted to choose the long or short limit when you install your system.

6 Generally, you install the system with long file names (the 255 character limit) to gain flexibility in naming files. Also, files created on other systems that allow long file names can be moved to your system without being renamed.

Avoid long file names if:

- You plan to use applications that read directory file information and do not use portable directory routines like the those described in *directory(3C)* in the *HP-UX Reference* manual. If these applications assume that directories are in an array of fixed-size entries, they will not work with long file names. To correct this, rewrite the application to correct the assumptions about directories using the directory file information required by long file names.
- Programs (with no source code available) that were developed for or compiled on releases of HP-UX that do not support long file names will be run on the system.
- Other systems in your organization run version of HP-UX that impose a 14 character limit on file name length. In this environment, you might want uniformity across the systems so that files can be moved between systems.

See the section later in this chapter entitled "Modifying File Name Length."

Overriding the File System Parameters

The following sections discuss overriding the `mkfs` parameters. Table 6-2 provides a reference for all file system parameters.

Overriding the Default Block and Fragment Values

Configuring blocks and fragments represent a compromise between access speed and effective use of disk space. Larger block sizes enable greater access speed, but waste disk space.

The typical defaults, 8 kilobytes blocks size and 1 kilobyte fragment size, are appropriate for most file systems. On 4-way striped disk arrays, the minimum fragment size is 2 kilobytes.

You can use one of the following suggested block and fragment combinations for the listed file systems (block size/fragment size):

1. **/tmp** is usually 8K/8K to allow quick access. Most files in this directory are temporary. Therefore, wasting space is not a problem here.
2. **/usr** is 8K/1K, which has the median trade-off between speed and space utilization.
3. **/mnt** is usually 4K/1K because files that reside here are typically small and remain for a long time.

The `newfs` **-b** *blksize* parameter specifies the block size in kilobytes and the `newfs` **-f** *fragsize* parameter specifies the fragment size in kilobytes.

Note Although you can create a file system that uses a combination of a 64K block size and an 8K fragment size, it is recommended you do this only after verifying that performance is maintained. The 64K/8K combination is recommended for use only with magneto-optical media.

Overriding the Default Reserved Disk Space

The value in *minfree* is the percentage of disk space reserved for the superuser when the file system fills up. It allows the superuser to reserve space for system use. The file system throughput degrades as the number of choices for free blocks is reduced. By setting *minfree* at 10%, which is the default, you are ensuring that the file system throughput will not degrade significantly.

Decreasing the value of the *minfree* parameter lets you write to an additional percentage of file system space. The lower the percentage, the greater the possibility that your file's blocks will be scattered on the disk. Performance decreases as the disk fills up.

Overriding the Default Bytes Per Inode

The *nbpi* parameter dictates the relationship between the number of data bytes on the disk and the number of inodes allocated on the disk. If you increase the number of bytes per inode, you are asking for fewer inodes. The default is to create one inode for every 2048 bytes of data space.

If your system has many small files, you can decrease the average number of bytes per inode. This gives you more inodes, and lets you create more (but smaller) files. Having many inodes takes more space on your file system.

If your system has a few large files, you can increase the space available for data by increasing the average number of bytes per inode.

Table 6-2. File System Parameters

Parameter	/etc/disktab Names	Range	Default
section size	s#	defined in /etc/disktab	none
section block-size	b#	4096 bytes (4K), 8192 bytes (8K), or 65,536 bytes (64K)	8192 bytes (8K)
section frag-size	f#	1024 (1K) to 8192 (8K)	1024 bytes (1K); 2048 bytes (2K) on 4-way striped array
number of tracks per cylinder	nt	greater than 0	16
number of cylinders per disk	nc	1 to 32	16
% space reserved	N/A	0 to 100	10 %
revolutions per minute	rm	N/A	3600
number of bytes per inode	N/A	1 to (function of file system size and other parameters)	2048 bytes (2K)

Example: Creating a File System with Smaller Block Size

Let's assume that you want to set up a file system that will contain many small files. To avoid wasting space, you decide to set up the file system with a smaller block size.

You check the `/etc/disktab` file. From the description for your HP 7937 disk, you fin that section 7 is the right size, but you want to use a block size of 4096 bytes instead of the default 8192 bytes.

You can specify a block size of 4096 bytes at the time you create the file system when you issue the `newfs` command:

```
newfs -b 4096 /dev/rdsk/c3d0s7 hp7937
```

After you have created a file system, you can add it to your existing file system by using the `mount` command. Go to the section titled "Mounting File Systems Using HP-UX Commands."

Example: Creating a File System in a Disk Array

6

Assume that you have added an HP 2252B Disk Array. It is LU 3. You want to make new file systems in sections 7, 8, and 9.

To create a file system in section 7 using default file system parameters for a two-striped array (that, with an 8 kilobyte block size, a 1 kilobyte fragment size, a 10% reserved space, and 2048 kilobytes per inode) enter the following command:

```
newfs /dev/rdsk/c3d0s7 hp22520
```

Notice that the values for block size, fragment size, reserved space, and the number of bytes per inode do not have to be specified in the command because you want the default values.

Create file system in sections 8 and 9 using the following two commands:

```
newfs /dev/rdsk/c3d0s8 hp22520
newfs /dev/rdsk/c3d0s9 hp22520
```

Now you can mount the newly created file systems to the existing file system. You can use the `mount` command, described in the next section.

Example: Create a File System in a Logical Volume

Assume you have created a logical volume, `/dev/vg_accounts/act_lv`, and now you want to create a file system in that logical volume.

1. Find out the disk type.

   What you need to find out, if you do not know it, is which disk you are going to use for the file system. In a volume group, there might be different disk types and you need to know which disk contains the logical volume in which you are creating the file system. The `newfs` utility creates a file system based on the disk type, as it is defined in `/etc/disktab`.

Note If the logical volume has physical extents on more than one disk type, you will have to choose one disk type to use with the `newfs` command. It is possible that a file system could experience performance impact when it is created in a logical volume that spans different disk types.

See chapter 8, "Managing Logical Volumes."

For our example, you would do this as follows:

Enter:

```
/etc/lvdisplay -v /dev/vg_accounts/act_lv
```

In the display that follows, the device files for the disks used by the logical volume are listed immediatedly under the label, `PV Name`; for example:

```
.
PV Name            LE on PV  PE on PV
/dev/dsk/c1d0s2    5         15
.
```

Now, using the character device file for the disk (if the disk were `/dev/dsk/c3d0s2`, the character device file would be `/dev/rdsk/c3d0s2`), use `/etc/diskinfo` to find out the disk type. Look for the line containing `product id:`.

For example:

```
.
product id: 2213A
.
```

2. Create the file system as follows.

   Remember to use the logical volume's character device file with the `newfs` command.

```
newfs /dev/vgaccounts/ract_lv 2213A
```

6

Mounting and Unmounting File Systems

The process of adding an auxiliary file system is known as *mounting* a file system. Removing an auxiliary file system is known as *unmounting*.

Mounting a file system links the file system contained in a specific disk section or logical volume to a directory in the existing file system tree. Once mounted, the file system becomes accessible to users.

The directory in the existing file system becomes the root directory of the added file system and the files in the added file system become part of the existing file system hierarchy. Figure 6-2 earlier in this chapter shows the relationship of a mounted file system to the existing file system system.

You can create, modify, delete, and move files within the added auxiliary file system. When you no longer need to use the files on that file system, you can unmount it.

Unmounting a file system removes its files from the existing file system's hierarchy. The files themselves remain on the disk and can be accessed by mounting the file system again. Mounted file systems are automatically unmounted at shutdown time.

The Mount Directory

The mount directory becomes the root directory for the file system you add. The mount directory should be an empty subdirectory on the existing file system. Indicate the full absolute path name.

If you specify a non-empty directory for the mount point, any files in the mount point directory will be inaccessible when the new file system is mounted. The files will still exist, but they will remain inaccessible until you unmount the overlaying file system. Therefore, do not mount a file system over data that you will need later.

You can use SAM or use HP-UX commands directly to mount or unmount file systems.

Mounting and Unmounting Local File Systems Using SAM

If you used SAM to create a file system on a newly added disk, you probably had SAM mount that file system by selecting `Now` to the choice `When to Mount`.

However, if you did not have SAM mount the file system then, or if you have since unmounted the file system, you can use SAM to mount it.

You can also use SAM to unmount file systems that are currently mounted.

The following sections describe mounting and unmounting a local HFS file system using SAM.

If you need to mount or unmount file systems over the network, see the section "Mounting NFS File Systems Using SAM".

The following list summarizes the steps you need to take to mount or unmount local file systems using SAM.

1. Run SAM.
2. Highlight `Disks and File Systems` and activate the (Open) control button.
3. Highlight `Local File Systems` and activate the (Open) control button.
4. Highlight the file system you want to add from the list of file systems SAM displays.
5. From "Actions" on the menu bar, highlight `Modify...`.
6. The field: `Modifying file system:` shows the mount directory.
7. Activate the `Now` control button below `When to Mount:`. Also activate `At Every System Boot` if you want the system mounted each time you reboot the system.

   If you want to unmount a mounted file system, deactivate the `Now` control button.
8. Examine and modify, if necessary, the other options for the file system.
9. Activate (OK) to mount or unmount the file system.

Mounting File Systems Using HP-UX Commands

If you do not use SAM to mount an auxiliary file system, you can use the *mount*(1M) command. The **mount** command attaches an auxiliary file system to an existing directory in the file system tree. The auxiliary file system can be on a disk or disks connected directly to your system or it can be part of a remote file system, in either a disk section or a logical volume.

To mount a file system:

1. If necessary, connect the disk(s) containing the file system to your computer; refer to the manual *Installing Peripherals*. If you are planning to NFS mount file systems, there are other requirements. See the section "Mounting NFS File Systems."
2. Determine the block special device file for the disk section or logical volume containing the file system.

   Or, when mounting file systems via NFS, know the name of the host machine and the file system's mount directory on the server machine.
3. Determine the mount point directory for the file system. Use the **mkdir** command to create the directory if it does not yet exist.
4. Determine which **mount** command options—if any—to use when you mount the file system.
5. Use the **mount** command to add the file system, using options you select.
6. Edit the file **/etc/checklist** to have the file system mounted when you boot your system.

Mounting File Systems Using HP-UX Commands

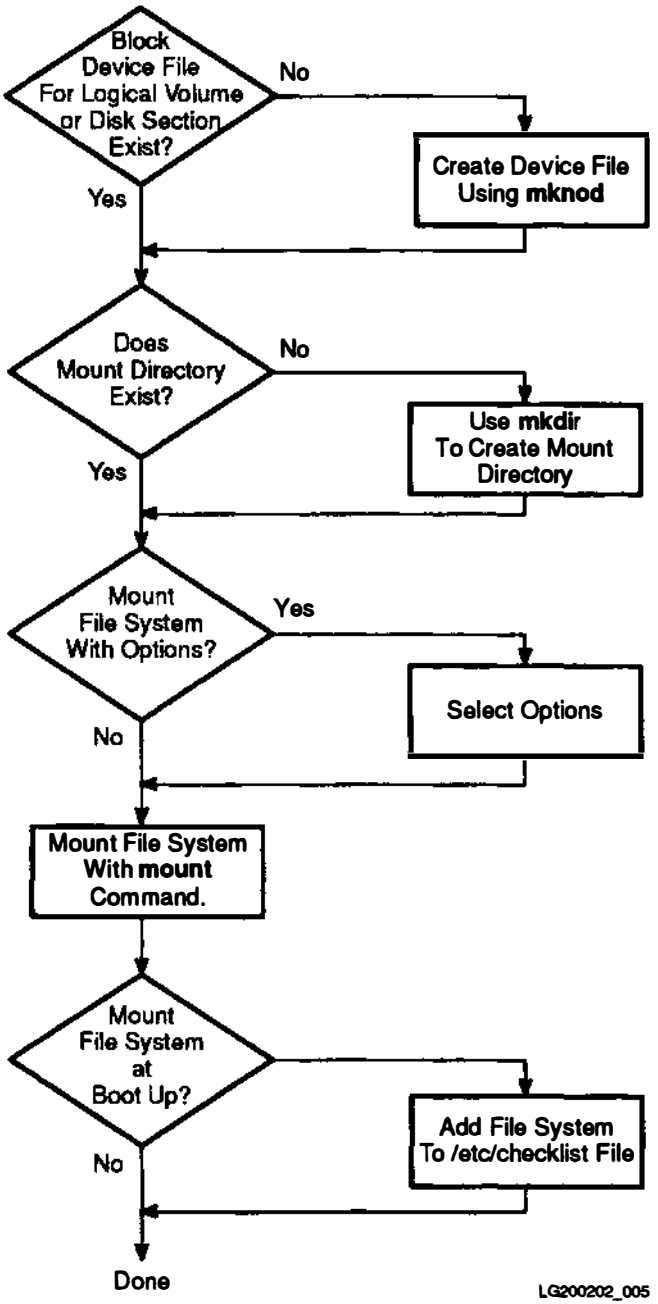


Figure 6-4.

Syntax of the `mount` command

The *mount*(1M) command has the following syntax:

`mount` *sfname directory*

where:

sfname is the name of the block special file (device file) associated with the device containing the file system to be mounted.

For example, to specify the file system in section 4 of the disk assigned the logical unit number 0, use the block device file name:

```
/dev/dsk/c0d0s4
```

To specify the block device file for a logical volume, for example, logical volume `lvol4` in the volume group `vg00`:

```
/dev/vg00/lvol4
```

To specify the name of a remote file system, use the form: *host:path*. For example,

```
hpfcd:/build
```

See the section, "Mounting NFS File Systems."

When mounting a CD-ROM file system, the device file must specify section 2. For example,

```
/dev/dsk/c1d0s2
```

directory the mount point directory, that is, the directory in the existing file system tree where the file system is to be mounted.

Example: Mounting a File System

For example, the following command,

```
mount /dev/dsk/c0d0s4 /projects
```

mounts (attaches) the file system contained in section 4 of the disk LU 0 to the mount point directory `/projects`.

After mounting the file system, you can edit the `/etc/checklist` file to have the file system automatically mounted each time you boot the system. Refer to the section "Automatically Mounting Your File Systems at Bootup" for details concerning adding entries to the `/etc/checklist` file. For the file system mounted above, you would add a line in the `checklist` file such as:

```
/dev/dsk/c0d0s4  /usr   hfs  defaults  0  3  # users
```

Example: Mounting a File System, Logical Volume

To mount the file system existing in the logical volume `/dev/vg3/lvol3` to the mount directory `/users`, enter

```
mount /dev/vg3/lvol3 /users
```

6

Adding the file system to the `/etc/checklist`, you would enter:

```
/dev/vg3/lvol3  /users/  hfs  defaults  0  3  # users
```

Note If an HFS file system has been unmounted improperly and not checked for inconsistencies with the utility `fsck(1)`, the `mount` command will not be able to mount the file system. Run `fsck` on that file system before attempting to remount it. For information on using the `fsck` utility, refer to Chapter 6, "File System Problems," and appendix B, "Using the fsck Command," in the *Solving Problems* manual.

Example: Mount a File System, Read-Only Permission

Suppose you wanted to mount a file system and have that file system be read-only. You would use the command:

```
mount /dev/dsk/c0d0s4  /users -o ro
```

In this example, the read-only option is specified by `-o ro`; the `-o` signifier must precede all `mount` command options.

For this file system, you can add a line in the `/etc/checklist` file that would be, for example:

```
/dev/dsk/c0d0s4  /users   hfs  ro  0  0  # users
```

See the section "Automatically Mounting Your File Systems at Bootup" for a detailed discussion of making entries for the `/etc/checklist` file.

Example: CD-ROM file system

Before you can mount CD-ROM file systems, your system's kernel must be configured for the `cdfs` subsystem.

Mounting a CD-ROM file system is specified in the same way as an HFS file system, except that the section number must be 2. If you wanted to mount the CD-ROM file system in the disk section with the block device file `/dev/dsk/c0d0s2` to the mount directory `/userdoc`, you would use the command:

```
mount /dev/dsk/c0d0s2  /userdoc
```

Note CD-ROM file systems are read-only.

Using `umount` to Unmount a File System

To unmount a file system, use the *umount*(1) utility. All files on the particular file system to be unmounted must be closed. Attempting to unmount a file system that has open files (including your working directory) causes the *umount*(1) utility to fail without unmounting the file system.

Use the *ps*(1) utility with **-ef** options or the *fuser(1M)* utility to check for open files. If there are open files, take necessary actions to close the files. You might need to execute these commands more than once to ensure that all files are closed.

Note Use the the `mount` or `umount` commands from a client in an HP-UX cluster only for locally mounted file systems or NFS file systems.

Syntax of the umount Command

The *umount*(1M) command has the following syntax:

6

```
umount sfname
```

or

```
umount directory
```

where:

sfname is the path name of the block device file for the device containing the file system to be unmounted or the name of a remote file system in the form: *host:path*.

directory is the directory where the file system is mounted.

When using `umount` to unmount a file system, you can specify either the name of the mount directory or the block special file for the disk section containing the file system.

Note Always unmount file systems contained on a mass storage device before removing the device from the system. Removing a device containing mounted file systems (for example, disconnecting or turning off the power to a disk, or removing a disk pack from a mass storage device) will likely corrupt the file systems.

Examples: Unmounting a File System

For example, to unmount a local HFS file system in section 13 of disk lu 0 mounted at the directory `/users`, issue the command:

```
umount /dev/dsk/c0d0s13
```

or

```
umount /users
```

or

```
umount /dev/vg3/lvol4
```

To unmount a remote NFS file system:

```
umount hpfcf8:/users
```

Unmounting More than One File System

The `umount` command removes the specified file system from the `/etc/mnttab` file. If you wish to unmount all of your currently mounted file systems contained in the `/etc/mnttab` file execute:

```
umount -a
```

Or, to unmount currently mounted file systems of a particular type, you can use the `-t` option. For example, to unmount all NFS file systems, use the command:

```
umount -a -t nfs
```

Note You cannot unmount the root (/) file system.

If you have unmounted a file system and do not want it mounted the next time the system boots, you can retain the entry in the `/etc/checklist` file, replacing the current entry in the *type* field with `ignore`. You can manually mount the file system, and later, you can change the `type` entry to have automatically mounted again.

Automatically Unmounting Your File Systems at Shutdown

When you execute the *shutdown(1M)* command, the system unmounts all of the file systems listed in the `/etc/mnttab` file. The *shutdown* utility uses `umount -a` to unmount file systems. For more information on `shutdown(1M)`, refer to Chapter 3, "Starting and Stopping HP-UX".

Mounting NFS File Systems

You can use either SAM or HP-UX the command *mount*(1M) to mount file systems located on a remote system.

Mounting and Unmounting NFS File Systems Using SAM

You can mount and unmount NFS file systems using SAM.

When configuring NFS file systems in SAM,

- Note that a machine can be an NFS server (export file systems), an NFS client (mount remote file systems), or both.
- Both for the **NFS Server** (the system where the file systems reside, which is the machine exporting the file sytems) and for the **NFS Client** (the machine where remote file systems are mounted), entries must exist in the `/etc/hosts` file. SAM makes these entries automatically.

Configuring a Local File System for Export

Configuring a local file system for export means that the local machine will be the server for a local file system that can be mounted by remote clients.

1. Highlight `Disks and File Systems->` and activate the (Open) control button.
2. Highlight `Networked File Systems (NFS)->` and activate the (Open) control button.
3. Highlight `Local File Systems Exported` and activate the (Open) control button.
4. From "Actions" on the menu bar, choose `Add...`.
5. Enter the name of the local file system to export in `Local Directory Name:` field. Accept the default settings on the remainder of the fields, unless you want to modify them.
6. Activate (OK) to set up the local file system to be exported.

Enabling or Disabling the Server File System

In the "Local File System Exported" Screen:

1. Highlight the file system for export.
   a. If the NFS server is "Disabled," from "Actions" on the menu bar, choose `Enable NFS Server`. This enables a local file system to be mounted on remote systems.
   b. If the NFS server is "Enabled," from "Actions" on the menu bar, choose `Disable NFS Server`. This disables a local file system to be exported to remote systems.

To Make Remote Systems Clients of a Local File System

1. Highlight the file system for export.
2. From "Actions" on the menu bar, choose `View Directory Access`.
3. Add the machine name in the appropriate field for the type of access desired.

6

Mounting Remote File Systems

Mounting a remote file system means that a local machine becomes a client for a file system that is being exported from a remote server.

1. Highlight `Disks and File Systems->` and activate the [Open] control button.
2. Highlight `Network File Systems (NFS)->` and activate the [Open] control button.
3. Highlight `Remote File Systems Mounted` and activate the [Open] control button.
4. From "Actions" on the menu bar, choose `Add Remote Directory...`.
5. Enter a name in the `Remote System Name` field.
6. Enter a name in the `Remote Directory Name` field.
7. Enter a name in the `Local Directory Name` field.

Accept the default settings on the remainder of the fields, unless you want to modify them.

8. Activate [OK] to enable the local file system to be exported.

Unmounting a Remote File System

1. Highlight `Disks and File Systems->` and activate the [Open] control button.
2. Highlight `Network File Systems (NFS)->` and activate the [Open] control button.
3. Highlight `Remote File Systems Mounted` and activate the [Open] control button.
4. Highlight the remote file system to be unmounted.
5. From "Actions" on the menu bar, choose `Modify...`. Deactivate the `now` control button in the "When to Mount" field.
6. Activate [OK] to unmount the remote file system.

Unmounting an Exported File System on a Server

6

To unmount a file system that is currently being exported to one or more clients:

1. Have the file system unmounted from each and every client. From each client:
   a. Run SAM.
   b. Unmount the remote file system. Use the procedure described in "Unmounting a Remote File System".

Note Unmounting an exported file system on which users of remote client systems have open files could result in lost data.

---

2. From the server, disable the exported file system.

Mounting NFS File Systems Using HP-UX Commands

Before you can mount file systems located on a remote system, NFS Services software must be installed and configured on both local and remote systems. Refer to the manual *Installing and Administrating NFS Services* for the necessary and detailed information.

Note the following about using `nfs` file systems.

Preparations for Mounting Remote File Systems

1. Both for the **NFS Server** (the system where the file systems reside, which is the machine exporting the file sytems) and for the **NFS Client** (the machine where remote file systems are mounted), entries must exist in the `/etc/hosts` file. (See *hosts*(4) in *HP-UX Reference Manual.*)

   Note that a machine can be an NFS server (export file systems), an NFS client (mount remote file systems), or both.

2. The **NFS Server** must list both the file systems and the NFS Clients that can mount the file systems in the file `/etc/exports`. (See *exports*(4) in *HP-UX Reference Manual.*)

   Entries in the `/etc/exports` file require the following syntax: the file system name must be flush left and the client machine names are to follow, separated by single spaces. An entry in `/etc/exports` that lists only a file system name (no machine names) indicates that all machines can access the file system. (Review *exports*(4) in *HP-UX Reference Manual* for guidelines and cautions concerning the use of the `-async` option.)

   For example, to allow machines called `rolf` and `egbert` to remotely mount the `/usr` file system, edit the file `/etc/exports` on the server machine and add the line:

```
.
/usr rolf egbert
.
```

Example: Mounting an NFS File System

To mount the remote file systems `/users/jpsouza` on the remote system `hpedc2` to the mount directory `/users/jpsouza` on your local file system, enter:

1. Create the mount directory if it does not already exist:

   ```
   mkdir /users/jpsouza
   ```

   Note that the mount directory should be empty. A mounted file system will mask files in a mount directory, making them unavailable until you unmount the file system.

2. Mount the remote file system:

   ```
   mount hpedc2:/users/jpsouza /users/jpsouza
   ```

   This will make the files in the directory `/users/jpsouza` on remote machine `hpedc2` accessible from your local machine.

When mounting NFS file systems, you can use options that affect mounting conditions, user permissions, and so on. All options are thoroughly described in the manual *Installing and Administrating NFS Services*. (See also *mount*(1M) in *HP-UX Reference Manual*.)

6

NFS Mount Problems

When the `mount` command succeeds, it is silent. Otherwise, you will receive an error message. If the attempt to mount a remote file system fails, be sure to verify that:

- The client machine has an entry in NFS server's `/etc/exports` file that allows it to mount the remote file system.
- The mount directory exists and is not currently being used as a mount point, or that no files in the directory are in use.

Automatically Mounting Your File Systems at Bootup

To automatically mount all your file systems at bootup, do the following:

- List the file systems that you want automatically mounted in the `/etc/checklist` file. See "Making Entries for Mounting a File System in the Checklist File" and the entry for *checklist*(4) in *HP-UX Reference Manual*.
- In the `/etc/rc` file, enter the `mount -a` command (if it is not already present).
  - For `hfs` type file systems, add the line

    ```
    /etc/mount -a -t hfs
    ```

    in the `hfsmount()` section of the `/etc/rc` file.
  - For `cdfs` file systems, you can add the line

    ```
    /etc/mount -a -t cdfs
    ```

    in the `localrc()` section of the `/etc/rc` file.

Most of the earlier examples of creating and mounting file systems using HP-UX commands include adding an entry to the `/etc/checklist` file. Also, when you use SAM to create a file system and want the file system mounted when you boot your system, SAM makes entries in `/etc/checklist`. The following sections discuss the `/etc/checklist` file in greater detail.

6

Reading/Modifying the /etc/checklist File

The `/etc/checklist` file is an ASCII file that lists all file systems and swap devices. The `/etc/rc` script refers to it when executing the `mount -a` and `swapon -a` commands and when running the `fsck(1M)` utility.

This file is also described in detail in section 4 of the *HP-UX Reference*. If you want to automatically enable swap devices, see "Enabling Swap Areas on Boot Using the Checklist File" in Chapter 7.

SAM automatically creates entries in `/etc/checklist` when you indicate you want a file mounted `on boot` when you add a file system.

If you choose to add file systems using the `mount` command, you must manually edit and maintain this file.

Making Entries in the Checklist File for Mounting a File System at Boot

Before reading the discussion on making `/etc/checklist` entries, you can enter the command:

```
/etc/mount -p
```

This displays information about all of the currently mounted file systems in `/etc/checklist` format.

You can also see the same information by simply typing the `mount` command without options; the display will not be formatted, however.

Look at the following example entry in an `/etc/checklist` file:

```
/dev/dsk/c2d0s0    /users    hfs    defaults 0 2 # users
```

An entry for a file system in the `/etc/checklist` file has seven fields separated by blank spaces. In each field, you need to put either an entry or a placeholder. The following describes making entries for each field:

- First field: Enter the *block device file* corresponding to the disk used for the file system.
- Second field: Enter the mount point *directory*; this directory is located in the existing file system and will serve as the root directory for the mounted file system.
- Third field: Enter the *type* of the entry. The *type* must be one of the following depending on the type of file system:

  `hfs` (local high-performance file system)
  `nfs` (file system available through NFS Services)
  `cdfs` (CD-ROM file system)

  You can also enter `ignore` in this field when you want to retain the entry for the file system in the `checklist` file, but do not want the file system mounted at bootup. You can mount the system later.

- Fourth field: Enter *options*, which include the following:

defaults	use all default options (rw,suid).
rw	read-write (default).
ro	read-only.
suid	set user ID execution allowed (default).
nosuid	set user ID execution not allowed.
quota	enable disk quotas.
noquota	disable disk quotas (default).

  These options correspond to the options used with `mount` and are described in *mount*(1M) in *HP-UX Reference Manual*.

  Disk quotas are discussed later in detail. See the section "Using Disk Quotas to Limit Users' Disk Space."

- Fifth field: This field is reserved for future use by backup utilities.
- Sixth field: Assign a pass number of 1 for the root file system and larger numbers to other file systems. The pass number is used by `fsck` issued with the `-p` command. The *fsck*(1M) command ignores file systems with pass numbers of 0.
- Seventh field: Enter a comment preceded by the `#` character.

The following is a sample `checklist` file for a file system residing on an HP 7935 with six disk sections:

```
/dev/dsk/c0d0s4     /       hfs   defaults   0 1  # root
/dev/dsk/c0d0s15    /swap   swap  defaults   0 0  # swap
/dev/dsk/c0d0s14    /tmp    hfs   defaults   0 2  # /tmp
/dev/dsk/c0d0s3     /mnt    hfs   defaults   0 3  # /mnt
/dev/dsk/c0d0s5     /usr    hfs   defaults   0 4  # /usr
/dev/dsk/c0d0s10    /users2 hfs   defaults   0 5  # /users2
```

What Happens When the System Boots

At boot up, the `/etc/bcheckrc` and `/etc/rc` file system scripts are executed. The `/etc/bcheckrc` script checks each file system in `/etc/checklist` during start up and uses the clean byte to see if the system was properly shut down. If a file system appears to have been previously shutdown improperly, `bcheckrc` runs `fsck` during the startup process. Then `/etc/rc` mounts all file systems listed in `/etc/checklist`.

Modifying File Name Length

Here are some potential problems and troubleshooting suggestions:

1. A program opens directories and reads the directory entries directly. Change the program to use directory library routines. See `directory (3)` in the *HP-UX Reference Manual.*
2. A program assumes that the maximum length of a file name (in a buffer) is 14 characters. For example, **char filename**[14] or **char filename[DIRSIZ]** (**MAXNAMLEN** should be used for the buffer size, if only a few buffers are involved).

   If you want to store more than a few file names, enable the DIRSIZ_MACRO compilation flag. When DIRSIZ_MACRO is enabled, DIRSIZ is a macro instead of a constant of 14. The macro accepts an argument that is a pointer to a struct direct and returns the size of the file name rounded to a 4-byte boundary. You can then allocate more memory for the file names.
3. A program includes <**dir.h**> and uses **struct direct**. The struct direct for systems that support long file names is a variable length structure and the struct direct for systems that support only short file names is a fixed-size structure.

   You can include **ndir.h** and use directory libraries.
4. A program assumes there is only one file system magic number. (The magic number for a system that supports long file names is different from the magic number for a system that supports only short file names.)

   Change the program to allow the new magic number for long file names.
5. A program uses **MAXNAMLEN** and assumes it has a value of 14 (when you convert to long file names, you need a **MAXNAMLEN** of 255).

   Recompile the program.
6. A program uses **DIRSIZ** and assumes it is a constant of 14 (meaning the maximum file name length is 14 characters).

   Instead of **DIRSIZ**, use **MAXNAMLEN** to dictate the maximum file name length on a system that supports long file names. (For systems that support only short file names, use **DIRSIZ_CONSTANT** to dictate the maximum file name length.)

Enabling Long File Names Using SAM

1. Back up your entire file system before you use SAM to convert the length of your file names. (You should do a backup before you perform any operation that alters the file system).
2. Get into single-user mode. You can use SAM to get to single-user mode (via `Routine Tasks->` in SAM's main control window), although you must reenter SAM after you reach single-user mode to convert to long file names.

   Type:

   `shutdown`

3. Run SAM.
4. Highlight `Disk and File Systems->` and activate (Open).
5. Highlight `Local File Systems` and activate (Open).
6. Highlight the file system with short names that you want to convert to long file names and from "Actions" on the menu bar, choose `Convert to Long File Names` .

   SAM will issue a warning that converting to long file names is irreversible and ask you confirm the change.

   Answering y will perform the task on the selected file system.

   If you plan to convert some file systems to long file names, the recommendation is to convert *all* file systems to long file names to prevent inconsistencies and undesired events (such as the overwriting of files) that could result when using both long and short file names on the same system.

Caution Converting a file system to long file names is not an easily reversible process. You must actually back up the data, recreate file systems with short file names, and restore data to new file systems

Changing a file system back to short names requires that you perform the series of steps described in the section "Disabling Long File Names" .

Enabling Long File Names Using convertfs

If you have an HP-UX file system that allows only short file names, use the *convertfs*(1M) utility to convert the file system to allow long file names.

Follow these steps:

1. Back up your entire file system before you use the *convertfs* utility. (You should do a backup before you perform any operation that alters the file system).
2. Shut your system down to single-user state as follows:

   `shutdown`

3. Unmount all of your file systems:

   `umount -a`

4. Execute the *convertfs* utility:

   `convertfs`

You will receive these messages:

```
Warning:  certain programs might not work with long file names.

Converting the file system will cause a system reboot.  The system
should be shut down into single user state and all non-root file systems
should be unmounted before this utility is run.

Do you wish to continue? [y/n]
```

If you have your system in single-user state and have all non-root file systems unmounted, answer **y**.

It will then ask you if you want to convert all of the normally mounted file systems listed in `/etc/checklist`. If you answer "no" to this prompt, `convertfs` will ask whether you want to convert each file system in `/etc/checklist`. Respond to the prompt for each file system.

The `convertfs` utility modifies the superblocks and reformats the directories in the file systems you want to convert. After modifying each file system, `convertfs` executes an `fsck` so that the file system can again be mounted.

Note Although the `convertfs` utility allows just one (or a few) file systems to be selected for conversion to long file names, HP recommends that you convert all or none of your normally mounted file systems in `/etc/checklist`. This prevents inconsistencies and undesired events (such as the overwriting of files) that can occur if you mix long and short file names on the same system.

If you have converted the root file system, `convertfs` reboots the system so that the changes made to the file system superblock will not be overwritten by an update of the superblock in the system memory.

You can also execute `convertfs` with the name of the specific file system you want to convert:

```
convertfs /dev/rdsk/c2d0s10
```

The `convertfs` utility converts the named file system without prompting for input.

After you reboot the system or remount the converted file systems, you can use long file names on the converted file systems. The *newfs*(1M) and *mkfs*(1M) utilities create new files of the same type as the root file system. If you converted the root file system, all new file systems you create allow long file names. If you need a file system with short file names, use the `-S` option to either *newfs*(1M) or *mkfs*(1M). (Refer to either *newfs*(1M) or *mkfs*(1M) in the *HP-UX Reference Manual* for more details.)

Disabling Long File Names

When you use `convertfs(1M)` to convert to long file names, you cannot use the utility to convert back to short file names. If you must convert back to short file names after using `convertfs`, the file system should not have any file names longer than 14 characters. Recreate the file system with short names using the **-S** option to `newfs` or `mkfs` and then recover the original files from the backup media. If the root file system needs to be converted back to short file names, reinstall it from the installation tape. Be sure to save any files customized for your system so these files can be recovered after the reinstallation.

The following four steps describe the process of converting your file system to short file names:

1. Examine all file names to make sure they have 14 or fewer characters. Use the `mv` command to rename files with long names so that they have names of 14 or less characters.
2. Back up your entire file system after you have shortened the file names.
3. Recreate the file system with short file names by executing `newfs` with the **-S** option.
4. If the root file system must be changed, reinstall it from the installation tape.

Checking File System Consistency

The HP-UX file system can develop inconsistencies over a period of time. For example, turning off the computer without previously shutting down or unmounting the file system will cause some corruption of the system. Except for obvious events, such as a power failure, the causes of file system corruption are difficult to determine.

You should check the file system for consistency periodically and anytime you suspect a problem. Some commands, such as `/etc/update` or `convertfs`, will not function properly unless the file system is free of inconsistencies.

Using the fsck Utility

You can use the `fsck` utility to check file systems for any inconsistencies and to make any necessary repairs. The manual *Solving HP-UX Problems* contains detailed information about using the `fsck` command (see Appendix A, "Using the fsck Command").

Caution When you run `fsck`, make sure the file systems are inactive. The best way to proceed is to:

1. Make sure all users are logged off the system.
2. Issue the command:

   ```
   /etc/shutdown
   ```

   This terminates running processes in an orderly and cautious manner and places the system in single-user mode.
3. Run `fsck` in single-user state. Refer to the manual *Solving HP-UX Problems* when running `fsck`.

Displaying Current Disk Usage Information

1. Use the *bdf*(1M) to list all currently mounted file systems and key information about them. Enter:

   ```
   bdf
   ```

   The output resembles:

   ```
   Filesystem          kbytes    used   avail capacity Mounted on
   /dev/dsk/c0d0s13    484960  243777  192687     56%  /
   /dev/dsk/c5d0s8     237810   48481  165548     23%  /graphics
   /dev/dsk/c4d0s2     277954  129787  134269     49%  /users
   /dev/dsk/c3d0s10    121663   23257   86239     21%  /tmp
   /dev/dsk/c2d0s2     277954  177941   72217     71%  /projects
   /dev/dsk/c1d0s11    461664  320936   94561     77%  /usr
   /dev/dsk/c3d0s7      70979    2985   60896      5%  /usr/spool
   /dev/dsk/c1d0s7      71599   44562   23457     66%  /usr/contrib
   hpcupt3:/usr/lotus   50775   22244   23453     49%  /usr/lotus
   ```

   The column headed "Filesystem" lists the block device files for all mounted file systems, whether they are locally or remotely mounted.

   The values in the next three columns are reported in kilobytes. You can multiply these values by 1024 to find the values in bytes. Divide them by 1000 and round the result upwards to find the value in megabytes.

   The values under "used" and "avail" add up to the total space available to users, and the percentage under "capacity" corresponds to the value under "used." The total of "used" and "avail" equals 90% of the value under "kbytes." The difference is the "minfree" area that, by default, is reserved for system administration use and allows for efficient file system performance.

   The "Mounted on" column lists the file systems' mount directories.

2. Use *df*(1M) to see the amount of free space left in a file system. For example, you can use `df -t /users` to see usage information for a mounted file system, `/users`:

```
df -t /users

/users  (/dev/dsk/c1d0s0 ):  288480 blocks        128796 i-nodes
                             879996 total blocks  147456 total i-nodes
                             503516 used  blocks   18660 used i-nodes
                                 10 percent minfree
```

   The top line shows the available space (in 512-byte sized blocks) and the number of available file system inodes. Divide by the number of 512-byte blocks by 2 to get the number in kilobytes.

While the examples provide suggestions, HP-UX has many commands for exploring the system. In general, you can find them in sections 1 and 1M of the *HP-UX Reference Manual*. The following commands explain how to get information about disk usage. Some commands generate long lists, which you might want to redirect to a file or pipe to `more` to view the information one screenful at a time.

- Use *du*(1) to monitor users who are increasing their disk usage. The `du` utility displays information in 512-byte blocks.

```
du -s /users/*

184 /users/jamieo
92 /users/michelem
10 /users/rykl
10 /users/alisonm
```

- Use *du*(1) and *sort*(1) to list files in decreasing size.

```
du -s /users/michelem | sort -nr

92 /users/michelem
24 /users/michelem/checklist.man
14 /users/michelem/shutdown.man
12 /users/michelem/swapon.man
10 /users/michelem/umount.man
10 /users/michelem/mount.man
6 /users/michelem/convertfs.man
```

```
2 /users/michelem/testgraph
2 /users/michelem/tempfile
```

- Use *find*(1) to locate files over a particular size. The following example displays files larger than .5 Mbytes:

```
find / -size +1000 -print > big-files
```

6

- Use `find` to locate files older than *n* days. The following example displays files not written or accessed in 90 days:

```
find / -mtime +90 -atime +90 -print > aging-files
```

Displaying File System and Disk Usage when in SAM

In SAM, you can obtain information about the usage of disks and mounted file systems. When you are displaying lists of disks or file systems, you can highlight an item on the list and then, from "Actions," choose `View More Information...`.

Using SAM to Recover Disk Space

SAM provides the means to remove old or useless files that can accumulate on your system, wasting valuable disk space.

1. Run SAM.
2. Highlight `Routine Tasks->` and activate the (Open) control button.
3. Highlight `Disk Space Recovery->` and activate the (Open) control button.
4. You can now choose, using the (Open) control button, one of the following activities:
   - `Log File Trimming->`
   - `Large File Removal`
   - `Core File Removal`

   These tasks are straightforward. SAM can help you decide how to remove these files on the basis of type, size, and date.

Using Disk Quotas to Limit Users' Disk Space

Disk **quotas** provide a system administrator the means to limit how many files and file blocks a user can own on an HFS or NFS file system. Disk quotas are established on a per-file-system basis. You must have superuser privileges to set up disk quotas. For each user, you can set limits for the number of files (by limiting the number of inodes) that can be created and for the number of file system blocks that can be used. Each user can have a **soft limit** and a **hard limit**. A soft limit is a preferred limit that a user can exceed for a limited time, while a hard limit is an absolute limit. If a user reaches hard limits or fails to reduce usage below soft limits before a specified time, he or she will be unable to create files or increase file system block usage.

For each file system enabled with quotas, the operating system maintains statistics on limits and usage. A user can check his or her quota status at any time. As system administrator, you can increase or decrease a user's limits at any time.

Caution By using the `chown` command to change the ownership of files, a user can bypass disk quota limits. For example, a user can use the `chown` command to make `root` the owner of a file; this file, now owned by `root`, will not be considered in the file system usage computed for the user who created it.

A solution to this potential problem is to reserve the use of the `chown` command to privileged users. See `setprivgrp (1m)`. Also, see the chapter "Controlling System Access."

By editing the `/etc/checklist` file, you can have quotas turned on automatically for each file system when you boot your system. You can turn quotas off and turn them back on again at any time, though this is discouraged because of the system overhead required to recompile usage.

SAM does not have the capability to perform tasks involving quotas.

Caution In an HP-UX cluster, some behaviors differ from those seen on a standalone system. Those differences are indicated in the following discussion.

Planning for Disk Quotas

Choosing file systems for disk quotas. Typically, you will want to set disk quotas on file systems that would otherwise become full without limitations on their use. This means that file systems containing the home directories and files of several users are suitable for disk quotas. It is not recommended that you set up quotas for the `/tmp` file system, unless you set the hard limits very large, the soft limits small, and the time limits short to prevent users from using `/tmp` as storage .

Choosing limits. You can choose to set uniform soft and hard limits for most or all users, or you can set limits for each user individually. The recommended method for setting uniform limits is to assign limits for one or more prototype users and apply those limits to actual users.

You can also set a limit to the time users are permitted to exceed the soft limits set for them. These time limits are set for a file system, applying to all users on the file system.

Disk quotas and performance. Because the disk quota statistics are resident in memory, the use of disk quotas involves minimal computation and seldom results in the transfer of data to and from a disk. The time required to reboot a system that has crashed will take somewhat longer because of the time required to run `quotacheck`. (See the section, Checking Consistency of File System Usage Data.)

6

Setting Up Disk Quotas on a File System

To set up disk quotas, use the steps outlined below. Details and examples will follow.

1. Mount the file system for which you will use disk quotas.
2. Edit the file `/etc/checklist` to have disk quotas enabled on the file system the next time you start up the system.

   For an auxiliary file system not listed in the `/etc/checklist` file, you can enable disk quotas for it when you mount it using the `quotas` option with the `mount` command. However, you must perform the following steps before turning quotas on.

3. Create the empty file `quotas` in the root directory of the file system for which you are enabling quotas.
4. Set user quotas using `edquota` command.

   Set quotas for a prototype user and apply these quotas to all users of that file system, or set individual quotas user by user.

Mount the File System

The file system you want to set up with quotas must be mounted.

Let's suppose you want to want to implement quotas on the HFS file system `/users` that has the device file `/dev/dsk/c1d0s8`. This file system would already be mounted if you have it listed in your `/etc/checklist` file. If it is not mounted, you can mount it by entering:

```
mount /dev/dsk/c1d0s8 /users -t hfs
```

Edit /etc/checklist to Add "quota" Option

When the entry for a particular file system listed in `/etc/checklist` contains the option `quota`, the file system will be enabled for disk quotas at the time the system is started. For example, if the line in your `/etc/checklist` file for the `/users` file system looks like:

```
/dev/dsk/c1d0s8 /users hfs rw,suid 0 1
```

modify that line to include the quota option. Note that options in `/etc/checklist` entries are separated by commas and no spaces. The line should then look like:

```
/dev/dsk/c1d0s8 /users hfs rw,suid,quota  0 1
```

Create the "quotas" File in the File System's Root Directory

Each file system using disk quotas must have an initially empty file—named `quotas`—in its root directory. The `quotas` file will contain the limits and usage statistics for each user in the file system in binary form.

To create the `quotas` file for the `/users` file system (`/users` must be mounted), use the `cpset` command. For example, you might enter:

```
cpset /dev/null /users/quotas 600 root bin
```

In the above command, `/dev/null` specifies that the file created is empty. `/users/quotas` specifies that the file `quotas` is to be at the root of the file system mounted on `/users`. See `cpset(1m)` in the *HP-UX Reference Manual*.

Note **Assigning user IDs.**

The file `quotas` keeps file system usage information in binary form for users on the basis of their user ID numbers. The `quotas` file can become large if users on a file system have user IDs that are large. The `quotas` file will contain data space for the number of possible users. A high user ID number (a four-digit number, for example) means the file must create space for a large number of possible users; this could result in wasted space if there are only a few users. So, to control the size of the `quotas` file, refrain from using large numbers for user IDs. If you are using SAM to create user IDs, you will not have the problem.

While HP-UX supports "sparse" files and does not allocate disk blocks for non-existent users, the act of restoring a backup or of making a copy of the `quotas` file causes expansion of the sparse file.

Establish Quotas for Users of the File System

To establish quotas for the users of a file system, you can use the `/etc/edquota` command to edit a character representation of the contents of the `quotas` file for that file system. See the `edquota(1m)` in the *HP-UX Reference Manual*. The editor you will use is the one specified by the `EDITOR` environment variable. If you have not specified EDITOR, `vi` will be invoked by default.

Note The `edquota` utility converts binary data from a `quotas` file into a text representation, creates a temporary file, invokes the editor on that file, and converts the edited text back to binary form before storing the data back to the `quotas` file.

Using edquota -p. Let's assume you want to set uniform limits for users in the `/users` file system. To do this, you can set limits for a prototypical user using the `/etc/edquota` command. Then, using the `-p` option, use the `/etc/edquota` command to replicate these limits for other users owning files in the file system.

In the following example, a prototypical user in the file system `/users` is assigned a soft limit of 1000 blocks, a hard limit of 1200 blocks, a soft limit of 250 files, and a hard limit of 300 files.

1. Set the limits for a prototype user. Use the `/etc/edquota` command.

   a. Type: `/etc/edquota protojoe`

   b. While in the editor, type the following:

   ```
   fs /users blocks (soft = 1000, hard = 1200) inodes (soft = 250, hard = 300)
   ```

   After you save the text file, the `quota` file is updated.

2. Now, run the `/etc/edquota` command with the `-p` option to implement the prototype user's limits for other users of the `/users` file system:

   ```
   /etc/edquota -p protojoe alice george
   ```

   This assigns the prototype limits of the prototypical user, `protojoe`, to the actual users, Alice and George. Notice that you can include more than one user on the command line.

Setting quotas for an Individual User. Use the `/etc/edquota` command to set quotas for individuals. For example, suppose you want to set quotas for Ted that would allow him to have higher limits than the prototypical user. Type:

```
/etc/edquota ted
```

When you get into the editor, you would enter a line such as:

```
fs /users blocks (soft = 1200, hard = 1500) inodes (soft = 300, hard = 350)
```

Save the file. Ted now has limits different from the prototypical user.

Note When removing a user from the system, run `/etc/edquota` to set the user's limits to 0. Then, when the user is removed from the system, there will be no entry for that user in the `quotas` file.

Setting Time Limits for a File System's Users

The soft time limits—that is, the times a user is allowed to reduce the number of blocks or files to values below soft limits—are set using the `/etc/edquota` command with the `-t` option. The time limits apply system wide, that is, to all users in a file system. You can set different time limits for files and file system blocks.

Note Users of client machines of an HP-UX cluster can expect a message at a regular time interval of 60 seconds when they have exceeded their file system usage soft limits for either files or blocks.

For example, you could set the soft time limits of ten days for file system blocks and 15 days for files in the file system `/users`.

To edit the `quotas` file to specify time limits, type:

```
/etc/edquota -t
```

When you are in the editor, type the line:

```
fs /users blocks time limit = 10.00 days, files time limit = 15.00 days
```

Saving the file establishes the specified time soft limits for the file system.

The default time limit for both file system blocks and files is seven days. You can specify the default time limits by entering 0 (zero) in fields where you would specify the limits. For example, to implement default limits for the `root` file system, you could enter the line:

```
fs / blocks time limit = 0, files time limit = 0
```

Turning On Disk Quotas

After you have set up disk quotas on a file system, you need to put quotas into operation.

Turning On Disk Quotas by Mounting the File System

When you mount a file system with the `quota` option, disk quotas are turned on, provided, of course, that the `quotas` file exists in the root directory of the file system.

You can:

1. Include the `quota` option in the file system entry in the `/etc/checklist` file so that the system will enable quotas when the `mount -a` command is issued or when it mounts the file system the next time you start up the system.
2. Interactively mount a file system and include the `quota` option in the mount command line.

   For example,

   ```
   mount /dev/dsk/c1d0s8 /users -o quotas
   ```

Turning On Disk Quotas Using the quotaon Command

1. Check the file system for consistency.

   Having to enable disk quotas with the `quotaon` command implies that the file system has been used with disk quotas off and that the `quotas` file has out of date information. That means that the `quotacheck` command should be run on that file system.

   For example, running

   ```
   /etc/quotacheck /dev/dsk/c1d0s8
   ```

   updates the `quotas` file for that file system to reflect any usage while disk quotas was turned off. See the later section "Checking Consistency of File System Usage Data" for a detailed discussion of running `quotacheck` interactively.

2. Use the `quotaon` command.

   Use the `quotaon` command to turn on disk quotas for a mounted file system for which disk quotas is set up but not currently turned on. Remember, the file `quotas` must exist in the root directory of the file system.

   For example, issuing the command

   ```
   /etc/quotaon -v /users
   ```

   will start up quotas on the `/users` file system and print a message that states that quotas are being turned on for the file system.

   You can also use the `/etc/quotaon -v -a` command, in which case the `-a` option turns on disk quotas for each mounted file system with the `quota` options in the file `/etc/checklist`. The `-v` (verbose) option specifies that the affected file systems be listed in a message to the screen.

   If you issue the `/etc/quotaon` command specifying a file system for which quotas are already turned on, there is no effect.

Turning Off Disk Quotas

When you unmount a file system, the system automatically turns off disk quotas.

You can turn off disk quotas for a file system without unmounting that file system by using the `/etc/quotaoff` command. However, the use of `/etc/quotaoff` command is not recommended because once quotas are turned off, the actual disk usage will be inconsistent with the usage recorded in the `quotas` file, making the information in the `quotas` file invalid.

If you use the `/etc/quotaoff` command, you must run the `/etc/quotacheck` command to check the consistency between the actual file usage and the `quotas` file before turning on disk quotas for the file system again. See the section "Checking Consistency of File System Usage Data" for a discussion of using the `/etc/quotacheck` command. If `quotaoff` is not used and file systems are not unmounted improperly, `quotacheck` overhead is minimal.

Displaying File System Usage, Soft and Hard Limits

Commands are available to show information about disk usage and quotas.

Reporting File System Usage

The `repquota` command is available to the system administrator to determine quota information about file systems. For example,

```
repquota /users
```

shows the usage for each user of the file system `/users`. You might get a report that looks like:

```
/dev/dsk/c1002d0s7 (/users):

        Block limits                        File limits
User            used   soft   hard   timeleft    used   soft   hard   timeleft
bill      --     59    100    200                  24     30     40
fred      +-    199    100    200   1.7 weeks      10     30     40
joe       --     63    100    200                   9     30     40
dan       ++    173    100    200   1.4 weeks      32     30     40   1.4 weeks
```

Entering the command,

```
repquota -a
```

displays usage for each user on all file systems listed in the `/etc/checklist` file with the 'quota" option. See `repquota(1m)`.

Reporting a Summary of Ownership

You can use the `quot` command to display the number of 1024-byte blocks in a file system that are currently owned by each user. For example, for the file system `/dev/dsk/c1d0s5`, you could issue the command:

```
/etc/quot /dev/rdsk/c1d0s5
```

You would receive the following output:

```
/dev/rdsk/c1d0s5 (/users):
 2843   benny
 2429   fisher
 1102   ariel
  164   #220
   25   anitasz
   15   nanda
```

You could have specified the mount point directory instead of the device file in the command described above and have received the same output.

Reporting Individual Usage

6

A user can display his or her usage by using the `quota` command. For example, if Joe types `quota`, he will receive warnings about file systems where his usage exceeds limits:

```
quota -v

Disk quotas for joe (uid 203):
Filesystem     usage  quota  limit    timeleft  files  quota  limit timeleft
/users           159    110    210    .8 weeks     24     10     25
```

Only a user with superuser privileges can use the *user* option for the `quota` command to view specific usage and quota information about other users. See `quota(1)`.

Using `quota -v` shows a user all of the quota information on all file systems (including remotely mounted systems) where he or she has limits.

What to Do When Reaching a Hard Limit

When you (or users on the system) reach a hard limit or fail to reduce usage below soft limits, an error message appears on the terminal. For example, if a user reaches the block limit, the following message appears:

```
DISK LIMIT REACHED -- WRITE FAILED
```

When a user reaches the file limit, the following message appears:

```
FILE LIMIT REACHED -- CREATE FAILED
```

When reaching these limits, a user is unable to create files or use additional file system blocks.

Recovery from this condition requires a sequence of steps, depending on whether or not the user is in an editor when receiving the message.

When Not in an Editor

If a user is *not* in an editor when the limit is reached, the user must:

1. Abort the process or processes that are using the file system.
2. Remove enough files to take the number of files and file system blocks well below the soft limits established in the `quotas` file.

   The `quota` command reports whether a user is above or below the limit in the specific file system. The `du` command can help in determining the current number of blocks in files and directories.
3. Try running the aborted processes again.

When In an Editor

When *in* an editor, the user needs to remove files to a level below the quota limits and still preserve the recent changes made to the file being edited. The user can do this by:

1. Writing the file to another file system (such as `/tmp`) where quotas are not exceeded, and getting out of the editor.
2. Removing files until the number remaining is well below the file and file system block quotas, that is, soft limits.
3. Moving the file back into the original file system.

Or, if a job-control shell is being used, the user can do this by:

1. Going to the shell.

   Typing the "susp" character (usually **CTRL-Z**, that is, [CTRL] and [Z] at the same time) moves the user to the shell and suspends the editor.
2. Removing files until the number remaining is well below the file and file system block quotas.
3. Typing `fg` at the shell prompt return to the editor.

6

Checking Consistency of File System Usage Data

You can run the `quotacheck` command to check for inconsistencies between the `quotas` file and actual usage and to update the `quotas` file.

The system updates the `quotas` file when users logout and when file systems are unmounted. In this way, the usage information stored in the `quotas` file matches actual usage. But, if disk quotas are turned off for a file system, then, the `quotas` file and the actual usage will become inconsistent when the file system is used.

Checking Quotas When Starting the System

When `quotacheck` runs, the file system being checked must not be in use. The best way to run `quotacheck` is from the `/etc/bcheckrc` file during boot up, after file systems are mounted. It can be run with the `-a` option so that it checks all file systems listed `/etc/checklist` with the read-write and disk quotas options.

Running quotacheck Interactively

Before you interactively mount a file system, run `quotacheck` to check that file system for any inconsistencies. Make sure the file system you are checking is unmounted.

You can run `quotacheck` with a `-v` option to have `quotacheck` report the quotas information for each user in the file system. Otherwise, it reports only the changed quotas.

For example, you run:

```
quotacheck -v /dev/dsk/c1d0s8
```

The output might look like:

```
*** Checking quotas for /dev/dsk/c1d0s8 (/users)

/dev/dsk/c1d0s8: dan      fixed    files  12 -> 13     blocks  103 -> 128
```

The columns to the right of the Dan's name indicate the `quotas` file for Dan has been fixed, that is, changed to incorporate any changes in the number of files and blocks.

Moving File Systems from One Device to Another

You might want to move the data in a file system on one type of disk to another disk, from one disk section to a logical volume, from one logical volume to another, or from one disk section to another. For example, you might want to move a file system to a larger disk, or you might want to move a file system from an HP-IB disk to an HP-FL disk.

The following steps outline how to copy data from one device to another. An example follows.

1. Back up files from the current device onto tape.
2. Add your new disk, if necessary or create the logical volume.
3. Create new file systems on your new disk, section, or logical volume.
4. Edit the `/etc/checklist` file.
5. Mount your new file system.
6. Restore the files backed up on tape to your new file systems.

Example, Moving File Systems from Disk to Disk

6

Let's assume you want to copy data from your current HP-IB disk, a Model HP 7935, to a Model HP 7937FL disk. You have file systems in sections 8 and 10 of the HP 7935 disk, which has been assigned logical unit number 3. By looking at the file `/etc/checklist`, you can determine the mount point directory and the device file name for each of the sections containing your file systems. For example,

```
more /etc/checklist
.
.
/dev/dsk/c3d0s8    /usr/users    hfs rw 0 3 # database
/dev/dsk/c3d0s10   /mnt/tools    hfs rw 0 4 # tools
.
.
```

Copy this file system to the new disk as follows:

1. Back up your file system onto tape.

   Suppose the files you want to move to the new disk are now in the `/usr/users` and `/mnt/tools` file systems. To back up these files to 6250 bpi magnetic tape, load your tape on the tape drive and use the following commands:

   ```
   /etc/fbackup -0f /dev/rmt/0h -i /usr/users
   ```

   Detailed information about performing backups is in Chapter 9, "Backing Up and Restoring Your Data".

2. Add the new disk to your system.

   The manual *Installing Peripherals* explains how to install your new disk.

   When you reboot your system after physically connecting and turning on your disk, the system will automatically add the disk to its I/O configuration, make all the necessary device files, and assign the disk a logical unit number.

   Let's assume your new disk is the seventh disk on your system and it is assigned logical unit 7.

3. Make file systems on the new disk.

   To determine in which sections of your new disk to build the new file systems, refer to the file `/etc/disktab`. For example, `/etc/disktab` shows you that on the HP 7935, section 8 has 190460 units of 1024 bytes and section 10 has 129024 units of 1024 bytes. On an HP 7937FL disk, section 8 has 353778 units of 1024 bytes and section 10 also has 129024 units of 1024 bytes. The comparison shows that the sizes of sections 8 and 10 on the new disk are equal to or larger than the equivalent sections on the current disk. So, you can use the same numbered sections on the new disk for your file systems.

   Use `newfs` to make the new file systems on your new disk. For example,

   ```
   newfs /dev/rdsk/c7d0s8 hp7937
   newfs /dev/rdsk/c7d0s10 hp7937
   ```

   See the earlier section "Using the `newfs` Command.".

4. Edit `/etc/checklist`.

   After you have created the file systems on your new disk, edit the file, `/etc/checklist`. Change the device files to reflect your new disk. For example, change `/dev/dsk/c3d0s8` to `/dev/dsk/c7d0s8` and change `/dev/dsk/c3d0s10` to `/dev/dsk/c7d0s10`. Editing `/etc/checklist` tells the system to automatically mount the new file systems each time you boot your system.

5. Mount new file systems.

   Now, mount the two new file system by running the command:

   ```
   mount -a
   ```

6. Restore your backed up files to the file systems on the new disk.

   Load the tape with your files on the tape drive, and restore the files to their new file systems. You can use the following commands.

   ```
   /etc/frecover -xf /dev/rmt/0h -i /usr/users
   /etc/frecover -xf /dev/rmt/0h -i /mnt/tools
   ```

HP-UX System Files

The `root` (`/`) file system contains `hp-ux`, which is the configured operating system (also known as the kernel), and the operating system files. These files reside in subdirectories described in Table 6-3.

- The subdirectories `/dev`, `/lib`, `/bin`, and `/etc` form the core of the `/` file system. Because they reside in `/`, they cannot be inadvertently unmounted.
- The subdirectories `/usr` and `/tmp` typically undergo modification and it is recommended that they be set up (at installation time, preferably) in logical volumes or disk sections that are separate from the core root subdirectories.

Because the operating system depends on these files, you should keep them separated from the directories and files required for general use on your system. As you add users and applications to your system, it's best to organize space for them in other file systems.

Table 6-3. Root File System Subdirectories

Directory	Contents
/bin	Compiled, often-used commands.
/dev	Block and character special device files used to communicate with devices. See *mknod*(1M) in the *HP-UX Reference Manual*.
/etc	Most system administrator commands and configuration (customization) files.
/etc/newconfig	Customized configuration files and shell scripts during an update so you can use them for reference. You typically copy many of these files back into `/etc`. The `/etc/newconfig/README` file contains useful information about files in `/etc/newconfig`.
/etc/conf	Kernel configuration description files.
/etc/filesets	A list of loaded filesets.
/lib	Object code libraries and related utilities.
/mnt	User's home directories (usually).
/system	Revision lists and customize scripts from installation.
/tmp	Temporary files.
/usr	Commands and log files.
/usr/adm	System administration data files.

Table 6-1. Root File System Subdirectories (Continued)

Directory	Contents
/usr/bin	Commands not required to boot, restore, or repair the file system.
/usr/contrib	Files and commands contributed by user groups.
/usr/contrib/bin	Contributed commands.
/usr/contrib/lib	Contributed object libraries.
/usr/contrib/man	On-line documentation for contributed commands.
/usr/diag	Diagnostic tools.
/usr/include	High-level C language header files; the shared definitions.
/usr/include/local	Site-specific C language header files.
/usr/include/sys	Low-level, kernel-related C language header files.
/usr/lib	Less-used object-code libraries, utilities, lp commands, and miscellaneous data files.
/usr/lib/uucp	Configuration files for UUCP at install.
/usr/local	Localized, site-specific files.
/usr/local/bin	Localized, site-specific commands.
/usr/local/lib	Object code libraries for the site-specific commands.
/usr/local/man	On-line documentation for the site-specific commands.

Table 6-1. Root File System Subdirectories (Continued)

Directory	Contents
/lost+found	Orphaned files and directories created by **newfs** and used by **fsck**.
/usr/mail	Used by the mail facilities for your mail box.
/usr/news	System-wide news files.
/usr/spool	Spooled (queued) files for various programs.
/usr/spool/cron	Spooled jobs for **cron** and **at**.
/usr/spool/lp	Control and working files for the **lp** spooler.
/usr/spool/uucp	Queued work files, lock files, log files, status files, and other files for UUCP.
/usr/spool/uucppublic	Files freely accessible to remote systems via LAN and **uucp**.
/usr/tmp	Temporary large files.
/usr/man	All shipped on-line documentation.
/usr/man/cat1 ... cat9	On-line documentation that has already processed to speed up access.
/usr/man/cat1.Z ... cat9.Z	Compressed versions of **cat** directories.
/usr/man/man1 ... man9	The unformatted on-line documentation pages.
/usr/man/cat1.Z ... cat9.Z	Compressed versions of the on-line documentation pages.

The Location of the Root File System

The system needs to know where to locate the `root` file system each time it starts up.

The disk that contains `root` is the "root disk" and the disk section and or the logical volume that contains `/` is the root device.

The system kernel configuration file, `/etc/conf/gen/S800`, contains an entry that specifies the hardware address of the root disk. The following describes how to specify the location of `root`, depending on whether `root` is in a logical volume or a disk section.

LVM

If the root disk on your system is an LVM disk, then `root` (the root file system) exists in the root logical volume on that disk. Most of the time, the root disk is the same disk as the *boot* disk.

By default, the `S800` file should contain an entry for `root` that resembles:

```
.
root on lvol;
.
```

With this instruction, the bootstrap program, `hpux`, locates the HP-UX kernel (the HP-UX program) based on information contained in the "`LABEL`" LIF file on the boot disk. The kernel then runs and locates the `root` logical volume by reading data stored in the Boot Disk Reserved Area (BDRA), an LVM data structure located on the boot disk.

The BDRA contains information about the location of the `root`, primary swap, and dump logical volumes. The commands *lvlnboot*(1M) and *lvrmboot*(1M) provide the means to specify or change the information that the BDRA contains.

Chapter 8, "Managing Logical Volumes," discusses how to use the *lvlnboot*(1M) and *lvrmboot*(1M) commands to change the information in the BDRA about the location of `root`, primary swap, and dumps logical volumes.

Non-LVM

The `root` disk contains root in a fixed section (the default is section 13). If you look in the `S800` file, you will find an entry that either reads:

```
.
root on default;
.
```

or an explicit statement that specifies the location of `root`.

The statement that specifies the location of the root file system in the `S800` file has the form:

`root on` *software_module_path* `at` *hardware_path* `section` *n*`;`

For example, if root were on section 0 of an HP-IB disk that had its bus address set to 2 and that was connected to an HP-IB interface card in slot 1 of the CIO channel adapter, the entry in the `S800` file would be:

```
root  on  cio_ca0.hpib0.disc1  at  4.1.2  section 0;
```

6

Moving the Root File System from Section to Section

If you run out of space in the `root` section of your system disk, you can move the `root` directory to a larger section. (Moving `root` to another logical volume is discussed in Chapter 8, "Managing Logical Volumes.")

SAM does not provide a means of doing this, and you should attempt it only if you are an experienced system administrator.

If you need to move `root`, read through this section to the end. If there's anything in the procedure you are not confident you can do, don't attempt it. Specifically, you should be thoroughly familiar with the following commands and utilities:

- *uxgen*(1M)
- *isl*(1M)
- `hpuxboot(1M)`
- `newfs(1M)`
- `fbackup(1M)`
- `frecover(1M)`
- `lif(4)`
- `lifcp(1)`
- `mount(1M)`
- `umount(1M)`

6

Summary of Steps to Move the Root Disk Section

1. Edit the `S800` file to show the new location of `root`.

   For example:

   ```
   root on cio_ca0.hpib0.disc1 at 8.0.0 section 4;
   ```

2. Run uxgen to build the new kernel.
3. Copy the existing root file system to the new location.
4. Update `hp-ux`, `/etc/checklist`, and `/etc/mnttab` in the new root file system.
5. Reboot the system but interact with ISL to select the new kernel and make sure that all changes were made correctly.

6. Modify the boot partition to automatically boot from the new root section.

Example

The following example exchanges the locations of the `root` and `/tmp` file systems on an HP 7935 disk.

Assume your disk has the following sections:

```
/        on        /dev/dsk/c0d0s0          # 24 MB
swap     on        /dev/dsk/c0d0s1          # 49 MB
/tmp     on        /dev/dsk/c0d0s4          # 110 MB
/usr     on        /dev/dsk/c0d0s3          # 30 MB
/mnt     on        /dev/dsk/c0d0s5          # 54 MB
/extra   on        /dev/dsk/c0d0s10         # 132 MB
```

The `root` file system is nearly full but the `/tmp` space is larger than needed, so we are going to move `root` to section 4 and `/tmp` to section 0.

First, build a new kernel

6

1. Change directories:

   ```
   cd /etc/conf/gen
   ```

2. Make a new `uxgen` input file and edit it:

   ```
   cp S800 S800.BCKUP        back up the existing uxgen input file
   vi  S800.BCKUP            edit the new uxgen input file
   ```

3. Change the line specifying the root location to:

   ```
   root  on  cio_ca0.hpib0.disc1 at 8.0.0 section 4;
   ```

   Save the new `uxgen` input file.

4. Now (assuming that you have enough space to build the kernel):

   ```
   uxgen S800.BCKUP          execute uxgen using new input file
   ```

   This creates `/etc/conf/S800.BCKUP/hp-ux`.

Then, back up your system completely

```
/etc/fbackup -u0f /dev/rmt/0h -g /usr/adm/fbackupfiles/graphs/g1\
-c /usr/adm/fbackupfiles/fb_config
```

This example recursively backs up all files and directories listed in the text file `/usr/adm/fbackupfiles/graphs/g1` to tape at `/dev/rmt/0h`.

Put the system into single-user mode

```
shutdown
```

Copy the root file system

1. Remove any unwanted files from `/tmp`.
2. Unmount the `/tmp` file system:

   ```
   /etc/umount /dev/dsk/c0d0s4
   ```

3. Build a file system and create a directory for it:

   ```
   /etc/newfs -n -v -b 8192 -f 1024 /dev/rdsk/c0d0s4 hp7935
   mkdir /newroot
   ```

4. Mount the new file system and create a graph file to use with *fbackup*(1M) based on a listing of currently mounted systems:

   ```
   /etc/mount /dev/dsk/c0d0s4 /newroot
   /etc/mount > /graph
   ```

5. Edit the graphfile that frecover uses during backup:

   ```
   vi /graph
   ```

   Enter text, such as:

   ```
   i /
   e /newroot
   e /mnt
   e /usr
   e /extra
   ```

6. Now copy the root file system:

   ```
   cd /newroot
   ```

```
/etc/fbackup -f - -g /graph | /etc/frecover -x -X -f -
```

7. Now create directories on which to mount the other file systems, such as those excluded in the graph file. From the directory `/newroot`, type:

```
mkdir tmp extra mnt usr
```

Update two files on the new root

1. Modify `/newroot/etc/mnttab` so the device name for `/` is `c0d0s4` and the one for `/tmp` is `c0d0s0`.
2. Also change the device names in `/newroot/etc/checklist`.

Bring the new kernel over

```
cp /etc/conf/S800.BCKUP/hp-ux /newroot/hp-ux
```

Reboot the system to test the new kernel and root file system

1. Type:

```
/etc/reboot
```

6

2. Interrupt the automatic boot process. When you see:

```
Autoboot from primary path enabled.
To override, press any key within 10 seconds.
```

press a key. The ISL program on partition 6 does not know that **hp-ux** was moved.

3. Boot from primary path. Answer y to the question:

```
Boot from primary boot path (Y or N)?>Y
```

4. Interact with IPL. Answer Y to the question:

```
Interact with IPL (Y or N)?>Y
```

5. When the system displays the ISL prompt, type:

```
hpux disc1(8.0.0;4)hp-ux
```

The kernel (hp-ux) is now booted from the new root on section 4. The system should come up normally except that the old root file system still exists in `/tmp`.

If the system does not come up normally, reboot without interrupting the automatic boot. This will bring up the old kernel and root file system.

Create new AUTO file for autoboot

You can then modify the AUTO file in the boot partition on disk to allow for automatic boot of the root file system from section 4.

1. Change to a directory (*dirname*) with more than 2 MB of free space:

   ```
   cd dirname
   ```

2. Copy the lif image of the AUTO file in the root partition to `my.AUTO` so you can read it:

   ```
   lifcp  /dev/rdsk/c0d0s6:AUTO  my.AUTO
   ```

3. Read the autoboot file:

   ```
   more my.AUTO
   ```

   You will see something like:

   ```
   hpux disc1(8.0.0;0)hp-ux
   ```

4. Write down this old boot path. Change the boot path in the autoboot file so it boots from Section 4:

   ```
   mkboot -a "hpux disc1(8.0.0;4)hp-ux" /dev/rdsk/c0d0s6
   ```

   This command overwrites the old autoboot file.

5. Reboot the system:

   ```
   /etc/reboot
   ```

The system should come up normally.

If the system does not come up correctly, reboot the system from the copy of the old root file system. You need to interact with ISL and type in the old boot path.

Make a new /tmp file system over the old root file system

1. Unmount the old root file system:

```
/etc/umount /dev/dsk/c0d0s0
```

2. Make the/**tmp** file system:

```
/etc/newfs -n -v -b 8192 -f 8192 /dev/rdsk/c0d0s0 hp7935
```

3. Mount the /tmp file system:

```
/etc/mount /dev/dsk/c0d0s0 /tmp
```

4. Restore any files you saved from /tmp.

Creating a LIF Volume

The LIF format is an HP standard format used for directories and files on flexible and hard disks. It is used on Series 200, 300, 500, and 800 computers that use the HP BASIC or Pascal operating systems, especially systems using 5.25- and 3.5-inch flexible disks made by Hewlett-Packard Company.

Prerequisites and Conditions

- If your media has never been initialized, you must use `mediainit` to initialize the media, then use `lifinit` to create a LIF volume.
- A LIF volume can be created directly on a disk or within the HP-UX file system.
- Test previously initialized media to see if it is a LIF volume *before* you initialize a disk. Assuming the media to check is associated with `/dev/rdsk/c2d0s10`, execute:

  ```
  lifls /dev/rdsk/c2d0s10
  ```

- You get one of the following responses:
  - For an initialized disk that is a LIF volume and contains files, you see a listing. For example, you might see:

    ```
    PLOT_DEMO HELLO_TEST MY_FILE
    ```

  - For an initialized disk that is a LIF volume and has no files, you see an empty line.
  - For an uninitialized disk, you see:

    ```
    lifls: Can't list /dev/rdsk/c2d0s10; not a LIF volume
    ```

  You should continue, or not, depending on what you see.
- There is no SAM procedure.

Manual Procedure for Making a LIF Volume

1. Become the root user.
2. Use `lifinit` to create a LIF volume according to the following syntax:

   `lifinit` [`-v`*nnn*] [`-d`*mmm*] [`-n` *VOL_NAME*] *FILE_NAME*

   If you use the last two parameters, capitalize them.

   a. The following example writes a LIF volume header to the disk associated with the device file named `/dev/rdsk/c2d0s8`:

      ```
      lifinit /dev/rdsk/c2d0s8
      ```

   b. The following example writes a LIF volume header (named WORK) to an HP-UX file (named TMP) that will be copied to a flexible disk where the volume size (270336) is the number of bytes contained on a 5.25-inch flexible disk and the disk can contain 240 directories.

      ```
      lifinit -v270336 -d240 -nWORK TMP
      ```

3. The *HP-UX Reference Manual* has more information about the options used with `lifinit`.

6

7

Managing Swap Space

The **CPU** (Central Processing Unit) divides its time among all **processes** (running programs) that are active at any given time. When a process has received its share of CPU time, or when it reaches a point where the next instruction can't be executed immediately (for example, when the process is waiting for data that has to be retrieved from disk), the process might be **swapped out**, that is, all or part of a process might be moved from main memory to a reserved area, usually on disks, known as a **swap area**. The `root` (system) disk usually contains the **primary swap** area, and other disks (either in disk sections or logical volumes) contain secondary swap areas.

Whether a process actually is swapped out depends on how busy the system is. If the system is not busy (that is, if the active processes are few enough and/or small enough for all of them to fit in main memory), no swapping occurs. There is a daemon that monitors free memory and keeps it to a maximum by cleaning out data that is no longer needed. If it cannot free up enough space to meet demand, it enables swapping.

This chapter explains how to manage your system's swap space. Managing swap space on your system involves determining how much and what type of swap space the system needs, and also involves adding or removing it as the system's needs change.

A complete conceptual discussion of swap space is in the manual *How HP-UX Works: Concepts for the System Administrator*.

Terms Used in this Chapter

The following terms appear frequently in discussions in this chapter. The list of terms appears here so it can be located easily. You may scan the list now and refer to it later.

device file A file used by the computer to communicate with a device. It is sometimes referred to as a "special file" or "device special file." For swapping, the device is usually a logical volume or a disk section.

device swap A disk or disk section reserved exclusively for swap space.

disk section On Series 800/600 computers, a logical division or partition on a hard disk. To HP-UX, each section appears as an independent file system or swap area, that is, a separate entity with its own device file. Disk sections are fixed in size.

dumps area A disk section or logical volume designated to receive the dump of the core memory image during a system crash. The core dump can be used to troubleshoot and restore the system to normal working order. The dumps area can be normally used as a swap area or it can be reserved exclusively for dumps.

7

dynamically allocatable swap space Disk space that can be allocated while the system is running, that is, without having to configure it into the kernel and reboot. Both device swap and file system swap can be dynamically allocated.

file system The organization of files on disk into hierarchical directories; file systems can exist in logical volumes or disk sections.

file system swap Space within a file system that is used for swapping. File system swap shares disk space with a file system, whereas device swap has exclusive use of a logical volume or a disk section.

interleaved swapping Swapping where pages are swapped to and from two or more devices on an interleaved basis.

logical volume A *logical* (rather that physical) construction that is a map of data stored on physical volumes (disks). A logical volume can be conceptualized as a storage device of flexible size. The data in a logical volume can map to one or more physical volumes (see Chapter 8, "Managing Logical Volumes").

primary swap A device swap area which has been configured in the system's kernel and is available when the system boots.

process A program running in memory. The image of a process may be temporarily stored on disk, by means of swapping, to free up space in physical memory.

pseudo swap System memory used for swap space.

secondary swap Device swap space used in addition to primary swap. May be configured into the system or added dynamically.

swap in The process of reading the process's image from the swap space on the disk into the computer's physical memory.

swap out The process of writing the process's image from the computer's physical memory onto the swap space on the disk.

swap space Space on a disk used for temporarily storing the process image.

Types of Swap Space

There are two types of swap space: device swap space and file system swap space. Each type of swap space is used differently by the system and each has advantages and disadvantages.

Device Swap

Device swap space occupies a logical volume or a disk section. The logical volume or disk section is reserved expressly for swapping purposes. Your system must have at least one device swap area available to the system when it boots. This area is known as the **primary** swap area. The primary swap area is configured into the system's kernel and is specified in your system's `S800` file. You can add **secondary** swap devices, either on the root disk or, preferably, on another disk.

Device swapping has the advantage of being fast because the system accesses the logical volume or disk section directly without going through a file system (as it does in the case of file system swap; see the next section).

You can allocate device swap dynamically using either SAM or the `/etc/swapon` command; see the section "Adding Secondary Device Swap."

File System Swap

Setting up **file system** swap space allows a process to use space within an existing file system if it needs more than the allocated device swap space. Although file system swap space coexists with the device swap space and it is used by the system *after* device swap is used.

When your system only occasionally needs additional swap space, file system swap provides an efficient way to increase it, because it uses unused file system space rather than reserving an entire dedicated logical volume or disk section for occasional use. On the other hand, because file system swap requires the system to perform a greater amount of processing and is usually slower than device swap, it is not a good permanent solution.

How Much Swap Space Do You Need?

Because swap space is a disk-based component of virtual memory, it must be large enough to hold all the processes that could be running at peak times.

Your system has a default amount of swap space initially allocated to it; in most cases, the system could not run without it. If system performance is good, and, in particular, if you are not getting swap errors such as

`Sorry pid` *pid number* `was killed due to no swap space`

then your system has adequate swap space.

If you know or suspect that you will have to increase (or decrease) your swap space, there are two ways to figure out how much swap space you need:

1. Use the precise but detailed formula in appendix B, "Swap Space Computation."

   Or,

2. Use a "rule of thumb" method to estimate your swap space requirements. The following section describes a method of estimation.

How Much Swap Do You Have?

To find how much swap space is configured on your system, run the command: *swapinfo*(1M).

```
swapinfo

   Kb      Kb      Kb   PCT  START/      Kb
TYPE   AVAIL    USED    FREE  USED   LIMIT RESERVE  PRI  NAME
dev    48560    1456   47104    3%       -       -    0 /dev/dsk/c0d0s15
.
dev     3072    1024    2048   33%       -       -    1 /dev/dsk/sales_lv
.
.
fs         0       0       0  100%    4096    1000    2  /lpspmg
.
hold       0   11204  -11204
```

The output tells you the type of swap by location, how much of it is available, how much is used, and how much is free.

Under type, in addition to `dev` (device) and `fs` (file system), you might see "`hold`." The operating system can "hold," or reserve, an amount of swap space (from no specific device or file system) based on the possible requirements of currently running processes. This is a means to ensure currently running processes do not run out of swap space.

It is possible that a new process will not be able to start until another process terminates. Until a process terminates, swap space held for it cannot be allocated or held for new processes.

7

Estimating Swap Space Needs (A Rule of Thumb)

You can estimate the amount of swap space you need by adding the space required by your largest application to the current amount of swap space.

Use the following to determine the swap space. Remember, 1 Kbyte = 1024 bytes.

1. Determine the swap space (in Kbytes) required by your largest application (look in the manual supplied with your application or ask the manufacturer). If you will be running several applications concurrently, you should add their swap space requirements together.	_____
2. Add the current amount of swap space based on the output of the `swapinfo` command.	_____
TOTAL swap space needed (in Kbytes); sum of 1 and 2.	_____

Swap Space Defaults

The default maximum amount of swap space you can configure—for both device swap and file system swap combined—is approximately 537 Mbytes. The tunable system parameter *maxswapchunks* controls this maximum.

The parameter *maxswapchunks* (default value of 256) limits the number of swap space chunks. The size of each chunk of swap space is the product of *swchunk* (default value of 2048) and `DEV_BSIZE`, the size of a disk sector. Typically, `DEV_BSIZE` (see the file `/etc/disktab`) has a value of 1024 bytes.

For example, when the value of the parameter *maxswapchunks* is 256, the maximum configurable device swap space (*maxswapchunks* x *swchunk* x `DEV_BSIZE`) is:

```
256 x 2048 x 1024 bytes = 537 megabytes
```

If you need to increase the limit of configurable swap space beyond the default, increase the value of the *maxswapchunks* operating system parameter. To change the values of any operating system parameters, you will need to reconfigure your system's kernel. You can use either SAM or HP-UX commands to change system parameters and regenerate a new kernel.

Appendix A, "System Parameters" contains more detailed discussions of the system parameters.

Allocating Swap Areas: Guidelines

After determining how much swap space your system needs, you need next to consider how you are going to allocate the swap space.

Distributing Device Swap Areas

From the point of view of performance, two swap areas on different disks are better than one swap area with the equivalent amount of space. This allows **interleaved swapping**; that is, where swapping I/O transactions (writes) are alternated or interleaved between the devices. Two memory pages, if they are contiguous, are written to only one device.

Device swap areas should be of similar size for best performance because, otherwise, when all space in the smaller device swap area is used, the larger swap area is all that is available and interleaving is no longer possible.

When increasing your swap space:

- If you have only one disk, move the current primary swap area to a larger section if at all possible.
- If you have more than one disk, choose a section or a logical volume on a disk other than the one with the primary swap area.
- The maximum number of devices that you can configure for swap is controlled by the `nswapdev` tunable system parameter. Appendix A, "System Parameters," describes using `nswapdev`.

Don't create a separate device swap area on the disk containing the primary swap area, because this will cause excessive head movement on that disk and slow the system down.

There are examples and procedures later in this chapter to show you how to move and add swap space.

If you are using LVM, you should set up multiple device swap areas in logical volumes that are on different disks (physical volumes).

See the section "Setting Up Logical Volumes for Swap" in Chapter 8, "Managing Logical Volumes" for a description of how to set up logical volumes on specific disks.

Choosing Device Swap Areas

For device swap, you must allocate logical volumes or disk sections.

Disk Sections

The sizes of the sections on each drive are listed in the `/etc/disktab` file. For example, the amount of space in section 15 of an HP 7937 is 48560 blocks or approximately 48 Mb (a block is 1024 Kb).

An easier approach is to let SAM list any available disk sections and their sizes on each disk at the time you add device swap (see "Adding Device Swap in a Disk Section Using SAM").

Logical Volumes

You can create logical volumes the size you require. If you are using LVM, you can create the logical volume right before you enable it for swap. You can do this with either SAM or HP-UX commands.

Guidelines for Setting Up File System Swap Areas

When you need more swap space and you have no devices available for additional device swap, you can dynamically add file system swap to your system. Use the following guidelines.

- To keep good system performance, avoid using heavily used file systems such as `root` for file system swap.

- Use the `bdf` command to check file systems for available space. For example,

```
bdf

Filesystem          kbytes     used    avail   capacity   Mounted on
/dev/vg0/lvol1      22623     19275     1082     95%       /
/dev/vg2/lvol6      152031    105319    31508    77%       /usr
.
.
/dev/dsk/c1d0s10    121771    31166     78427    28%       /extra
/dev/dsk/c1d0s14    121663    2073      107423   2%        /tmp
/dev/dsk/c0s0s3     179565    153566    8042     95%       /mnt
```

  You can also use the *swapinfo*(1m) command to show information about file systems for which swap might be already enabled.

- Enabling file system swap allows more processes to run simultaneously. This may result in slower system response time.

- The maximum number of file systems you can swap to is controlled by the value of the `nswapfs` parameter. Appendix A, "System Parameters," contains information about the `nswapfs` parameter.

  If you enable two file system swap areas, it is preferable to enable them on file systems that are located on separate disks or, in the case of LVM, separate physical volumes. If you enable two different file system swap areas on one disk, excessive disk head movement will slow your system's performance.

Guidelines for Assigning Swap Priority

Each swap area requires an assigned priority. Priorities range from 0 (the highest) to 10 (the lowest). The system uses the swap areas with higher priority first. If you assign the same priority to two different swap areas, the system uses each of them on an alternating, or interleaved basis. The system gives device swap priority over file system swap when each has the same priority.

In general, it's best to assign highest priorities to the swap areas which afford the fastest performance. This means: give device swap areas priority over file system swap areas, give faster devices priority over slower devices, and give lower-use file systems priority over higher-use file systems.

The primary swap area has priority 0. Device and file system swap areas enabled dynamically default to a priority of if no priority is specified.

Adding, Removing, or Changing Device Swap Space

As you manage your device swap space, you will be performing at least one of the following tasks:

- Increase device swap space.

  Do this by extending or adding to the existing logical volumes used for swap, or, if you are not using logical volumes, use an available disk section for swap. You can do this dynamically, while the system is running.

- Reduce device swap space.

  Do this by reducing the size or number of logical volumes used for swap, or, if you are not using logical volumes, discontinue the use of a disk section for swap. Swapping to devices cannot be done dynamically. You must reboot the system to put the device swap changes you make into effect.

The "System Administration Manager," SAM, provides the capability to perform these tasks. You can also use HP-UX commands.

Changing Primary Swap from Section to Section

You can move your **primary swap** area to a new disk or new disk section.

Before you contemplate doing this, however, refer to the earlier sections about estimating swap space needs. You might not need to change primary swap space, but add device swap instead.

Configuring Primary Swap with SAM

You can use SAM to change the primary swap device.

You can review the following sections in this chapter to help you plan how to set up primary swap and then refer to the section "Reconfiguring the Kernel to Modify Special Devices" in Chapter 2, "Constructing an HP-UX System".

Example, Changing the Primary Swap Area Using HP-UX Commands

1. Decide on the new location for primary swap.

   You might choose another section on the disk where your primary swap is currently located, or you might choose another disk. If you are installing a new disk, refer to the *Installing Peripherals* manual.

   Let's suppose you decide to add another disk and use it entirely for swap. For example, you might add an HP-IB disk—an HP7937—and connect it to the interface card in CIO slot 0 and set its bus address to 3.

   Type `/etc/ioscan -f` to display information that shows the added disk. For the disk you added in this example, you should see a line that resembles:

```
Class       LU  H/W Path Driver                H/W Status        S/W Status
==========================================================================
.
.
disk         3  4.0.3    cio_ca0.hpib0.disc1   ok(0x22e)         ok
.
```

2. Edit the `S800` file to redefine the primary swap area.

   When you edit the `S800` file, look for the line that reads:

   ```
   swap   on   default;
   ```

   Instead of specifying the default, you would replace the line in the existing file with the explicit specification for primary swap. For our example, follow the words `swap on` with:

   a. the software module path (this is "`Driver`" in the `ioscan` output)
   b. "at"
   c. the hardware module path (this is "`H/W Path`" in the `ioscan` output)
   d. "section"
   e. the section number; for our example, were are using "2", the entire disk

   For *this example*, the line would read:

   ```
   swap  on  cio_ca0.hpib0.disc1 at 4.0.3 section 2;
   ```

3. Run `uxgen` to rebuild the new kernel. Refer to the section in Chapter 2, "Reconfiguring the Kernel," which describes running *uxgen*(1M). That section also contains important information that will help prepare you for possible problems booting with your new kernel.

4. Shut down your system and turn off the power.

   After building the new kernel, shut down the system to install your disk. Refer to the manual *Installing Peripherals* for disk installation instructions.

5. Turn on the device and reboot.

Caution If the system detects a file system present in the disk section you plan to use for primary swap, it will issue you a warning during the reboot process. The warning will tell you that the file system will be destroyed if primary swap is configured in that section.

When you receive the warning, the bootup process will pause for 15 seconds, giving you time to decide whether to:

a. allow the boot process to continue, allowing the primary swap area to be configured (as you intend) in the area now occupied by a file system
b. stop the boot process by shutting down the computer

Even if you let the boot process continue, the file system will not be overwritten until any swapping takes place. If you elect to halt the boot process, you can reboot using your backup kernel (See the section "If the New Kernel Doesn't Boot"): then, you can take one of the following actions:

- use another disk or disk section for primary swap in place of the one that contains the file system; if you use another disk section for primary swap, you will need to re-edit your S800 file to specify that section and generate a new kernel;
- use the disk with the file system, but save the desired files in the disk section before using it for swap

Try your new configuration again.

Your new primary swap area is installed. You can now use the previous primary swap disk section for file storage or as secondary swap.

Changing Primary Swap Logical Volume

Typically, you cannot extend the primary swap logical volume because its physical extents must be contiguous, usually (but not necessarily) on the root disk.

You can attempt to increase the size of the primary swap logical volume by using the *lvextend*(1M) command. If this should fail, it is likely because disk space (physical extents) contiguous with the swap space had been previously allocated.

You can then create a new logical volume for primary swap within the root volume group, which is the volume group that contains the `root` logical volume.

You can find the procedures for extending a logical volume or creating a new primary swap logical volume in chapter 8, "Managing Logical Volumes."

Adding/Removing Device Swap

You can add (or enable) and remove device swap areas (other than the primary device swap) either by using SAM or by using HP-UX commands. With either method, you can enable device swap dynamically, that is, while the system is running.

We recommend you use SAM when possible because SAM can accomplish tasks more easily and with less chance of error.

Both logical volumes and disk sections can serve as swap devices.

Are You Using LVM?

If you are not sure whether you are running logical volumes or not, before you run SAM, type:

```
/etc/vgdisplay
```

- You are *not* using LVM if you get the message,

  ```
  /etc/vgdisplay: Command not found.
  ```

  You need to use a disk section for device swap. Go to the section "Adding Device Swap in a Disk Section Using SAM".

- You do use LVM if you get no response at all or any other message, usually a message that discusses LVM features. Proceed to the next section, "Adding Device Swap in a Logical Volume Using SAM".

(See chapter 8, "Managing Logical Volumes," for a detailed description of LVM.)

Adding Device Swap Using SAM

With SAM you can add device swap in either logical volumes or disk sections.

How to Use SAM

To use SAM,

- Ensure that you have superuser capabilities.
- Type:

  `/usr/bin/sam`

  Activating the [Help] button from a dialog or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

  Choosing an item from the "Help" menu within a functional area gives you information about:

  - the current functional area
  - keyboard navigation within SAM
  - using the SAM help system
  - displaying the version of SAM you are currently running

Pressing the [f1] key gives you context-sensitive information for the object field at the location of the cursor.

See Chapter 1, "Introduction to System Administration" for additional information about using SAM. If you aren't familiar with SAM, take a moment to look at that overview material. There is a help system in SAM to assist you.

Also, remember the section later in this chapter, "Guidelines for Using Logical Volumes," can help you make optimal decisions.

Adding Device Swap in a Logical Volume Using SAM

Unless you have an empty logical volume to use for device swap, you will have to create the logical volume first. If you already have an empty logical volume to use for device swap, you can skip to the section, "Adding Device Swap in a Logical Volume Using SAM".

Creating a Logical Volume for Device Swap

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate [Open].
3. Highlight `Logical Volumes` and activate the [Open] control button.
4. From "Actions" on the menu bar, choose `Create....`
5. Highlight a volume group in which to create the logical volume.

   Note that on the displayed list, you can see how much space is available in each of the existing volume groups.
6. Enter a name, `swap1`, in the "Logical Volume Name:" field, or leave the field blank and let LVM assign the logical volume name.
7. Enter the size of your logical volume in the `Logical Volume Size (MB)` field. Note that values are rounded up to the nearest multiple of the physical extent size.

   (If you want to mirror the logical volume, refer to the mirroring discussion in Chapter 8, "Managing Logical Volumes".)
8. Activate the [OK] control button to create the logical volume.

   Note that once SAM creates the logical volume, you can proceed to the subarea in SAM that enables you to add swap.

7

Adding Device Swap in a Logical Volume Using SAM

After creating a logical volume for device swap, add the device swap as follows:

1. From "List" on the menu bar, choose `Swap`.
2. From "Actions" on the menu bar, choose `Add Device Swap` and then choose `Using the Logical Volume Manager...`
3. Highlight the logical volume you want to use for swap in the field: "Select Logical Volume." If no empty logical volume exists, you must create one; see the previous section, "Creating a Logical Volume for Device Swap".

Indicate when you want to enable the swap in the appropriate box. If you want to modify the priority, do that now. See the earlier section, "Guidelines for Assigning Swap Priority".

Adding Device Swap in a Disk Section Using SAM

To enable device swap in a disk section, an empty disk section must be available. If no disk sections are available, then you must add a disk.

Refer to the *Installing Peripherals* manual for instructions on adding your disk. Then, refer to the section "Adding a Disk" later in this chapter.

When you have added the disk and restarted your computer, you can add swap using SAM as follows:

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate the (Open) control button.
3. Highlight `Swap` and activate the (Open) control button.
4. From "Actions" on the menu bar, choose `Add Device Swap` and then choose `Not Using the Logical Volume Manager`.
5. Highlight the disk you will use in the field labeled: "Select a disk."

   If the disk you are adding is not on the list of unused disks:

   - Activate the `Device Missing...` control box. SAM will now direct you through a diagnostic process that enables you to configure your disk.

     Some of the things SAM will ask you to check are:

     - Whether the device is turned on.
     - Whether the device was on when SAM started.
     - The connections.
     - Whether the driver for the disk is configured in the kernel.

   When you have successfully added your disk, you can resume.

6. Highlight the disk section you want to use for device swap in the field, "Select a section." Currently used sections (containing other swap areas or file systems) do not appear in the selection field.
7. Use the default priority, unless you want to assign a different priority (see "Guidelines for Assigning Swap Priority").
8. Indicate when you want the device swap enabled, "Now", "At Every System Boot", or both.
9. Activate [OK] to enable the device swap. When SAM has completed the task, you can see the new swap device on the swap object list.
10. At a later time, you can modify the priority and when you want it enabled, although once it is enabled, you must reboot to disable device swap.

Disabling Device Swap Using SAM

You can disable device swap as follows:

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate the Open control button.
3. Highlight `Swap` and activate the Open control button.
4. Highlight the swap device that you want to disable.
5. From "Actions" on the menu bar, choose `Modify Swap`.
6. In the "When to Enable" field, deactivate the `At Every System Boot` control button.

Note that the change does not take place until the next system boot, at which time this swap device is not enabled (that is, it will not be present in the `/etc/checklist` file).

You can re-enable the swap device by repeating steps 1 - 5 above and then activate the `Now` control button.

Adding Device Swap with the /etc/swapon Command

You can add secondary device swap in logical volumes or disk sections with the *swapon*(1M) command.

When you're using `swapon` to add device swap, use the following outline; details and examples follow.

1. Determine the block special device file name for the logical volume or section you want to use as a swap device. Make sure you have followed the guidelines for selecting the particular device for swapping.
2. Determine the priority you want to assign the secondary swap device. The system uses higher priority swap devices before it uses lower priority devices. If you don't assign a priority, the swap area will receive the default priority of 1.
3. Enable your secondary swap device using the `/etc/swapon` command. This makes your swap device immediately available for use.
4. Modify the `/etc/checklist` file by adding an entry for your secondary swap device if you want it enabled each time you boot your system.

Determining the Block Device File for the Chosen Section

For example, if you want to select section 10 on the disk assigned logical unit number (LU) 3, you would specify the block device file `/dev/dsk/c3d0s10`, where `dsk` specifies the directory for block special files associated with disks, `c3` specifies the LU 3, and `s10` specifies section 10.

Determining the Block Device File for a Logical Volume

Provide the full path name for the device file for the logical volume. For example, `/dev/vg3/lvol4` for logical volume `lvol4` in volume group `/dev/vg3`.

Note You need to first create the logical volume. If you cannot use SAM, refer to "Creating Logical Volumes" in Chapter 8, "Managing Logical Volumes." Also, refer to the section, "Setting Up Logical Volumes for Swap."

Setting the Priority for the Secondary Swap Device

You can assign priorities for swap devices (and file system swap areas, too; see "Specifying a Priority for File System Swap"). The range for priority is from 0 to 10, 0 having the highest priority. The system uses swap spaces with the highest priorities (that is, lowest priority number) first.

For example, you can assign the highest priorities to swap sections on disks that transfer data the fastest, and assign the lower priorities to swap sections on the slower disks.

Example: Enabling Device Swap in a Disk Section

After you have chosen a disk section for swap, determined its block device file name, and determined its priority, you can enable it using the `/etc/swapon` command.

For example, suppose you select section 10 of a disk that has been assigned lu 3. You want to assign it priority 1. Type the command as follows:

```
/etc/swapon /dev/dsk/c3d0s10 -p 1
```

The priority is assigned with the `-p` option.

Example: Enabling Device Swap in a Logical Volume

Suppose you want to use `/dev/vg3/lvol4` for swap; enter:

```
/etc/swapon /dev/vg3/lvol4
```

Once you have set up swap in a logical volume, you can increase or decrease the size of the logical volume and thus change available device swap space. SAM can help you do this, or you use either the *lvextend*(1M) or the *lvreduce*(1M) command. See Chapter 8, "Managing Logical Volumes" for discussion of these commands.

Verify that Device Swap is Enabled; the swapinfo Command

Type the command:

```
/etc/swapinfo
```

The listing will verify whether you have successfully added your device swap.

Enabling Your Swap Device Each Time You Boot the System

The device swap area will be enabled each time you boot the system if you have included an entry for it in the `/etc/checklist` file (`checklist(4)`). For example, for the device swap you enabled in the example above, add a line in the `/etc/checklist` as follows:

```
/dev/dsk/c3d0s10 /swap swap pri=1 0 0 #secondary swap
```

7

See a more detailed discussion of swap area entries in `/etc/checklist` in the section, "Enabling Swap Areas on Boot."

The system initialization file `/etc/rc` normally contains the command `/etc/swapon -a` so that all swap areas (both device swap and file system swap) listed in the `/etc/checklist` file become enabled when the system boots.

Removing Device Swap Using HP-UX Commands

You might want to remove device swap you have set up, perhaps to use a larger section or a smaller section instead. It's also possible you have two or more sections devoted to device swap and want to use one of them for a file system.

To remove any device swap you have set up:

1. Edit the `/etc/checklist` file to remove the entry for the specific device swap area you want to remove. Skip this step if you did not use SAM to add device swap or never added an entry in `/etc/checklist` for this device swap area.
2. Reboot your system by running `shutdown -r`; before issuing this command, make sure no users have either files open or processes running which could be corrupted by the shutdown.

Adding, Removing File System Swap

When no devices are available to use for swapping, you can use the vacant space within file systems for swapping. When you enable a file system for swap, the operating system can allocate the file system's free space when it needs it. However, once the operating system allocates a file system's space for swap, it does not relinquish it back to the file system.

Several file systems can be used for file system swap. The tunable system parameter `nswapfs` determines the maximum number of file systems you can enable for swap.

You can add file system swap by using either SAM or the `/etc/swapon` command. In either case, file system swap is enabled dynamically, that is, while the system is running without having to be previously configured into the system.

Note You cannot remove file system swap using SAM. You will need to use HP-UX commands described in "Removing File System Swap Added by SAM" and "Removing File System Swap Using HP-UX Commands."

Adding File System Swap Using SAM

1. Run SAM.
2. Highlight `Disk and File Systems->` and activate [Open].
3. Highlight `Swap` and activate [Open].
4. From "Actions" on the menu bar, choose `Add File System Swap...`.
5. Highlight the file system you want to use for swap from the list of file systems that appear under "Select a file system:".
6. Enter values in the fields for "Minimum Swap (Kbytes)" and "Maximum Swap (Kbytes):" These values specify, respectively:

   a. the minimum amount of the file system space you are designating as usable only for swap. This amount of space within the file system will not be used for file storage.
   b. the maximum amount of file system space you want used for swap. This value lets you keep a specific amount of file system space for file system only. This is important value to set because once the operating system allocates file system swap, it does not return it for file system use.

   Note, you must enter values in these fields. SAM will convert the entered value to a value in terms of the file system block size, usually four or eight kilobytes (Kbytes).

   However, disk space for swap is a function of the product of *swchunks* (a tunable system parameter, 2048 by default) and the value of `DEV_BSIZE` (defined in `/etc/disktab`, typically 1024 bytes). A typical swap space chunk is 2 megabytes (2048 kilobytes). If you specify 1000 kilobytes as a minimum, for example, the actual minimum chunk of swap space allocated would be 2048 kilobytes.

7. Designate the priority and when you want the swap enabled.
8. Activate [OK] to enable the file system swap.

Changing File System Swap Values Using SAM

The SAM `help` screens provide instructions and suggestions for changing the values currently set for file system swap. Once file system swap is enabled, the `minimum` value cannot be changed; the `maximum` swap value can be increased (but not decreased).

Removing File System Swap You Added With SAM

SAM cannot remove file system swap. You can remove file system swap by editing the file `/etc/checklist` and removing the entry that SAM created for the file system swap. The file system swap stays configured until the next time you reboot.

Adding File System Swap Using /etc/swapon

If you cannot use SAM to add file system swap, use the `/etc/swapon` command.

The following is a summary of the steps to enable file system swap on your system; details and examples are in the following sections.

1. Choose a file system for swap space use.
2. Determine the mount point directory (or the root directory) of the file system.
3. Choose values for the parameters of the `/etc/swapon` command. The parameters, all of which are optional, allow you to customize how your file system swap will work. See Table 7-1 for a description of how the parameters work.
4. Run `/etc/swapon` with desired options.
5. Add your file system swap to the `/etc/checklist` file if you want the new file system swap enabled on boot.

Choosing a File System for Swap

Follow the guidelines in the earlier section, "Selecting File System Swap Areas." The file system you designate for swap space use must be mounted. You can designate the file system you choose to use for swap by specifying the mount point directory of that file system.

Determine Mount Point Directory of the File System

Specify the absolute path name of the mount point directory for the swap file system in the `/etc/swapon` command line.

Choosing Values for /etc/swapon Parameters

The following table discusses the optional parameters you might use for the `/etc/swapon` command. An example, which makes use of these parameters, is in the following section, "Example, Setting Up File System Swap with /etc/swapon."

Table 7-1. /etc/swapon Parameters

Option	What it Does	How you use it
-m	Specifying an Initial Amount of Swap Space	You can use the `-m` option to specify the number of file system blocks to be initially taken for swap. File system blocks allocated for swap with `-m` option are never returned to the file system. The system allocates swap space in the file system when it requires it and frees it up when it no longer needs it. So, it may be unnecessary to use the `-m` option unless the file system is heavily used. The default amount of initial swap space is 0.
-l	Specifying a Limit to File System's Swap Space.	You can specify a limit to how much of the file system space you want to use for file system swap. The default limit is 0, that is, there is no limit to the amount of file system space that the system can use for swap. Use the `-l` option to specify an amount of file system blocks for a limit.
-r	Specifying Reserve Space for the File System.	You can specify that part of the swap file system be reserved exclusively for the file system. The default value is 0, meaning no space is reserved for file system. Use the `-r` option to specify a number of file system blocks to reserve for the file system.
-p	Specifying a Priority for File System Swap.	The system allocates swap space on devices or in file systems with the highest priority (that is, lowest priority number) first. The default priority value is 1. Use the `-p` option to specify another priority between 0 and 10.

Example, Setting Up File System Swap with /etc/swapon

Let's assume you want to set up swap in the `/extra` file system. You have run the `bdf` command and determined that `/extra`, a file system which is not used often, has ample space available for swap use. Because you also want to have this file system swap area available for use each time you boot, you must add an entry in the `/etc/checklist` file. You would do this as follows:

1. Evaluate the values you want to specify for the parameters of the `/etc/swapon` command. For this example, we will make the following assumptions:

Note The `swapon` parameters for file system swap requires values for file system blocks. The size of a file system block is determined when the file system is created, and is variable.

- You are not concerned that file system space will be unavailable initially for swap because the file system has more than enough space for both the file system and swapping. Therefore, you do not need to specify that space be initially enabled for swap. You will let the system take file system space as it needs it.
- You do want to limit the amount of file system space that can be used by the system to 5,000 file system blocks.
- You want to reserve 10,000 file system blocks for the file system's exclusive use.
- You want the file system swap to be accessed after a device swap area that you have previously set up with priority 0, so you decide you will assign the new file system swap priority 1.

Note Typically, file systems use block sizes of 4096 or 8192 bytes. If you are not sure of the block size used by the file system you want to set up swapping to, you can use the command:

```
dumpfs file_system | grep bsize
```

In the listing, look at the value to the right of "bsize" to see the current block size used by the file system. The value is in bytes.

Also remember that the operating system will allocate the swap area in chunks, typically 2 megabytes in size (see "Swap Space Defaults"). For reference, a typical 2-megabyte chunk of swap space is 512 4096-byte file system blocks.

2. To activate swapping to this file system immediately, run the `/etc/swapon` command using the parameter values you have chosen:

```
/etc/swapon -l 5000 -r 10000 -p 1 /extra
```

3. To verify that you have enabled your new file system, run the command *swapinfo*(1M). You should see a listing similar to the following after entering the command; note the line that begins "fs".

```
swapinfo

          Kb      Kb      Kb   PCT  START/      Kb
TYPE   AVAIL    USED    FREE  USED   LIMIT RESERVE  PRI  NAME
.
dev    48560    1888   46672    4%       -       -    0  /dev/dsk/c0d0s15
fs      6144       0    6144    0%    6144       0    1  /sizzle
.
.
```

4. Add an entry for file system swap in the `/etc/checklist` file if you want your file system swap to be enabled when you boot your system. The command `swapon -a` is contained in the `/etc/rc` file and causes all swap areas listed in `/etc/checklist` to be enabled. (Note that because the file system must be mounted before file system swap can be enabled, the file system itself must also have an entry in `/etc/checklist`.)

   An entry in `/etc/checklist` for the file system swap area in the previous example will look like:

   ```
   default /extra swapfs lim=5000,res=10000,pri=1 0 0
   ```

   A detailed discussion of file system swap area entries in `/etc/checklist` is located in the later section, "Making Entries for File System Swap in the Checklist File."

Removing File System Swap Using HP-UX Commands

To remove a file system swap area you have set up:

1. Edit the `/etc/checklist` file to remove the entry for the specific file system swap area you want to remove. Skip this step if you did not use SAM to add file system swap or never added a `swapfs` entry for this file system in `/etc/checklist`.
2. Reboot your system by running `shutdown -r`; before issuing this command, make sure no users have either files open or processes running which could be corrupted by the shutdown.

Enabling Swap Areas on Boot Using the Checklist File

You can enable swap areas on boot by including entries for them in the `/etc/checklist` file. See the entry for *checklist(4)* in the *HP-UX Reference Manual*.

Each of the earlier examples for adding swap by using HP-UX commands include adding entries to the `/etc/checklist` file. And, when you use SAM to add device or file system swap and want the swap area enabled at boot, SAM makes entries in `/etc/checklist`. The following sections discuss the `/etc/checklist` file in greater detail.

Making Entries for Device Swap in the Checklist File

Look at the following example entry in an `/etc/checklist` file:

```
/dev/dsk/c0d0s1   /swap    swap  pri=0 0 0 # swap area
```

Entries for device swap in `/etc/checklist` have seven fields separated by blank space. In each field, you need to put either an entry or a placeholder. The following describes making entries for each field.

- First field: Enter the *block device file* corresponding to the disk swap section.
- Second field: For device swap, enter a placeholder. For example, you could enter the mount point *directory* for the disk section corresponding to the special device file.
- Third field: Enter the *type* of the entry, which for device swap is `swap`.
- Fourth field: Enter *options* here. For device swap, the only possible option is priority. Priority is indicated by `pri=`*n* where *n* is between 0 and 10. Otherwise, use "defaults" as a placeholder. The default for priority is 1.
- Fifth and sixth fields: Enter the placeholder `0` in each of these fields, because they are not necessary for device swap.
- Seventh field: you may enter a comment preceded by the `#` character.

Making Entries for File System Swap in the Checklist File

The following example shows an entry in `/etc/checklist` for file system swap.

```
default   /tmp/swap swapfs lim=1500,res=2000,pri=1 0 0 #file swap
```

Entries for file system swap in `/etc/checklist` have seven fields separated by blank space. The following describes making entries for each field.

1. First field: For file system swap entries, enter a placeholder, such as "default." (For all other type entries, the first field contains a special device file name, which does not apply for file system swap.)
2. Second field: Enter the name of the mount point *directory* of the file system to be used for swapping. Specify this directory using its absolute path name.
3. Third field: Specify the type of entry, which for file system swap is `swapfs`.
4. Fourth field: Specify the *options* for the file system swap. These options are the same as the parameters used when enabling file system swap with the `/etc/swapon` command. (See the earlier section, "Choosing Values for /etc/swapon Parameters".)

   Briefly, those options are:

   a. min=*n*, where *n* is the initial number of file system blocks for swap
   b. lim=*n*, where *n* is the limit to the number of file system blocks for swap
   c. res=*n*, where *n* is the number of file system blocks reserved for the file system
   d. pri=*n*, where *n*, ranging from 0 to 10 (0 is high priority), indicates the swapping priority

   When you specify the options, leave no spaces anywhere in the entry for options and separate the options with commas.
5. Fifth and sixth fields: Enter the placeholder `0` in each of these fields, as these fields are not used for file system swap entries.
6. Seventh field: You may enter a comment preceded by the `#` character.

Adding a Disk

When you add your disk to the computer, refer to the *Installing Peripherals* manual.

The manual *Installing Peripherals* describes making the physical connections, determining the physical address of your disk, and adding the disk to your systems I/O configuration.

When you reboot your system after connecting and turning on your disk, the system automatically configures the disk into its I/O configuration, assigns the disk a logical unit number, and makes all its necessary device files. Be sure to remember the disk's model number because you will need to refer to it later.

Run mediainit (if necessary)

This step might not be necessary in the case of new Hewlett-Packard disks, which have been initialized and formatted, or disks that you have been using already without difficulty.

However, you can use the `mediainit` command to format a disk if you have doubts about its integrity or condition. It takes a while (possibly an hour, maybe more) to initialize a disk.

Refer to `mediainit(1)` in the *HP-UX Reference Manual*. Before initializing a disk, refer to the manual supplied with the particular disk for any interleaving instructions.

7

When issuing the `mediainit` command to initialize a disk, use the device file that specifies in the minor number that the diagnostic bit is set. These files reside in the `/dev/diag/dsk` directory. For example, if you want to initialize the disk that has been assigned logical unit number 3, you would specify the device file `/dev/diag/dsk/c3d0`.

Caution *Do not* initialize a disk that contains valuable data that you have not backed up. Initializing will destroy all data on a disk.

Using a New Disk for LVM

Before you can use the new disk you add as an LVM disk, you must add the disk to an LVM volume group. That is, you must include the disk in a group of disks that compose a volume group.

Use SAM to add the disk to a volume group; Chapter 8, "Managing Logical Volumes," shows how. Or, you can use HP-UX commands. See chapter 8, "Managing Logical Volumes," for the procedure to add a disk to a volume group using the *pvcreate*(1M) and *vgextend*(1M) commands.

Once the disk is part of a volume group, you can create a logical volume, and configure the logical volume for device swap. Refer to "Creating a Logical Volume for Device Swap" earlier in this chapter.

Setting Up Dumps Areas

You can designate disk space to be used by the system to write, or "dump," an image of the core memory after a system crash. This "core dump" is useful in troubleshooting and restoring the system to working order.

You need to configure the dumps area in the kernel so that the kernel can immediately use the dumps area during a crash, and so that the `savecore` command can locate the core dump during boot after a crash.

How Much Disk Space to Use For Dumps?

The amount of disk space available for core dumps needs to accommodate the amount of core memory.

Usually, the primary swap area can also serve as a dumps area. In fact, by default, primary swap will be a dumps area when no dumps area is specifically designated.

However, because the physical memory on the system often exceeds the space available in the primary swap area, you can configure additional disk space for the full core memory image. Core memory can be as great as 2 Gbytes.

Primary swap does not need to serve as a dumps area. Using primary swap as a dumps area conserves disk space, but recovery of the core dump could be faster if primary swap area is not used. This is a decision to make, weighing disk usage and recovery time.

7

Multiple dumps devices can be configured on the system. You can configure as a dumps device a logical volume or disk section that is used for also swap.

Finding Physical Memory

If you do not know the amount of physical memory on your system, you can enter:

```
/etc/dmesg | grep real
```

The output will resemble:

```
real mem = 50331648
```

Divide the value in bytes by (1024 * 1024) to obtain the value in Mbytes. The amount displayed in the example above equals 48 Mbytes, and that's how much space core dumps require.

Note The message buffer is always available after bootup, but it might not be available if the system has not been rebooted for awhile.

Configuring Dumps with SAM

You can use SAM to add or remove dumps devices.

You can review the preceding sections in this chapter to help you plan how to set up dumps areas and then refer to the section "Reconfiguring the Kernel to Modify Special Devices" in Chapter 2, "Constructing an HP-UX System".

Configuring Disk Sections as Dumps Devices

The kernel configuration file, `/etc/conf/gen/S800`, contains the specification for dumps devices in disk sections. (The next section describes using logical volumes as dumps devices.)

In the configuration file, the keyword "`dumps`" precedes a specified dumps device. For example, when using primary swap as a dumps device,

```
dumps on default;
```

Or, when using another device as a dumps device,

```
dumps on cio_ca0.hpfl0.disc2 at 4.4.2 section 15;
```

If you use primary swap and another section as dumps devices, the dumps statement resembles:

```
.
dumps on default
         cio_ca0.hpfl0.disc2 at 4.4.2 section 15;
.
```

Note the following rules about specifying disk section dumps devices:

- Primary swap used as a dump device must be the first dumps device specified.
- When primary swap is not used as a dumps device, then you must explicitly specify the location; you cannot specify "`on default`."
- If you do not specify primary swap, the line, "`dumps on default`," will be ignored, resulting in no configured dumps.
- You can specify any secondary swap device to be a dumps device.

If you have set up swap devices for dumps, during the system startup following a system core dump, `savecore` runs in the background after reading the dump on primary swap. Then, any secondary swap devices configured for dumps are not enabled for swap until `savecore` has read the dumped data.

Reconfigure the Kernel

After changing the dumps statements in the `S800` file, either to add or delete dumps devices, you will have to reconfigure the kernel and reboot to implement

the changes. Use the procedure described in Chapter 2, "Constructing an HP-UX System".

Configuring Logical Volumes to be Dump Areas

Note You must have LVM configured on your system to use logical volumes. See Chapter 8, "Managing Logical Volumes" for details about using and configuring LVM.

Dumps logical volumes can exist only within the root volume group. That is, dumps logical volumes can exist only in volume groups where `root` is in a logical volume.

A Boot Disk Reserved Area (BDRA) exists on each bootable disk within the root volume group. The BDRA maintains the information that the kernel requires about those logical volumes that contain `root`, primary swap, and dumps.

If you intend to make a logical volume a dumps device, then you must use the *lvlnboot*(1M) to update the BDRA. When discontinuing the use of a logical volume as a dumps device, you use the *lvrmboot*(1M) command.

Updating the BDRA to Specify Logical Volume Dumps Areas

Suppose, for example, the logical volume `/dev/vg00/lvol2` is the primary swap area; to make it a dumps area, enter:

```
lvlnboot -d lvol2 /dev/vg00
```

The `-d` (dumps) flag with `lvlnboot` updates the BDRA.

Specify other logical volumes as dumps areas in the same way. You can use any secondary swap logical volume as a dumps area, provided the swap area is in the root logical volume.

If you have set up swap devices for dumps, during the system startup following a system core dump, `savecore` runs in the background after reading the dump on primary swap. Then, any secondary swap devices configured for dumps are not enabled for swap until `savecore` has read the dumped data.

Updating the BDRA to Remove a Logical Volume as a Dumps Device

If you have logical volume `/dev/vg00/lvol3` currently configured as a dumps device, you can discontinue using it as a dumps device by entering:

```
lvrmboot -d lvol3 /dev/vg00
```

Reboot to Implement Changes to BDRA

Changes to the BDRA take effect the next time you reboot the system.

7

Specifying Logical Volumes Dumps in the S800 File

Because the BDRA on the boot disk contains the dumps area information, the dumps specification in the S800 file is:

```
root on lvol;
swap on lvol;
dumps on default;
```

Specifying Both Logical Volume and Disk Sections as Dumps Devices

The key rule in specifying dumps devices in the S800 file is that *if primary swap is to be a dumps device, it must be the first dumps device specified.*

Examples:

- Primary swap is a disk section and there are multiple dumps devices, some of which are logical volumes.

```
root on lvol;
swap on cio_ca0.hpfl0.disc2 at 8.0.0 section 1;
dumps on default
         lvol;
```

- Primary swap is a logical volume and there are multiple dumps devices, some on disk sections.

```
root on lvol:
swap on lvol;
dumps on default
         cio_ca0.hpfl0.disc2 at 8.0.2 section 1;
```

Reconfigure the Kernel

After making any changes to the dumps statements in the S800 file, whether you add or delete dumps device statements, you will have to reconfigure the kernel and reboot to implement the changes. Use the procedure described in Chapter 2, "Constructing an HP-UX System".

8

Managing Logical Volumes

The Logical Volume Manager (LVM) subsystem enables a system administrator to manage disk space in terms of **logical volumes** rather than (or in addition to) fixed-sized disk sections. Like disk sections, logical volumes can hold file systems, swap areas, and raw data. LVM provides several advantages:

- Logical Volumes can span disks.

  Logical volumes can range in size from one megabyte (smaller than a disk partition) to larger than a single disk. As a result, the file systems, swap areas, or areas of raw disk space contained within logical volumes can be very small or be as large as four gigabytes, spanning the space of more than one disk.

- Logical Volumes are extensible.

  The sizes of logical volumes can be changed; they can be extended or reduced when needs change. Likewise, file systems or other forms of disk usage residing in logical volumes are also extensible, not being limited to the sizes of traditional disk sections.

- Logical Volumes allow soft partitioning.

  You can create a logical volume of any size on any disk. Within a single group of disks (a volume group), you can create up to 255 logical volumes.

- Logical Volumes provide high data availability through **mirroring**.

  Through the use of the MirrorDisk/UX software, LVM can provide up to three copies of data, allowing one copy of data to be available in the event one or two copies are lost. LVM transparently ensures that the data on each mirrored copy is kept in a consistent state. By mirroring logical volumes, you can also back up one copy of data while other copies stay online.

  Mirroring the `root` file system, the primary swap area, and any other mission-critical data provides increased system availability.

Note You must use the MirrorDisk/UX product, part number B2491A, for mirroring.

- LVM provides for the recovery from media errors.

  LVM transparently uses hardware and software relocation mechanisms to recover from media errors, or **bad blocks** on a disk.

What This Chapter Covers

This chapter describes how to manage logical volumes by using either SAM or HP-UX commands. In addition to describing how to create, modify, and monitor logical volumes, this chapter recommends methods and guidelines for taking advantage of LVM's features while maintaining acceptable system performance.

With LVM, the standard HP-UX commands, such as `newfs`, `mkfs`, `swapon`, or `fsck`, use the same syntax as they do with file systems or swap areas created on traditional disk partitions. If the command syntax requires the disk model, you need to supply the model on which the logical volume exists. In place of the device file for the section, you need to supply the device file for the logical volume.

For specific information about managing file systems, refer to Chapter 6, "Managing the File System."

For information about managing swap space, refer to Chapter 7, "Managing Swap Space."

Terms Used in this Chapter

The following terms appear frequently in this chapter.

activation The process by which a volume group, its disks, or its logical volumes are made available for use.

allocation policy Allocation refers to how data is distributed on a disk. Allocation policy can be **contiguous**, that is, with no allowable gaps in the set of physical extents on a physical volume, or non-contiguous, with gaps allowed in the set of physical extents.

bad block relocation An LVM feature, employable at the system administrator's option, that provides for the transparent detection of bad disk sectors and the relocation of data from the bad to good disk sectors. The LVM bad block relocation feature can work in conjunction with disk controllers that perform bad sector relocation at the hardware level. Note that the `root` logical volume does not support software bad block relocation.

block A unit of space for data on a disk, typically having a size of 1024-bytes.

device file Used to identify and communicate with a device (such as a disk) or pseudo device (such as a logical volume). There are block device files (used for transmitting data in multiple-byte blocks) and character device files (used for transmitting data byte-by-byte. Device files are typically stored in the `/dev` directory.

By default, the full path name of block device files follows a convention for logical volumes: `/dev/vg`*nn*`/lvol`*n*.

By default, the full path name of character device files follows a convention for logical volumes: `/dev/vg`*nn*`/rlvol`*n*.

extent See **physical extent**, **logical extent.**

8

file system The organization of files on storage devices. The term file system can refer either to the entire file system tree or to a subsection of that file system, contained within a disk section or a logical volume, that can be mounted or unmounted from the tree.

group file The character device file for a volume group.

logical extent A contiguous set of virtual blocks of data contained within a logical volume. A logical extent maps to one (more, if LVM mirroring is used) physical extent. Within a volume group, logical extents are of a fixed size.

logical volume A *logical* (rather that physical) construction that is a map of data stored on physical volumes (disks). A logical volume can be conceptualized as a storage device of flexible size. The data in a logical volume can map to one or more physical volumes.

physical extent A specific, contiguous set of blocks within a region of a physical volume (disk) where data resides. A physical extent can range, in megabytes, by a power of two from one megabyte to 256 megabytes. Within a volume group, physical extents are of a fixed size.

physical volume A disk on which LVM has built data structures, making it an LVM disk. The physical device where data is stored. A physical volume is an entire disk, and has, therefore the device files assigned to section two of the disk. It is not possible to use only part of a disk as an LVM disk. The character device file, `/dev/rdsk/c`*n*`d0s2`, and the block device file, `/dev/dsk/c`*n*`d0s2`, are the appropriate device files for physical volumes.

quorum The number of physical volumes required to be available in order for a volume group to be activated or manipulated after one or more of a volume group's physical volumes becomes unavailable (because of a disk failure or because it contains stale data after a system crash). A volume group without a quorum of physical volumes remains unavailable.

raw disk I/O The reading and writing of data directly to and from a disk section or a logical volume via the character (raw) device file, independent of file system or swap management.

swap space Space within a disk section, a logical volume, or a file system that is designated to temporarily store the process image. See the chapter, "Managing Swap Space."

volume group A set of physical volumes (disks) whose space can be combined and logically divided up into logical volumes. Only logical volumes and physical volumes that are a part of a volume group can map together; a physical volume can belong to only one volume group.

Terms Used in this Chapter (Mirroring Only)

The following terms appear frequently in this chapter, but apply to the MirrorDisk/UX product, part number B2491A.

activation The process by which a volume group, its disks, or its logical volumes are made available for use.

allocation policy Allocation refers to how mirrored copies of data are distributed to disks (physical volumes). Allocation policy can be **strict**; that is, copies cannot reside on the same disk. With non-strict allocation, mirrored copies can reside on the same disk.

Allocation policy can be **contiguous**, that is, with no allowable gaps in the set of physical extents on a physical volume, or non-contiguous, with gaps allowed in the set of physical extents.

logical extent A set of virtual blocks of data contained within a logical volume. A logical extent maps to one (more, if MirrorDisk/UX is used) physical extent. In the case of MirrorDisk/UX, a logical extent maps to one, two, or three physical extents, depending on whether the stored data is unmirrored, mirrored once, or mirrored twice.

mirrored data The copies of data stored in physical extents that map to a unique logical extent. Data can be singly mirrored (two physical extents containing replicated data are allocated for each logical extent) or doubly mirrored (three physical extents contain replicated data are allocated for each logical extent).

mirror write cache (MWC) A MirrorDisk/UX mechanism, whose use is optional, that tracks outstanding mirror write requests and provides a basis for the resynchronization of data blocks after a system crash or power failure.

physical volume group Physical volumes within volume groups can be organized, usually on the basis of the I/O channel or interface adapter to which they are connected, into subgroups known as physical volume groups for the purpose of achieving higher availability of mirrored data. With strict allocation with respect to physical volume groups, mirror copies of a logical volumes cannot reside in the same physical volume group.

scheduling policy Determines whether data is written to mirrors in *parallel* or *sequential* manner.

strict allocation Mirrored copies of data cannot reside on the same disk; strict allocation is only possible when more than one physical volume is available in the volume group. Allocation can be set up to be strict with respect to groupings of disks; physical volumes are usually based on which I/O interface card or I/O bus they are connected to.

synchronization The process of updating "stale" copies of mirrored physical extents.

Are You Running LVM Now?

Type the command:

```
/etc/vgdisplay
```

If you see:

```
/etc/vgdisplay: Command not found.
```

then you don't have LVM on your system. Go to the next section, "Making Logical Volume Manager Available for Use," for information about getting LVM configured on your system.

No response, or any number of other different responses to the command, `/etc/vgdisplay`, means LVM *is* on your system.

A likely response might resemble:

```
--- Volume groups ---
VG Name                 /dev/vg00
VG Status               available
.
.
.
```

8

Making Logical Volume Manager Available for Use

Look at the following likely possibilities and do what applies in your situation.

If Hewlett-Packard Has Pre-Installed Release 9.0 on Your System

You have LVM available for use if you are using a system on which Hewlett-Packard has pre-installed HP-UX Release 9.0.

The root volume group, which includes a boot area, a root logical volume, and a primary swap logical volume, exists on your system. You can go ahead and use the available space within the volume group, or you can add disks to the volume group if you need more space.

If You Have Migrated Your Root Disk From Release 8.0 to Release 9.0

If you have migrated `/` and other system-related file systems, such as `/usr`, from Release 8.0 to Release 9.0 and Logical Volumes, then you have LVM running on your system. You have a root volume group, which includes a boot area, a root logical volume, and a primary swap logical volume.

If you need more space, you can use any existing space left in your root volume group, or you can add *new* disks to the root volume group.

Any other disks on the system, however, that still use traditional disk sections can become LVM disks only if you migrate them. You can use SAM to help you do this, or use HP-UX commands. See the section in this chapter, "Migrating Non-Root Disks From Traditional Sections to Logical Volumes".

If You Have Updated to Release 9.0 Without Migrating the Root File System

If you have run the */etc/update*(1M) program to get your system running Release 9.0 and have selected the LVM file set, then you have LVM on your system, but no disks on your system are LVM disks.

To use LVM, you can:

- Add new disks to your system and use them as LVM disks for file systems, swap, or raw data. SAM can help you add new disks as LVM disks.

  If you are planning not to use SAM, go to the section "Creating Volume Groups" if you are familiar with LVM, or go to the section "LVM-An Overview" to become familiar with LVM.

- Migrate existing file systems and raw data from their current location in sections to logical volumes on a disk-by-disk basis. See the section in this chapter, "Migrating Non-Root Disks From Traditional Sections to Logical Volumes".

If You Still Have Release 8.0 Running on Your System

You have two choices, if you want to use LVM.

- Migrate the root disk to LVM.

  This involves backing up your current system (all files and any raw data), installing Release 9.0 (creating a root volume group in the process) on your root disk(s), and restoring the data (as needed) to the logical volumes in the root volume group. (Refer to chapter 3 of the *Installing and Updating HP-UX*, part number B3108-90006, for complete discussion of migrating the root disk.)

  After you have migrated `/` and `/usr`, you can create logical volumes in the root volume group for any file systems or raw data that formerly used the disks now in the root volume group. You can add *new* disks to the root volume group if you need more space.

  You must migrate the file systems still using traditional disk sections. See the section in this chapter, "Migrating Non-Root Disks From Traditional Sections to Logical Volumes".

- Run */etc/update*(1M) and *not* create a root volume group.

  Run the `update` program to update your system to Release 9.0. You can add the LVM subsystem without making the root disk a logical volume disk. This allows you to put logical volumes on all *other* disks after completing the update to Release 9.0.

  Migrate existing file systems and raw data from their current location in sections to logical volumes on a disk-by-disk basis. See the section in this chapter, "Migrating Non-Root Disks From Traditional Sections to Logical Volumes".

8

LVM—An Overview

Refer also to the glossary of LVM-related terms at the beginning of this chapter.

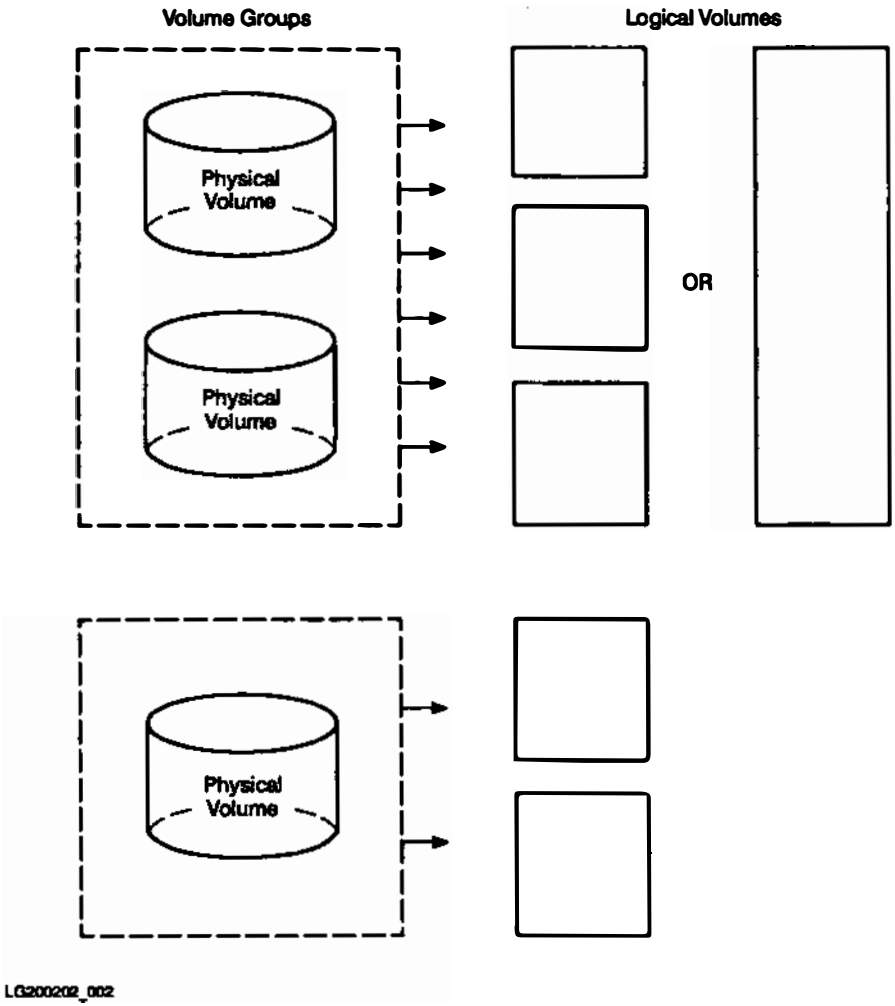
Note For a thorough discussion of LVM concepts, refer to the manual *How HP-UX Works*, part number B2355-90029. That manual describes the elements of LVM, describes how LVM works, and provides general guidelines for using LVM.

Overview of How Logical Volumes are Organized

LVM knows disks as **physical volumes**. You can combine physical volumes into **volume groups**. You can either create a new volume group that contains new disks (physical volumes), or you can add new disks (physical volumes) to an existing volume group. From one to 32 disks can belong to a volume group. Restrictions about grouping disks into volume groups are discussed in the section: "Organizing Your Disk Space into Logical Volumes."

Within a volume group, you can *logically* divide the total disk space into **logical volumes.** A volume group can contain several logical volumes of various sizes and configurations. A single logical volume can span disks and be larger than a disk.

Figure 8-1 shows how logical volumes can be organized within volume groups of physical volumes.



8

Figure 8-1. Organization of Logical Volumes Within Volume Groups

Displaying the LVM Organization on Your System

To explore the organization of logical volumes on your system, look at the output of the command:

```
vgdisplay -v
```

The list contains information about the volume groups, logical volumes, and physical volumes currently configured. Look at the following listing for a small system:

```
--- Volume groups ---
VG Name                 /dev/vgroot
VG Status               available
Max LV                  255
Cur LV                  4
Open LV                 4
Max PV                  16
Cur PV                  3
Act PV                  2
Max PE per PV           1016
VGDA                    4
PE Size (Mbytes)        4
Total PE                479
Alloc PE                352
Free PE                 249
Total PVG               0

   --- Logical volumes ---
   LV Name                 /dev/vgroot/LVroot
   LV Status               available/syncd
   LV Size (Mbytes)        92
   Current LE              23
   Allocated PE            46
   Used PV                 2

   LV Name                 /dev/vgroot/usr
   LV Status               available/syncd
   LV Size (Mbytes)        152
   Current LE              38
```

8

```
Allocated PE            76
Used PV                 1

LV Name                 /dev/vgroot/swap
LV Status               available/syncd
LV Size (Mbytes)        152
Current LE              38
Allocated PE            38
Used PV                 1

LV Name                 /dev/vgroot/scaff
LV Status               available/syncd
LV Size (Mbytes)        400
Current LE              100
Allocated PE            100
Used PV                 2

--- Physical volumes ---
PV Name                 /dev/dsk/c3d0s2
PV Status               available
Total PE                157
Free PE                 0

PV Name                 /dev/dsk/c5d0s2
PV Status               available
Total PE                322
Free PE                 127
```

In the example output shown above, you can see:

- One volume group: `/dev/vgroot`
- Four logical volumes: `/dev/vgroot/LVroot`, `/dev/vgroot/usr`, `/dev/vgroot/swap`, `/dev/vgroot/scaff`
- Two disks (physical volumes): `/dev/dsk/c3d0s2`, `/dev/dsk/c5d0s2`

The `vgdisplay -v` output also shows information about how large the logical volumes are and about how the data they contain is allocated to the disks in terms of **extents**. See the next section, "Physical Extents, Logical Extents."

Physical Extents, Logical Extents

LVM stores data on disks (physical volumes) within areas called **physical extents**, which are sets of contiguous disk blocks. At the time you form a volume group, the physical extent size is established. The extent size is the same, therefore, on all disks in a volume group. By default, each extent has a size of four megabytes.

Type:

```
vgdisplay
```

You will see a line resembling:

```
PE Size (Mbytes)        4
```

This means the size of each physical extent is 4 megabytes.

Logical volumes consist of **logical extents**, which map to the physical extents on the disks in the volume group. Therefore, within a volume group, physical extents and logical extents are of the same size, that is, have the same capacity. Except in the case of mirroring, within a volume group, one physical extent is allocated for each logical extent. In the case of mirrored logical volumes (see the next section, "Mirroring of Data"), there can be two or three sets of physical extents allocated for each set of logical extents.

Type:

```
vgdisplay -v
```

You will see a pair of lines:

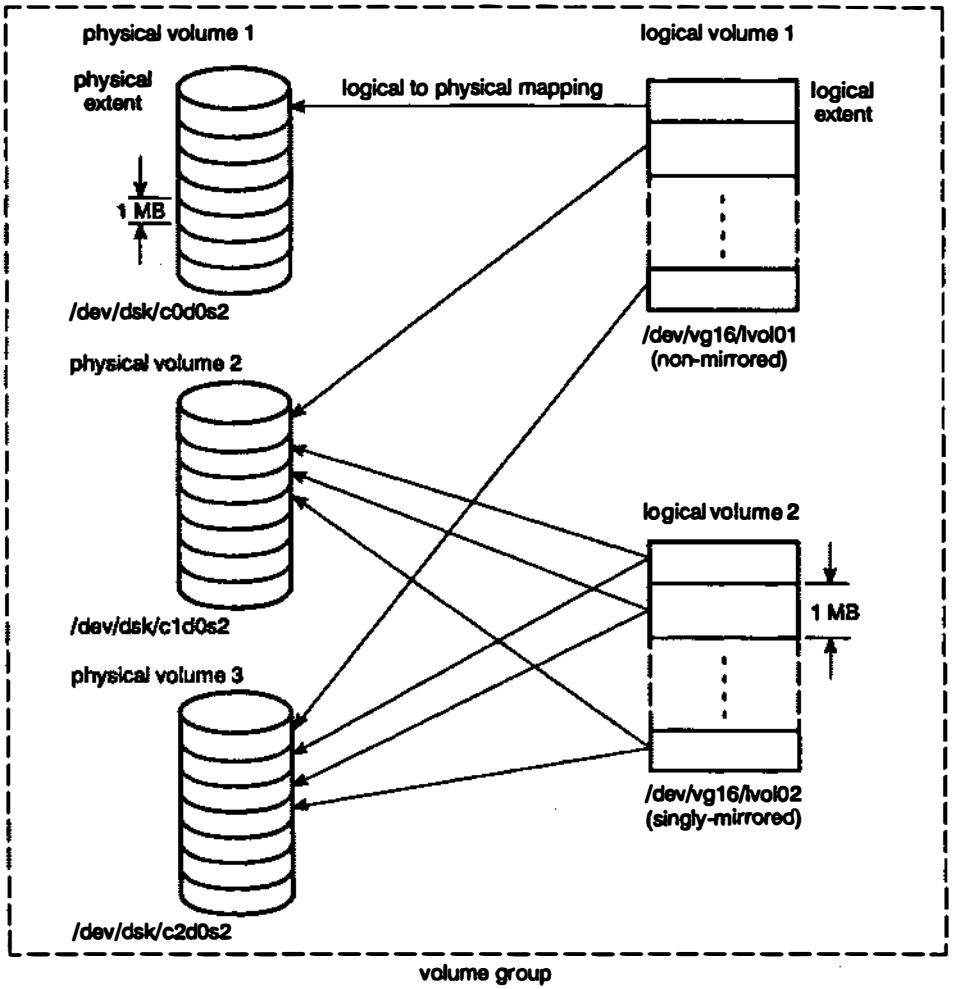
```
Current LE              23
Allocated PE            23
```

The LVM subsystem transparently handles the mapping of logical extents to physical extents, although, as a system administrator, you can control which disks are used by a logical volume, bypassing the default distribution method LVM uses. By default, LVM selects available physical extents from disks in the order in which you added physical volumes to the volume group.

The physical extents that map to the logical extents within a logical volume can be discontiguous on a physical volume and can, in fact, exist on different disks. Thus, it's possible that a file system created within one logical volume,

might exist on more than one disk. File system performance is best, however, when the physical extents used by a file system are contiguous on each disk they reside on.

Figure 8-2 shows the correspondence between physical extents and logical extents in a volume group.



LG200202_001

Figure 8-2. Physical Extents and Logical Extents

Mirroring of Data

The LVM mirroring product, MirrorDisk/UX, allows you to create mirror image copies of the data in a logical volume. When data within a logical volume is mirrored, the logical extents map to two or three identical physical extents. Refer to the section entitled "Managing Mirrors" for a detailed description of MirrorDisk/UX.

Listing What's In the Logical Volumes

Logical volumes can contain file systems, swap space, or raw data.

Listing File Systems in Logical Volumes

To list any mounted file systems in logical volumes, type:

```
bdf
```

The first column lists the logical volumes that contain mounted file systems and the last column displays where they're mounted.

```
Filesystem          kbytes    used   avail capacity Mounted on
/dev/vg00/lvol2       19643       9   17669     0%   /lptest
/dev/vg00/lvol4       39034       9   35121     0%   /temp3
.
.
```

If you are concurrently using logical volumes and disk sections for file systems, you will see them listed together.

For example,

```
Filesystem          kbytes    used   avail capacity Mounted on
/dev/dsk/c1d0s8     179559    4663  156940     3%   /mnt
/dev/vg2/lvol1        7675    6018     889    87%   /lpspmg
.
```

Listing Swap Areas in Logical Volumes

8

To list any swap areas contained in logical volumes, type:

```
swapinfo
.
   Kb      Kb      Kb   PCT  START/      Kb
TYPE   AVAIL    USED    FREE  USED   LIMIT RESERVE  PRI  NAME
dev    48560    4120   44440    8%       -       -    0 /dev/dsk/c0d0s15
dev    10240       0   10240    0%       -       -    1 /dev/vg00/lvol1
```

Listing Raw Data in Logical Volumes

No HP-UX command can explicitly list only those logical volumes that contain raw data; applications using the logical volumes to store raw data can possibly do that. Therefore, you might need to devise a means to chart logical volumes used for raw data. For example, when you create a logical volume to contain raw data, give the logical volume an easily recognizable name, such as `/dev/vg00/sales_data`, so that you can identify it and manage it later on.

LVM Naming Conventions

There are naming conventions that apply to physical volumes, volumes groups, and logical volumes. You need to refer to LVM objects by name when referring to them in SAM or with HP-UX commands or when viewing information about them.

Disk (Physical Volumes) Names

LVM disks (physical volumes) are identified by the device file names for their section 2: `/dev/dsk/c`*n*`d0s2`, `/dev/rdsk/c`*n*`d0s2` Note that each disk has a *block* device file and a *character* device file. Which name you use depends on what you are doing with the the disk.

When you create a physical volume, (that is, make a disk an LVM disk), you must use the raw device file for disk section two, which, for example, might be `/dev/rdsk/c3d0s2` if the disk were LU 3.

On the other hand, when you add the physical volume to a volume group, you would use the disk's *block* device file for section two, `/dev/dsk/c3d0s2`, for example.

(All special files for a disk are created automatically when it is physically added to the system.)

Volume Groups Names:

When you create a volume group, you assign the volume group name. You create a directory `vg`*nn* in the `/dev` directory. Typical volume group names would be `vg01`, `vgroot`, or `vg_sales`.

8

Logical Volume Names

Logical Volumes are identified by their device file names, which take the forms: `/dev/vg`*NN*`/lvol`*n* (the block device file form) and `/dev/vg`*NN*`/rlvol`*n* (the character device file form; note the "r").

By default, the number *n* starts at 1, and proceeds 2, 3, and so on in the order that logical volumes are created within each Volume Group.

When LVM creates a logical volume, it assigns it the symbolic name and creates both a block and character devices file for it. LVM places the device files for a logical volume in the appropriate Volume Group directory.

For example, the block name for the first logical volume in volume group `vg01` would have the full path name:

```
/dev/vg01/lvol1
```

You can assign a name to a logical volume when you create it. For example, if you create a logical volume to contain raw data for a sales database, you might want to name it:

```
/dev/vg01/sales_db_lv
```

After the logical volume in the above example has been created, it has two device files:

`/dev/vg01/sales_db_lv`	*(the block device file)*
`/dev/vg01/rsales_db_lv`	*(the character, or raw, device file; notice the "r")*

Physical Volume Group Names

Physical volume groups have no prescribed convention, although within a volume group, each physical volume group must have a unique name. You can name them PVG1, PVG2, and so on. For example, the volume group `/dev/vg02` might have the two physical volume groups: `/dev/vg02/pvg1` and `/dev/vg02/pvg2`.

Guidelines for Using Logical Volumes

Using LVM requires some planning. In the course of setting up or modifying logical volumes on your system, you will need to consider:

How will you use a logical volume? For a file system, for swap space, or for raw data storage?
How big do you want a logical volume to be?
What interface type do the disks require?
What models are your disks?
Is I/O performance extremely important?
Does your data require high availability?

The following sections will help you plan how to set up logical volumes.

Planning Logical Volumes for File Systems

File systems can exist in a logical volume just as they do in a disk section.

What Size Logical Volume Does a File System Require?

You can consider the space required by a file system as having three major components:

- Space for the file system's overhead, which includes all data structures required to maintain the file system. Typically, this is about 5 % of the total space required by a file system.
- Space for user data.
- A "minfree" space, which is part of the data space but reserved for use by the system administrator and to enhance performance; typically 10% of the data space.

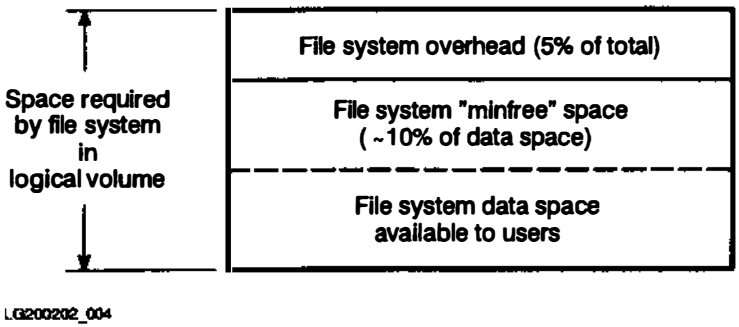


Figure 8-3. File System Space

To get a rough estimate of how big to make a logical volume for a file system, do the following:

1. Estimate how much disk space users will require for their data on an ongoing basis. Allow for some fluctuation and allow for some growth.
2. Increase that estimate by 10 % for the "minfree" area; this gives you an estimate of the total user data space.
3. Increase this estimate by yet another 5 % for file system overhead. This is the overall total space you estimate for the file system.
4. Divide this overall estimate by the logical extent size used in this logical volume. This gives you the size of the logical volume in logical extents.

These estimations are not extremely accurate, but you do not need an accurate value for how big to make a file system. Rather, you want to size your file system to be large enough for some useful time before having to increase its size.

Example, Estimating the Size of a Logical Volume for a File System. A group of users requires 60 megabytes space for file system data; this estimate allows for some upward fluctuation and some growth. You then add six megabytes for the "minfree" space and arrive at 66 megabytes. Then you add another four megabytes for file system overhead and arrive at a grand total estimate of 70 megabytes required by the file system.

Now, if you are creating the logical volume in a volume group that has an extent size of four megabytes, you will need a logical volume of 72 megabytes; LVM creates logical volumes in even multiples of the logical extent size.

Increasing the Size of a File System. You can increase the size of a file system by enlarging the logical volume it resides in and then using the *extendfs*(1M) command. This procedure is described later in the section "Extending a Logical Volume and Extending a File System."

Note

Decreasing the size of a file system involves several steps that include backing up the file system, creating a new file system, and restoring the file system data to the newly created file system.

Because increasing the size of a file system is much easier than reducing one, you might benefit by being conservative in estimating how large to create a file system.

Examples of creating and extending file systems in logical volumes are in this chapter and also in chapter 5, "Managing File Systems."

Guidelines for Spanning Disks With File Systems

Whenever possible, if you plan to have a file system span disks, have the logical volume span *identical* disk types.

Normally, by default, LVM will create logical volumes on available disks, not necessarily with regard for best performance. It's possible to have a file system span two disks with different characteristics, in which case the file system performance could possibly be impaired.

As a system administrator, you can exercise control over which disks (physical volumes) will contain the physical extents of a logical volume. You can do this by:

1. Creating a logical volume without specifying a size; by default, if you specify no size for a logical volume when you create it, no physical extents are allocated for the logical volume.
2. Extending the logical volume to specific physical volumes.

Specific examples for using the *lvcreate*(1M) and *lvextend*(1M) commands in sequence are shown later. You can also use SAM to create and extend logical volumes.

Guidelines for Creating Logical Volumes for Swap

If you plan multiple device swap areas, you will attain the best performance when the device swap areas are on different physical volumes (disks). This allows for the interleaving of I/O to the physical volumes when swapping occurs.

You can configure this swapping configuration by creating multiple logical volumes for swap, each logical volume on a separate disk.

You can use HP-UX commands to help you obtain this configuration; SAM does not allow you to create a logical volume on a specific disk.

Brief examples of using swap areas in logical volumes are in this chapter. Also see chapter 6, "Managing Swap Space."

Guidelines for Creating Logical Volumes for Raw Data Storage.

All that is necessary for a raw data logical volume (such as for a database partition) is to create the logical volume. You do need to specify how large to create the logical volume.

Typically, an application (such as a database) accesses the data in the logical volume that you create. You don't have to create any structures within the logical volume.

Sizing a Logical Volume for a Database Partition

How much space in a logical volume do you require for your database partition?

Typically, you specify the size of a logical volume in megabytes. However, a logical volume's size must be a multiple of the extent size used in the volume group. By default, the size of each logical extent is 4 megabytes. Type `vgdisplay` and look at the value following PE Size (Mbytes) to check.

So, if a database partition requires 33 megabytes and the default logical extent size is 4 megabytes, you must create a logical volume that is 36 megabytes (or 9 logical extents).

```
33 MB
-----------------      =  8.25 --> 9  logical extents
4 MB / logical extent
```

Distributing Logical Volumes Over Disks

If you plan to use logical volumes heavily for raw disk storage (such as for setting up database partitions) consider how the logical volumes are distributed over the disks.

By default, LVM will assign disk space for logical volumes from one disk, use up the space, and then assign space from the next disk. LVM uses the disks in the order in which they were added to the volume group. This means that the logical volumes are not evenly distributed over the disks.

As I/O access to logical volumes occurs, some disks in the volume group will be heavily used (those with the most logical volumes) while others will be lightly used. This arrangement does not provide optimum I/O performance.

However, you can place logical volumes on specific disks in an interleaved manner, thus balancing the I/O access and optimizing performance.

This chapter has examples of creating logical volumes within specific physical volumes.

Naming Logical Volumes for Raw Data

Because there are no direct HP-UX commands to track the raw data within a logical volume, it's a good idea to name the logical volumes you create for raw data with easily recognizable names. In this way, you can list the logical volume and its characteristics.

Guidelines for Using Disk I/O Interfaces

LVM accommodates disks that use SCSI, HP-FL, and, to a limited extent, HP-IB I/O interface types.

Note Except for HP-FL disk arrays on CIO computers, LVM can be used on all Series 800 supported disks .

The following table highlights the important differences (with respect to LVM) between HP-IB disks and other disk interface types:

Table 8-1. Disk Interface Types and LVM Support

	SCSI	HP-FL	HP-IB
Support mixing with other interface types in a volume group?	Yes	Yes	No
Support bad block relocation?	Yes	Yes	No
Support LVM mirroring?	Yes	Yes	No

As Table 8-1 shows, both HP-FL and SCSI type disks can be placed in one volume group; however, best I/O performance results when volume groups contain only one interface type.

Note HP-IB disks *cannot* be mixed with HP-FL or SCSI disks in a volume group.

Planning for Bad Block Relocation

LVM provides a software bad disk block relocation feature, which uses unused disk space at the end of the disk to store data that could not be stored at an intended location disk because of a defect on the disk. If your disk supports automatic bad block relocation (usually known as "sparing"), then LVM's bad block relocation mechanism is unnecessary.

Bad block relocation is in effect by default when a logical volume is created. You can use the `-r n` option of the *lvcreate*(1M) command to disable the bad block relocation feature.

Use SAM or HP-UX Commands to Manage Logical Volumes

This chapter describes tasks you will most likely be performing as you set up and manage logical volumes. There are two approaches, both of which this chapter describes, for managing logical volumes. You can:

- Use SAM, the System Administration Management subsystem.

  You can manage logical volumes from SAM's `Disk and File Systems` section. How to use SAM to manage logical volumes is described in the section "Managing Logical Volumes Using SAM."

- Use the HP-UX commands interface.

  HP-UX provides several commands (nearly 30) that you can use to create, configure, modify, and monitor logical volumes.

The description of how to use HP-UX Commands to manage logical volumes is in the section "Managing Logical Volumes Using HP-UX Commands."

Note

Guidelines for Using LVM

Whether you use SAM or HP-UX Commands, you will need to make some decisions when setting up or configuring LVM. Often, the decision you make will affect system performance or the availability of data.

The section, "Guidelines for Using Logical Volumes," provides suggestions for optimizing the way you use and set up LVM on your system.

Managing LVM Using SAM

SAM enables you to performs many LVM management tasks.

With SAM, you can

- Create volume groups.
- Add or remove disks to volume groups.
- Create or remove logical volumes.
- Increase the size of logical volumes.
- Increase the size of a file system in a logical volume.
- Change the configuration of logical volumes.
- Configure mirrors in logical volumes.
- Change mirror configuration in logical volumes.
- Split and merge mirrored logical volumes.

You must use HP-UX commands to accomplish the following tasks:

- Extend logical volumes to specific disks.
- Extend mirrors to specific disks.
- Reduce the size of logical volumes.
- Create or change physical volume groups.
- Back up and restore LVM disk information.

How to Use SAM

To use SAM,

- Ensure that you have superuser capabilities.
- Type:

  `/usr/bin/sam`

  Activating the (Help) button from a dialog or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

  Choosing an item from the "Help" menu within a functional area gives you information about:

  - the current functional area
  - keyboard navigation within SAM
  - using the SAM help system
  - displaying the version of SAM you are currently running

Pressing the (f1) key gives you context-sensitive information for the object field at the location of the cursor.

See Chapter 1, "Introduction to System Administration" for additional information about using SAM. If you aren't familiar with SAM, take a moment to look at that overview material. There is a help system in SAM to assist you.

Also, remember the section earlier in this chapter, "Guidelines for Using Logical Volumes," can help you make optimal decisions.

Adding Disks Using SAM

With LVM, disks exist within volume groups. So, when you install a disk on the system, you can:

- Add it to an existing volume group (extend a volume group by adding a disk to it).
- Use it to start a new volume group (create a volume group).

Refer to *Installing Peripherals* to install your new disk. Once you have your disk installed and turned on, you can run SAM and make the disk an LVM disk and assign it to a volume group.

Once the disk is an LVM disk and part of a logical volume, you can create logical volumes in the added space and use them for file systems or swap space.

Note **Pre-installed Systems**

If you are using a pre-installed version of HP-UX, Release 9.0, then you already have one volume group set up.

You can add disks to the existing volume group using SAM.

It is not necessary to set up another volume group unless the disks you are adding to your system use a different interface type or will adversely affect I/O performance.

Some Performance Reminders

If you are adding a disk to an existing volume group or creating a new volume group, remember the following key points.

- HP-IB disks cannot be mixed with disks of other interface types. HP-IB disks do not support bad block relocation or mirroring.
- Mixing disks of different types (either different models or SCSI with HP-FL) in a volume group can produce unpredictable performance when file systems span different disk types or mirrored copies of file systems are on different disk types.

Add a Disk to an Existing Volume Group (Extend a Volume Group)

In SAM, you can configure a new disk into an existing volume group as follows:

1. Highlight `Disks and File Systems->` and activate the (Open) control button.
2. Highlight `Volume Groups` and activate the (Open) control button.
3. Highlight the existing volume group to which to add your disk.
4. From "Actions" on the menu bar, choose `Extend...`.
5. On the list of disks, highlight the disk you added.
6. Activate the (OK) button to add the disk to the volume group.

You have now made the disk available for LVM use. But you must now create logical volumes on this added disk space before you can create file systems and swap space. To create logical volumes for new file systems or new swap areas in the added space, see "Creating Logical Volumes Using SAM".

Create a New Volume Group with a New Disk.

1. Highlight `Disks and File Systems->` and activate the (Open) control button.
2. Highlight `Volume Groups` and activate the (Open) control button.
3. From "Actions" on the menu bar, choose `Create....`
4. From the list, highlight the disk you added.
5. Enter a name for the new volume group in the `New Volume Group Name:` field.
6. Activate the `Optional Volume Group Settings.....` menu bar if you want to change any of the default volume group settings. See "Guidelines for Using Logical Volumes".

Note Note that if you want to use the disk you are adding as a boot disk, turn on the "Set Aside Boot Space on Physical Volume" checkbox.

---

7. Activate the (OK) button to create the new volume group that includes the new disk.

Now, you can:

- Add more disks to this volume group. See the previous section, "Add a Disk to an Existing Volume Group (Extend a Volume Group)". This is how you can create a new volume group consisting of several disks; choose one disk to create the volume group and add the other disks one at a time.
- Create logical volumes for new file systems or new swap areas in the new volume group. See "Creating Logical Volumes Using SAM".

Creating Logical Volumes Using SAM

You can create a logical volume in any available space within the volume group.

You can create a logical volume for

- a file system
- swap space
- raw data

After you create a logical volume, you can proceed immediately to the area in SAM that enables you to create a file system or a swap area in the logical volume. If you want to use the logical volume for raw data, you are done when the logical volume is created.

Before using SAM to create the logical volume, you should have some idea about the logical volume's characteristics, such as:

- In what volume group do you want to create the logical volume?
- What size do you want to make the logical volume?
- What custom name, if any, do you want to give the logical volume?
- Do you want to mirror the logical volume?
- Do you want to change any of the default logical volume characteristics?

Refer to "Guidelines for Using Logical Volumes" for discussions of these questions.

Creating a Logical Volume in SAM

To create a logical volume in SAM:

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate the [Open] control button.
3. Highlight `Logical Volumes` and activate the [Open] control button.
4. From "Actions" on the menu bar, choose `Create...`.
5. Highlight a volume group in which to create the logical volume.
6. Enter a name in the "Logical Volume Name" field, or leave the field blank and let LVM assign the logical volume name.
7. Enter the size of your logical volume in the `Logical Volume Size (MB)` field. Note that values are rounded up to the nearest multiple of the physical extent size.

   If you want to mirror the logical volume, refer to the the section, "Using SAM to Manage Mirrors".
8. Activate the [OK] control button to create the logical volume.

   Note that when the logical volume exists, you can proceed to subareas in SAM that enable you to build file systems or swap areas.

Go on to the next section if you want to create a file system, or skip to "Creating a Swap Area in a Logical Volume Using SAM" if you want to create a swap area.

When SAM displays the Logical Volume list confirming that the logical volume now exists, you can choose either the `Local File Systems` or the `Swap` subarea in SAM from "List" on the menu bar.

8

Creating a File System in a Logical Volume Using SAM

Chapter 6, "Managing the File System", covers file systems in detail.

creating in logical volumes using SAM| The following discussion covers the creation of a file system (using defaults) immediately after creating a logical volume. If you have just created a logical volume, you can:

1. From "List" on the menu bar, choose `Local File Systems`.
2. Without highlighting a file system on the `Local File Systems` list, choose `Add` and then choose `Using the Logical Volume Manager...` from "Actions" on the menu bar. (If no logical volume exists, you must create it in the `Logical Volumes` subarea.)
3. Enter a `Mount Directory`, such as `/projects`. The mount directory can be new or an existing empty directory. This is where the new file system will join the existing file system hierarchy. If the directory you specify does not exist, SAM will create it for you. (See Chapter 6, "Managing the File System," for more information on the mount directory.)
4. Highlight the logical volume from the selection list.
5. Turn on the "Create new file system" checkbox; accept the checkbox setting for file name length unless you want to specify otherwise.
6. Use defaults for `Set user ID`, `When to Mount:`, and `Access Permissions:` unless you want to change them.
7. Activate [OK] to create the file system.

   SAM creates the file system in the logical volume you created, mounts it, and adds it to the `/etc/checklist` file.

   The file system is now ready to use. Later on, you can increase the size of the file system you created (from the "Logical Volume" subarea in SAM), or mirror it (if you have MirrorDisk/UX) if you need to.

Increasing the Size of a File System Using SAM

You can use SAM to increase the size of a file system in a logical volume by increasing the size of the logical volume.

extending in logical volumes using SAM| SAM will automatically unmount the file system, extend the logical volume, extend the file system to the size of the logical volume, and remount the file system.

Note Because the file system is unmounted in this process, no users can be using the file system. If necessary, run SAM with the system in single user mode.

Choose `Routine Tasks->` on SAM's control box, then choose `System Shutdown`, and then activate the `Go to Single User State` control button.

To extend a file system in SAM,

1. Highlight the logical volume that contains the file system on the `Logical Volumes` list.
2. From "Actions" on the menu bar, choose `Increase Size...`.
3. Enter the new size in the `Increase to New Size (MB)` field. SAM will round your entry upward to the nearest whole physical extent size.
4. Activate [OK] to increase the size of the logical volume and the file system in it.
5. Choose `Local File Systems` from "List" on the menu bar to display the file system and verify its increased size.

8

Decreasing the Size of a Logical Volume Using SAM

You can decrease the size of a logical volume.

Note

Decreasing the size of a logical volume will probably result in the loss of some of the data in the logical volume, unless you first copy the data in the logical volume either to another logical volume or to a backup tape.

If you no longer need the data in the logical volume you are reducing in size, you can also remove the logical volume and create another one in its place.

You cannot reduce the size of a file system.

---

1. Highlight the logical volume that you want to decrease in size on the `Logical Volumes` list.
2. From "Actions" on the menu bar, choose `Decrease Size...`.
3. Enter the new size in the appropriate field. SAM will round your entry upward to the nearest whole physical extent size.
4. Activate [OK] to decrease the size of the logical volume and the file system in it.

After you have decreased the size of the logical volume, you can either restore the data it contained formerly or use it for a new purpose.

Creating a Swap Area in a Logical Volume Using SAM

Chapter 7, "Managing Swap Space", contains discussion of swapping in more detail. Chapter 7 covers such topics as how much swap you require, when to use file system swap, how you should allocate device swap over disks or logical volumes, how to assign priorities, and so on.

The following discussion covers the creation of a device swap area (using defaults) immediately after creating a logical volume. If you have just created a logical volume, you can:

1. Choose Swap from "List" on the menu bar.
2. Without highlighting any objects in the Swap list, choose Add Device Swap and then choose Using the Logical Volume Manager... from "Actions" on the menu bar.
3. Highlight the logical volume from the selection list.
4. You should not need to change any of the fields and checkboxes on the next screen. The logical volume you created and the defaults of enabling swap now and at each boot time are selected. The default "priority" is also set. Make changes if necessary.
5. Activate OK to add device swap in the new logical volume and enable it for use.

Managing Logical Volumes Using HP-UX Commands

This section describes how to set up, modify, and view the status of logical volumes using the set of LVM HP-UX commands. Numerous examples illustrate their use. Also, remember the section earlier in this chapter, "Guidelines for Using Logical Volumes," can help you make optimal decisions.

In addition to this chapter, you can refer to:

- Chapter 6, "Managing the File System," for examples of using LVM commands when creating logical volumes for file systems.
- Chapter 7, "Managing Swap Space," for examples of using LVM commands when creating logical volumes for swap.
- The manual *How HP-UX Works* for LVM concepts discussed in depth.
- The overview of LVM presented earlier in this chapter.
- The guidelines for using LVM presented later in this chapter.
- The LVM command entries in *HP-UX Reference* for syntax descriptions, explanations, and other usage examples.

LVM Commands - Quick Reference

Table 8-2. LVM Commands

Command	Description
extendfs	Extends the size of a file system.
lvcreate	Creates a logical volume in a volume group.
lvchange	Changes the characteristics of a logical volume.
lvdisplay	Displays information about logical volumes.
lvextend	Increases the disk space allotted to a logical volume.
lvlnboot	Prepares a logical volume to be a root, primary swap, or dump volume.
lvreduce	Decreases the disk space allocated to a logical volume.
lvremove	Removes one or more logical volumes from a volume group.

Table 8-2. LVM Commands (continued)

Command	Description
lvrmboot	Removes link that makes a logical volume a root, primary swap, or dump volume.
pvchange	Changes the characteristics of a disk in a volume group.
pvcreate	Makes a disk an LVM disk (a physical volume), such that it can be used as part of a volume group.
pvdisplay	Displays information about physical volumes in a volume group.
pvmove	Moves data from one logical volume disk to another.
vgcfgbackup	Backs up the volume group configuration information.
vgcfgrestore	Restores volume group configuration from a configuration file.
vgchange	Set availability or changes the characteristics of a volume group.
vgcreate	Creates a volume group.
vgdisplay	Displays information about volume groups.
vgexport	Exports a volume group and its associated logical volumes.
vgextend	Extends a volume group by adding disks to it.
vgimport	Imports a volume group onto the system.
vgreduce	Reduces a volume group by removing one or more disks from it.
vgremove	Removes definition(s) of volume group(s) from the system.
vgscan	Scans all disks and looks for logical volume groups.

Table 8-3. LVM Commands - Mirroring

Command	Description
lvmerge	Merges two logical volumes into one logical volume.
lvsplit	Splits a mirrored logical volume into two logical volumes.
lvsync	Synchronizes mirrors that are stale in one or more logical volumes.
vgsync	Synchronizes mirrors that are stale in one or more volume groups.

Creating Volume Groups Using HP-UX Commands

All logical volumes exist on disks that have been organized into **volume groups.**

Checklist of five steps for creating a volume group:

1. Select disks for the volume group.
2. Make the disks LVM disks (that is, make them physical volumes).
3. Create a control directory for the volume group.
4. Create a "group" file in the volume group directory.
5. Create the volume group.

Note **Pre-installed Systems**

If you are using a pre-installed version of HP-UX, Release 9.0, then you already have one volume group set up, the Root Volume Group.

You can add disks to the existing volume group by using the *vgextend*(1M) command. See the section "Adding Disks to an Existing Volume Group."

It is not necessary to set up another volume group unless the disks you are adding to your system use a different interface type or will adversely affect I/O performance.

1. **Select disks for the volume group.** A volume group typically contains more than one disk, although it can contain only one. Ideally, a volume group contains identical disks; this optimizes file system performance when the file system spans disks.

   You can combine SCSI and HP-FL disks in a volume group, but the combination of dissimilar interfaces might affect performance.

   You cannot mix HP-IB disks with SCSI or HP-FL disks; HP-IB disks will not work with other disk interface types in a volume group.

2. **Make the disks LVM disks (physical volumes).** This enables them to be members of a volume group. Use the *pvcreate*(1M) command.

```
pvcreate /dev/rdsk/cnd0s2
```

- Be sure the disk you want to use as a physical volume has been installed and is turned on.
- Know the *raw* device file name for section 2 of the disk (`/dev/rdsk/cnd0s2`).
- If you want to use the disk as a system disk, that is, to hold the root file system logical volume, you need to specify a boot section with the `-B` flag.

3. **Create a directory for the volume group** using the pattern `/dev/vgnn`.

   ```
   mkdir /dev/vgnn
   ```

   In the directory name, *nn* is the volume group number. Use a sequence for the volume groups on your system. If you have a pre-installed system, you already have volume group. You can also use a customized name for the volume group, such as `/dev/report_data`.

4. **Create a control file named** `group` **in the directory** `/dev/vgnn` with the *mknod*(1M) command.

   ```
   mknod /dev/vgnn/group c 64 minor_number
   ```

   - You need to specify that the `group` file is a character device file by using a "`c`" following the device file name.
   - The major number for the `/dev/vgnn/group` device file, regardless of the volume group number, is always 64; that's the major number for the LVM pseudo driver.
   - The minor number for the `group` file is hexadecimal, always ends in "0000," and has the form:

     ```
     0x__0000
     ```

     where the two numbers preceding the 0000 are the hexadecimal representation of *nn*, which is the volume group number.

   You can verify that you have created the "group" file by listing the contents of the `/dev/vgnn` directory. For example,

   ```
   ll /dev/vg01
   total 0
   crw-rw-rw-   1 root     other     64 0x010000 Feb  7 12:53 group
   ```

   The `group` special file provides the means by which the LVM kernel and the LVM commands communicate within the volume group you create.

5. **Create the volume group** and specify the disks (physical volumes) it will contain; use *vgcreate*(1M).

```
vgcreate /dev/vgnn /dev/dsk/cnd0s2
```

- Use the name for the volume group,vg*nn*, that you used in step 3 to create the volume group directory.
- When you create a volume group, you can assign several physical volumes to it at one time; the physical volumes must exist and not be part of an existing volume group.

  Note that when you the disk an LVM disk (physical volume), you used the *character* device file for the disk; now you use the *block* device file to include the disk in a volume group.
- You can use the default values for the numerous flags for the *vgcreate*(1M) command.

Verify that you have created the volume group by using the `vgdisplay` command; for example:

```
vgdisplay /dev/vg01

--- Volume groups ---
VG Name                 /dev/vg01
VG Status               available
Max LV                  255
Cur LV                  0
Open LV                 0
Max PV                  32
Cur PV                  1
Act PV                  1
Max PE per PV           1016
VGDA                    2
PE Size (Mbytes)        4
Total PE                632
Alloc PE                0
Free PE                 632
Total PVG               0
```

Points to Remember

- The disk you want to use as a physical volume must be installed on your system and be turned on. (See the Installing Peripherals manual if you need information for installing a disk).
- You need to consider what disks you include together within a volume group. A volume group consisting of the same disk type will exhibit best performance. A volume group with differing disk types might exhibit unpredictable file system performance. (Refer to the later discussion on "Guidelines for Organizing Disk Space in Logical Volumes.")

Example: Setting Up a Volume Group with Two Disks

You've added two new disks on your system, LU numbers 2 and 3. They are identical SCSI drives, Model HP2213A. You already have one volume group, and you plan now to set up the second one, `vg01`.

1. Make the disks LVM disks (physical volumes).

   ```
   pvcreate /dev/rdsk/c2d0s2
   pvcreate /dev/rdsk/c3d0s2
   ```

   LVM confirms.

2. Now, create the volume group directory.

   ```
   mkdir /dev/vg01
   ```

3. Then, create the character device file `/dev/vg01/group`.

   ```
   mknod /dev/vg01/group c 64 0x010000
   ```

4. Create the volume group, using defaults for the *vgcreate*(1M) command.

   ```
   vgcreate /dev/vg01 /dev/dsk/c2d0s2 /dev/dsk/c3d0s2
   ```

   Or, you could create the volume group and set it up with a two megabyte extent size.

   ```
   vgcreate -s 2 /dev/vg01 /dev/dsk/c2d0s2 /dev/dsk/c3d0s2
   ```

5. Finally, verify that you succeeded in making the volume group by typing:

   ```
   vgdisplay /dev/vg01
   ```

6. Now you can create a logical volume for a file system or swap area. See "Creating Logical Volumes".

Adding Disks to an Existing Volume Group

To add disks (physical volumes) to an existing volume group:

1. First, make the disks that you want to add to a volume group LVM disks; use the *pvcreate*(1M) command.
2. Use the *vgextend*(1M) command to include one or more physical volumes.

You can also remove disks from a volume group using the *vgreduce*(1M) command, but you must first remove any logical volumes on the physical volume. See the later section, "Removing Logical Volumes".

Caution Whenever you add or remove disks from the **root volume group** (the volume group that contains **/**), you must run the *lvlnboot*(1M) command to update the boot data stored on the boot disks in the volume group. See the later discussion about the "The Root Volume Group."

Example, Adding Two Disks to an Existing Volume Group

Suppose you have an existing volume group, `/dev/vg01`, and you want to add two disks to it. You have installed the disks, LUs 4 and 5.

1. Make the disks LVM disks (physical volumes); remember to use the disk's character device file.

```
pvcreate /dev/rdsk/c4d0s2
pvcreate /dev/rdsk/c5d0s2
```

2. Add the LVM disks to the volume group `/dev/vg01` (remember to use the disk's block device file).

```
vgextend /dev/vg01 /dev/dsk/c4d0s2 /dev/dsk/c5d0s2
```

3. To verify that the volume group now contains the disks, use the command:

```
vgdisplay /dev/vg01
```

You can see the disks under the heading: `--- Physical Volumes ---`.

8

Displaying Information About Volume Groups, Physical Volumes

Once a volume group is created, you can display useful information about it using the *vgdisplay*(1M) and *pvdisplay*(1M) commands. Even though the examples that follow show information about topics discussed later, they are immediately useful LVM commands.

Listing a Specific Volume Group

Use `vgdisplay` to list information about a volume group or all volume groups. For example, to display information about the particular volume group, `/dev/vg2`:

```
vgdisplay /dev/vg2

--- Volume groups ---
VG Name                 /dev/vg2
VG Status               available
Max LV                  255
Cur LV                  1
Open LV                 1
Max PV                  32
Cur PV                  3
Act PV                  3
Max PE per PV           1016
VGDA                    6
PE Size (Mbytes)        4
Total PE                469
Alloc PE                2
Free PE                 467
Total PVG               0
```

8

Listing All Volume Groups

Notice that the display above shows information for the volume group specified. If you had simply entered `vgdisplay`, however, you would see the same type of information for each "available" volume group in the system (see the later discussion "Setting Volume Group Availability").

For volume groups that are unavailable, you will see a message such as:

```
vgdisplay: Volume group not activated.
vgdisplay: Cannot display volume group "/dev/vg3".
```

The later discussion "Setting Volume Group Availability" provides details about activating and deactivating volume groups.

Displaying Detailed Volume Group Information

Use `vgdisplay -v` to show the detailed information about a volume group.

In addition to general information about the volume group, the display shows information about each logical volume and each physical volume in the volume group.

Use the **pvdisplay** command to show information about specific physical volumes. For example, you could see information about two physical volumes, **/dev/dsk/c2d0s2** and **/dev/dsk/d3d0s2**.

```
pvdisplay /dev/dsk/c2d0s2 /dev/dsk/c3d0s2
--- Physical volumes ---
PV Name                 /dev/dsk/c2d0s2
VG Name                 /dev/vg3
PV Status               available
Allocatable             yes
VGDA                    2
Cur LV                  0
PE Size (Mbytes)        4
Total PE                125
Free PE                 125
Allocated PE            0
Stale PE                0

PV Name                 /dev/dsk/c3d0s2
VG Name                 /dev/vg2
PV Status               available
Allocatable             yes
VGDA                    2
Cur LV                  1
PE Size (Mbytes)        4
Total PE                125
Free PE                 124
Allocated PE            1
Stale PE                0
```

Creating Logical Volumes

You can create logical volumes in the disk space available within a volume group by using the *lvcreate*(1M) command. The general form for using the command is:

```
lvcreate volume_group
```

The above command creates a logical volume with default characteristics in the specified volume group.

Note Note that, in addition to the examples in this section, Chapter 6, "Managing the File System", has examples of creating logical volumes for file systems and Chapter 7, "Managing Swap Space", has examples of creating logical volumes for swap areas.

Creating a Logical Volume, Simple Case

Create a logical volume in the volume group `vg00`

```
lvcreate vg00
Logical volume "/dev/vg00/lvol4" has been successfully created
with lv number 4.
```

When LVM creates the logical volume, it creates the block and character device files and places them in the directory `/dev/vg01`.

Note By default, the logical volume is created without size. You can use the *lvextend*(1) command to extend the logical volume to a specific size.

8

Creating Logical Volumes, Optional Characteristics

The guidelines discussed earlier in this chapter can help you decide how to set up a logical volume depending on how you intend to use it.

Before you create a logical volume, you should determine the characteristics you want it to have, and choose the appropriate flags of the *lvcreate*(1M) command. You can create a logical volume using the default characteristics and change them later. (The entry for *lvcreate*(1M) in *HP-UX Reference* contains a complete description of all options.)

Options Available When Creating Logical Volumes with `lvcreate`

-L The size of the logical volume in megabytes. The size specified will be rounded up to the nearest whole logical extent size (the default size of a logical extent is four megabytes).

By default, logical volumes have no size when they are created.

Note that when you create a logical volume, you cannot specify which disk should hold the logical volume's data. Therefore, if you want to create a logical volume of a specific size that uses a specific disk, create the logical volume without the `-L` flag, and then extend the logical volume by using *lvextend*(1M) to a specific physical volume.

-l The number of logical extents in the logical volume. The default is zero; you can create a logical volume and later change its size using the *lvextend*(1M) command.

Note that when you create a logical volume, you cannot specify which disk should hold the logical volume's data. Therefore, if you want to create a logical volume of a specific size that uses a specific disk, create the logical volume without the `-l` flag, and then extend the logical volume by using *lvextend*(1M) to a specific physical volume.

-p Read-write access permissions of the logical volume. By default the permissions are set to read-write (`w`); use `-p r` to set read-only permission.

-n A custom name you want to assign to the logical volume. By default, LVM will name it.

-r The bad block relocation policy. By default, bad block relocation is in effect; `-r n` turns it off. (Note that the bad block relocation policy cannot be used for either the `root` or primary swap logical volumes.)

-C Contiguous allocation policy. Contiguous allocation (`-C y`) means that physical extents of a logical volume are allocated in an unbroken, ordered sequence on one physical volume (disk). (Contiguous allocation can apply to mirrored logical volumes also, in which case each mirrored copy must be contiguous.) By default, allocation is not contiguous. (Note that the `root` and primary swap logical volumes must be created with continuous allocation in effect.)

Note When you create logical volumes for either `root` or primary swap, you must issue the `lvcreate` command specifying continuous allocation (`-C y`) and no bad block relocation (`-r n`).

See the later discussion on the "The Root Volume Group."

More Examples: Creating Logical Volumes

- To create a logical volume with default characteristics in the volume group `/dev/vg01`:

  ```
  lvcreate /dev/vg01
  ```

  Not only does LVM create the logical volume, it creates the block and character device files and places them in the directory `/dev/vg01`.

- To create a logical volume of 200 megabytes in the volume group `/dev/vg01`, and set the permissions to read-only, use:

  ```
  lvcreate -L 200 -p r /dev/vg01
  ```

  LVM reports its activities, first as it creates the logical volume (naming it using the next available number in sequence) and then as it extends it. You can check the directory `/dev/vg01` and find that the block and character device files are also created.

- To create a logical volume named `projects` that is 10 logical extents in size, use:

  ```
  lvcreate -l 10 -n projects /dev/vg01
  ```

  Note that LVM first creates the logical volume (naming it projects) and then extends it. You can check the directory `/dev/vg01` and find that the block and character device files are also created.

Displaying Information About Logical Volumes

You can view the characteristics of logical volumes by using the *lvdisplay*(1M) command.

- To display information about one or more logical volumes:

```
lvdisplay /dev/vg2/usr
--- Logical volumes ---
LV Name                  /dev/vg2/usr
VG Name                  /dev/vg2
LV Permission            read/write
LV Status                available/syncd
Mirror copies            0
Consistency Recovery     MWC
Schedule                 parallel
LV Size (Mbytes)         100
Current LE               25
Allocated PE             25
Bad block                on
Allocation               strict
```

- You can also use the `-v` option with `lvdisplay` to display detailed information about one or more logical volumes (it might be helpful to pipe the display command to `more`):

```
lvdisplay -v /dev/vg2/usr | more
```

Along with the general information about the logical volume, you can see the distribution of its logical extents on physical volumes, how each logical extent corresponds to each physical extent, and the status of that physical extent, current or stale, if it is a mirror copy. (See "Managing Mirrors".)

For example, a verbose display might include something like:

```
# lvdisplay -v /dev/vg2/usr
--- Logical volumes ---
LV Name
.
.
   --- Distribution of logical volume ---
   PV Name            LE on PV  PE on PV
   /dev/dsk/c5d0s2    25        25
```

8

```
    --- Logical extents ---
LE   PV1                PE1  Status 1
0000 /dev/dsk/c5d0s2    0028 current
0001 /dev/dsk/c5d0s2    0029 current
0002 /dev/dsk/c5d0s2    0030 current
0003 /dev/dsk/c5d0s2    0031 current
0004 /dev/dsk/c5d0s2    0032 current
0005 /dev/dsk/c5d0s2    0033 current
0006 /dev/dsk/c5d0s2    0034 current
.
.
```

In this listing, we can see that there are 25 logical extents in `/dev/vg2/usr`. Under the `Distribution` heading, we can see that there is one disk, `/dev/dsk/c5d0s2` (under `PV Name`), that has the copies of the logical extents.

In the listing "`--- Logical extents ---`" we can see exactly how the data in the logical volume are distributed to physical extents. Logical extent ("`LE`") "0000" corresponds to physical extent ("`PE1`") "0028". If the data had been mirrored, there would be other columns headed "`PE2`" and "`PE3`" to show where the other physical copies are located. See the section, "Managing Mirrors."

Changing the Characteristics of a Logical Volume

You can view the current characteristics of a specific logical volume by using the *lvdisplay*(1M) command, previously discussed, and you can change the current characteristics of a specific logical volume by using the *lvchange*(1M) command.

The characteristics of a logical volume that you can change are those that you could define when you created the logical volume. Refer to the section "Options Available When Creating Logical Volumes with `lvcreate`".

8

Example, Changing Allocation Policy from Contiguous to Non-Contiguous

Change the current allocation policy of a logical volume to be non-contiguous.

```
lvchange -C n /dev/vg02/proj_lv
```

Increasing the Size of a Logical Volume

You can increase the size of a logical volume by using the *lvextend*(1M) command. You can also specify that you want the increased disk space (either in terms of extents or megabytes) allocated to a specific disk (physical volume), or you can let LVM determine where to allocate it.

Example, Increasing the Size of a Logical Volume

Suppose you want to increase a logical volume, `/dev/vg01/lvol4`, which is currently 48 megabytes, to 100 megabytes. (If it contains a file system, make sure the file system is unmounted. An example of extending a logical volume and extending a file system in it follows.)

```
lvextend -L 100 /dev/vg01/lvol4
```

Note that you specify the new size rather than the amount of increase. When LVM implements the new size, it will use whole extent sizes. For example, if the extent size was four megabytes and you specified `-L 98`, LVM extends the logical volume to 100 megabytes.

Example: Increasing the Number of Logical Extents in a Logical Volume.

You can also increase the size of a logical volume by increasing the number of its logical extents.

Suppose the logical extent size is four megabytes and you need the logical volume to be 100 megabytes; this requires 25 logical extents. You would then enter:

```
lvextend -l 25 /dev/vg01/lvol4
```

Note that when specifying extents, you use the `-l` option (lowercase L) rather than `-L`, which is for specifying size in megabytes.

Extending a Logical Volume to a Specific Disk

Suppose you have several disks in a volume group and two of them are identical models. You want to extend a 275 megabyte logical volume that resides on one of the disks to 400 megabytes. Also, you want to make sure the increase is allocated to the other identical disk.

To extend a logical volume (e.g., `/dev/vg01/lvol4`) to a specific disk (e.g., `/dev/dsk/c3d0s2`), explicitly specify the disk's block device file.

```
lvextend -L 400 /dev/vg01/lvol4 /dev/dsk/c3d0s2
```

Extending a Logical Volume and Extending a File System

If you have a file system which is nearing capacity, you can extend the logical volume that contains the file system, and then use the *extendfs*(1M) command to increase the file system.

Note If the logical volume you are extending has a contiguous allocation policy in effect, the command to extend the logical volume will fail if no contiguous disk space is available.

---

1. Unmount the file system you want to extend.

   ```
   umount /dev/vg4/lvol2
   ```

2. Use *lvextend*(1M) to extend the logical volume first. For example, to extend a logical volume by 50 megabytes to 100 megabytes,

   ```
   lvextend -L 100 /dev/vg4/lvol2
   ```

3. Use the *extendfs*(1M) command to increase the file system capacity in proportion to the increase in the logical volume. Note that *extendfs*(1M) requires the use of the character device file.

   ```
   extendfs /dev/vg4/rlvol2
   ```

   The results of file system extension are reported.

4. Mount the file system again; specify the mount point, `/projects`, for example.

   ```
   mount /dev/vg4/lvol2 /projects
   ```

5. Run `bdf` and note the increased capacity.

Reducing the Size of a Logical Volume

You can reduce the size of a logical volume by using the *lvreduce*(1M) command. (You can also reduce the number of mirror copies using the `lvreduce` command; see "Reducing the Number or Mirror Copies.")

Reducing the size of a logical volume is appropriate when you want to use the logical volume for another purpose that requires less space.

Caution When you reduce the size of a logical volume, you might lose data as LVM deallocates disk space.

Reduce the size of a logical volume only if you no longer need its current contents, or if you have safely backed up the contents to tape or to another logical volume.

You cannot reduce the size of a file system.

Example, Reducing the Size of a Logical Volume

Suppose you have a logical volume of 80 megabytes. You no longer need the current data in the logical volume, but you would like to use it for another purpose that requires only 40 megabytes.

```
lvreduce -L 40 /dev/vg03/lvol4
```

When you issue the command, LVM asks for confirmation. If you answer `y` to proceed, you get a confirming message.

```
Logical volume "/dev/vg03/lvol4" has been successfully reduced.
```

Moving Data in Logical Volumes From Disk to Disk

You can use the *pvmove*(1M) command to move data contained in logical volumes from one disk to another disk within a volume group.

For example, you can move a logical volume's data from one disk to another to use the space on the first disk for some other purpose. You might also move a logical volume's mirror copy from one disk to another (you can find an example of this under "Moving a Logical Volume Mirrored Copy from One Disk to Another").

Also, you can move all data from one disk to another. You might want to do this, for example, so you can remove a disk from a volume group. After removing the logical volume data off a disk, you can then remove the disk from the volume group. See "Reconfiguring Disks in a Volume Group."

Example, Moving Logical Volume Data from One Disk to Another

Suppose you want to move the data in a logical volume, `/dev/vg01/markets`, from the disk `/dev/dsk/c3d0s2` to the disk `/dev/dsk/c4d0s2`.

Note that when you issue the command, you must specify a specific logical volume on the source disk with the `-n` flag. You must also specify the source disk first in the command line. For example,

```
pvmove -n /dev/vg01/markets /dev/dsk/c3d0s2 /dev/dsk/c4d0s2
```

8

Example, Moving All Data on One LVM Disk to Another.

If you want to move all the data on a given disk within a volume group, you can use the *pvmove*(1M) command to move the data to other specific disks or let LVM move the data to available space within the volume group. See "Reconfiguring Disks in a Volume Group" to remove the disk from the volume group.

To move data off disk `/dev/dsk/c4d0s2` and let LVM transfer the data to available space in the volume group, you could enter:

```
pvmove /dev/dsk/c4d0s2
```

To move data off disk `/dev/dsk/c4d0s2` to the destination disk `/dev/dsk/c5d0s2`, enter:

```
pvmove /dev/dsk/c4d0s2 /dev/dsk/c5d0s2
```

If space doesn't exist on the destination disk, the `pvmove` command will not succeed.

Removing Logical Volumes

You can remove a logical volume with the *lvremove*(1M) command. For example, to remove an empty logical volume,

```
lvremove /dev/vg3/lvol2
```

If the logical volume contains data, LVM prompts you for confirmation. You can use the `-f` flag to remove the logical volume without a confirmation request.

Managing Mirrors

MirrorDisk/UX provides the means to store identical copies of data on separate physical volumes (disks). This means that data is not lost if a single disk fails when a disk head crashes or power supply fails.

In addition to mirroring copies of data on different disks, you can configure mirrors on disks that use different controllers or different I/O channels, that is, on disks in separate **physical volume groups**.

Note

MirrorDisk/UX is a separately orderable product. MirrorDisk/UX functionality can be added to the system via the *update*(1M) utility.

If you do not have MirrorDisk/UX, then you will observe the following:

- In SAM, any mirroring aspects of the LVM subareas in SAM are disabled.
- When you use mirroring related flags with LVM HP-UX commands, the system will either ignore them or will issue syntax error messages.

Mirroring with MirrorDisk/UX is appropriate for:

- Making copies of valuable data on two or more disks, thus providing increased data availability.
- Protecting the root disk to provide increased system availability by mirroring the root file system and primary swap.
- Offsetting the risk posed to a single large file system that spans two disks by maintaining multiple copies of the file system data; the failure of one of the disks containing part of the file system does not make the entire file system unaccessible.

It is also possible to increase system performance with MirrorDisk/UX if the I/O on your system consists predominantly of reads; on the other hand, if the I/O on your system is balanced between reads and writes, using mirrors can degrade performance.

Guidelines for Managing Mirrors

The following sections provide guidelines for using MirrorDisk/UX. A variety of examples are shown in later sections that discuss configuring and managing logical volumes.

Number of Mirror Copies

You specify the number of mirror copies of a logical volume when you create a logical volume. The default is *no* mirrored copies. Therefore, by specifying one or two mirror copies of a logical volume, you are specifying mirroring.

Singly-mirrored data is an original plus a copy, that is, two copies of data. **Doubly-mirrored** data consists of an original and two copies.

Doubly-mirrored data provides higher availability than singly-mirrored data, but consumes correspondingly more disk space. You might want to set up doubly-mirrored data so that one copy can be taken off-line (to back up the logical volume, for example), while the other two mirror copies are still online.

Guidelines for Setting Allocation Policy

- Strict Allocation

  When you set up mirrors, you can specify that LVM create mirror copies on separate physical volumes. This is known as **strict allocation** and is the default. HP recommends the strict allocation policy. Non-strict allocation doesn't afford the higher availability benefits of mirroring, yet it does allow you to split and merge mirrored logical volumes.

  You can also set strict allocation to apply to physical volumes in separate physical volume groups (PV groups). This can provide data availability in the event of a failure of a disk interface card, or perhaps, a failure in one I/O channel.

  You set the allocation policy when you create the logical volume, either with SAM or with the *lvcreate*(1m) command.

When you use SAM to create a mirrored logical volume, you have the choices to set up mirrors as follows:

- `Separate Disks`; this is strict allocation.
- `On Separate PV Groups`; this is strict allocation with respect to physical volume groups (PV groups).
- `Same Disk O.K.`; this is non-strict allocation.

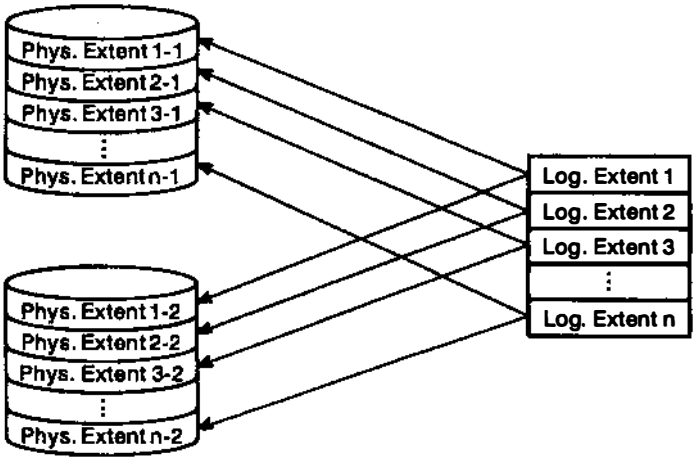
With HP-UX command *lvcreate*(1M), the strict allocation is implemented with the `-s` option.

- Contiguous Allocation

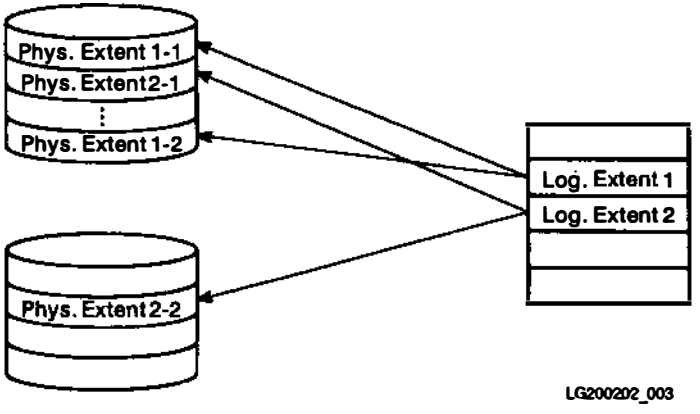
You can also specify that the physical extents of mirror copies be allocated in a **contiguous** manner, that is, with no gaps within the set of physical extents in a mirror copy, and with all physical extents of a mirror copy existing on one physical volume.

By default, the allocation of physical extents is not contiguous. But, if a logical volume is to be used for a root, swap, or dump device, the physical extents must be contiguous for all mirror copies.

Strict Mirror Copy Allocation



Non-Strict Mirror Copy Allocation



8

Figure 8-4. Mirror Copy Allocation, Strict and Non-strict

Mirror Write Scheduling Policy

You can specify whether the mirror copies of a logical extent are written to the physical volumes using a **parallel** or a **sequential** scheduling policy.

Sequential scheduling provides for slightly higher data integrity than does parallel scheduling. When you specify sequential scheduling, the data is written one copy at a time, writing the next copy only when the previous copy is successfully written.

Parallel scheduling, on the other hand, is more efficient because data is transferred simultaneously to all mirror copies.

HP recommends using parallel (the default) scheduling because of the performance loss associated with sequential scheduling.

Configuring Mirrors for Crash Recovery

Three choices are available for configuring mirrored logical volumes to recover consistent data after a system crash:

- Enable the **Mirror Write Cache** (MWC). The system recovers consistent data as fast as possible after a crash, but at the cost of some impact on routine performance.

  The Mirror Write Cache tracks each write of mirrored data to the physical volumes and maintains a record of any mirrored writes not yet successfully completed at the time of a system crash. Therefore, during the recovery from a crash, only those physical extents involved in unsuccessful mirrored writes need to be made consistent.

- Employ the **Mirror Consistency Recovery** mechanism . This means no impact on routine performance, but a longer recovery period after a system crash.

  During recovery, the Mirror Consistency Recovery mechanism examines each logical extent in a mirrored logical volume, ensuring all copies are consistent. It does not use data in stale copies to achieve consistency.

- Provide for no consistency recovery. By using neither the MWC nor the Mirror Consistency Recovery mechanism, you keep performance and get fast recovery. After a crash, the logical volumes are available, but there is no guarantee of consistent data from mirror copy to mirror copy.

The choice of which recovery means to set up depends on the relative importance of performance, the speed of system recovery in the event of a crash, and the consistency of recovered data among mirror copies.

You can configure a logical volume to use the MWC or the Mirror Consistency Recovery method at the time you create a logical volume or at the time you change the characteristics of a logical volume. You can implement the configuration using SAM or the HP-UX commands *lvcreate*(1M) or *lvchange*(1M).

Although you can decline to use either or both of the mirror consistency recovery methods, be aware that any system crash will jeopardize the integrity of mirrored data. Many databases use a log recovery scheme to ensure mirror consistency, in which case a logical volume containing only records of user data, for example, needs no mirror consistency recovery scheme. On the other hand, the logical volume containing the database's log does require a mirror consistency recovery scheme (HP recommends using the MWC).

Guidelines: Mirroring Logical Volumes with File Systems

Detailed discussion of mirroring can be found in "Planning for Mirroring." To set up mirroring, you need to add the MirrorDisk/UX product.

Whenever possible, configure the mirrored copies of file systems in logical volumes on identical disk types. By comparison, mirrored copies of a file system on different disk types will yield unpredictable and probably inferior performance.

As a system administrator, you can control the way in which physical volumes are allocated to contain the mirrored logical volume copies. The best way to do this is by:

1. Creating an unmirrored logical volume of no logical extents. By default, if you specify no size for a logical volume when you create it, no physical extents are allocated for the logical volume.
2. Extending the logical volume by one logical extent to a specific physical volume of the desired disk type.
3. Extending the logical volume by the number of mirrored copies, specifying the physical volumes for each mirror copy.
4. Extending the mirrored logical volume by the number of extents required for the file system to each of the physical volumes.

Specific examples for using the *lvcreate*(1M) and *lvextend*(1M) commands in sequence are shown in the section, "Creating Logical Volumes." While you can use SAM to create and extend logical volumes, you cannot extend logical volumes to specific disks with SAM.

Guidelines for Mirroring Swap

8

You can mirror logical volumes with swap areas and gain by having higher availability. The recovery time for mirrors of swap logical volumes is not an important consideration, so mirrors of swap logical volumes should be created with Mirror Write Cache disabled (`-M n`).

Using SAM to Manage Mirrors

In SAM, the Logical Volume subareas related to mirroring do not function unless the MirrorDisk/UX subsystem has been added to the system via *update*(1M).

SAM can help you with the following mirroring tasks.

- Setting up a logical volume with mirrors.
- Setting and changing the characteristics of a logical volume's mirrors; for example, you can specify:
  - The mirroring policy: same disk, separate disks, or separate PV groups.
  - The mirror recovery method.
  - Scheduling Policy.
- Changing the number of a logical volume's mirror copies.
- Splitting a mirrored copy away from a logical volume (for backup purposes).
- Merging copies of a split mirrored logical volume.

As you attempt to perform tasks with logical volume mirrors in SAM, be sure to refer to SAM's online [Help], if necessary, and review the previous section, "Guidelines for Managing Mirrors."

Creating a Mirrored Logical Volume Using SAM

The task of creating a mirrored logical volume using SAM is not much different from creating a non-mirrored logical volume. All you need to do is specify one or two mirror copies instead of the using the default zero mirror copies when you create the logical volume. The key thing to remember is that each mirror copy consumes disk space equivalent to the size of the logical volume. For example, a ten megabyte logical volume that is singly mirrored requires 20 megabytes of disk space overall.

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate the [Open] control button.
3. Highlight `Logical Volumes` and activate the [Open] control button.
4. From "Actions" on the menu bar, choose `Create...`
5. Highlight a volume group in which to create the logical volume.
6. Enter a name in the "Logical Volume Name:" field, or leave the field blank and let LVM assign the logical volume name.
7. Enter the size of your logical volume in the `Logical Volume Size (MB)` field. Note that values are rounded up to the nearest multiple of the physical extent size.

   Normally, when creating a logical volume without mirrors, you can create an empty logical volume, but you must specify a size when you set up a mirrored logical volume.
8. Choose the number of mirror copies. If you want to mirror the logical volume, choose a value other than zero for the number of "Mirror Copies." "0 Copies" means there is only one set of data. "1 Copy" means data exists in duplicate, and "2 Copies" means data exists in triplicate.

9. Skip the `Optional Logical Volume Settings...` unless you want to view or modify them. The defaults are:

   a. Scheduling Policy: "Parallel" scheduling is the default. "Sequential" scheduling impacts performance slightly while affording slightly more reliability.

   b. Mirror Policy: "On Separate Disks" is the default.

   c. Mirror Consistency: "Mirror Write Cache" is the default.

   Before changing the default settings for mirroring, be sure to review the "Guidelines for Managing Mirrors" section.

10. Activate [OK] to have SAM create the mirrored logical volume. SAM reports its activities and adds the logical to the list of logical volumes.

    Note that once the logical volume exists, you can proceed to subareas in SAM that enable you to build file systems or swap areas.

Changing the Number of Mirror Copies of a Logical Volume Using SAM

In SAM, you can change the number of mirror copies of a logical volume as follows:

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate the [Open] control button.
3. Highlight `Logical Volumes` and activate the [Open] control button.
4. Highlight the logical volume for which you want to change the number of mirror copies.
5. From "Actions" on the menu bar, choose `Change # of Mirror Copies...`.

8

6. Choose the number of mirror copies you want for this logical volume.

Note If you are increasing the number of mirror copies, there must be disk space available for the added mirrored data. The available disk space must be compatible with the allocation policies in effect for the logical volume.

a. If the logical volume has `Contiguous allocation`, then there must be a contiguous space available on the disk that will hold the added copy.

b. If the mirror allocation is strict (that is, the Mirror Policy is `Separate Disks`), you must have one disk available, each with sufficient space, for each mirror copy of the logical volume.

If the space is not available, then SAM will not succeed in increasing the number of mirror copies.

---

7. Activate [OK] to change the number of mirror copies. When SAM completes the task, it will list the logical volume, reflecting the new number of mirror copies.

Splitting and Merging Mirrored Logical Volumes Using SAM

LVM provides you the capability to split a mirrored logical volume into two logical volumes. For example, if you have a doubly mirrored logical volume, you can split off one copy and create a new logical volume, while the formerly doubly mirrored logical volume is now singly mirrored.

The advantage of this capability is that you can split mirrored logical volumes that contain file systems, for example, to perform activities (such as backup) on one of the split-off logical volume while the other logical volume remains online for use. When you complete the activity on the split-off logical volume, you can merge the two logical volumes back into one.

Splitting a Logical Volume Using SAM

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate the (Open) control button.
3. Highlight `Logical Volumes` and activate the (Open) control button.
4. Highlight the logical volume for which you want to change the number of mirror copies.
5. From "Actions" on the menu bar, choose `Split a Mirror Copy Offline...`

   When you make this choice, SAM presents you with the logical volume you had previously highlighted. SAM indicates it will create a new logical volume with the same name, adding the suffix `b`. You can choose another suffix.

6. Activate [OK] to split the mirrored logical volume. When SAM reports its actions, activate [OK] again.
7. When SAM refreshes the list of logical volumes, you will see two logical volumes with similar names, one with the suffix assigned in the splitting operation.

   At this time, you can perform an activity on the split-off logical volume. Typically, you will check its file system for consistency, create a mount point directory for it and mount it. One way to do this is:

   a. Exit SAM to run *fsck*(1M) on the split-off logical volume.

      If the split-off logical volume was **/dev/vgsales/saleslvb**, you would enter:

      ```
      /etc/fsck /dev/vgsales/saleslvb
      ```

      See *Solving HP-UX Problems* for instructions if **fsck** should not succeed.

   b. Create a mount directory for the file system in the split-off logical volume.

      ```
      cd /
      mkdir sales_temp
      /etc/mount /dev/vgsales/saleslvb /sales_temp
      ```

   Next, you can return to SAM and do the following:

8. Back up the file system in the split-off logical volume; you can use the new mount point directory as the directory to be included in the backup. See Chapter 9, "Backing Up and Restoring Your Data."

8

Merging Split Logical Volumes

When you are ready to merge the split-off logical volume with the original logical volume, do the following:

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate the [Open] control button.
3. Highlight `Local File Systems` and activate the [Open] control button.
4. Highlight the file system that is in the split-off logical volume.
5. From "Actions" on the menu bar, highlight `Remove...`.
6. From "List" on the menu bar, highlight `Logical Volumes`.
7. From the list of logical volumes, highlight the original logical volume that you want to merge.
8. From "Actions" on the menu bar, highlight `Merge Offline Mirror Copy Back Online...`.
9. Look at the fields that describe the logical volumes you are merging. Make sure the top field contains the name of the original logical volume, and the bottom field contains the name of the split-off logical volume.
10. Activate [OK] to merge the logical volumes.
11. When the task is complete, the original logical volume will be listed with its original number of mirror copies.

Modifying Mirrored Logical Volumes Using SAM

You can use SAM to modify certain characteristics of the mirrored logical volumes you have set up. With SAM, you can:

- Increase or decrease the number of mirror copies of a logical volume.
- Change the optional logical volume settings that pertain to mirroring, namely:
  - Scheduling Policy.
  - Mirror Policy (Allocation).
  - Mirror consistency method.

To change logical volume mirrors,

1. Run SAM.
2. Highlight `Disks and File Systems->` and activate the (Open) control button.
3. Highlight `Logical Volumes` and activate the (Open) control button.
4. Highlight the logical volume you want to change.
5. From "Actions" on the menu bar, choose `Modify...`.

   Make your change and activate the (OK) control button to put the change into effect.

8

Using HP-UX Commands to Manage Mirrors

Several HP-UX commands have flags specifically designed to create and manage mirrored logical volumes. Other commands apply only to mirrored logical volumes.

Creating Mirrored Logical Volumes

If you are using MirrorDisk/UX, and are setting up mirrored logical volumes, then in addition to the standard options available for the *lvcreate*(1M) command, you can use several command options to customize your mirrors.

Also, remember that the section in this chapter, "Guidelines for Managing Mirrors," can help you make optimal decisions. There is a section devoted to setting up MirrorDisk/UX.

Options Available When Creating Mirrored Logical Volumes

-m Sets the number of mirror copies of logical volume data. Use `-m 1` for single mirroring (the original plus one copy), use `-m 2` for double mirroring (the original plus two copies).

-s Sets the mirror copy allocation policy. The default is strict allocation (`-s y`), which means that mirror copies exist on separate physical volumes. Set non-strict allocation with `-s n`.

If you are using physical volume groups, you can set allocation to be strict within groups with `-s g`.

-C Sets the contiguous allocation policy. Contiguous allocation (`-C y`) means that physical extents of a logical volume are allocated in an unbroken, ordered sequence on one physical volume (disk). Contiguous allocation applies to each mirror copy of a logical volume. By default, allocation is not contiguous.

-M Set the Mirror Write Cache operation. `-M n` sets the MWC to off; by default, MWC is on.

-c Sets the Mirror Consistency Recovery mechanism operation. `-c y` enables the Mirror Consistency Recovery mechanism; `-c n` is the default.

Example: Creating a Mirrored Logical Volume

To create a logical volume of 40 megabytes in `/dev/vg01`, singly mirrored with strict allocation, with the Mirror Write Cache off, and the Mirror Consistency Recovery mechanism enabled, type:

```
lvcreate -L 40 -m 1 -M n -c y /dev/vg01

A logical volume with name "lvol1" will be created.
Logical volume "/dev/vg01/lvol1" has been successfully created
with lv number 1.
Logical volume "/dev/vg01/lvol1" has been successfully extended.
```

The above command creates a logical volume `/dev/vg01/lvol1` since this is the first logical volume created in this volume group. The logical volume has a size 40 megabytes; if the default size of each logical extent is four megabytes, there are two sets of ten physical extents created on each of two disks in the volume group `vg01`. For this logical volume, the Mirror Write Cache is disabled and the Mirror Consistency Recovery mechanism is enabled.

Note that you must use either the `-L` (megabytes) or the `-l` (logical extents) option when creating a new mirrored logical volume; you cannot mirror a logical volume of zero size.

Increasing the Number of Mirror Copies

To mirror an unmirrored logical volume, or increase the number of mirrored copies from one to two, you can use the *lvextend*(1M) command.

You can also, if necessary, specify to which disk LVM is to allocate the new mirror copies.

Examples: Increasing the Number of Mirror Copies

- To mirror `/dev/vg3/lvol2`, which is currently unmirrored, you would type:

```
lvextend -m 1 /dev/vg3/lvol2
The newly allocated mirror is now being synchronized. This operation will
take some time. Please wait ....
Logical volume "/dev/vg3/lvol2" has been successfully extended.
```

  The `-m 1` option specifies that you want to create one mirror copy in addition to the original copy of the logical volume. After LVM creates the mirror, it synchronizes the data in the newly created physical extents with the original data. This can take some time, depending on how large the logical volume is.

- To mirror `/dev/vg3/lvol2`, which is currently unmirrored, and have LVM place the mirror copy on the specific physical volume `/dev/dsk/c4d0s2`, type:

```
lvextend -m 1 /dev/vg3/lvol2 /dev/dsk/c4d0s2
```

Example: Creating Mirrored Logical Volumes that Use Specific Disks

Let's suppose you want to create a singly mirrored logical volume of 60 megabytes for a file system. You want the logical volume to use two specific disks (physical volumes) of the same type; the volume group contains several disk types. By having the logical volume's mirrored copies on disks of the same type, you intend to optimize the file system's performance.

1. Create a logical volume, no size, no mirroring, with the name **saleslv**.

   ```
   lvcreate -n saleslv /dev/vg5
   ```

   Let's assume logical volume **/dev/vg5/saleslv** is successfully created.

2. Extend the file system by 1 extent to one of the specific disks.

   ```
   lvextend -l 1 /dev/vg5/saleslv /dev/dsk/c4d0s2
   ```

3. Then extend the logical volume again, this time specifying that a mirror copy be created on the second desired disk.

   ```
   lvextend -m 1 /dev/vg5/saleslv /dev/dsk/c5d0s2
   ```

4. Now, extend the logical volume on both physical volumes to the total number of desired extents (or a specific size in megabytes).

   ```
   lvextend -L 60 /dev/vg5/saleslv /dev/dsk/c4d0s2 /dev/dsk/c5d0s2
   ```

   This last step occurs quickly because the extents being added to the logical volume contain no data. You could have eliminated the last step by extending the logical volume to 60 megabytes in step two, but the subsequent mirroring step (step 3) would require time for resynchronization.

Note In the future, when you extend a mirrored logical volume and want to continue to use the same specific physical volumes, remember to specify those specific physical volumes when you issue the command.

Mirroring Across Separate I/O Channels - PV Groups

You can create subgroups of disks within a volume group. These subgroups are called physical volume groups, or "PV groups." (SAM does not provide the capability to create physical volume groups.)

Why Create Physical Volume Groups?

Putting disks into subgroups within a volume group provides increased mirroring protection. For example:

- You can group disks on the basis of the interface cards they are connected to and mirror data between groups. This means greater data availability in the event of an interface card failure.
- You can also group disks on the basis of which I/O bus they use. Disks mirrored across I/O busses have still greater protection because a failure of one of the busses will not necessarily make the data unavailable.

Setting Up Physical Volume Groups

To create physical volume groups, you need to:

1. Create one physical volume group as you create a volume group.
2. Extend the volume group to contain a second physical volume group.

Creating Physical Volume Groups with `vgcreate`, `vgextend`, Example:

Let's assume you have just added four disks to your system, and connected two of them (LUs 3 and 4) to one I/O card and the other two (LUs 5 and 6) to another card. Your plan is to create a large file system in a logical volume that spans two disks connected to one card, and mirror the data on the disks connected to the other card.

1. Use the `pvcreate` command to make the disks LVM disks.

```
pvcreate /dev/rdsk/c3d0s4
pvcreate /dev/rdsk/c4d0s4
pvcreate /dev/rdsk/c5d0s4
pvcreate /dev/rdsk/c6d0s4
```

2. Create the volume group `vg_accounts` with one physical volume group, `iocard1`.

```
vgcreate -g iocard1 /dev/vg_accounts /dev/dsk/c3d0s2 /dev/dsk/c4d0s2
```

3. Then, extend the volume group `vg_accounts`, adding the other physical volume group, `iocard2`.

```
vgextend -g iocard2 /dev/vg_accounts /dev/dsk/c5d0s2 /dev/dsk/c6d0s2
```

Creating Logical Volumes in Physical Volume Groups

When you create a logical volume that you want mirrored across physical volume groups (which is why the PV groups exist), you can create the logical volume with the `-s g` option. The `-s g` option specifies that mirrored copies of the logical volume must exist in separate physical volume groups.

For example, let's assume you are creating a singly mirrored logical volume, `accountslv` (24 megabytes in size), in the volume group created in the previous example.

You would enter:

```
lvcreate -m 1 -s g -n accountslv -L 24 /dev/vg_accounts
```

The mirrored copies will exist in separate PV groups.

The /etc/lvmpvg File

The physical volume group data is stored in the ASCII file called **/etc/lvmpvg**. As you create physical volume groups, LVM updates the file.

For example, for the PV groups created above, the file would contain:

```
.
VG vg_accounts
PVG iocard1
/dev/dsk/c3d0s2
/dev/dsk/c4d0s2

PVG iocard2
/dev/dsk/d5d0s2
/dev/dsk/d6d0s2
.
.
```

Because /etc/lvmpvg" is an ASCII file, you can recreate it if necessary. Refer to *lvmpvg*(4) in *HP-UX Reference Manual*.

Splitting and Merging Mirrored Logical Volumes

You can split a mirrored logical volume into two logical volumes using the *lvsplit*(1M) command.

The original logical volume retains the original name, while the split-off logical volume has the suffix `b`. If you want to assign a specific name to the split-off logical volume, you can use the `-s` option to create a custom suffix.

You merge the split pair back into one mirrored logical volume using the *lvmerge*(1M) command.

Discussion, Splitting and Merging a Mirrored Logical Volume

As an example, you can split a mirrored logical volume to back up one of the mirrored copies of a file system while leaving the other mirrored copy of the file system open for use. When you complete the backup activity, you can merge the split pair back to one mirrored logical volume.

For each mirrored logical volume that you split into two logical volumes, LVM dynamically creates a table that it uses to resynchronize the two when you merge them back again. The table reduces synchronization time. The table exists as long as neither of the split pair is extended, reduced, or split again, or until the system is rebooted. If no table exists, all data is resynchronized.

When LVM merges the split pair, it resynchronizes the data, updating the physical extents in the split-off copy based on changes made to the copy that remained in use. No resynchronization takes place if no changes occurred in either copy.

Online Backup Using `lvsplit`, `lvmerge`, Example:

Suppose you want to back up the file system `/lptest` in the doubly mirrored logical volume `/dev/vg2/lvol1`. Because you want to access the file system even while backing it up (that is, you do not want to unmount it), you plan to temporarily split the logical volume in two.

1. Split the logical volume. (You could first use `lvdisplay -v /dev/vg2/lvol1` to verify the number of current logical extents and allocated physical extents.)

```
lvsplit -s backup /dev/vg2/lvol1
Logical volume "/dev/vg2/lvol1backup" has been successfully created
with lv number 5.
Logical volume "/dev/vg2/lvol1" has been successfully split.
```

8

The original logical volume `/dev/vg2/lvol1` continues to be available for use; using the display commands, you can verify that it now has one fewer mirror copies of each logical extent; it would now have one mirror if it had two mirrors before you split it, and it would now have no mirrors if it had one mirror before.

The new logical volume `/dev/vg2/lvol1backup`, split off from the original, now exists.

2. Perform backup (or other file system operation) on the file system in `/dev/vg2/lvol1backup`.

   First perform file system consistency check using *fsck*(1M). Refer to the manual *Solving HP-UX Problems* if `fsck` fails to resolve inconsistencies. Then, you can mount it (refer to chapter 6, "Managing the File System") and back it up (refer to chapter 9, "Backing Up and Restoring Your Data").

3. Unmount the file system (if you had mounted it) in the split-off logical volume when you are ready to merge the split pairs.

4. Merge the split-off logical volume and the original logical volume back into one mirrored logical volume.

```
lvmerge /dev/vg2/lvol1backup /dev/vg2/lvol1
```

Caution Notice the sequence:

`lvmerge` *split-off_logical_volume original_logical_volume*

Do not reverse the order, or else the more recent data could be overwritten.

The split-off logical volume merges back with the original logical volume, now modified.

After the merge, you can verify that the logical volume's mirrored state matches the pre-split state and that the split-off logical volume no longer exists.

Reducing the Number of Mirror Copies

You can reduce the number of mirror copies that a logical volume has by using the *lvreduce*(1M) command. By default, LVM removes the most recently created copy of data. However, you can name the specific disk from which you want the mirrored copy of data removed.

(You can also use the *lvreduce*(1M) the size of a logical volume; see "Reducing the Size of a Logical Volume.")

You can use the `lvreduce` command to *un*mirror a logical volume.

Example, Reducing the Number of Mirror Copies

Suppose you have a singly mirrored logical volume that you no longer want to mirror.

```
lvreduce -m 0 /dev/vg3/lvol1
Logical volume "/dev/vg3/lvol1" has been successfully reduced.
```

Example, Reducing a Mirror Copy from a Specific Disk

You can use the *lvreduce*(1M) command to reduce the number of mirror copies and choose the disks from which you want the copies removed.

Suppose you have a file system in the logical volume `/dev/vg1/users`, which is doubly mirrored. You can remove the data from the disk `/dev/dsk/c4d0s2` and keep the file system singly mirrored.

```
lvreduce -m 1 /dev/vg1/users /dev/dsk/c4d0s2
```

8

Moving a Logical Volume Mirrored Copy from One Disk to Another

You can move a logical volume's mirror copy from one disk to another by using the *pvmove*(1M) command.

For example, suppose you have a doubly mirrored logical volume (`/dev/vg01/lvol4`). One of the copies is on a disk you want to remove from the system (`/dev/dsk/c7d0s2`). There is room on another disk (`/dev/dsk/c5d0s2`) for the mirror copy. To move the copy, you could issue the following command:

```
pvmove -n /dev/vg01/lvol4 /dev/dsk/c7d0s2 /dev/dsk/c5d0s2
```

You must cite the logical volume using the `-n` flag and follow with the specification of the source disk. The first disk cited is the source disk; it must be specified.

If you do not specify a destination disk, LVM will transfer the data from the source disk to available space on other disks in the volume group, following the allocation policy you had set for the logical volume. Refer to "Reconfiguring Disks in a Volume Group" when removing the disk from the volume group.

Synchronizing a Mirrored Logical Volume

A mirrored logical volume with **stale** physical extents requires synchronization so that mirrored copies can contain identical data. Data in physical extents can become **stale** when, for various reasons, LVM cannot access the physical volume (disk power failure, for example) containing those physical extents when it attempts to update them.

Automatic Synchronization

When you activate a volume group, either at boot from the `lvmrc` script or later on manually with the *vgchange*(1M) command, LVM synchronizes the mirrored copies of all logical volumes, replacing data in physical extents marked as stale with data from non-stale extents.

LVM automatically resynchronizes mirrored data in the following cases:

- When a disk comes back online after experiencing a power failure.
- When you extend a logical volume by increasing the number of mirror copies; the newly added physical extents require synchronization.

Manual Synchronization

You can manually synchronize the data in a logical volume using the *lvsync*(1M) command and the *vgsync*(1M) command.

When you look at the status of a logical volume, using the `lvdisplay -v` command for example, you can see if the logical volume contains any stale physical extents. You can also identify which disk contains the stale physical extents.

```
lvdisplay -v /dev/vg3/lvol1

--- Logical volumes ---
LV Name                 /dev/vg3/lvol1
VG Name                 /dev/vg3
LV Permission           read/write
LV Status               available stale
Mirror copies           1
Consistency Recovery    MWC
Schedule                parallel
Current LE              3
```

```
Allocated PE          6
Bad block             on
Allocation            strict

   --- Distribution of logical volume ---
   PV Name            LE on PV  PE on PV
   /dev/dsk/c3d0s2    3         3
   /dev/dsk/c4d0s2    3         3

   --- Logical extents ---
   LE   PV1                PE1  Status 1 PV2                PE2  Status 2
   0000 /dev/dsk/c3d0s2    0000 current /dev/dsk/c4d0s2     0000 stale
   0001 /dev/dsk/c3d0s2    0001 current /dev/dsk/c4d0s2     0001 stale
   0002 /dev/dsk/c3d0s2    0002 current /dev/dsk/c4d0s2     0002 stale
```

Example: Synchronizing a Logical Volume

Using the *lvsync*(1M) command, you can synchronize the data.

```
lvsync /dev/vg3/lvol1
```

Synchronization takes time; how much time depends on the amount of data and the hardware involved. When you issue the command, you must wait for synchronization to complete; synchronization is complete when the prompt returns.

Synchronizing All Mirrored Logical Volumes in a Volume Group

If necessary, you can synchronize all mirrored logical volumes having stale extents in a volume group using the *vgsync*(1M) command.

Working with Volume Groups

Once you have set up volume groups, there are certain tasks that you might need to perform on volume groups, such as:

- Deactivating volume groups to perform certain maintenance tasks.
- Changing the hardware locations of the disks that make up volume groups.
- Adding or removing the disks in volume groups to and from other systems.
- Recovering from the loss of the `/etc/lvmtab` configuration file.

Setting Volume Group Availability

A newly created volume group is active and available. You use the *vgchange*(1M) command to make a volume group temporarily unavailable (that is, deactivate it) and to make it available (activate it) again.

You might want to deactivate a volume group, for example, to perform maintenance on a disk that is part of that volume group.

Deactivating Volume Groups

To temporarily deactivate a volume group:

1. Unmount any file systems in any logical volumes in the volume group. See Chapter 6, "Managing the File System" for a discussion of using the *umount*(1M) command.
2. Deactivate the volume group using the *vgchange*(1M) command.

At shutdown, volume groups are automatically deactivated.

Example: Deactivating a Volume Group

8

Deactivate a volume group `/dev/vg02`. For example, the volume group contains one logical volume `/dev/vg02/lvol1` with a file system, `/georgeb`.

1. Unmount the file system.

```
umount /georgeb
```

2. Make the volume group unavailable.

```
vgchange -a n /dev/vg02
```

Activating Volume Groups

A volume group might need to be activated if it was not automatically activated at boot up, or if you have purposely deactivated it.

If a volume group is unavailable, attempts to use it might result in a message such as:

```
lvdisplay: Couldn't query logical volume "/dev/vg2/lvol1":
Volume group not activated.
lvdisplay: Cannot display logical volume "/dev/vg2/lvol1".
```

To activate an unavailable volume volume group, `/dev/vg2`, for example, use:

```
vgchange -a y /dev/vg2
```

Reactivating an Active Volume Group

You might also reactivate an already active volume group to activate disks that were unavailable, either at boot time or when the volume group was first activated, but which are now available.

Suppose a disk, which had not been turned on when the system started, is now turned on and online. Running `vgchange -a y` causes the status of all disks to be checked. The changed status of the disk will be entered in the kernel data structures and the disk and its data will now be available.

If MirrorDisk/UX is present on your system, reactivating a volume group causes LVM to perform any necessary mirror consistency recovery activities, depending on whether the Mirror Write Cache or Mirror Consistency Recovery mechanisms are enabled on any logical volumes. After any mirror consistency activities, LVM resynchronizes the physical extents, unless otherwise specified.

Editing the `lvmrc` File for Automatic Volume Group Activation

Typically, all volume groups listed in the `lvmtab` file made available when you boot your system via the `vgchange -a y` command in the `lvmrc` file.

By default, `/etc/lvmrc` activates all volume groups and synchronizes the extents for all volume groups in parallel. You can customize the `/etc/lvmrc` file to:

- specify synchronization of extents one volume group at a time (this takes a longer time for synchronization but has less effect on I/O performance).
- select certain volume groups to activate.

The file `/etc/lvmrc` contains within it examples for making edits. Be careful when editing this file.

Using Options When Activating Volume Group

You can also activate volume groups for special purposes and under specific circumstances. You can use other flags along with `vgchange -a y`:

- To activate a volume group but prevent any opens of its logical volumes, use the `-l` flag. The volume group will be "available" but cannot be used; this is useful for maintenance activities. You can deactivate and then reactivate the volume group to resume normal logical volume use.
- To activate a volume group on the condition that all physical volumes it contains are available, use the `-p` flag. If any one physical volume is unavailable, the attempt to activate the volume group will fail.
- To activate a volume group and bypass the quorum requirement, use the `-q n` flag. (The quorum requirement sets the minimum number of a volume group's physical volumes that must be available for activation; see the section, "Recovering From Disk Failures in a Volume Group.") This action might be necessary to make data available after a severe system crash.

Caution Use of this option might result in loss of user data.

- To activate a volume group without synchronizing mirrored physical extents, use the `-s` flag (applies only to volume groups with mirrored logical volumes). This option allows fast availability of a volume group, but leaves mirrored physical extents unsynchronized.

Reconfiguring Disks in a Volume Group

There are occasions when you might need to do the following:

- Move the disks in a volume group to different hardware locations on a system.
- Move entire volume groups of LVM disks from one system to another.

When you do either of the above tasks, the LVM configuration file, `/etc/lvmtab`, must change to reflect the new hardware locations and device files for the disks.

The file `/etc/lvmtab` maintains information about all LVM disks in the system on a volume group basis. You cannot edit this file directly, since it is not a text file. Instead, you can use the *vgexport*(1M) and the *vgimport*(1M) commands to reconfigure the volume groups and record the configuration changes in the `/etc/lvmtab` file.

Moving a Volume Group's Disks

The procedure for moving the disks in a volume group to different hardware locations or different systems is illustrated in the following example.

Exporting and Importing a Volume Group

Suppose you want to move the three disks in the volume group `/dev/vg_planning` to another Series 800 computer.

1. Make the volume group unavailable.

   ```
   vgchange -a n /dev/vg_planning
   ```

2. Use *vgexport*(1M) to remove the volume group information from the `/etc/lvmtab` file. You can first preview the actions of `vgexport` with the `-p` option.

   ```
   vgexport -p -v -m plan_map vg_planning
   ```

   With the `-m` option, you can specify a **mapfile** to hold the information that is removed from the `/etc/lvmtab` file. This file is important because it contains the names of all logical volumes in the volume group.

   Later, you can use the mapfile when you set up the volume group on the new system.

   If the preview is satisfactory, run the command without `-p`.

   ```
   vgexport -v -m plan_map vg_planning
   ```

   When `vgexport` runs, it removes the device files for the disks in the volume group, removes the `vg_planning` information from the `/etc/lvmtab` file, and creates the mapfile `plan_map`.

   Once the `/etc/lvmtab` file no longer has the `vg_planning` volume group configured, you can shutdown the system, disconnect the disks, and reinstall the disks on the new system. Transfer the file `plan_map` to the `/` directory on the new system.

3. Add the disks to the new system.

   Once you have the disks installed on the new system, note their new LU numbers so you can refer to the device files created for them. Suppose, for our example, the new LU numbers are 6, 7, and 8.

4. On the new system, create a new volume group directory and **group** file. These steps are required for an imported volume group just as they are required for creating a new volume group (see "Creating a Volume Group" earlier in this chapter).

```
cd /
mkdir dev/vg_planning
cd dev/vg_planning
```

When you create the **group** file, specify a minor number that reflects the volume group number. (Volume group numbering starts at 0; the volume group number for the fifth volume group, for example, is 04.)

```
mknod /dev/vg_planning/group c 64 0x040000
```

5. Now, issue the *vgimport* command. To preview, use the **-p** option.

```
vgimport -p -v -m plan_map /dev/vg_planning /dev/dsk/c6d0s2 /dev/dsk/c7d0s2 /dev/dsk/c8d0s2
```

To actually import the volume group, re-issue the command omitting the **-p**.

6. Finally, activate the newly imported volume group:

```
vgchange -a y /dev/vg_planning
```

8

Backing Up and Restoring the Volume Group Configuration

Periodically (at least once per day), you should use the *vgcfgbackup*(1M) command to record a volume group's configuration.

This is important because, unlike fixed disk sections that begin and end at known locations on a given disk, each volume group configuration is unique, frequently changes, and uses the space on several disks.

With the LVM volume group information backed up, the corrupted or lost LVM configuration on a given disk can be restored in the event of a disk failure.

When to Use vgcfgbackup

Use `vgcfgbackup` periodically (at least once per day) and whenever you make *any* change to the LVM configuration including:

- Add or remove disks to a volume group.
- Change the boot disks in a volume group.
- Create or remove logical volumes.
- Extend or reduce logical volumes.

When You Might Need to Use vgcfgrestore

It will be necessary to use `vgcfgrestore` anytime the LVM configuration information is destroyed by the accidental or incorrect use of commands such as `newfs` or `dd`, which can destroy valuable data.

Running `vgcfgbackup`

With all disks in the volume group online, run:

```
vgcfgbackup /dev/vgsales
```

This command saves the configuration of volume group `/dev/vgsales` to the file `/etc/lvmconf/vgsales.conf`.

You can use the `-u` option to update the configuration information to reflect the addition of new disks or disks that were unavailable during the original `vgcfgbackup`: (as long as new disks are online, other disks in the volume group can be offline):

```
vgcfgbackup -u /dev/vgsales
```

Running `vgcfgrestore`

If a disk has corrupt LVM configuration data, you can use the *vgcfgrestore*(1M) to restore it. Before running `vgcfgrestore`, make the volume group unavailable:

```
vgchange -a n /dev/sales
```

For example, to restore LVM configuration data for `/dev/dsk/c4d0s2`, a disk in the volume group `/dev/vgsales`, enter:

```
vgcfgrestore -n /dev/vgsales /dev/dsk/c4d0s2
```

This restores the LVM configuration to the disk from the backup in `/etc/lvmconf/vgsales.conf`.

Listing Backed Up Configuration Information

You can list the contents of the backed up configuration file the `-l` option; enter:

```
vgcfgrestore -n /dev/vgsales -l
```

Restoring Configuration Information, Changed Device File

You can also use `vgcfgrestore` with the `-o` option. The `-o` (meaning "old") can be used to restore the LVM configuration data to a disk whose device file has changed since the last backup.

For example,

```
vgcfgrestore -n /dev/vgsales -o /dev/dsk/c4d0s2 /dev/dsk/c5d0s2
```

Recovering an lvmtab File - Using vgscan

The file `/etc/lvmtab` lists all volume groups and the physical volumes that are matched to each volume group. During boot, for example, the disks and volume groups are activated based on the contents of the `/etc/lvmtab` file. If the `/etc/lvmtab` file should be inadvertently destroyed or corrupted, the proper activation of volume groups and disks cannot occur.

You might know that you have destroyed the `/etc/lvmtab` file, but it might have been corrupted inadvertently. Clues that the `/etc/lvmtab` file has been corrupted or is missing might include messages to the effect that a volume group or disk that you know is present cannot be found; or, you might run a display command, such as `/etc/vgdisplay`, and LVM reports that it is creating an `/etc/lvmtab` file, when one has always existed.

You can use *vgscan*(1M) command to recreate the `/etc/lvmtab` file. When the command is run, it scans each disk in the system, looking for logical volume and volume group information. It also scans the `/dev` directory, looking for matches between the volume group device files, the logical volume device files, and the disks.

It's a good idea to run `vgscan` as follows:

- Run `vgscan` as soon as you realize the `/etc/lvmtab` has been damaged or destroyed. Remove a corrupt `/etc/lvmtab` file before recreating it with `vgscan`.
- Run `vgscan` with the system running, before you reboot.
- Make sure all disks on your system are online.
- Run `vgscan` with the `-p` and `-v` options first. The `-p` option allows you to preview the new `/etc/lvmtab` file but does not create it. The `-v` option lets you view any message `vgscan` generates during the preview.

If `vgscan` finds disks that it cannot match with a volume group, or finds volume groups for which disks cannot be found, it instructs you to run `vgimport` to record the correct volume group and disk information in the `/etc/lvmtab` file. (See "Reconfiguring Disks in a Volume Group" for examples and discussion about running `vgimport`.)

Recovering From Disk Failures in a Volume Group

Each LVM disk within a volume group contains identical information about the configuration of the volume group that it belongs to (in reserved volume group descriptor and status areas). LVM uses this information to verify that the configuration is current and sufficient for the volume group to operate.

LVM Quorum

LVM requires a "quorum" of disks in a volume group be available. At boot time, LVM requires a quorum of one more than one-half of the disks in the volume group, which means, for example, that LVM cannot activate a two-disk volume group with one of the disks offline.

At runtime, LVM requires a quorum of half of the disks in a volume group for activation. If, during runtime, a disk fails and causes quorum to be lost, LVM alerts you with a message to the console, but keeps the volume group active.

Recovering Quorum

Operating without a quorum presents several limitations. For example, the Mirror Write Cache does not function, certain logical volumes might become inaccessible, and changes you make to the configuration do not become recorded on the failed disks.

There are various causes for the loss of quorum. Procedures for recovering an active volume group will vary. Refer to the manual *Solving HP-UX Problems*. That manual presents methods you can use to regain an active volume group.

The Root Volume Group

The root volume group is a special volume group for the following reasons:

- It is the volume group that contains the **system boot disks**; typically the boot area is at a specific region on the same disk (physical volume) that contains the root logical volume.

  One part of the boot area, the Boot Data Reserve Area (BDRA), contains information about:

  - all bootable disks in the volume group.
  - the root logical volume and its mirror copies, if any.
  - the primary swap logical volume and its mirror copies, if any.
  - the dumps logical volume(s).
  - all disks in the volume group.

- The root volume group contains the **root file system (/) logical volume.**

  The root logical volume, vital because it contains the operating system software, has certain restrictions that do not apply to regular logical volumes.

  The root logical volume:

  - must be located at a specific location on a bootable disk so the root file system can always be found on the disk containing it, even if the information in the BDRA is not accessible or if the LVM data is corrupt.
  - must be created with contiguous allocation of physical extents.
  - cannot support software bad block relocation.
  - must be identified in the BDRA (using `lvlnboot -r`).
  - cannot be increased (or decreased) in size because unmounting **/** is impossible (a requirement of extending a file system in a logical volume is to unmount it).

- It contains the **primary swap logical volume**. The primary swap logical volume can be located on any disk in the volume group.

  Like the root logical volume, the primary swap logical volume:

  - must be created with contiguous allocation of physical extents.
  - must be identified in the BDRA (using `lvlnboot -s`).
  - cannot be easily extended and cannot be reduced.

- The root volume group can contain **dumps logical volumes**. All logical volumes used for dumps can reside in the root volume group only. Although a dumps device is optional (but highly recommended by HP), you can create a dumps logical volume(s) in the root volume group. It is possible (desirable when a possible core dump is expected to be very large) to have more than one dumps logical volume. The following constraints apply to dumps logical volumes; they:
  - must be created with contiguous allocation of physical extents.
  - must be identified in the BDRA (using `lvlnboot -d`).

*The root volume group also must contain the file systems that contain `/usr` and `/tmp`.

The `/usr` and `/tmp` directories, if they are not included in the `/` file system logical volume, must exist in logical volume(s) within the root volume group. Logical volumes created for `/usr` and `/tmp` need not be on a bootable disk or contiguous.

Run lvlnboot When Making Changes to the Root Volume Group

You must run the *lvlnboot*(1M) command anytime you do the following:

- Add a disk to the root volume group (using `vgextend`).
- Remove a disk from the root volume group (using `vgreduce`).
- Make changes to *any logical volume for* `root`*, primary swap, or dumps*, such as:
  - Mirror or increase the number of mirrors of any logical volume in the root volume group.
  - Increase the size of a logical volume.
  - Decrease the size of a logical volume.
  - Make changes to the characteristics of a logical volume.
  - Move a logical volume.
  - Split a mirrored logical volume and plan to reboot before merging the split pair.
- Change the location of the root logical volume.
- Change the location of the primary swap.

Note Changes you make to the root, swap, or dump logical volumes will not take effect until you reboot the system.

Running lvlnboot -v to Show Current Root Volume Group

Running:

```
lvlnboot -v
```

shows the disks contained in the root volume group, lists the boot disks, and shows the locations of the root, swap, and dumps logical volumes.

Running lvlnboot When Adding Non-Root Disks to the Root Volume Group

Whenever you add or remove disks from the root volume group, you need to update the BDRA(s) on disks within the root logical volume. Do this after adding or removing the disk by using the *vgextend*(1M) or *vgreduce*(1M) commands by running the `lvlnboot -r` on the root logical volume.

Note If you add or remove the disk to the root volume group using SAM, SAM will automatically update the BDRA.

Suppose you add a disk, `/dev/dsk/c4d0s2`, to the root volume group (`vgroot`) and the root logical volume is `rootlv`.

```
vgextend /dev/vgroot /dev/dsk/c4d0s2
lvlnboot -r -v rootlv /dev/vgroot
```

The `lvlnboot -r` command updates the BDRA(s) throughout the root volume group that another disk has been added to the volume group.

Note If you do not run the `lvlnboot` command when adding or removing a disk from the root volume group, even when the change does not involve a root or primary swap logical volume, the change will not be reflected in the BDRA of the boot disks. This could possibly lead to a loss of quorum and make the system unbootable without manual intervention.

Creating Boot Disks in the Root Volume Group

You must specify any boot disks before you include them in a volume group. Since the existing root volume group already has a boot disk, the procedure for creating another boot disk is appropriate for adding a disk that will be used to:

- mirror the `root` logical volume, or
- contain a new root logical volume

Briefly, the steps to add a boot disk to the root volume group are:

1. Use the `-B` option with the `pvcreate` command when you make the disk an LVM disk. `-B` creates an area for boot utilities.
2. Add the disk to a volume group with `vgextend`.
3. Use `mkboot` to place boot utilities in the boot area.
4. Use `mkboot -a` to add an `AUTO` file in boot LIF area.

In the following sections, example procedures demonstrate these steps. See "Example: Mirroring the Root File System Logical Volume" and "Moving Root from a Disk Section to a Logical Volume".

Making Root, Swap, and Dump Logical Volumes

When you create logical volumes that you intend to use for root, swap, or dump, you usually want to place the logical volume on a specific disk (physical volume). SAM does not provide the capability to create a logical volume on a specific disk, so you must use the *lvcreate*(1M) and *lvextend*(1M) commands.

Creating Root Logical Volumes

The disk intended for the root logical volume must be a bootable disk. You need to make the disk an LVM disk with the `pvcreate -B` command and add it to the root volume group. You must have added the boot utilities with the `mkboot` command. (See the example, "Example: Mirroring the Root File System Logical Volume", which shows the sequence of commands to make a bootable LVM disk.)

Also, the `root` logical volume must be the first logical volume found on a bootable LVM disk; that is, the root logical volume must begin at physical extent `0000`. This is important in the event it is necessary to boot the system in maintenance mode; the root file system's location is known. This means the disk that will contain a root logical volume or a mirror copy of it should not have data in the region following the boot area.

You can use the *pvmove*(1M) command to move an existing logical volume to another disk, if it's necessary to make room for `root`.

Creating a New Logical Volume for Root

It might be necessary to create a new root logical volume in the event the current logical volume containing root is too small. The root logical volume cannot be extended.

When you create the root logical volume, remember that the root logical volume must be created with contiguous extents (`-C y`) and bad block relocation disabled (`-r n`). For example, to create a logical volume of 160 megabytes for `root` in the volume group `/dev/vgroot`, enter:

```
lvcreate -C y -r n -n root /dev/vgroot
lvextend -L 160 /dev/vgroot/root /dev/dsk/c4d0s2
```

8

Once the `root` logical volume is created, you can move the data into it. See the example procedure "Moving Root from a Disk Section to a Logical Volume".

Mirroring an Existing Root Logical Volume

When you mirror the root logical volume, you will be extending a copy of the existing root logical volume to a specific disk that has contiguous space available immediately following a boot area created with `pvcreate -B`.

For example, to mirror `/dev/vgroot/rootlv` to the disk `/dev/dsk/c4d0s2`, enter:

```
lvextend -m 1 /dev/vgroot/rootlv /dev/dsk/c4d0s2
```

Updating Boot Information in the BDRA, Root Logical Volumes

When you create or mirror a root logical volume, you need to update the boot information on the disks in the volume group. Use the *lvlnboot*(1M) command with the `-r` option.

For example, if you have set up `root` in `/dev/vgroot/rootlv`, you would enter:

```
lvlnboot -r rootlv /dev/vgroot
```

You can use the `-v` option to have LVM report its activities.

Note Any time you make a change to the root logical volume, whether to add or reduce mirrors, to change any of the mirror characteristics, and so on, you must update the BDRA by running the `lvlnboot -r` command.

If you no longer want to use a logical volume as the root logical volume, use *lvrmboot*(1M) to update the boot information on the disks in the volume group. For example:

```
lvrmboot -r charoot /dev/vgroup
```

Note Using `lvrmboot -r` also removes the BDRA information entirely from *all* boot disks in the volume group, thereby removing primary swap and dump logical volume definitions as well.

Primary Swap Logical Volumes

The primary swap logical volume does not have to be on the same disk as the `root` logical volume, nor does it need to be on a bootable disk. It does need to be part of the root volume group.

Creating a New Logical Volume for Primary Swap

The primary swap logical volume needs to be contiguous. For example:

```
lvcreate -C y -L 48 -n pswap /dev/vgroot
```

As long as you create a contiguous logical volume for primary swap, you don't need to designate a specific disk.

Mirroring an Existing Primary Swap Logical Volume

Extend the existing primary swap logical volume with *lvextend*(1M) and use the `-m option`. Also, because mirror consistency is undesirable with swap, you must use `-M n` and `-c n` to disable the Mirror Write Cache and mirror consistency recovery mechanisms.

Let's assume you want to extend primary swap to a specific disk (physical volume) `/dev/dsk/c4d0s2`.

```
lvextend -m 1 -M n -c n /dev/vgroot/pswap /dev/dsk/c4d0s2
```

If contiguous space doesn't exist on the disk, the attempt to mirror will fail.

Updating Boot Information in the BDRA, Primary Swap Logical Volumes

When you create or mirror a primary swap logical volume, you need to update the boot information on the disks in the volume group. Use the *lvlnboot*(1M) command with the `-s` option.

For example, if you have set up primary swap in `/dev/vgroot/pswaplv`, you would enter:

```
lvlnboot -s pswaplv /dev/vgroot
```

Note Any time you make a change to the primary swap logical volume, whether to add or reduce mirrors, to change any of the mirror characteristics, and so on, you must update the BDRA by running the `lvlnboot -s` command.

If you no longer want to use a logical volume as the primary swap logical volume, use *lvrmboot*(1M) to update the boot information on the disks in the volume group. For example:

```
lvrmboot -s swaplv /dev/vgroot
```

Creating Dumps Logical Volumes

A dumps logical volume can exist on any disk in the volume group (but only in the root volume group), and do not need to be on a boot disk. Mulitiple dumps devices (up to 32) can exist on any system. The primary swap area can also serve as a dumps area.

Creating a New Logical Volume for Dumps

The dumps logical volumes needs to be contiguous. For example, to create two dumps logical volumes named `dumps1` and `dumps2`, each 20 megabytes in size, in the volume group `/dev/vgroot`.

```
lvcreate -C y -L 20 -n dumps1 /dev/vgroot
lvcreate -C y -L 20 -n dumps2 /dev/vgroot
```

Updating Boot Information in the BDRA, Dumps Logical Volumes

When you create a dumps logical volume, you need to update the boot information on the disks in the volume group. Use the *lvlnboot*(1M) command with the `-d` option.

For example, if you have set up dumps in `/dev/vgroot/dumps1` and `/dev/vgroot/dumps2`, you would enter:

```
lvlnboot -d dumps1 /dev/vgroot
lvlnboot -d dumps2 /dev/vgroot
```

Note Any time you make a change to the dumps logical volumes, whether to change any of the logical volume characteristics, and so on, you must update the BDRA by running the `lvlnboot -d` command.

If you no longer want to use a logical volume as a dumps logical volume, use *lvrmboot*(1M) to update the boot information on the disks in the volume group. For example:

```
lvrmboot -d dumps1 /dev/vgroup
```

Using primary swap for dumps. It's possible to use the primary swap logical volume as a dumps area. In fact, primary swap will be used as dumps by default. (This choice conserves disk space but can slow the process of recovering dumps data; see the discussion "Setting Up Dumps Areas" in chapter 7.)

If the primary swap logical volume is `/dev/vgroot/pswap`, you would enter:

```
lvlnboot -d pswap /dev/vgroot
```

You can also use:

```
lvrmboot -d pswap /dev/vgroot
```

to discontinue the use of the primary swap logical volume for dumps, but continue using it for primary swap.

Example: Mirroring the Root File System Logical Volume

Since `root` and any mirror copies of `root` must begin at physical extent `0000` on a bootable disk, `root` needs to be mirrored on a either new disk or on a disk with the space created for it. The following example shows the addition of a disk to the volume group for the purpose of mirroring `root`.

The example also includes creating a mirror copy of the primary swap logical volume, since that is a logical companion step to the mirroring of `root`. Primary swap does not need to be on a specific disk or at a specific location, but it does need to be contiguous. The recommended mirror policy for primary swap is to have the Mirror Write Cache and the Mirror Consistency Recovery mechanisms disabled.

1. Create another bootable LVM disk.

   ```
   pvcreate -B /dev/rdsk/c3d0s2
   ```

   Note the **r** in `rdsk`; use the disk's character (raw) device file. The `-B` option creates a region for a LIF volume and boot utilities, and creates a BDRA (Boot Data Reserved Area) on this disk.

2. Include this disk in the current root volume group.

   ```
   vgextend /dev/vg00 /dev/dsk/c3d0s2
   ```

   Note `/dsk`; this is the disk's block device file.

3. Now, make the disk a boot disk by adding boot utilities:

   ```
   mkboot /dev/rdsk/c3d0s2
   ```

   For LVM disks, the *mkboot*(1) command requires the character device file.

4. Copy a correct `AUTO` file into the new LIF area. Do this as follows:

   ```
   mkboot -a "hpux (52.3.0;2)/hp-ux" /dev/rdsk/c3d0s2
   ```

   The "52.3.0" is the hardware path of the new disk; it varies, depending on the system type and location of the disk. The "2" following the semi-colon is the disk section number that represents the whole disk; "hp-ux" is the name of the kernel that will be booted.

   Note the use of the disk's character device file.

5. Now, mirror the root logical volume to the new bootable disk.

   ```
   lvextend -m 1 /dev/vg00/lvol1 /dev/dsk/c3d0s2
   ```

   Note the use of the block device files.

6. Mirror the primary swap logical volume.

   ```
   lvextend -m 1 /dev/vg00/prswaplv /dev/dsk/c3d0s2
   ```

   Note the use of the block device files. The contiguous mirror copy of primary swap logical volume will be placed on the added disk.

7. Now, update the boot information contained in the BDRA (Boot Data Reserved Area) of the boot disks in the root volume group with the locations of the mirror copies of `root` and primary swap.

   ```
   lvlnboot -r lvol1 /dev/vg00
   lvlnboot -s prswaplv /dev/vg00
   ```

   Also, you can use `-v` option to have LVM report its activities.

Now that you have created mirror copies of the root logical volume and the primary swap logical volume, should either of the disks fail, the system can use the copy of `root` or of primary swap on the other disk and continue.

When the failed disk comes back online, it will be automatically recovered, provided the system has not been rebooted.

If the system is rebooted before the disk is back online, you will need to reactivate the disk and update the LVM data structures that track the disks within the volume group. You can use `vgchange -a y` even though the volume group is already active.

For example, you can reactivate the disk using the following:

8

```
vgchange -a y vg01
```

At this command, LVM scans and activates all available disks in the volume group, including the disk that came online after the system rebooted.

Moving Root from a Disk Section to a Logical Volume

The following procedure describes making an available disk an LVM root disk. The procedure is appropriate in the following cases:

- The root file system is currently in a fixed disk section and you want to move it to a logical volume so that you can mirror root with MirrorDisk/UX.
- You want to move the root file system from one logical volume to a larger logical volume (the root logical volume cannot be extended).

To use this procedure, you must have:

- LVM currently configured in the system's kernel.
- An available empty disk you can use as an LVM disk.

Note If you do not have a second disk available, you will need to "migrate" `root` to LVM. Migration involves backing up the data in the `root` disk section and all other sections of the root disk, installing a new Release 9.0 HP-UX operating system, creating logical volumes on the disk, and then restoring the backed up data. Migration procedures are described in chapter 3 of the manual *Installing and Updating HP-UX*.

This procedure requires you to be thoroughly familiar with several commands, some of which are discussed in detail earlier in this chapter. Read the procedure first. Do not attempt the procedure if you aren't confident you can do it.

Example: Creating a Root Logical Volume on a New Disk

This procedure is described through the use of an example. In the example, the new disk that is being made a bootable LVM disk is a SCSI HP2213A, connected to the SCSI interface card in slot 13, and assigned bus address 3. The hardware address is 52.3.0 and the LU for the new disk is 2. We will assume we have added the disk using the procedures described in *Installing Peripherals*.

Also, note in this example that a new primary swap logical volume is also being created because, typically, primary swap also resides on the root disk. The primary swap area is also being configured for dumps. Doing these activities in parallel is an efficient way to set up a new root disk on logical volumes.

1. Perform the following procedure in single-user mode. After making sure users are not on the system, put the system into single-user mode by entering:

   ```
   shutdown
   ```

2. Run the command: `/etc/ioscan -f -C disk` so you can determine the correct device files.

   The list generated enables you to identify the hardware addresses and LU numbers of all disks on the system.

   With the LU number, you can determine the device file corresponding to the disk that you want make an LVM boot disk. For example, if the LU is 2, then the character device file for that disk is: `/dev/rdsk/c2d0s2`.

3. Make the new disk a bootable physical volume.

   For example,

   ```
   pvcreate -B /dev/rdsk/c2d0s2
   ```

   The `pvcreate` command sets up a disk to be an LVM disk. The `-B` option ensures the disk will have room for a LIF area to hold boot utilities and a BDRA (Boot Data Reserved Area).

4. Create a new directory and control file for the new root volume group, which we call `vgroot`. (This step is not necessary in the case of moving root from one logical volume to another because the volume group already exists).

```
mkdir /dev/vgroot
mknod /dev/vgroot/group  c 64 0x010000
```

When you create the control file for `/dev/vgroot`, the minor number you provide must be unique. The method for using HP-UX commands to create a volume group is described earlier in the section "Creating Volume Groups Using HP-UX Commands".

5. Create the new volume group and add the disk to it (not required if we are moving root from logical volume to logical volume).

```
vgcreate /dev/vgroot /dev/dsk/c2d0s2
```

We now have one volume group, `vgroot`, with one physical volume, `/dev/dsk/c2d0s2`.

6. Or, extend the existing root volume group an add the disk to it (not applicable if you are moving root from a disk section).

If we are moving root from a logical volume to another logical volume, we would extend the existing volume group as follows:

```
vgextend /dev/vgroot /dev/dsk/c2d0s2
```

7. Put the boot utilities on the disk using *mkboot*(1M).

```
mkboot /dev/rdsk/c2d0s2
```

The boot utilities go in the LIF area previously created by the "`pvcreate -B`" command.

8. Add the `AUTO` file in the boot area with the `-a` option of the `mkboot` command.

```
mkboot -a "hpux (52.3.0;2)/hp-ux" /dev/rdsk/c2d0s2
```

This command creates the file in the boot area that tells the boot loader on which device and by what name to look for the kernel.

9. Create the new logical volume that will contain `root`.

```
lvcreate -C y -n root /dev/vgroot
lvextend -L 200 /dev/vgroot/root /dev/dsk/c2d0s2
```

The first command creates a contiguous logical volume (`-C y`) named "`root`" (`-n root`) in the root volume group, `/dev/vgroot`. The `-r n` option turns off bad-block relocation. The logical volume actually has no size yet, but its name and characteristics are defined. As LVM creates the logical volume, it will report its name. LVM also creates the block and character device files for the logical volume; for example, it would create `/dev/vgroot/root` and `/dev/vgroot/rroot`.

The second command extends the `root` logical volume by 200 megabytes (`-L 200`) to the disk (`/dev/dsk/c2d0s2`) you added. By default, each extent is equivalent to four megabytes, so the size of the logical volume is also 50 logical extents. If you require a different size for the root file system you plan to move, you can use an appropriate value.

10. If appropriate, create a new logical volume for primary swap at this time, also.

```
lvcreate -C y -L 64 -r n -n swap /dev/vgroot
```

This command creates a contiguous logical volume of 64 megabytes. Assume, for this example, it created the logical volume `swap` (`-n swap`).

11. Update the information in the BDRA to reflect that `root` and primary swap will use these logical volumes. Also, define `dumps` (`lvlnboot -d`) to use the primary swap logical volume.

```
lvlnboot -r root /dev/vgroot
lvlnboot -s swap /dev/vgroot
lvlnboot -d swap /dev/vgroot
```

8

12. Create a file system in the logical volume `/dev/vgroot/root`.

```
newfs /dev/vgroot/rroot hp2213A
```

Note that you need to supply the character device file (`rroot`) for the logical volume.

13. Create a mount directory for the new file system.

```
mkdir /newroot
```

14. Mount the new file system.

```
mount /dev/vgroot/root /newroot
```

15. Move to / and copy the root file system to the /newroot.

```
cd /
find . -xdev -depth -print | cpio -pxdm /newroot
```

This command copies the file system in /, the current directory, to /newroot which is in the new logical volume. The -xdev option prevents file systems at other mount points from being copied, but it does create the mount points.

16. Modify the /newroot/etc/checklist file.

Modify the /newroot/etc/checklist file. For example, edit the file /newroot/etc/checklist to have /dev/vgroot/root mounted at / and /dev/vgroot/swap set up as swap. The following lines are examples for use in /newroot/etc/checklist.

```
/dev/vgroot/root /     hfs  rw,suid 0 1
/dev/vgroot/swap /swap ignore
.
.
```

8

Build a New Kernel, with Root in a Logical Volume

You now need to build a new kernel to specify the new location of `root` in a logical volume.

1. Move to the directory: `/etc/conf/gen`.
2. Make a new `uxgen` input file and edit it.

   ```
   cp S800 S800.BCKUP
   vi S800.BCKUP
   ```

3. In the input file, change the line that begins "`root on`" to:

   ```
   root on lvol;
   ```

4. For primary swap, change the line that begins "`swap on`" to:

   ```
   swap on lvol;
   ```

5. For dumps, change the line that begins "`dumps on`" to:

   ```
   dumps on default;
   ```

   This specifies that primary swap is now also to be in a logical volume. See chapter Chapter 7, "Managing Swap Space", if you want to see more information about configuring primary swap.

6. Build a new kernel using `uxgen`.

   ```
   uxgen S800.BCKUP
   ```

   This creates the kernel file with the pathname `/etc/conf/gen/S800.BCKUP/hp-ux`.

7. Copy the new `uxgen` input file and put it in place.

   ```
   cp /etc/conf/gen/S800.BCKUP /newroot/etc/conf/gen/S800
   ```

8. Copy the new kernel into the `/newroot` directory.

   ```
   cp /etc/conf/S800.BCKUP/hp-ux /newroot/hp-ux
   ```

Reboot and Try the New Configuration

1. Enter:

   ```
   /etc/reboot
   ```

2. Interrupt the automatic boot process. When you see:

   ```
   Autoboot from primary path enabled.
   To override, press any key within 10 seconds.
   ```

   press a key. The ISL utility in section 6 does not know that you want to boot `hp-ux` from `/newroot`.

3. Don't boot from primary boot path. Answer `n` to the question:

   ```
   Boot from primary boot path (Y or N)?> n
   ```

4. Don't boot from the alternate boot path. Answer `n` to the question:

   ```
   Boot from alternate boot path (Y or N)> n
   ```

5. Specify the boot path and Interact with IPL. For our example, at the prompt, type:

   ```
   Enter boot path, command, or ?: 52.3.0
   ```

   Then, when you get the message:

   ```
   Interact with IPL?>
   ```

   Answer "n".

   The kernel (`hp-ux`) now boots from the new root logical volume on the new disk at the hardware address specified. The old root file system resides on the original disk in the original section.

6. If the system does not come up correctly, reboot without interrupting the automatic boot. This brings up the old kernel and the root file system.

After Successfully Moving Root

After successfully booting from root on the new logical volume, do the following to complete the process of moving `root`.

Change the Primary Boot Path in ISL

To reboot automatically from the new root file system at its new hardware address, change the `PRIMPATH` variable in the initial system loader ISL (see *isl*(1M) in *HP-UX Reference Manual*).

1. Reboot.

   ```
   /etc/reboot
   ```

2. Interact with ISL by pressing a key when you see:

   ```
   Autoboot from primary path enabled.
   To override, press any key within 10 seconds.
   ```

3. Answer "n" to the questions concerning booting from the primary boot path and alternate boot path, and enter the hardware path to the new root disk at the next prompt:

   ```
   Enter boot path, command, or ?> 52.3.0
   ```

4. At the prompt, interact with IPL, specify the new PRIMPATH variable, and boot:

   ```
   Interact with IPL?> y

   ISL> primpath 52.3.0
   ISL> hpux (;2)/hpux
   ```

8

Create Configuration File and Perform Critical Backups

The following backup operations are extremely important and should be done as soon as possible after you have moved `root` to a new logical volume.

Caution Do not omit the following operations. They are intended to keep your configuration and data safe and recoverable in case of accidental disk failure.

Create the Volume Group Configuration File

With the *vgcfgbackup*(1M) command, create a binary backup file that contains information about the configuration of the new volume group.

Issue the `vgcfgbackup` command while in the `/` directory.

```
cd /
vgcfgbackup /dev/vgroot
```

The `vgcfgbackup` command saves the configuration of the volume group in the file: `/etc/lvmconf/vgroot.conf`. You should also use *vgcfgbackup*(1M) to keep the configuration current. See "Backing Up and Restoring the Volume Group Configuration" earlier in this chapter.

Back Up the New Root File System

The new root file system has not yet been backed up. You can back up the new `/` file system using a graph file based on the listing in the `/etc/checklist` file, including `/` and excluding the other mounted file systems. Refer to Chapter 9, "Backing Up and Restoring Your Data" for information about using *fbackup*(1M).

Transfer Other Data from the Root Disk to Logical Volumes

After moving `root` and primary swap to logical volumes, you can move other data on the former root disk, whether in the form of file systems or raw data, to logical volumes on another disk by migrating them.

You can do this at your leisure, using either SAM or HP-UX commands to create the new logical volumes, and then moving the data either by copying it or by backing it up and restoring it.

Note It is best to be in single-user mode when performing operations that involve moving data from device to device. Being in single-user mode avoids data loss.

LVM Kernel Configuration Requirements

The LVM subsystem requires the following entries in the kernel configuration file, `/etc/conf/gen/S800`:

- An "include" statement for the LVM subsystem:

  ```
  include lvm;
  ```

- An "include" statement for the LVM pseudo driver, `lv`

  ```
  include lv;
  ```

LVM-Related Tunable Operating System Parameters

- maxvgs

  By default, you can configure a maximum of ten volume groups. You can reset the *maxvgs* tunable operating system parameter. The number of volume groups cannot exceed 256.

- no_lvm_disks

  If you are *not* running LVM, you can set the *no_lvm_disks* operating system parameter to 1, which indicates to the system that there are no LVM disks. This speeds up the boot process very slightly by allowing the system to skip checking disks for LVM usage.

  By default (*no_lvm_disks* value = 0), the system checks root, swap, and dump disks for LVM use.

Configuring Kernel Devices, LVM

Together, the root file system, primary swap, a dumps device are the key kernel devices located on disk. The S800 file, used to create the kernel along with LVM disk data structures, is where these devices are specified.

These devices can exist in logical volumes or in disk sections. It is also possible that on one system, some of the devices can be in logical volumes and others in disk sections. For example, you could have / and primary swap in logical volumes, and dumps in standard disk sections.

More discussions about primary swap device and dumps devices are contained in chapter 6, "Managing Swap Space."

Configuring Root, Swap, and Dump Logical Volumes in Kernel

The volume group that contains the root file system logical volume is the **root volume group**. The root volume group also contains the primary swap logical volume and any dumps logical volumes. The previous section, "The Root Volume Group," has detailed discussions of the root, primary swap, and dumps logical volumes. The root volume group is created at HP-UX installation time.

By default, the kernel devices portion of the S800 file would contain:

```
root on lvol;
swap on lvol;
dumps on default;
```

The keyword "lvol" indicates that the kernel must use the information about the root volume group from the BDRA (the Boot Data Reserved Area) on the boot disk. With this information, the kernel can activate the root volume group. It will mount the root file system, enable the primary swap logical volume, and set up the dump device to accept the system core dumps.

8

Because the BDRA stores the information that the kernel requires, you must modify the information in the BDRA when you make any changes to the root volume group, including adding or removing disks, or changing the root, swap, or dumps logical volumes. You can use the *lvlnboot*(1M) and *lvrmboot*(1M) commands to do this.

Mixed LVM and Non-LVM Kernel Devices

You can have `root` in a logical volume, and have swap and dumps in standard disk sections. The exception is that if root is not in a logical volume, then neither can other kernel devices be in logical volumes.

For example, the `S800` file could contain:

```
root on lvol;
swap on cio_ca0.hpfl0.disc2 at 8.6.3 section 7;
dumps on default;
```

The above shows `root` in a logical volume, primary swap in a traditional section, and dumps that also uses the primary swap device.

Another example,

```
root on lvol;
swap on lvol;
dumps on cio_ca0.hpfl0.disc2 at 8.6.4 section 7;
```

Where `root` and swap are in logical volumes and dumps is in a traditional disk section. More discussion of the primary swap and dumps can be found in chapter 7, "Managing Swap Space."

Non-LVM Root, Swap, and Dumps in Kernel

It is possible to use LVM on your system and configure kernel devices in traditional disk sections rather than in logical volumes. In this case, the kernel devices are specified in the `S800` file, either explicitly, or by the term `default`. For example,

```
root on default;
swap on default;
dumps on default;
```

These lines specify that the system determines these devices from the kernel loaded from the `autoboot` file contained in the `LIF` volume on the boot disk.

Migrating Non-Root Disks From Traditional Sections to Logical Volumes

If you currently have data in traditional disk sections and you want to move that data into logical volumes, then you must "migrate" the data. It's best to migrate disks to logical volumes on a disk-by-disk basis.

Briefly, migration entails the following:

- Planning the eventual layout of logical volume on your disks.
- Backing up the data that currently resides in the disk sections to tape. The sections could contain either file systems or raw data.
- Making the disk an LVM disk (physical volume) and adding it to a volume group. This action destroys any data on the disk.
- Creating logical volumes on the disk (now an LVM disk) for the data you backed up to tape.
- If you are migrating file systems, create and mount new file systems in the logical volumes.
- Restoring the backed up data from tape to the newly created file system.

Note Migrating the system disk.

If you want to migrate the system disk, or the "root" disk, to LVM, you must reinstall the system. See chapter 3, "Migrating the Root Disk to Logical Volumes," in the *Installing and Updating HP-UX* manual for the procedure to migrate the root disk to LVM.

8

Migrating File Systems

You can use either SAM or HP-UX commands to perform the series of tasks involved in migrating data from traditional disk sections. We recommend you use SAM.

The LVMMIGRATE File.

If you migrated your root disk to logical volumes and used the `lvmmigrate` tool, you can find the `LVMMIGRATE` file in the directory `/local`. That file should give you a list of the disks and file systems as they existed in disk sections before the reinstallation of HP-UX 9.0.

After the installation of HP-UX 9.0, you can run the `/etc/lvmmigrate` command with the `-p` option to get a list of file systems in disk sections and recommendations for their migration to logical volumes. Look in `/tmp` for the `LVMMIGRATE` file.

Print out a copy of that file so you can refer to it as you work through the following migration procedure.

Using SAM to Migrate File Systems to Logical Volumes

If you are not familiar with how to use SAM, refer to Chapter 1, which provides an overview of interacting with SAM.

Run SAM.

- Listing the file systems on the disk.

  Suppose, for example, you want to migrate a certain file system, `/projects`. You must also migrate the other file systems on the same disk.

  In SAM, you can see the file systems on a disk by:

  1. Highlighting and activating `Disks and File Systems`.
  2. Highlighting and activating `Local File Systems`.
  3. Review the list of file systems under "Mount Directory".

     When you see the file system on the list, look under the column "Device File" and determine the device file for the file system. Assume the file system `/projects` is in the disk section with device file `/dev/dsk/c3d0s7`. Other file systems on the same disk as `/projects` will have device files that begin `/dev/dsk/c3.....` Those are the file systems on the disk with LU 3.

     Suppose you find that on disk LU 3 there is one other file system, `/graphics`, in addition to `/projects`.

- Back up the file systems on the disk you are migrating.
  1. Highlight and activate `Backup and Recovery->` .
  2. Highlight and activate `Backup Devices` . Then select the backup device.
  3. Indicate that you want to interactively back up files. From the `Select Backup Scope...` menu, indicate you want the selected files `/graphics` and `/projects` included in the backup.
  4. Making sure the tape is loaded on the backup device, proceed to back up the file systems.
- Now that the file systems are safely backed up on tape, remove the file systems in the disk sections and make the disk an LVM disk.
  1. From SAM's main menu, highlight and activate `Disks and File Systems` .
  2. Highlight and activate `Local File Systems` .
  3. Highlight a file system (`/projects`, for example) and choose `Remove` from "Actions" on the menu bar. Repeat this for each file system you are migrating.
  4. From "List" on the menu bar, highlight `CD-ROM and Hard Disks` .
  5. From "Actions" on the menu bar, choose `Add a Hard Disk Drive` and then choose `Using the Logical Volume Manager` .
  6. Choose the disk from the list displayed.
  7. Highlight the `Volume Group Name.....` control button, if you want to add the disk to an existing volume group, and select the volume group from the displayed list.

     If you are creating a new volume group with the migrating disk, type the name in the field to the right of `Volume Group Name.....` .
  8. Skip the `Optional Volume Group Settings.....` unless you want to change the defaults. You can review them.
  9. Activate the [OK] control button to add the disk to your volume group.

8

- Create the logical volume for the file system.

  1. From "List" on the menu bar, choose `Logical Volumes`.
  2. When the list of existing logical volumes is displayed, choose `Create` from "Actions" on the menu bar.
  3. Now, choose a volume group from the list of volume groups displayed. Choose the volume group that contains your migrating disk.
  4. Enter a name for the logical volume you want to create, `proj_lv` for example, or skip this field to accept a default name.
  5. Enter the size of the logical volume. The recommended value listed in the `/local/LVMMIGRATE` file (under the heading "Recom. LV Size in MB") can guide you here. The logical volume should be large enough to hold the data you had in the disk section file system.
  6. Skip the `Optional Logical Volume Settings...` unless you want to change the default logical volume characteristics.
  7. Activate the [OK] control button to create the logical volume.

     The logical volume should appear on the list of logical volumes. Repeat steps 2 through 7 to make another logical volume at this time if you have another file system to migrate.

- Create the file system(s) in the logical volume.

  1. From "List" on the menu bar, choose `Local File Systems`.

  2. When the list of existing file systems is displayed, choose `Add` and then `Using the Logical Volume Manager` from "Actions" on the menu bar.

  3. In the `Mount Directory:` field, type in the name of a directory that will be the mount point of the file system. In the `/local/LVMMIGRATE` file, this is the directory under the heading "Filesystem." You could use `/projects`.

     SAM will create the directory, unless it already exists.

  4. Choose the appropriate logical volume you created from the logical volume list.

  5. Activate the `Create new file system` control button.

  6. Accept the default settings for `Set user ID`, `When to Mount`, and `Access Permissions` unless you want to change them.

  7. Activate the [OK] control button to create and mount the new file system.

     The file system should appear on the list of file systems. Repeat steps 2 through 7 to create another file system at this time if you have another file system to migrate.

- Restore backed up data into file systems.

  1. Highlight and activate `Backup and Recovery->`.

  2. Highlight and activate `Backup Devices`. Then select the backup device.

  3. Indicate that you want to recover the files in `/projects` and `/graphics` and recover them.

  You can now use the file systems in logical volumes that had been in disk sections.

Migrating Data From DataPair/800 to LVM's MirrorDisk/UX

Users who are currently using DataPair/800 to mirror disks for high data availability can use the following procedure to migrate the data to logical volumes and use LVM-based MirrorDisk/UX to mirror the data.

Also, refer to the manual *Managing Disk Mirrors Using DataPair/800*, part number 92453-90003.

Example: Migrating Two DataPair/800 Disks to LVM

With DataPair/800, you set up mirrors between like disk sections on separate disks. It's possible to have mirrored sections and unmirrored sections on a disk.

When you migrate a disk that currently contains sections mirrored with DataPair/800 to LVM, you must migrate the *entire* disk; the process of creating an LVM disk overwrites all of the data on the disk. The implication is that you must consider *all* data on a migrating disk, not just the mirrored sections.

The following example shows the case where two file systems are mirrored on two disks by DataPair/800. The goal in the example is to convert the disks to LVM disks, include them in a volume group, and set up the file systems in mirrored logical volumes.

1. Back up all of the data that you will be migrating. Use *fbackup*(1M); see Chapter 9, "Backing Up and Restoring Your Data". This is very important.

2. List the mirrors on your system. Identify the mirrored disks you want to migrate:

   ```
   mirror -l
   ```

   Each line will show information for a mirror. The line shows a pair of device files, one for each mirrored section. The left hand device file (column 1) lists the primary mirror, and the right hand device file (column 3) lists the secondary mirror. For example:

   ```
   /dev/dsk/c3d0s7   ONLINE   /dev/dsk/c4d0s7   OFFLINE  GOOD
   /dev/dsk/c3d0s8   ONLINE   /dev/dsk/c4d0s8   ONLINE   GOOD
   ```

8

3. Make sure each mirror copy has the same data.

```
mirror -r /dev/rdsk/c3d0s7 &
```

4. Verify.

```
mirror -l
.
/dev/dsk/c3d0s7   ONLINE    /dev/dsk/c4d0s7     REIMAGE  GOOD
.
```

5. The first disk you want to migrate is the secondary disk of a mirrored pair. The primary mirrored sections are on LU 3 and the secondary sections are on LU 4.

   Run *lvmmigrate*(1M). This will produce a file that you can use to determine the required sizes for the logical volumes you need to create.

```
lvmmigrate -v
```

   Look in the file `/tmp/LVMMIGRATE` that is produced by `lvmmigrate -v`.

   The file lists the device files for the disks, their hardware paths, and indicates which disk section contains a specifc file system by mount point. The listing also indicates a recommended logical volume size under the heading "Recom. LV Size in MB."

```
.
 Disk Name        LU    HW Path  Size-MB    Model  In Root VG ?
================================================================
.
/dev/dsk/c4d0s2     4     52.5.0   624       2213A      N
.
   HW Path Section   Section/   Recom. LV  Backup In Root  FileSystem
                     Size in MB  Size in MB           VG ?
=====================================================================
.
    52.5.0     7         73          16       -       N    /parts_list
    52.5.0     8        433         240       -       N    /drawings_list
.
```

8

6. Unmount all file systems accessing the disk you are migrating. Let's assume you are going to migrate the file systems `/parts_list` in section 7 and `/drawings_list` in section 8 of the mirrored disk LU 4. The primary mirrored disk is LU 3.

```
umount /dev/dsk/c4d0s7
umount /dev/dsk/c4d0s8
```

7. Make sure the mirror copies are online and current.

```
mirror -l
.
/dev/dsk/c3d0s7   ONLINE    /dev/dsk/c4d0s7    ONLINE   GOOD
/dev/dsk/c3d0s8   ONLINE    /dev/dsk/c4d0s8    ONLINE   GOOD
.
```

8. Unconfigure the mirror(s), by specifying the primary mirror disk.

```
mirror -u /dev/dsk/c3d0s7 /dev/dsk/c3d0s8
```

9. Make the migrating disk (the secondary disk) an LVM disk. Because there are file systems on the disk, you will be asked to confirm.

```
pvcreate /dev/rdsk/c4d0s2
```

10. Extend an existing volume group or create a new volume group with the disk. In our example, we will create the volume group `/dev/vendorvg`. (See "Creating Volume Groups Using HP-UX Commands.")

```
mkdir /dev/vendorvg
mknod /dev/vendorvg/group c 64 0x040000
vgcreate /dev/vendorvg /dev/dsk/c4d0s2
```

11. Create a logical volume for each of the file systems you are migrating and extend them to the new LVM disk. Suppose you want to create a logical volume for the file system `parts` and another for `drawings`.

```
lvcreate -n partslv /dev/vendorvg
lvextend -L 16 /dev/vendorvg/partslv /dev/dsk/c4d0s2
lvcreate -n drawingslv /dev/vendorvg
lvextend -L 240 /dev/vendorvg/drawingslv /dev/dsk/c4d0s2
```

12. Create the file systems. Assuming the disk is an HP 2213A (you can see this in /tmp/LVMMIGRATE file).

```
newfs /dev/vendorvg/rpartslv hp2213A
newfs /dev/vendorvg/rdrawingslv hp2213A
```

13. Mount the new file systems to the mount directories.

```
mount /dev/vendorvg/partslv /parts_list
mount /dev/vendorvg/drawingslv /drawings_list
```

14. Now, mount the file system in sections 7 and 8 of LU 3, the primary mirror disk (you unmounted them in step 3). Make temporary mount directories for them. This enables you to copy the file system data into the newly created logical volumes.

```
cd /
mkdir parts_list.old
mount /dev/dsk/c3d0s7 /parts_list.old
mkdir drawings_list.old
mount /dev/dsk/c3d0s8 /drawings_list.old
```

15. Now copy the data from the sections to the logical volumes.

```
cp -r /parts_list.old/*.* /parts_list
cp -r /drawings_list.old/*.* /drawings_list
```

16. Unmount the file systems in the sections of the primary disk and remove the temporary mount point directories.

```
umount /dev/dsk/c3d0s7
umount /dev/dsk/c3d0s8
rmdir parts_list.old
rmdir drawings_list.old
```

17. Make the migrating primary DataPair/800 disk an LVM disk. You will be asked to confirm since file systems still exist on the disk.

```
pvcreate /dev/rdsk/c3d0s2
```

18. Extend the volume group to include the disk.

```
vgextend /vg/vendorvg /dev/dsk/c3d0s2
```

19. Now, mirror the logical volumes `partslv` and `drawingslv` to the new disk.

```
lvextend -m 1 /dev/vendorvg/partslv /dev/dsk/c3d0s2
lvextend -m 1 /dev/vendorvg/drawingslv /dev/dsk/c3d0s2
```

20. Finally, update the volume group configuration to reflect the addition of the new disks and logical volumes to the volume group by running *vgcfgbackup*(1M).

```
vgcfgbackup /dev/vendorvg
```

21. Update the `/etc/checklist` file to reflect the new device files for the file systems now in logical volumes.

9

Backing Up and Restoring Your Data

Of all the tasks that system administrators perform, among the most important are creating **system backups**. The most effective way to insure against the loss of your system's data is to copy the data from your system onto storage media (such as magnetic tape, cartridge tape, optical disk, or another hard disk). You should store copies of your data *away from your system* so that you can recover the data should something happen to your primary data. Data can also be backed up over a network to a computer at a different location. The important thing is to have copies of all your important files *somewhere other than on your system*.

This chapter will include the following topics and tasks:

- Terms used in this chapter
- Determining which data to back up
- Determining how often to back up data
- Choosing the type of storage device to use
- Backing up your data
  - using SAM.
  - using HP-UX commands.
- Automating your backup process
  - using SAM.
  - using HP-UX commands.
- Restoring data
  - using SAM.
  - using HP-UX commands.
- Other backup and restore utilities

The most commonly used backup commands for backing up and restoring your data are `fbackup` and `frecover`. SAM also uses these commands. This chapter primarily focuses on SAM and the use of the `fbackup` and `frecover` commands. Other back up and restore commands are listed in "Other Backup and Restore Utilities".

Note For backing up and restoring your logical volume configuration, refer to Chapter 8, "Managing Logical Volumes".

Terms Used in this Chapter

The following terms appear in this chapter. This list of terms appears here so that you can easily locate it. You can scan (or skip over) the list now and, if you need to, refer to it later.

backup levels A mechanism used to distinguish between varying degrees of incremental backup. Each backup level has a date associated with it that indicates when the last backup at that level was created. These dates are stored in the backup database file (`/usr/adm/fbackupfiles/dates` unless the `-d` option to `fbackup` is used to specify a different file). See "Full Backups vs. Incremental Backups".

excluded files A way of specifying *exceptions* in a group of files to be included in a backup (see **included files** below). For example, if you want to back up all of the files in a directory except for one, you can include the directory in your backup and indicate that you want to *exclude* that file.

extraction The process of recovering *specific files* from a group of files in a backup, as opposed to recovering *all* of the files from the backup.

full backup A backup that includes *all* of the files in a specified set, *regardless of the date they were created, modified or previously backed up.*

graph file	A file that contains instructions about which files and directories to include in a backup, and of those, which files and directories to exclude from a backup.
included files	A list of files to be included in a backup.
incremental backups	A backup that includes only those files *that have been modified* since a previous backup. Backups can be "layered" (for example monthly backups, weekly backups, daily backups) using different backup levels (see **backup levels**, above).
local device	A backup device that is physically connected to the same computer as the data you are backing up or restoring. The other type of a device is a **remote device**.
recovery	The process of recovering *all* the files from a backup, as opposed to specific files (see **extraction**, above).
remote device	A backup device that is physically connected to the a another computer. The other type of a device is a **local device**.
system backups	The process of storing copies of important files on your system in a safe location not on your system so that you can restore the files if the primary copies on your system are lost or corrupted.

Determining Which Data to Back Up

To determine which data to back up, consider what files you need to recover if your file system becomes corrupt or your disk drive fails. Returning your system to the state just prior to the problem is determined by the information contained on your backups. If your backups include all of the files, you will be able to restore your system. You can only restore your system files as of the date of the most recent backup. Frequency of performing a backup is discussed in "Determining How Often to Back Up Data". If you prepare for the worst case, a total system failure and data corruption, you will be prepared for minor difficulties.

Note

- If you are running LVM, you must maintain the backup configuration files for each volume group. After making any changes to the configuration of the disks or the logical volumes within a given volume group, you must run the *vgcfgbackup*(1M) command. The availability of the backup file in the `root` file system makes it possible to easily recover the volume group configuration after a system crash or after the inadvertent use of destructive commands such as `newfs` or `dd`.
- The `/etc/lvmtab` file records the configuration of all volume groups and LVM disks on the system and is needed to enable the activation volume groups at system startup or later. In case `/etc/lvmtab` is destroyed, you can use the *vgscan*(1M) command to recreate it.

See Chapter 8, "Managing Logical Volumes", for a discussion of recreating the `/etc/lvmtab` file using `vgscan`, and a description (and examples) of running `vgcfgbackup` and `vgcfgrestore`, the command used to restore the configuration.

When you back up your system, you must define which files in your directory tree you want to back up. As you read, consider the following directory tree:

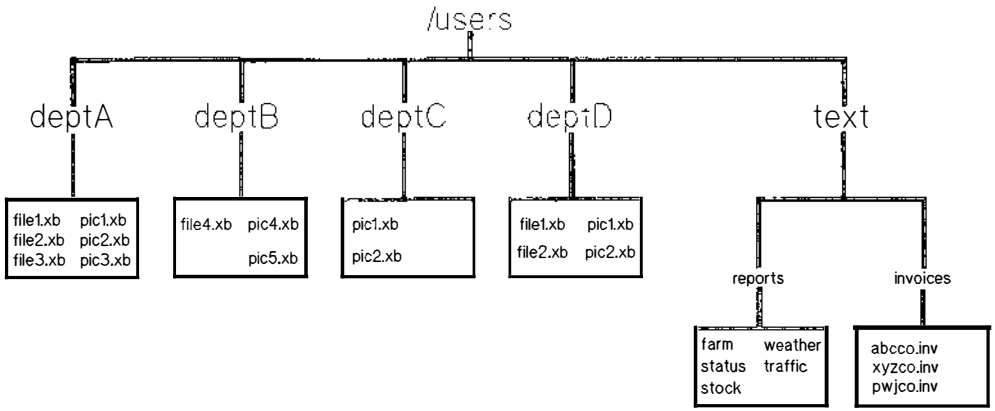


Figure 9-1. An Example Directory Tree

Included Files

Included files are directories and files that you want to include in your backup. When you specify a directory, all of the files and subdirectories are included in the backup.

Suppose you need to back up the files for Departments A, B and C. You explicitly specify that you want the following three directories backed up:

- `/users/deptA`
- `/users/deptB`
- `/users/deptC`

Included files are defined with the `-i` option to the `fbackup` and `frecover` commands, or with a graph file. See "Graph Files".

Excluded Files

Excluded files are those files *within your list of included directories and files* that you want to exclude from the backup. In other words, they are the exceptions.

Suppose that you have 100 departments and you need to back up the files for all of the departments *except* one. Entering in each department's directory name would be a lot of work.

In the example from the previous section on "included files," we listed out each directory from the example directory tree *except* for Department D's directory. You can tell the `fbackup` and `frecover` utilities to exclude a subset of the files from a group of files that are to be included in a backup. Another way to accomplish this task is to tell `fbackup` or `frecover` to:

```
INCLUDE:  /users

EXCLUDE:  /users/deptD
```

Excluded files are defined with the `-e` option to the `fbackup` and `frecover` commands, or with a graph file. See "Graph Files".

Graph Files

A file containing a list of directories and files to back up is a **graph file**. A graph file determines the scope of the backup. It is common to use different graph files for full and incremental backup schedules. If you use SAM to back up your system, you do not need to create graph files. SAM creates a graph file for you when you specify which files to back up.

When you run the `fbackup` and `frecover` commands, you can specify which files to include by using the `-i` option, and you can specify which files to exclude by using the `-e` option. More than one `-i` option and more than one `-e` option can be specified; however, if you have a long list of entries to specify, this can be a tedious and error prone way of entering the command.

It is possible to enter the information about which files to include/exclude in a file and then tell `fbackup` and `frecover` to look in that file for the list of files to back up or restore. Below are examples of graph files based on the examples in the previous sections "Included Files" and "Excluded Files".

Here is an example of a graph file for the list of files in the previous section, "Included Files":

```
i /users/deptA
i /users/deptB
i /users/deptC
```

Here is an example of a graph file for the example in the previous section, "Excluded Files":

```
i /users
e /users/deptD
```

Graph files contain one entry per line. If an entry begins with the two characters `"i "` (i,space), the files or directories represented on that line are *i*ncluded in the backup (or restoration). If an entry begins with the two characters `"e "` (e,space), the files or directories represented on that line are *e*xcluded from the backup (or restoration).

To tell `fbackup` or `frecover` to use a graph file, use the `-g` option followed by the name of the graph file.

Determining How Often to Back Up Data

A question closely related to determining *which* data to back up is determining *how often* to back it up. The critical question is, "*how much data can you afford to lose*?"

Evaluate the applications running on your system and the needs of your users to determine how critical the data on your system is to them. This will give you a guideline as to how often to back up the various files on your system. Consider the following things when determining how frequently to back up a particular file (or type of file):

- How often do the contents of the file change?
- How critical is it that the file's contents be up to date.

Note You can back up different groups of files at different frequencies and you can perform **incremental backups**, that back up *only those files which have changed (or are new) since the last time you performed a backup*. Incremental backups will be discussed later in this chapter.

You should create a backup schedule for your system that describes how often you will perform full backups and incremental backups of the various files on your system.

It is best to back up your system when there are few or no users logged in. Files that are in use when the `fbackup` or `frecover` command encounters them will not be backed up or restored. If you can afford to do so, you should change your system's run-level to the system administration state (single-user mode) before initiating the backup procedure. This will ensure that you are the only one logged in. For information about changing run-levels, see Chapter 4, "Controlling Access to the System".

Full Backups vs. Incremental Backups

Once you have identified the list of files to *include* and *exclude* you need to decide whether you want *all of the files* represented by your list to be backed up (a **full backup**), or *only those files that have changed (or that are new) since the last time you backed up this set of files* (an **incremental backup**).

Note A full backup does *not* mean a backup of every file on your system. It means a backup of every file on your "include list", regardless of when it was last backed up.

Backup Levels

The previous section stated that an incremental backup is a backup that includes *only those files that have changed (or that are new) since the last time you backed up this set of files*. This brings up the question, "how does `fbackup` know when the previous backup was created?" This information is contained in the file `/usr/adm/fbackupfiles/dates`, a file that is updated only when *all* of the following conditions are true:

- The "`-u`" option is used with `fbackup`.
- A graph file is used to indicate which files should be included/excluded when a backup is performed.
- Neither the "`-i`" nor the "`-e`" option is used (graph file used instead)
- The backup completed successfully

Note The `fbackup` command considers different graph files as separate backups. Backing up `graph_file_A` will have no effect on an incremental backup of `graph_file_B`.

Backup levels are a way of specifying varying degrees of incremental backup. For example, suppose you wanted to set up the following backup schedule:

- On the first day of the month, back up an entire set of selected files.
- Every Friday, back up all files in the selected set that have changed since the first of the month.
- Every day except Friday, back up all of the files in the selected set that have changed since the last Friday or first of the month, which ever is most recent.

There are three levels associated with the above schedule (the once per month level, the once per week level, and the once per day level). The once per month level is a *full backup*. The other two are *incremental backups*. The problem is how to distinguish between the two types of incremental backups. This is accomplished with backup levels.

The file `/usr/adm/fbackupfiles/dates` contains information about when the last backup at each backup level was performed. The `dates` file contains:

- the graph file used for the backup
- the level of the backup
- the date of the backup
- the start and end time for the backup

This information is used by `fbackup` to determine which files defined in the graph file are included in the backup. The `fbackup` command uses the following search sequence on the `dates` file to determine the base backup on which to build an incremental backup:

1. matching graph file
2. next lowest level number
3. most recent date

If no lower level is found, a full backup at the specified level is performed. If there are duplicates of a lower level found, the most recent is used as the base for the incremental backup.

You can have up to ten backup levels (0 - 9). Your backup strategy varies based on the level of activity on your system and the capacity of your media.

Recovery Example Using Three Backup Levels

To implement the earlier example of monthly, weekly, and daily backups use following backup levels:

level 0 - full monthly backup

level 1 - weekly backup on Friday

level 2 - daily backup, except Friday

Figure 9-2 illustrates the level numbers for implementing this example.

```
Date of the month: 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15    ... 1
Day of the month: Su  M  T  W Th Fr Sa Su  M  T  W Th  F Sa Su    ...
-----------------------------------------------------------------------
backup level       0  2  2  2  2  1  2  2  2  2  2  2  1  2  2    ... 0
```

Figure 9-2. Example of Using Incremental Backups

If your data became corrupt on Thursday the 12th, on Friday the 13th you would follow the following sequence to restore your system to it's Wednesday the 11th state:

1. Restore the monthly full backup from Sunday the 1st.
2. Restore the weekly incremental backup from Friday the 6th.
3. Restore the incremental backup from Wednesday the 11th.

Recovery Example Using Two Backup Levels

The following example illustrates a weekly full backup and daily incremental backup, two backup levels. When implementing your backup strategy using SAM, only two levels of backups are supported. Figure 9-3 illustrates the level numbers supported by SAM:

```
Date of the month: 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15    ... 1
Day of the month: Su  M  T  W Th Fr Sa Su  M  T  W Th  F Sa Su    ...
-------------------------------------------------------------------------
backup level       0  1  1  1  1  1  1  0  1  1  1  1  1  1  0    ... 0
```

Figure 9-3. Example of Using Incremental Backups Supported by SAM

If your data became corrupt on Thursday the 12th, on Friday the 13th you would follow the following sequence to restore your system to it's Wednesday the 11th state:

1. Restore the full backup from Sunday the 8th.
2. Restore the incremental backup from Wednesday the 11th.

Choosing the Type of Storage Device to Use

Once you have determined which files you need to back up, and how often (see the previous two sections for details on how to do this), you need to determine where to copy your files for backup purposes.

When you evaluate where to back up your data, consider the following:

- How much data do you need to back up (rough estimate)?
- How quickly will you need to retrieve the data when you need to?
- What types of storage devices do you have access to?
- How automated do you want the process to be (for example, will you have an operator change tapes when they fill up or will you be using an automatic changing device such as the 35401A cartridge tape changer or one of the optical disk library devices)?
- How quickly will you need to complete a backup?

All backup utilities require some overhead on the media for data structures. The amount of raw data contained on the media is always less than the total capacity of the media.

Use your answers to the above questions along with the following table to help you determine which storage device to use for your backups.

Table 9-1. Determining Which Device to Use for a Backup

Device Type	Holds Lots of Data	Recover Data Quickly	Backup Data Quickly	Unattended Backup?[1]
Reel to Reel Magnetic Tape	Good	Good	Good	No
HP-format Cartridge Tape Single Drive	Fair	Poor	Poor	No
HP Format Cartridge Tape Autochanger[2]	Excellent	Poor	Poor	Yes
QIC Format[3] Cartridge Tape	Fair	Good	Good	No
DDS Format (DAT) Tape Drive	Excellent	Good	Good	No
Optical Disk Single Drive	Good	Good	Good	No
Optical Disk Multi-disk Library	Excellent	Good	Good	Yes
Hard Disk	Good	Excellent	Excellent	No
Flexible (Floppy) Disk	Poor	Fair	Fair	No
DDS Format or Magnetic Tape over the Network[4]	*see specific device type*	*see specific device type*	*see specific device type*	*see specific device type*

1 You can perform an unattended (automatic) backup for the devices marked "no" in this column IF all of the data will fit on one reel, cartridge, floppy disk etc.

2 Autochanger must be set to "selective" mode (not "sequential" mode) for automatic backup.

3 QIC is the abbreviation of Quarter Inch Cartridge

4 Only magnetic tape and DDS format (DAT) tape drives are supported for remote backups.

When you are planning your backup strategy, it is also necessary to know how much data each media type will hold. Here are several tables containing approximate capacities for commonly used storage media.

Table 9-2. Disk Drive Storage Capacities

Media Type	Storage Capacity (Megabytes)	Comments
Optical Disk	325/side	An optical disk has two surfaces, each side has a capacity of 325 megabytes. See Table 9-6 for disk section capacities per surface. For optical disk library devices, multiply 650 megabytes by the number of disks installed in your device. SAM only supports using section 2.
Hard Disk	⟹	Hard disk capacities vary depending on the model you have.[1]

1 Consult the documentation that came with your disk drive for capacity information.

Table 9-3. Magnetic Tape Storage Capacities

Tape Density (bits/inch)	600 foot Reel[1]	1200 foot Reel[1]	2400 foot Reel[1]
800 BPI	5 Mbytes	10 Mbytes	20 Mbytes
1600 BPI	10 Mbytes	20 Mbytes	40 Mbytes
6250 BPI	40 Mbytes	80 Mbytes	140 Mbytes
HP7980/XC Compression Mode[2]	105 Mbytes	210 Mbytes	420 Mbytes

1 Capacities listed are approximations. Actual storage capacities vary with the number of errors encountered on each tape, the number of inter-record gaps on the tape and variations in tape length.

2 The Model HP7980/XC tape drive has a special data compression capability that allows more data to be stored on a given length of tape. The amount of data compression that occurs is data dependent. These values are averages.

Table 9-4. HP Format Cartridge Tape Storage Capacities

Number of Tracks on Tape[1]	150 foot Tape[2]	600 foot Tape[2]
16-Track Tape	16 Mbytes	67 Mbytes
32-Track Tape	32 Mbytes	133 Mbytes

1 Cartridge tapes come in 16-track and 32-track styles. A device (such as a Model 9145 Cartridge Tape drive) that is capable of handling 32-track tapes can read but not write 16-track tapes. 16-track devices cannot use 32-track tapes.

2 Capacities listed are approximations. Actual storage capacities vary with the number of errors encountered on each tape. Capacities are listed in megabytes.

Table 9-5. DDS Format (DAT) Tape Storage Capacities

	60 meter tape (Mbytes)[1]	90 meter tape (Mbytes)[1]
uncompressed mode	1300 (1.3 Gigabytes)	2000 (2 Gigabytes)
compressed mode	1300 to 5200 (1.3 to 5.2 Gigabytes)	2000 to 8000 (2 to 8 Gigabytes)

1 Capacities listed are approximations. Actual storage capacities vary with the type of data and the number of errors encountered on each tape.

Note

- Do not mix compressed and uncompressed data on the same tape.
- Half-height (3.5 inch) tape drives can read and write to 90 meter and 60 meters tapes.
- Full-height (5.25 inch) tape drives can only read and write to 60 meter tapes. They cannot read or write to 90 meter tapes.

Table 9-6. Optical Disk Surface Section Storage Capacities

Section Number	1024 bytes/sector Capacity (Mbytes)	512 bytes/sector Capacity (Mbytes)	Section Number	1024 bytes/sector Capacity (Mbytes)	512 bytes/sector Capacity (Mbytes)
0	23	23	7	42	41
1	16	16	8	265	239
2	307	281	9	246	220
3	19	19	12	305	279
4	37	37	13	289	263
5	209	184	14	23	23
6	1	1	15	16	16

Note

- These capacities are based on the entries in the `/etc/disktab` file. The calculations are rounded down to the nearest whole number. For exact calculations, divide the block size for the section by 1024. Refer to *disktab*(4) of the *HP-UX Reference Manual* manual and the Chapter 6, "Managing the File System" chapter of this manual for additional information.
- SAM only supports using section 2 (the entire surface) for backing up and recovering data.

Table 9-7. QIC-Format Cartridge Tape Storage Capacities

Media Type	Maximum Capacity (Mbytes)	Supported Write Formats	Supported Read Formats
DC6526	525[1]	QIC-525/320 QIC-150 QIC-120	QIC-525/320 QIC-150 QIC-120 QIC-24
DC6320	320[1]	QIC-525/320 QIC-150 QIC-120	QIC-525/320 QIC-150 QIC-120 QIC-24
DC6250	250[2]	QIC-150 QIC-120	QIC-150 QIC-120 QIC-24
DC6150	150[2]	QIC-150 QIC-120	QIC-150 QIC-120 QIC-24
DC600XTD	150[2]	QIC-150 QIC-120	QIC-150 QIC-120 QIC-24
DC600A	125[3]	QIC-120	QIC-120 QIC-24
DC300XL/P	60[4]	not supported	QIC-24

1 Capacity based on QIC-525 format.

2 Capacity based on QIC-150 format.

3 Capacity based on QIC-120 format.

4 Capacity based on QIC-24 format.

Note The `fbackup` and `frecover` commands only supports QIC-525 format. For any other QIC format use the `cpio` or `tar` utilities.

Automating the Backup Process

Because of the need to have the activity on the system as low as possible while files are being backed up, many system administrators create their backups during the middle of the night. It is therefore desirable to have a way to automate the process so that the backup can occur unattended. This eliminates the need to have someone manually start the backup in the middle of the night.

For backups to be truly unattended:

- All of the files must fit on one media unit, or
- Several backup devices must be specified, or
- You must use an autochanging device such as an HP35401 cartridge tape changer or an optical disk library device.

If none of the above are true, the backup cannot be automated.

Note SAM does not support multiple devices per automated backup entry. See "Setting Up an Automated Backup Schedule Using SAM".

Restoring Your Data

The primary reason for making backup copies of your data is so you can restore needed files that have been removed (or damaged).

There are two types of situations you are likely to encounter:

1. One or a few files need to be recovered, usually as a result of an accidental deletion or because the file has been overwritten by the wrong data.
2. You need to recover *all* of the files. This is usually part of the system crash recovery process. If you have experienced a file system failure and you suspect that you have corrupt data, refer to Chapter 6, "File System Problems" of the *Solving HP-UX Problems* manual. If your root disk failed and all the data on the disk is lost, you need to re-install HP-UX, refer to the *Installing and Updating HP-UX* manual.

   After you have repaired the file system or replaced the hardware, you can restore your data from your most recent backups.

   Ensure that your system can access the device from which you will restore the files from your backup. This can involve adding a disk or tape drive to your system, refer to the *Installing Peripherals* manual.

If your backup was created by the `fbackup` command (SAM uses the `fbackup` command), you can use SAM or the `frecover` command to restore the files from your backup.

The `frecover` command has several modes of operation and it is important to know the difference between them. The process of recovering *all* files from a backup is called **recovery**. The mode of operation for restoring *individual files* is called **extraction**.

Most of the concepts associated with backing up your system also apply to restoring files. Particularly, "including files", "excluding files", "graph files", and "device files." Refer to "Determining Which Data to Back Up" and "Choosing the Type of Storage Device to Use" for more information.

Backing Up and Restoring Tasks

There are two methods for performing backups and restoring data:

1. SAM (System Administration Manager)
2. HP-UX commands

Generally you should use the SAM method because it is simpler and faster than performing the task with commands. For information about running SAM and navigating within SAM, refer to Chapter 1, "Introduction to System Administration".

SAM allows you to control access to your system through its menu-selection and data-entry screens. By combining multiple "manual commands" into single tasks, SAM can save you time and keystrokes. SAM also eliminates the need to know command names and options.

Although HP-UX commands require you to learn more details than SAM does, you might need or prefer to use HP-UX commands, for the following reasons:

- HP-UX commands give you a greater degree of control.
- SAM might not be configured into your system. You *have* to use HP-UX commands.
- You might be more comfortable using HP-UX commands.

HP-UX has *many* utilities that can be used for creating backups. Each utility has its advantages and disadvantages. The pair of utilities that contain the best mix of features and that give you the greatest flexibility are `fbackup(1M)` and `frecover(1M)`.

The System Administration Manager (SAM) uses the `fbackup` and `frecover` commands to back up your system.

When backing up and restoring files that are NFS mounted to your system, `fbackup` and `frecover` can only backup and restore those files having "other user" read permission. `fbackup` and `frecover` normally operate in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `fbackup` and `frecover` can back up and restore the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

Backing Up Your System To a Local Device Using SAM

Note To backup files on disks physically connected to another computer, enter the `Remote Administration` functional area of SAM and refer to Chapter 1, "Introduction to System Administration", for additional information.

Gather a list of files you want to back up (see "Determining Which Data to Back Up") before you begin.

SAM only supports one level of incremental backups.

If you are setting up your system for performing an unattended backup in the future, see "Setting Up an Automated Backup Schedule Using SAM".

To back up your system right now:

1. Ensure that files you intend to back up are not being accessed.

Note The `fbackup` command will not back up files that are active (open) or locked when it encounters them.

2. Ensure that you have superuser capabilities.
3. Run the `fsck` command to check your file system for inconsistencies before you perform a backup. This ensures that you to do not back up data that is corrupt. Refer to Appendix A, "Using the fsck Command" in the *Solving HP-UX Problems* manual and *fsck*(1M) in the *HP-UX Reference Manual*.

Caution Do not run `fsck` on a file system that is mounted and active. This could introduce data corruption. Run `fsck` in single-user mode when checking the root file system. For file systems other that the root file system, unmount the file system, run `fsck`, and then remount the file system.

4. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the [Help] button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the [f1] key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.

5. Highlight **Backup and Recovery** and activate the [Open] control button.
6. Highlight **Backup Devices** and activate the [Open] control button.
7. Highlight the backup device from the object list.
8. Choose **Backup Files Interactively** from the "Actions" menu.

   SAM may ask you to provide additional information about:

   - magnetic tape density.
   - DDS format tape with data compression or without data compression.
   - optical disk number in an autochanger library system.
   - HP format cartridge tape number to start with in an autochanger.
9. Activate [Select Backup Scope].
10. Turn on the **Entire System** or **Selected Files** checkbox.

    If you turned on the **Entire System** checkbox, activate the [OK] control button.

    If you turned on the **Selected Files** checkbox, enter each file name in the "Included" and "Excluded" boxes and activate the [Add] control button. If you make a mistake, highlight the entry with the error and use the [Modify] or [Remove] control buttons to correct the mistake.

    When you have completed determining the selecting files to be recovered, activate the [OK] control button.

11. To:

    - cancel creating an index log or
    - backup data that is NFS mounted to your system,

    activate [Set Additional Parameters...] and turn off and on the appropriate checkboxes.

    Creating an on-line index log is performed by default and is very useful for tracking problems if your backup did not complete successfully. The log files are `/usr/sam/log/br_index.full` for full backups and `/usr/sam/log/br_index.incr` for incremental backups.

    When backing up files that are NFS mounted to your system, `fbackup` can only back up those files having "other user" read permission. `fbackup` normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `fbackup` can back up the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

    When you have finished setting the additional parameters, activate the [OK] control button.

12. Activate the [OK] control button to begin the backup process.

    If confirmation messages appear, read the message(s) and activate the [OK] control button to proceed in each case. SAM displays a window containing the output of the executed `fbackup` command. If you created an index log, the information displayed will appear in the index log.

Backing Up Your System To a Remote Device Using SAM

Gather the following information before you begin:

- A list of files you want to back up (see "Determining Which Data to Back Up").
- The name of the system to which the backup device is attached.
- The device file, on the remote system, for the backup device.

Note

- To backup files on disks physically connected to another computer, enter the `Remote Administration` functional area of SAM and refer to Chapter 1, "Introduction to System Administration", for additional information.
- SAM only supports backing up remotely to magnetic and DDS format tapes.

SAM only supports one level of incremental backup.

If you are setting up your system for performing an unattended backup in the future, see "Setting Up an Automated Backup Schedule Using SAM".

To back up your system right now:

1. Ensure that files you intend to back up are not being accessed.

Note

The `fbackup` command will not back up files that are active (open) or locked when it encounters them.

---

2. Ensure that you have superuser capabilities.

3. Run the `fsck` command to check your file system for inconsistencies before you perform a backup. This ensures that you to do not back up data that is corrupt. Refer to Appendix A, "Using the fsck Command" in the *Solving HP-UX Problems* manual and *fsck*(1M) in the *HP-UX Reference Manual.*

Caution Do not run `fsck` on a file system that is mounted and active. This could introduce data corruption. Run `fsck` in single-user mode when checking the root file system. For file systems other that the root file system, unmount the file system, run `fsck`, and then remount the file system.

---

4. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.

5. Highlight **Backup and Recovery** and activate the [Open] control button.

6. Highlight **Backup Devices** and activate the [Open] control button.

7. Choose **Use Remote Backup Device** and **Backup Files Interactively** from the "Actions" menu.

8. Activate [Specify Remote Backup Device...]

9. Fill in the hostname (machine name) to which the backup device is attached and the associated device file. Activate the [OK] control button.

10. Activate [Select Backup Scope].

11. Turn on the **Entire System** or **Selected Files** checkbox.

    If you turned on the **Entire System** checkbox, activate the [OK] control button.

    If you turned on the **Selected Files** checkbox, enter each file name in the "Included" and "Excluded" boxes and activate the [Add] control button. If you make a mistake, highlight the entry with the error and use the [Modify] or [Remove] control buttons to correct the mistake.

    When you are satisfied with the included and excluded lists, activate the [OK] control button.

12. To:

    - cancel creating an index log or
    - backup data that is NFS mounted to your system,

    activate [Set Additional Parameters...] and turn off and on the appropriate checkboxes.

    Creating an on-line index file is performed by default and is very useful for tracking problems if your backup did not complete successfully. The log files are `/usr/sam/log/br_index.full` for full backups and `/usr/sam/log/br_index.incr` for incremental backups.

    When backing up files that are NFS mounted to your system, `fbackup` can only back up those files having "other user" read permission. `fbackup` normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `fbackup` can back up the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

    When you have finished setting the additional parameters, activate the [OK] control button.

13. Activate the [OK] control button to begin the backup process.

    If confirmation messages appear, read the message(s) and activate the [OK] control button to proceed in each case. SAM displays a window containing the output of the executed `fbackup` command. If you created an index log, the information displayed will appear in the index log.

Setting Up an Automated Backup Schedule Using SAM

The easiest way to set up an automated backup schedule is to use SAM's "Backup and Recovery" capabilities. SAM allows you to:

- View the entries in the automated backup schedule
- Add an entry to the automated backup schedule
- Remove entries in automated backup schedule
- Override the backup schedule to perform the backup now

You can schedule backups to happen on specific days of the week at a specific time, or you can schedule them to happen on specific days of the month at a specific time.

You can have multiple entries in your automated backup schedule to accommodate combinations of full and incremental backups and to allow you to back up different groups of files at different intervals.

Note

- SAM uses the `fbackup` utility to perform its backups (automated or otherwise). If you use SAM to set up an automated backup schedule, you should be aware that `fbackup` is a highly interactive utility. If it should need attention (tape change, device not on line, etc.), `fbackup` will attempt to prompt for the input it needs. This may cause an automated backup to fail (or not complete).
- SAM supports one full and incremental backup level per graph file.
- SAM does not support multiple backup devices in a single graph file entry. To back up to multiple device, create multiple entries and divide the scope of the backup (file systems) among different backup devices.

If for some reason a particular backup could not proceed, you can override the schedule to perform the backup immediately. For example, if you have a backup scheduled for late at night and you notice that the tape drive was not operational during this time, you can perform the backup immediately. This does not affect the automated backup schedule. Your backups will proceed as scheduled in the future. Refer to "Override the Backup Schedule to Perform the Backup Now".

When backing up files that are NFS mounted to your system, `fbackup` can only back up those files having "other user" read permission. `fbackup` normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `fbackup` can back up the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

Viewing Entries

To view entries currently defined in your automated backup schedule:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:
   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.
3. Highlight `Backup and Recover` and activate the (Open) control button.
4. Highlight `Automated Backups` and activate the (Open) control button.

   The object list displays the following attributes for each automated backup entry:
   - The time of day that the backup will occur.
   - The days of the month *or* week on which the backup will occur.
   - Whether the backup will be a full backup or an incremental backup.
   - Files that are included/excluded in the backup.

   You can view the device files to be used for your automated backups by rearranging the object list attributes. See "Changing the Format of the Object List" in Chapter 1, "Introduction to System Administration".

Adding an Entry

To add an entry to your automated backup schedule:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.

3. Highlight **Backup and Recover** and activate the [Open] control button.

4. Highlight **Automated Backups** and activate the [Open] control button.

5. Choose **Add an Automated Backup** and **Local Backup Device** or **Remote Backup Device** from the "Actions" menu.

   To add local device entry:

   a. Activate [Specify Backup Device].

   b. Highlight the backup device and activate the [OK] control button.

      SAM may ask you to provide additional information about:

      i. Magnetic format tape density.

      ii. DDS format tape with data compression or without data compression.

      iii. Optical disk number in an autochanger library system.

      iv. HP format cartridge tape number to start with in an autochanger.

      If there are no local devices detected or the particular device you want is not listed, you will need to add the device to the system. To add the device to the system, exit SAM and refer to the *Installing Peripherals*.

Note If the local backup device does not appear in the object list and you know the device file, you can enter the remote backup device and specify the local hostname and device file. SAM only supports backing up remotely to magnetic and DDS format tapes.

   To add a remote device entry:

   a. Activate the [Specify Remote Backup Device].

   b. Fill in the hostname (machine name) to which the backup device is attached and the associated device file.

   c. Activate the [OK] control button.

6. Activate [Select Backup Scope...].
7. Turn on the `Entire System` or `Selected Files` checkbox.

   If you turned on the `Entire System` checkbox, activate the [OK] control button.

   If you turned on the `Selected Files` checkbox, enter each file name in the "Included" and "Excluded" boxes and activate the [Add] control button. If you make a mistake, highlight the entry with the error and use the [Modify] or [Remove] control buttons to correct the mistake.

   When you are satisfied with the included and excluded lists, activate the [OK] control button.

8. Choose `Select Backup Time...`. Fill in the time of day to perform a full backup. Turn on "Days of the Week" or "Days of the Month" checkboxes to define the date for the backup. Highlight from the list the day(s) or date(s).

   Optionally choose `Enabled` or `Disabled` to schedule an incremental backup. SAM only supports one level of incremental backup.

   For any backup type that is visible, you must specify the time parameter.

   When you are satisfied with the entries, activate the [OK] control button.

9. To:

   - cancel creating an index log or
   - backup data that is NFS mounted to your system,

   activate [Set Additional Parameters...] and turn off and on the appropriate checkboxes.

   Creating an on-line index log is performed by default and is very useful for tracking problems if your backup did not complete successfully. The log files are `/usr/sam/log/br_index.full` for full backups and `/usr/sam/log/br_index.incr` for incremental backups.

   When backing up files that are NFS mounted to your system, `fbackup` can only back up those files having "other user" read permission. `fbackup` normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `fbackup` can back up the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

   When you have finished setting the additional parameters, activate the [OK] control button.

10. Activate the [OK] control button to add an entry to the crontab file. If `cron` is not running, SAM will display a message. Refer to *cron*(1M) and *crontab*(1) in the *HP-UX Reference Manual* manual.

    You should see your entry in the object list.

Caution Prior to the scheduled backup, run the `fsck` command to check your file system for inconsistencies. This ensures that you do not back up data that is corrupt. Refer to Appendix A, "Using the fsck Command" in the *Solving HP-UX Problems* manual and *fsck*(1M) in the *HP-UX Reference Manual*.

Do not run `fsck` on a file system that is mounted and active. This could introduce data corruption. Run `fsck` in single-user mode when checking the root file system. For file systems other that the root file system, unmount the file system, run `fsck`, and then remount the file system.

Removing Entries

To remove an entry from the automated backup schedule:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.
3. Highlight `Backup and Recover` and activate the (Open) control button.
4. Highlight `Automated Backups` and activate the (Open) control button.
5. Highlight the entry or entries to be removed.
6. Choose `Remove an Automated Backup...` from the "Actions" menu.

Override the Backup Schedule to Perform the Backup Now

To override the backup schedule to perform the backup now:

1. Ensure that you have superuser capabilities.
2. Run the `fsck` command to check your file system for inconsistencies before you perform a backup. This ensures that you to do not back up data that is corrupt. Refer to Appendix A, "Using the fsck Command" in the *Solving HP-UX Problems* manual and *fsck*(1M) in the *HP-UX Reference Manual.*

Caution Do not run `fsck` on a file system that is mounted and active. This could introduce data corruption. Run `fsck` in single-user mode when checking the root file system. For file systems other that the root file system, unmount the file system, run `fsck`, and then remount the file system.

---

3. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the [Help] button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the [f1] key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.

4. Highlight `Backup and Recover` and activate the [Open] control button.
5. Highlight `Automated Backups` and activate the [Open] control button.
6. Highlight the backup entry you want to perform now.
7. Choose `Perform Backup Now...` from the "Actions" menu.

Restoring All Files From a Local Device Using SAM

Note To restore data to disks physically connected to another computer, enter the `Remote Administration` functional area of SAM and refer to Chapter 1, "Introduction to System Administration", for additional information.

When restoring files that are NFS mounted to your system, `frecover` can only restore those files having "other user" write permission. `frecover` normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `frecover` can restore the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

To restore all files using SAM:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.

3. Highlight `Backup and Recover` and activate the [Open] control button.
4. Highlight `Backup Devices` and activate the [Open] control button.
5. Highlight the device in the list from which the data is to be restored.
6. Choose `Recover Files or Directories`
7. Activate [Select Recovery Scope].
8. Turn on the `All Files on Media` checkbox and activate the [OK] control button.
9. To:

   - overwrite new files,
   - maintain original ownership,
   - recover files using full path name,
   - place files in a non-root directory

   activate [Set Additional Parameters] and turn on the appropriate checkbox.

   To restore files relative to a particular directory, fill in the directory.

   Activate the [OK] control button to set the additional parameters.
10. Activate the [OK] control button to start the restore process.

    If confirmation messages appear, read the message(s) and activate the [OK] control button to proceed in each case. SAM displays a window containing the output of the executed `frecover` command.

Restoring Individual Files From a Local Device Using SAM

Gather the following information and materials before you begin:

- A list of files you need.
- The media on which the data resides.
- The location on your system to restore the files (original location or relative to some other location).
- The device and device file for restoring the data.

Note To restore data to disks physically connected to another computer, enter the `Remote Administration` functional area of SAM and refer to Chapter 1, "Introduction to System Administration", for additional information.

When restoring files that are NFS mounted to your system, `frecover` can only restore those files having "other user" write permission. The `frecover` command normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `frecover` can restore the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

To restore individual files:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.
3. Highlight `Backup and Recover` and activate the (Open) control button.
4. Highlight `Backup Devices` and activate the (Open) control button.
5. Highlight the device in the list from which the data is to be restored.
6. Choose `Recover Files or Directories` from the "Actions" menu.

7. Activate [Select Recovery Scope].
8. Turn on the `Selected Files` checkbox.
   - Fill in the filename containing a list of files to restore. The filenames should be full pathnames. This file is *not* a graph file. This file is used to create a graph file. You can use the on-line index file created by a previous backup, but it must be edited to containing only the full pathnames of the files to be restored.

     or

   - Enter each file name in the "Included" and "Excluded" boxes and activate the [Add] control button. If you make a mistake, highlight the entry with the error and use the [Modify] or [Remove] control buttons to correct the mistake.

     You can use both the file and the included/excluded method simultaneously to specify files to be restored.

   When you have completed determining the selecting files to be recovered, activate the [OK] control button.
9. To:
   - overwrite new files,
   - maintain original ownership,
   - recover files using full path name, or
   - place files in a non-root directory

   activate [Set Additional Parameters].

   Turn on the appropriate checkbox.

   To restore files relative to a particular directory, fill in the directory.

   Activate the [OK] control button to set the additional parameters.
10. Activate the [OK] control button to start the restore process.

    If confirmation messages appear, read the message(s) and activate the [OK] control button to proceed in each case. SAM displays a window containing the output of the executed `frecover` command.

Restoring All Files From a Remote Device Using SAM

Note

- To restore files to disks physically connected to another computer, enter the `Remote Administration` functional area of SAM and refer to Chapter 1, "Introduction to System Administration", for additional information.
- SAM only supports restoring data remotely from magnetic and DDS format tapes.

When restoring files that are NFS mounted to your system, `frecover` can only restore those files having "other user" write permission. The `frecover` command normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `frecover` can restore the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

To restore all files using SAM:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.

3. Highlight `Backup and Recover` and activate the [Open] control button.
4. Highlight `Backup Devices` and activate the [Open] control button.
5. Choose `Use Remote Backup Device` and `Recover Files or Directories`
6. Activate [Specify Remote Backup Device...]
7. Fill in the hostname (machine name) to which the backup device is attached and the associated device file. SAM only supports restoring data remotely from magnetic and DDS format tapes.
8. Activate [Select Recovery Scope].
9. Choose `All Files on Media` and activate the [OK] control button.
10. To:

    - overwrite new files,
    - maintain original ownership,
    - recover files using full path name,
    - place files in a non-root directory

    activate [Set Additional Parameters] and turn on the appropriate checkbox.

    To restore files relative to a particular directory, fill in the directory.

    Activate the [OK] control button to set the additional parameters.
11. Activate the [OK] control button to start the restore process.

    If confirmation messages appear, read the message(s) and activate the [OK] control button to proceed in each case. SAM displays a window containing the output of the executed `frecover` command.

Restoring Individual Files From a Remote Device Using SAM

Gather the following information and materials before you begin:

- A list of files you need.
- The media on which the data resides.
- The location on your system to restore the files (original location or relative to some other location).
- The device and device file for restoring the data.

Note

- To restore files to disks physically connected to another computer, enter the `Remote Administration` functional area of SAM and refer to Chapter 1, "Introduction to System Administration", for additional information.
- SAM only supports restoring data remotely from magnetic and DDS format tapes.

When restoring files that are NFS mounted to your system, `frecover` can only restore those files having "other user" write permission. The `frecover` command normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `frecover` can restore the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

To restore individual files:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.

3. Highlight `Backup and Recover` and activate the [Open] control button.

4. Highlight `Backup Devices` and activate the [Open] control button.

5. Choose `Use Remote Backup Device` and `Recover Files or Directories` from the "Actions" menu.

6. Activate [Specify Remote Backup Device].

7. Fill in the hostname (machine name) and the assciated device file and activate the [OK] control button. SAM only supports restoring data remotely from magnetic and DDS format tapes.

8. Activate [Select Recovery Scope].

9. Turn on the `Selected Files` checkbox.

   - Fill in the filename containing a list of files to restore. The filenames should be full pathnames. This file is *not* a graph file. This file is used to create a graph file. You can use the on-line index file created by a previous backup, but it must be edited to containing only the full pathnames of the files to be restored.

     or

   - Enter each file name in the "Included" and "Excluded" boxes and activate the [Add] control button. If you make a mistake, highlight the entry with the error and use the [Modify] or [Remove] control buttons to correct the mistake.

     You can use both the file and the included/excluded method simultaneously to specify files to be restored.

   When you have completed determining the selecting files to be recovered, activate the [OK] control button.

10. To:

    - overwrite new files,
    - maintain original ownership,
    - recover files using full path name, or
    - place files in a non-root directory.

    activate [Set Additional Parameters].

    Turn on the appropriate checkbox.

    To restore files relative to a particular directory, fill in the directory.

    Activate the [OK] control button to set the additional parameters.

11. Activate the [OK] control button to start the restore process.

    If confirmation messages appear, read the message(s) and activate the [OK] control button to proceed in each case. SAM displays a window containing the output of the executed `frecover` command.

Viewing the Index File on the Local Media Using SAM

To view the index stored on the media:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.
3. Highlight `Backup and Recover` and activate the (Open) control button.
4. Highlight `Backup Devices` and activate the (Open) control button.
5. Highlight the device in the list from which the index is to be read and displayed.
6. Choose `Get List of Files on Device` from the "Actions" menu.

9

7. Fill in the filename to which you want the index file restored and activate the [OK] control button.

   If the device is an optical autochanger library system or an HP format cartridge tape autochanger, SAM asks you for the disk number or the cartridge tape number. Unless otherwise specified, SAM looks on the media at location one.

   If you backup is contained on multiple media and you specified to create an index, each media has an index file listing the contents of the media itself and all previous media. For example, If your backup is on three (A, B, and C) optical disks, the index file on disk A contains only the files contained on disk A; the index file on disk B contains the files on disk A and B; the index file on disk C contains the files on disk A, disk B, and and disk C.

   SAM will open a window to execute the the `frecover` command to restore the index file.

Important! Although every volume in the backup set has an index, indexes are completely accurate only for the previous volumes in the same set. For example, the index on the last volume may indicate that a file resides on that volume, but it may not have been backed up (it may have been removed after the index was created, but before fbackup attempted to back it up). *The only index guaranteed to be correct in all cases* is the on-line index, that is produced after the last volume has been written. See the next section for information about viewing SAM's on-line index files.

Viewing the Index File on a Remote Device Using SAM

Note

- To restore an index file to disks physically connected to another computer, enter the `Remote Administration` functional area of SAM and refer to Chapter 1, "Introduction to System Administration", for additional information.
- SAM only supports recovering the index file remotely from magnetic and DDS format tapes.

To view the index stored on the media:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.

3. Highlight `Backup and Recover` and activate the [Open] control button.
4. Highlight `Backup Devices` and activate the [Open] control button.
5. Highlight the device in the list from which the index is to be read and displayed.
6. Choose `Use Remote Backup Device` and `Get List of Files on Device` from the "Actions" menu.
7. Fill in the hostname (machine name), device file, and filename to which you want the index file restored and activate the [OK] control button. SAM only supports restoring data remotely from magnetic and DDS format tapes.

   SAM will open a window to execute the the `frecover` command to restore the index file.

Important! Although every volume in the backup set has an index, indexes are completely accurate only for the previous volumes in the same set. For example, the index on the last volume may indicate that a file resides on that volume, but it may not have been backed up (it may have been removed after the index was created, but before fbackup attempted to back it up). *The only index guaranteed to be correct in all cases* is the on-line index, that is produced after the last volume has been written. See the next section for information about viewing SAM's on-line index files.

Viewing the Index Files on Your System Using SAM

Whenever you use SAM to back up your system, the `fbackup` utility creates an on-line index file. SAM keeps a separate index file for each of the following:

- The most recent *full* backup
- The most recent *incremental* backup
- A history log of all operations
- The most recent file recovery operation

To view these on-line index files:

1. Ensure that you have superuser capabilities.
2. Run SAM; type:

   `/usr/bin/sam`

   SAM provides an on-line help system to assist you when you need additional information.

   Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   From within a functional area, choosing an item from the "Help" menu gives you information about:

   - the current functional area
   - keyboard navigation within SAM
   - using the SAM help system
   - displaying the version of SAM you are currently running

   From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

   See Chapter 1, "Introduction to System Administration", for additional information about using SAM.

3. Highlight **Backup and Recover** and activate the [Open] control button.
4. Highlight **Backup Devices** or **Automated Backups** and activate the [Open] control button.
5. Choose **View Log Files** and one of the following from the "Actions" menu:

   Full Backup Log...

   History Log...

   Incremental Backup Log...

   Recovery Log...

   After you have viewed the information, activate the [OK] control button.

Backing Up Your System Using HP-UX Commands

Gather the following information before you begin:

- A list of files you want to back up (see "Determining Which Data to Back Up").
- The type of backup (see "Full Backups vs. Incremental Backups").
- The device file for the device on which to create your backup. All configured HP-UX devices have a *device file* associated with them that tells HP-UX the hardware address of the device and that which driver to use when communicating with the device. For more information about device files, see Chapter 11, "System Configuration" in the *How HP-UX Works: Concepts for the System Administrator* manual or the *Installing Peripherals* manual.

When backing up files that are NFS mounted to your system, `fbackup` can only back up those files having "other user" read permission. `fbackup` normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `fbackup` can back up the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

The `fbackup` Command

To back up files using the **fbackup** command:

1. Ensure that files you intend to back up are not being accessed.

Note The **fbackup** command will not back up files that are active (open) or locked when it encounters them.

---

2. Verify that the connections to the backup device are correct.
3. Verify that the backup device is turned on.
4. Ensure that the backup device is loaded with media. If the backup requires you to use additional media, the **fbackup** command will prompt you when to load change media.
5. Ensure that you have superuser capabilities.
6. Run the **fsck** command to check your file system for inconsistencies before you perform a backup. This ensures that you to do not back up data that is corrupt. Refer to Appendix A, "Using the fsck Command" in the *Solving HP-UX Problems* manual and *fsck*(1M) in the *HP-UX Reference Manual.*

Caution Do not run **fsck** on a file system that is mounted and active. This could introduce data corruption. Run **fsck** in single-user mode when checking the root file system. For file systems other that the root file system, unmount the file system, run **fsck**, and then remount the file system.

---

7. Create a backup using the **fbackup** command:

   /etc/fbackup -f *device* [**-f** *device*] ... [**-***level*] [**-uH**]
   [**-i** *path*] [**-e** *path*] [**-g** *graph_file*] [**-I** *index_file*]

   The command syntax is shown on two lines because of its length. When you execute an actual **fbackup** command, do *not* enter a carriage return until you have entered the entire command. Let the characters wrap around onto additional lines, if necessary.

   The most frequently used **fbackup** command options are described in Table 9-8. Refer to **fbackup(1M)** in the *HP-UX Reference Manual* manual for details on additional options.

Table 9-8. fbackup command options

Option	What it is	Why you need to use it
`-f` *device*	Device File Name	The destination for the information you are backing up. More than one `-f` option can be used. When multiple devices are specified, backups are created on the devices in the order they are specified on the command line. The device access sequence repeats in a cyclical pattern until the backup is finished. If all of the device files have been used and there is still more data to be backed up, `fbackup` will prompt you to load additional media. For creating remote or network backups, the device file format is **machine:device-file**. The **machine** name is restricted to eight characters. Only magnetic tape and DDS format (DAT) tape drives are supported for remote backups. Refer to *fbackup*(1M) in the *HP-UX Reference Manual* for additional information. If you are using an HP format cartridge tape drive (such as a 9144 , 9145 or 35401), you will need to pipe the output of `fbackup` into the `tcio` utility that properly blocks the output for the cartridge tape drive. If you do not do this, the activity of handling the output from `fbackup` will cause excess wear on the cartridge tape drive mechanism which may result in mechanical failure. The `tcio` commadn also optimizes the data transfer rate between certain cartridge tape units and the computer. To send the output of `fbackup` to a HP-UX command like `tcio`, use the -f option and the device file name "-" (dash). This tells `fbackup` to send its output to the standard output device (that can be piped to an HP-UX command). Do not use the `tcio` command with QIC format cartridge tapes.

Table 9-8. fbackup command options (continued)

Option	What it is	Why you need to use it
`-0 ... -9`	Backup Level	This option tells **fbackup** the backup level that you want to use for this backup. See the section "Backup Levels" earlier in this chapter for information on how backup levels are used.
`-u`	Update "dates" file	You need to use this option when you want **fbackup** to record information about this backup in the file **/usr/adm/fbackupfiles/dates** so that future incremental backups can be based on the date of this backup.
`-i` *path*	Include	This option tells **fbackup** which files you want to include in your backup. See "Determining Which Data to Back Up". You can use as many `-i` options as you want.
`-e` *path*	Exclude	This options tells **fbackup** which files (from the list of included files) are to be excluded from the backup (the "exceptions"). See "Determining Which Data to Back Up". You can use as many `-e` options as you want.
`-g` *graph_file*	Graph	If you have a complicated structure of files to back up, or if you want to use the incremental backup features that **fbackup** provides (see "Backup Levels") you should use a "graph file" instead of using the `-i` and `-e` options to specify which directories and files to back up.
`-I` *index_file*	Index File	You can have the **fbackup** utility create an index file for you. This file will contain a complete list of the files that it has just backed up.

Table 9-8. fbackup command options (continued)

Option	What it is	Why you need to use it
`-n`	Cross NFS Mount Points	Back up data this is NFS mounted to your system. When backing up files that are NFS mounted to your system, **fbackup** can only back up those files having "other user" read permission. **fbackup** normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that **fbackup** can back up the files exported from the NFS server, login as superuser on the NFS file server and use the **root=** option to the **/usr/etc/exportfs** command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

Examples Using the `fbackup` Command

This section contains a series of examples showing a variety of ways to use the **fbackup** command. The examples are based on the directory in Figure 9-4 unless otherwise stated. Figure 9-4 is the same as Figure 9-1, it is repeated for your convenience.

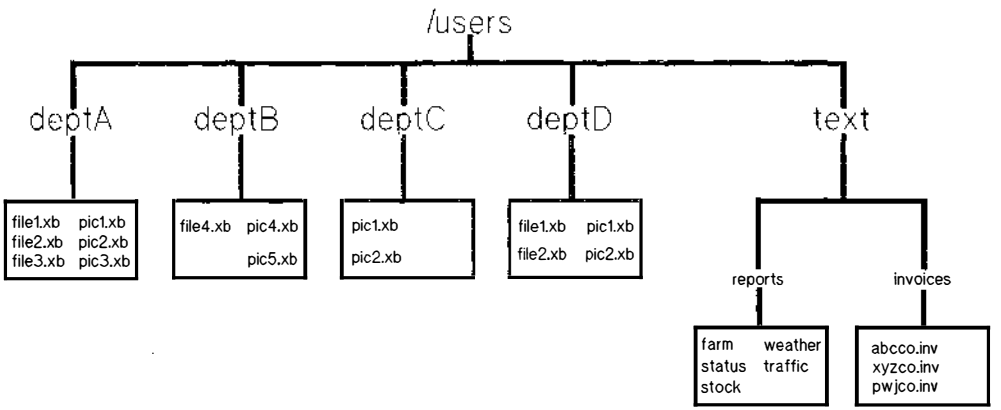


Figure 9-4. An Example Directory Tree

The following are example device files:

`/dev/rmt/0m`	represents a magnetic tape drive in 1600 bpi mode or a DDS (DAT) tape drive in uncompressed mode.
`/dev/rmt/1h`	represents a magnetic tape drive in 6250 bpi mode or a DDS (DAT) tape drive in compressed mode.
`/dev/rct/0s0`	represents an HP format HP-IB cartridge tape drive.
`/dev/rmt/0qic525`	represents a QIC format SCSI cartridge tape drive. If your cartridge tape uses QIC format, the `fbackup` command only supports the QIC-525 format. If your cartridge tape is not a QIC-525 format drive, refer to "Other Backup and Restore Utilities" for utilities that can help you create a backup. Ensure that your device file was created using the correct major and minor number. Refer to the *Installing Peripherals* manual for creating device files and minor number formats.
`/dev/ac/c0d1as2`	represents an optical disk in an optical library. This device file represents the library at LU zero, disk number 1, and surface "a" using section 2 (the entire surface).

For additional information on device files, see Chapter 11, "System Configuration" in *How HP-UX Works: Concepts for the System Administrator* and the *Installing Peripherals* manual.

To back up the entire directory structure in Figure 9-4 with the following criteria:

media:	DDS format (DAT) tape in uncompressed format
device file:	`/dev/rmt/0m`
backup type:	full
index file:	no
graph file:	no
update `dates` file:	no

type:

```
/etc/fbackup -f /dev/rmt/0m -i /users
```

To back up the entire structure *except* the `/users/text/invoices` directory with the following criteria:

media:	magnetic tape drive in 6250bpi mode
device file:	`/dev/rmt/0h`
backup type:	full
index file:	no
graph file:	yes (`/usr/adm/fbackupfiles/graphs/g1`)
update `dates` file:	yes

type:

```
/etc/fbackup -f /dev/rmt/1h -0 -u -g /usr/adm/fbackupfiles/graphs/g1
```

The contents of `/usr/adm/fbackupfiles/graphs/g1` is:

```
i /users
e /users/text/invoices
```

To perform an incremental back up (level 5) of the previous example, type:

```
/etc/fbackup -f /dev/rmt/1h -5 -u -g /usr/adm/fbackupfiles/graphs/g1
```

To back up the data as defined in the the previous example over the network to a device connected to remote machine `tulip`, type:

```
/etc/fbackup -f tulip:/dev/rmt/1h -5 -u -g /usr/adm/fbackupfiles/graphs/g1
```

To back up the entire structure *except* the `/users/text/invoices` directory with the following criteria:

media:	optical disk library system
device file:	`/dev/ac/c0d1as2`
backup type:	full
index file:	no
graph file:	no
update `dates` file:	no

type:

```
/etc/fbackup -f /dev/ac/c0d1as2  -i /users -e /users/text/invoices
```

To back up the entire structure *except* the `/users/text/invoices` directory with the following criteria:

media:	HP format cartridge tape drive
device file:	`/dev/rct/0s0`
backup type:	full
index file:	no
graph file:	no
update `dates` file:	no

type:

```
/etc/fbackup -f - -i /users -e /users/text/invoices | tcio -oe /dev/rct/0s0
```

Caution When you use `fbackup` to back up files to an HP format cartridge tape drive (such as a Model 9144 Model 9145, or Model 35401/Autochanger), you should always pipe the output from `fbackup` through the tape blocking utility known as `tcio`. If you do not do this, the activity of handling the output from `fbackup` will cause excess wear on the cartridge tape drive mechanism which may result in mechanical failure. The `tcio` command also optimizes the data transfer rate between certain cartridge tape units and the computer. The `tcio` command also handles multiple cartridge tapes.

Do *not* use the `tcio` command with QIC format cartridge tapes.

The `-f` option with the value "-" tells `fbackup` to send its output to the standard output file. In this example, that output is piped to the utility `tcio`. The `tcio` options direct the output to the cartridge tape drive represented by the device file `/dev/rct/0s0`. For complete information about valid options for `tcio`, see the `tcio(1)` manual reference page in the *HP-UX Reference Manual* manual.

To back up the entire file system (from `/`) using two magnetic tape drives in different modes with the following criteria:

media:	two magnetic tape drives (one in 1600bpi mode and the other in 6250bpi mode)
device file:	`/dev/rmt/0m` and `/dev/rmt/0h` respectively
backup type:	full
index file:	yes (`/tmp/index`)
graph file:	no
update `dates` file:	no

type:

```
/etc/fbackup -f /dev/rmt/0m -f /dev/rmt/1h -i / -I /tmp/index
```

This example will shows it is possible to specify more than one device to receive the output from `fbackup`. When more than one device is specified, the second one is used when the media on the first device fills up. This allows for an unattended backup if the media on the second device can hold the remaining data to be backed up.

Also in this example, an index file will be created called `/tmp/index`. An index is written to the beginning of each tape, listing all files in the specified "graph" being backed up. However, if a file is removed *after* the index is written but *before* the file is backed up to tape (or something else happens that prevents the file from being backed up), the index will not be completely accurate. If you tell `fbackup` to make an on-line index file (using the `-I` option), it will create the file *after* the backup is complete. Therefore, the on-line index file will be completely accurate with respect to which files are on each volume of the backup.

Backing Up to a Hard Disk Using the `cpio` Command

To back up to a hard disk using `cpio`:

1. Ensure that you have superuser capabilities.
2. Run the `fsck` command to check your file system for inconsistencies before you perform a backup. This ensures that you to do not back up data that is corrupt. Refer to Appendix A, "Using the fsck Command" in the *Solving HP-UX Problems* manual and *fsck*(1M) in the *HP-UX Reference Manual*.

Caution Do not run `fsck` on a file system that is mounted and active. This could introduce data corruption. Run `fsck` in single-user mode when checking the root file system. For file systems other that the root file system, unmount the file system, run `fsck`, and then remount the file system.

---

3. Add a disk to your system. This requires that you:
   - physically connect the disk drive to your system.
   - ensure that the device driver for the disk is part of the kernel.
   - ensure that there is a device file for the disk.

   Refer to the *Installing Peripherals* to add the disk drive to the system.
4. Create a file system on the newly added disk with the `newfs` command, for example:

   `newfs /dev/rdsk/`*disk_device_file* `hp2213A`
5. Mount the newly added disk and file system to your main file system:

   `mount /dev/dsk/`*disk_device_file* `/disk_mount_point`

9

6. Transfer the data from you main file system to the file system on the newly added disk:

```
cd /
find .  -xdev -hidden -print | cpio -padmuvx /disk_mount_point
```

   The `-xdev` option to the `find` command that avoids crossing any file system mount points that exist below starting points listed in the pathname list. This avoids a recursive backup of the `disk_mount_point` to the `disk_mount_point`. Refer to *find*(1) in the *HP-UX Reference Manual*.
7. Unmount the disk containing the backed up data:

```
umount /disk_mount_point
```

Note For additional information regarding the `newfs`, `mount`, or `umount` commands, refer to Chapter 6, "Managing the File System".

Backing Up To a Hard Disk or DDS Format Tape Drive Using the `dd` Command

The recommended utility for quickly backing up your entire file system to a hard disk is the `dd` utility. The `dd` utility is preferred over other utilities because it quickly creates a mirror image of your currently mounted file system. This method of copying your file system is typically used for copying your HP-UX system from one hard disk to another.

Note

- You can use `dd` to transfer data from a:
  - logical volume to a logical volume.
  - logical volume to a non LVM disk.
  - non LVM disk to a logical volume.
- Once a disk is part of LVM, `dd` cannot be used to transfer data from the disk. The data transfer must be made from the logical volume level.

It is also important to note that the `dd` utility only performs full backups (complete disk or disk section copies) and a backup log file is not automatically maintained. Using the `dd` command does not allow for individual file recovery. Only the entire disk section containing the file system can be restored.

To perform a full backup using `dd`:

1. Ensure that you have superuser capabilities.
2. Run the `fsck` command to check your file system for inconsistencies before you perform a backup. This ensures that you to do not back up data that is corrupt. Refer to Appendix A, "Using the fsck Command" in the *Solving HP-UX Problems* manual and *fsck*(1M) in the *HP-UX Reference Manual*.

Caution

Do not run `fsck` on a file system that is mounted and active. This could introduce data corruption. Run `fsck` in single-user mode when checking the root file system. For file systems other that the root file system, unmount the file system, run `fsck`, and then remount the file system.

---

3. Ensure that your system is configured to recognize the destination hard disk or DDS format tape drive. This requires ensuring the device driver is part of the kernel and that there is an appropriate device file. Refer to the *Installing Peripherals* manual for details.

4. Get your source file system device file with the `bdf` command, for example:

```
bdf
Filesystem        Kbytes    used    avail  capacity  Mounted on
/dev/dsk/c0d0s13  324546  175018   117073       60%      /
/dev/vg00/mktg    209437  175018   117073                /users/mktg
```

   For logical volumes you can also use the `lvdisplay` command. See Chapter 8, "Managing Logical Volumes".

5. Identify your destination hard disk device file

6. Invoke the `dd` command. The command syntax is:

   `/bin/dd if=`*source* `of=`*destination* `bs=`*block_size*

   where:

source	is the device file for the disk drive to be copied.
destination	is the device file for the destination (target) device.
var|block_size|	is the block size for the device as specified by the `/etc/disktab` file.

   For example,

   ```
   dd if=/dev/dsk/c0d0s13 of=/dev/dsk/c1d0s13 bs=1024k
   ```

7. Label the backup hard disk to identify the date, sections used, and their contents. If you do not label the disk with the sections containing your files, you will have to systematically mount every section of the disk to find your files.

Warning

- **The `dd` utility will overwrite data on the destination disk drive. This type of backup is only recommended for use between disks of the same type and size.**
- **Do not interchange the *source* and *destination* parameters! If you do, you will destroy the data you are trying to backup located on your *source* disk.**

It is recommended that you use the same type of backup hard disk as your primary hard disk and use the same section on the destination hard disk to create your backup. Additional hard disk sectioning information is contained in the `/etc/disktab` file and in Chapter 6, "Managing the File System".

Setting Up an Automated Backup Schedule Using HP-UX Commands

Setting up an automated backup schedule using HP-UX commands is more involved than using SAM to do it. This is an area where using SAM is recommended.

You may automate your backup procedure using the `crontab(1)` utility, which interfaces with the HP-UX process scheduling facility, `cron(1M)`. You can find additional information about `cron` and `crontab` in the *HP-UX Reference Manual* manual.

To automate your backup procedure, you need to:

1. Create a file that defines the process you want to automate.
2. Activate the processes that are defined in the file you created.

Note If you schedule `fbackup` via the `crontab` utility you should be aware that `fbackup` is a highly interactive utility. If it should need attention (tape change, device not on line, etc.), `fbackup` will attempt to prompt for the input it needs. This may cause an automated backup to fail (or not complete).

Creating an Automated Backup Schedule File

The `crontab` utility allows you to specify an input file containing the date, time, and runstrings of the backup procedures (processes) that you want to automate (see "Examples Using the `fbackup` Command"). This file (the input to the `crontab` utility) contains lines that have six required fields each. The fields are separated by spaces or tabs. Each entry in this file has the following format:

```
minute hour dates months days runstring
```

where:

`minute`	*specifies the minute (0-59)*
`hour`	*specifies the hour (0-23)*
`dates`	*specifies particular dates of the month (1-31)*
`months`	*specifies particular months of the year (1-12)*
`days`	*specifies particular days of the week (0-6 with 0 representing Sunday)*
`runstring`	*specifies the command line or script file to execute*

An entry of "*" in any crontab field represents all legal values.

Therefore, to schedule the `ps(1)` command to execute at 5:10pm (17:10) on every Friday and Monday during June, July, and August, you would make an entry in your crontab file that looks like this:

```
10 17 * 6,7,8 1,5 ps >> psfile 2>&
```

The `2>&` redirects any error messages to the file `psfile`.

Note When scheduling processes with `crontab`, you must redirect any output that is normally sent to the terminal to a file. In the above example, the file containing the output is `psfile`.

An example backup strategy may consist of a full backup (performed once a week on Saturday) and an incremental daily backup (every week day). Assume that the backups are to be performed at 4:03 am and the media is DDS format (DAT) tape. The following crontab file implements the example backup strategy:

```
3 4 * * 1 /usr/adm/incrback  >>  monbackup
3 4 * * 2 /usr/adm/incrback  >>  tuebackup
3 4 * * 3 /usr/adm/incrback  >>  wedbackup
3 4 * * 4 /usr/adm/incrback  >>  thubackup
3 4 * * 5 /usr/adm/incrback  >>  fribackup
3 4 * * 6 /usr/adm/fullback  >>  satbackupfull
```

In the above example "incrback" and "fullback" are shell scripts that contain the `fbackup` commands.

Activating Your Automated Backup Schedule

Once you have created the file defining your time-scheduled backups, you must inform `cron` that it has new jobs to schedule. You do this with the `crontab` utility. For example:

`crontab` *your_crontab_file*

This will activate all of the processes defined in *your_crontab_file*. This cancels any previously scheduled processes not defined in *your_crontab_file*.

Note Before activating a new crontab file, you should view currently scheduled processes. Consider adding these processes to *your_crontab_file*. See "Displaying Your Automated Backup Schedule".

After your cronfile backup has been activated, you must always ensure that:

- Run the `fsck` command to check your file system for inconsistencies before you perform a backup. This ensures that you to do not back up data that is corrupt. Refer to Appendix A, "Using the fsck Command" in the *Solving HP-UX Problems* manual and *fsck*(1M) in the *HP-UX Reference Manual*.

Caution Do not run `fsck` on a file system that is mounted and active. This could introduce data corruption. Run `fsck` in single-user mode when checking the root file system. For file systems other that the root file system, unmount the file system, run `fsck`, and then remount the file system.

- The system clock is set properly.
- The backup device is properly connected and the HP-UX I/O system recognizes the device file specified in the `fbackup` runstring.
- Adequate media has been loaded in the backup device.
- The backup device is connected your system and turned on.
- The NFS mounted files you want backed up have the correct permissions

  When backing up files that are NFS mounted to your system, `fbackup` can only back up those files having "other user" read permission. `fbackup` normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `fbackup` can back up the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

Displaying Your Automated Backup Schedule

To list your currently scheduled processes, execute the command:

```
crontab -l
```

This will display the contents of your activated crontab file. You should always view the currently scheduled processes *before* activating a new crontab file to be sure that you do not cancel processes that need to remain active. If there are such processes, add them to the definitions in your new crontab file before executing the `crontab` command to add new processes.

Deactivating Your Automated Backup Schedule

To deactivate all of your currently scheduled processes, execute the command:

```
crontab -r
```

Changing Your Automated Backup Schedule

To change your currently scheduled processes:

1. Edit your crontab file, which defines the jobs to time schedule to incorporate the changes you want.
2. Activate the edited file as you did the original. See "Activating Your Automated Backup Schedule".

Restoring Files Using HP-UX Commands

Gather the following information and materials before you begin:

- A list of files you need.
- The media on which the data resides.
- The location on your system to restore the files (original location or relative to some other location).
- The device file corresponding to the device being used to for restore the files.

When restoring files that are NFS mounted to your system, `frecover` can only restore those files having "other user" read permission. `frecover` normally operates in user-mode when crossing NFS mount points; not superuser-mode. To ensure that `frecover` can restore the files exported from the NFS server, login as superuser on the NFS file server and use the `root=` option to the `/usr/etc/exportfs` command to export the correct permissions. Refer to *exportfs*(1M) in the *HP-UX Reference Manual* and the *Installing and Administering NFS Services* manual.

The `frecover` Command

The **frecover** command restores backup files made using the **fbackup** utility. If your files on the media did not get created using **fbackup**, refer to "Other Backup and Restore Utilities".

To restore files from backups using **fbackup**:

1. Ensure that files you intend to back up are not being accessed.

Note The **frecover** command will not restore files that are active (open) or locked when it encounters them.

---

2. Verify that the connections to the device are correct.
3. Verify that the device is turned on.
4. Ensure that the device is loaded with media. If the restore process requires you to use additional media, the **frecover** command needs to be executed separately for each media.
5. Ensure that you have superuser capabilities.
6. Restore files using the **frecover** command.
   - The **frecover** command syntax generally used when recovering *all* files from your backup is:

     ```
     /etc/frecover -r -o [-f device]
     ```

     or

     ```
     /etc/frecover -o [-f device]
     ```

     where:

-r	Recovers all of the files on the media.
-o	Recovers the files from the media irrespective of age. Normally **frecover** will not overwrite an existing file with an older version.
-f	Specifies the device file from which to recover the data. If not specified the default, **/dev/rmt/0m**, is used. See "The **fbackup** Command".

- The `frecover` command syntax generally used when restoring *individual* files to your system is:

  `/etc/frecover -x -o[F | X] [-f` *device*`] [-i` *path*`] [-e` *path*`] [-g` *graph_file*`]`

  where:

`-x`	Extracts files from the backup.
`-o`	Recovers the files from the media irrespective of age. Normally `frecover` will not overwrite an existing file with an older version.
`-F`	Recovers files without leading directories. Normally `frecover` recovers files to their absolute path name.
`-X`	Recovers files with paths relative to the current directory.
`-f`	Specifies the device file from which to recover the data. If not specified the default, `/dev/rmt/0m`, is used. See "The `fbackup` Command".
`-i`	Defines specific files to include in the recovery.
`-e`	Defines specific files to exclude from the recovery.
`-g`	Specifies the graph file to be used. The lists of files to be included and excluded are with respect to the structure of files represented on the *backup*, not the structure of files currently on your system (although the two may be similar).

  The syntax is presented here on two lines because of page width limitations. When you enter an `frecover` command that is longer than one screen width, do *not* press carriage return until the end of the command. Let the characters wrap around onto the next line if necessary. Items listed in brackets ([ ]) are optional.

`frecover` has other options. For complete details on the options for `frecover`, see *frecover*(1M) in the *HP-UX Reference Manual*.

Caution When you use `frecover` to restore files from a cartridge tape drive (such as a Model 9144 , Model 9145, or Model 35401/Autochanger), you should always pipe the input to `fbackup` through the tape blocking utility known as `tcio`. If you do not do this, the activity of handling the input from the device will cause excess wear on the cartridge tape drive mechanism, which may result in mechanical failure. The `tcio` command also optimizes the data transfer rate between certain cartridge tape units and the computer.

Do *not* use the `tcio` command with QIC format cartridge tapes.

Examples Using the `frecover` Command

The following examples are based on a "full backup" of the directory structure listed in Figure 9-5. Figure 9-5 is the same as Figure 9-4 and Figure 9-1, it is repeated for your convenience.

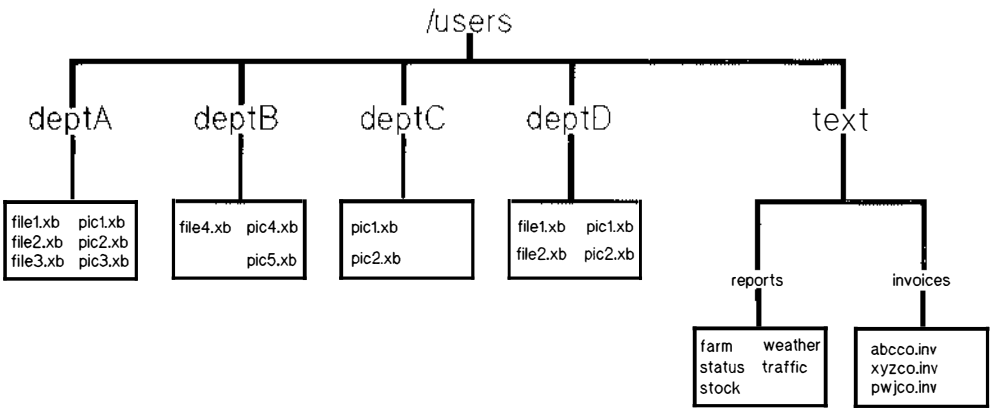


Figure 9-5. An Example Directory Tree

The following are example device files:

`/dev/rmt/0m`	represents a magnetic tape drive in 1600 bpi mode or a DDS format (DAT) tape drive in uncompressed mode
`/dev/rmt/1h`	represents a magnetic tape drive in 6250 bpi mode or a DDS format (DAT) tape drive in compressed mode.
`/dev/rct/0s0`	represents an HP format HP-IB cartridge tape drive.
`/dev/rmt/0qic525`	represents a QIC format SCSI cartridge tape drive. If your cartridge tape uses QIC format, the `fbackup` command only supports the QIC-525 format. If your cartridge tape is not a QIC-525 format drive, refer to "Other Backup and Restore Utilities" for utilities that can help you create a backup. Ensure that your device file was created using the correct major and minor numbers. Refer to the *Installing Peripherals* manual for creating device files and minor number formats.
`/dev/ac/c0d1as2`	represents an optical disk in an optical library. This device file represents the library at LU zero, disk number 1, and surface "a" using section 2 (the entire surface).

For additional information on device files, see Chapter 11, "System Configuration" in *How HP-UX Works: Concepts for the System Administrator* and the *Installing Peripherals* manual.

To restore the files from all of the directories under **/users/text** from a DDS format (DAT) tape into the **/tmp** directory on the system, type:

```
cd /tmp
/etc/frecover -x -oF -i /users/text
```

First change your working directory to **/tmp**. The **-F** option to **frecover** removes leading path names from all files on the tape. If there are files in the directory **/tmp** whose names match those coming from tape, specifying the **-o** option overwrites the version on disk, even if the copy on disk is newer. The **/tmp** directory now contains all of the files without the leading directories, for example:

```
ls /tmp

./          abcco.inv   pwjco.inv   stock       weather
../         farm        status      traffic     xyzco.inv
```

If there are duplicate filenames found on the tape, files are overwritten with the most recent one extracted from the tape.

To recover all of the files from a full backup using the default device (**/dev/rmt/0m**), type:

```
/etc/frecover
```

You do not need to specify the **-r** option since it is the default. Omitting the **-o** option will not overwrite any newer files on disk.

To recover all of the files from a full backup on an HP format cartridge tape, type:

```
tcio -i /dev/rct/0s0 | /etc/frecover  -f -
```

To restore the files belonging to the directory **/users/deptA** from DDS format (DAT) tape, type:

```
/etc/frecover -x -i /users/deptA
```

If files are currently in that directory (on the disk) that are newer than the corresponding files on the tape, **frecover** will *not* overwrite the newer version on disk because the **-F** option is not specified. Because we are restoring the files from the device represented by device file **/dev/rmt/0m**, we do not need to use the **-f** option.

To restore the files belonging to the directory `/users/deptA` from an HP format cartridge tape, type:

```
tcio -i /dev/rct/0s0 | frecover -x -f - -i /users/deptA
```

If files are currently in that directory on the disk that are newer than the corresponding files on the tape, we will *not* overwrite the newer version on disk. Because we are restoring the files from cartridge, we need to do two things:

1. Use the tape blocking utility, `tcio`, to filter the input from the HP format cartridge tape drive.
2. Do *not* use `tcio` with QIC format cartridge tapes.
3. Use `frecover's` -f option along with the "device" name "-", which indicates that `frecover` is to get its input from the standard input device (the piped output from `tcio`).

Restoring Data From a Hard Disk Using the `cpio` Command

1. Add the disk to your system. This requires that the device driver for the disk is part of the kernel and that there is a device file on the system to communicate to the disk. Refer to the *Installing Peripherals* to add the disk drive to the system.
2. Mount the disk to your root file system:

   ```
   mkdir /disk_mount_point
   mount /dev/dsk/disk_device_file /disk_mount_point
   ```

   For additional information refer to Chapter 6, "Managing the File System".
3. Restore data from the file system on the backup disk to your main file system:

   ```
   cd /disk_mount_point
   find /disk_mount_point -print | cpio -padmuvx /
   ```

Restoring Data From a Hard Disk Using the `dd` Command

Note

- You can use `dd` to transfer data from a:
  - logical volume to a logical volume.
  - logical volume to a disk that is not part of LVM.
  - disk that is not part of LVM to a logical volume.
- Once a disk is part of LVM, `dd` cannot be used to transfer data from the disk. The data transfer must be made from the logical volume level.

The recovery steps for restoring a file system from your backup hard disk depends on the file system you want to restore.

For restoring the root(/) file system, HP recommends the following five steps if you created your backup with the `dd` utility:

1. Get your current, destination, root (/) file system device file with the `bdf` command.
2. Get your backup hard disk, source, root(/) file system device file from the label. If you did not label your disk with the sections used for the backup, you must systematically mount all sections of the backup disk to find your files.
3. Reboot your system from the backed up file system on the backup hard disk. Rebooting your system is covered in the "Selecting a System to Boot" section of Chapter 3, "Starting and Stopping HP-UX".
4. Restore the currently mounted, backed up file system to your main disk drive with the `dd` utility.
5. Reboot your system from restored file system on your main hard disk. Rebooting your system is covered in the "Selecting a System to Boot" section of Chapter 3, "Starting and Stopping HP-UX".

For restoring a file system other than your root (/) file system, HP recommends the following five steps if you created your backup with the `dd` command:

1. Get your current, destination, file system device file with the `bdf` command.
2. Get your backup hard disk, source, file system device file from the label. If you did not label your disk with the sections used for the backup, you must systematically mount sections of the backup disk to find your files.
3. Unmount the file system you want to restore from your backup hard disk. See Chapter 6, "Managing the File System", for details on unmounting a file system.
4. Restore the file system from the backup hard disk with the `dd` command. This will overwrite the existing data in the specified destination section of your main hard disk.
5. Remount the recovered file system. See Chapter 6, "Managing the File System", for details on mounting a file system.

Viewing the Index File on the Local Device Using HP-UX Commands

The utility `frecover` has an "index" mode of operation. Using this mode of operation, you can have `fbackup` list the contents of the index at the beginning of a backup volume. To do this, use the `-I` option (instead of either `-r` or `-x`).

The syntax for this is:

```
/etc/frecover -I path -f device
```

Where:

path is the file in which to put the directory listing. The default is to print the listing to the standard output (usually the terminal screen).

device is the device file name for the device containing the backup medium.

Important! Although there is an index at the beginning of each volume in a backup set, indexes are completely accurate only for the previous volumes in the same set. Hence, the index on the last volume may indicate that a file resides on that volume, but it may not have been backed up (for example, if it was removed after the index was created, but before fbackup attempted to back it up). *The only index guaranteed to be correct in all cases* is the on-line index, which is produced after the last volume has been written (the one created using `fbackup's -I` option).

Here are two examples: Assuming that the device file for the magnetic tape drive is `/dev/rmt/0h` and we want to put the listing of the index in the file `/tmp/index2`, we would use the following command:

```
/etc/frecover -I /tmp/index2 -f /dev/rmt/0h
```

Assuming that the device file for the HP format cartridge tape drive is `/dev/rct/0s0` and we wanted to put the listing of the index in the file `/usr/adm/indexlog42`, we would use the following command:

```
tcio -i /dev/rct/0s0 |/etc/frecover -I /usr/adm/indexlog42 -f -
```

Other Backup and Restore Utilities

There are other utilities that can be used to back up files from your system. Some of them are listed below. Those mentioned here are *usually* available on other vendors' machines, so these utilities may be useful in transferring files between systems of various vendors.

Run the `fsck` command to check your file system for inconsistencies before you perform a backup. This ensures that you to do not back up data that is corrupt. Refer to Appendix A, "Using the fsck Command" in the *Solving HP-UX Problems* manual and *fsck*(1M) in the *HP-UX Reference Manual*.

Caution Do not run `fsck` on a file system that is mounted and active. This could introduce data corruption. Run `fsck` in single-user mode when checking the root file system. For file systems other that the root file system, unmount the file system, run `fsck`, and then remount the file system.

The other backup utilities available are:

/usr/bin/ftio is designed for 9-track magnetic tape media. The `ftio` utility has increased throughput over `cpio` and `tar` due to multiple processes and a larger blocking factor. This utility is not recommended for backing up secured (trusted) systems.

/bin/cpio The `cpio` utility with piped input from the `find` command can be used to create a backup. This utility is not recommended for backing up secured (trusted) systems.

/bin/dd The `dd` command can be used to create an exact image copy of your disk to DDS format (DAT) tape. The tape can be used to boot from, if your system disk crashes (or its data is erased). See "Backing Up To a Hard Disk or DDS Format Tape Drive Using the `dd` Command".

Note

- You can use `dd` to transfer data from a:
  - logical volume to a logical volume.
  - logical volume to a disk that is not part of LVM.
  - disk that is not part of LVM to a logical volume.
- Once a disk is part of LVM, `dd` cannot be used to transfer data from the disk. The data transfer must be made from the logical volume level.

/usr/bin/tar does not have the same level of error handling as `fbackup` and `frecover`. This utility will include special device files on the backup if the **-N** option is specified. This utility is not recommended for backing up secure (trusted) systems.

10

Managing Printers and Printer Output

What is the Line Printer Spooling System?

The **Line Printer Spooling System** (**lp spooler**) is a set of programs, shell scripts, and directories that control your printers and the flow of data going to them. It:

- helps prevent **intermixed listings**.
- provides control of printout routing.
- allows users to cancel, restart and adjust the priority of print requests.

Once a printer has been added to your system (that is, its driver is in your kernel configuration, and an appropriate device file exists), it can be added to the lp spooler (for example you can redirect the output of a command to the device file associated with the printer). We recommend adding all printers to the lp spooler. If you do not add your printer or plotter to the lp spooler, there is no coordination between multiple users and intermixed listings can occur. Unspooled printing is not recommended. The purpose of the lp spooler is to *automatically* coordinate between multiple users and prevent intermixed listings.

Note The term "printer" can be interchanged with the term "plotter" for the tasks described in this chapter.

This chapter will cover the following topics:

- Components of the lp spooler
- Remote printing
- Controlling data flow through the lp spooler
- Priorities of printers and print requests
- Using plotters with the lp spooler
- Collecting and reporting statistics about data flow through the lp spooler

This chapter describes how to accomplish the tasks associated with these topics using SAM and HP-UX commands. The HP-UX commands method is provided for those who do not have access to SAM or choose not to use it.

If you are already familiar with the basic concepts of the lp spooler you may want to proceed directly to one of the tasks in the "Line Printer Spooler Tasks" section of this chapter.

Note If you are reading this chapter because you have a problem with the lp spooler, you should first refer to Chapter 2, "Line Printer Spooling System Problems" in *Solving HP-UX Problems*, HP part number B2355-90030.

If you are working with the lp spooler for the first time or you want to review key concepts before performing a particular task, you should continue reading the material found in "LP Spooler Overview".

Terms Used in this Chapter

The following terms appear in discussions in this chapter. You can scan the list now and, if you need to, refer to it later.

destination A print destination is a generic term used to describe a printer or **printer class**. Users can specify a print destination when they print something by using the "`-d`" option to the `lp` command. See **printer class**.

interface script A shell script, located in the `/usr/spool/lp/interface` directory, that is the final stage of the lp spooler. Each printer that is configured in the **lp spooler** has an interface script that, under the control of the line printer scheduler, sends a print job to the printer.

intermixed listings Multiple jobs printing on a printer simultaneously that result in printed pages with characters from different print jobs mixed together. The lp spooler is designed to prevent this.

line printer scheduler The line printer scheduler is the heart of the **lp spooler**. It is the central program that must be running to ensure coordination of requests from users to printers.

lp spooler The HP-UX software subsystem responsible for controlling output to the printers on your system. Its primary responsibility is to prevent **intermixed listings**. It can also prioritize print jobs and start and stop output to printers.

local printer A printer, configured into your lp spooler, that is physically connected to your computer. See **remote printer**.

logical printer Each printer that is defined in your lp spooler is given a name that users will use to refer to it. You can create more than one definition (printer name) for any given printer. The logical printer name refers, not to the printer itself, but to one of the lp spooler definitions used to access the printer.

model script When a printer is added to the lp spooler, an interface script must be created that can set up the communication to the printer and send data to it. HP supplies models for these interface scripts in the `/usr/spool/lp/model` directory. These models are used by the `lpadmin` command to build the interface script for a printer as it is being defined.

cancel model script The cancel model script, `/usr/spool/lp/cmodel/rcmodel`, is used to cancel a print request to a printer on a remote system.

status model script The status model script, `/usr/spool/lp/smodel/rsmodel`, is used to return the status of the remote printer and print requests for the remote

network-based printer or plotter A printer or plotter that is directly connected to the local area network.

print request A term used to refer to a specific print job in the lp spooler.

print request identification number The number the lp spooler uses to identify your print request. This identification number consists of the name of the printer or printer class followed by a sequence number.

print request priority See **priority**.

printer class A defined group of printers. A printer class can be used as a print destination instead of a printer name. The first available printer in the printer class will print the next job queued to that printer class.

printer fence priority See **priority**.

printer name When a printer is configured into the lp spooler, it is given a name that users can use to specify that printer as a print destination for their printout.

print queues Also known as request directories, print queues are directories used by the lp spooler to hold print jobs for each print destination until they can be printed.

priority A value associated with each printer and print request. Priorities are used to control which print requests can print on a given printer. Priorities can be adjusted and must have a range from 0 to 7. See "Priorities of Printers and Print Requests".

remote print requests A print request issued via the lp spooler on your system to be printed on a printer that is attached to a remote computer.

remote printer A printer that is defined in your lp spooler but is physically connected to another computer (and accessed over a network).

remote spooling The process used to allow printing to printers that are defined as part of your lp spooler but physically connected to another computer.

remote spooling daemon A "behind the scenes" (background) program that runs on a remote computer. The remote spooling daemon receives print requests via a network and submits the print requests to its local lp spooler on the network user's behalf.

request directories See **print queues**.

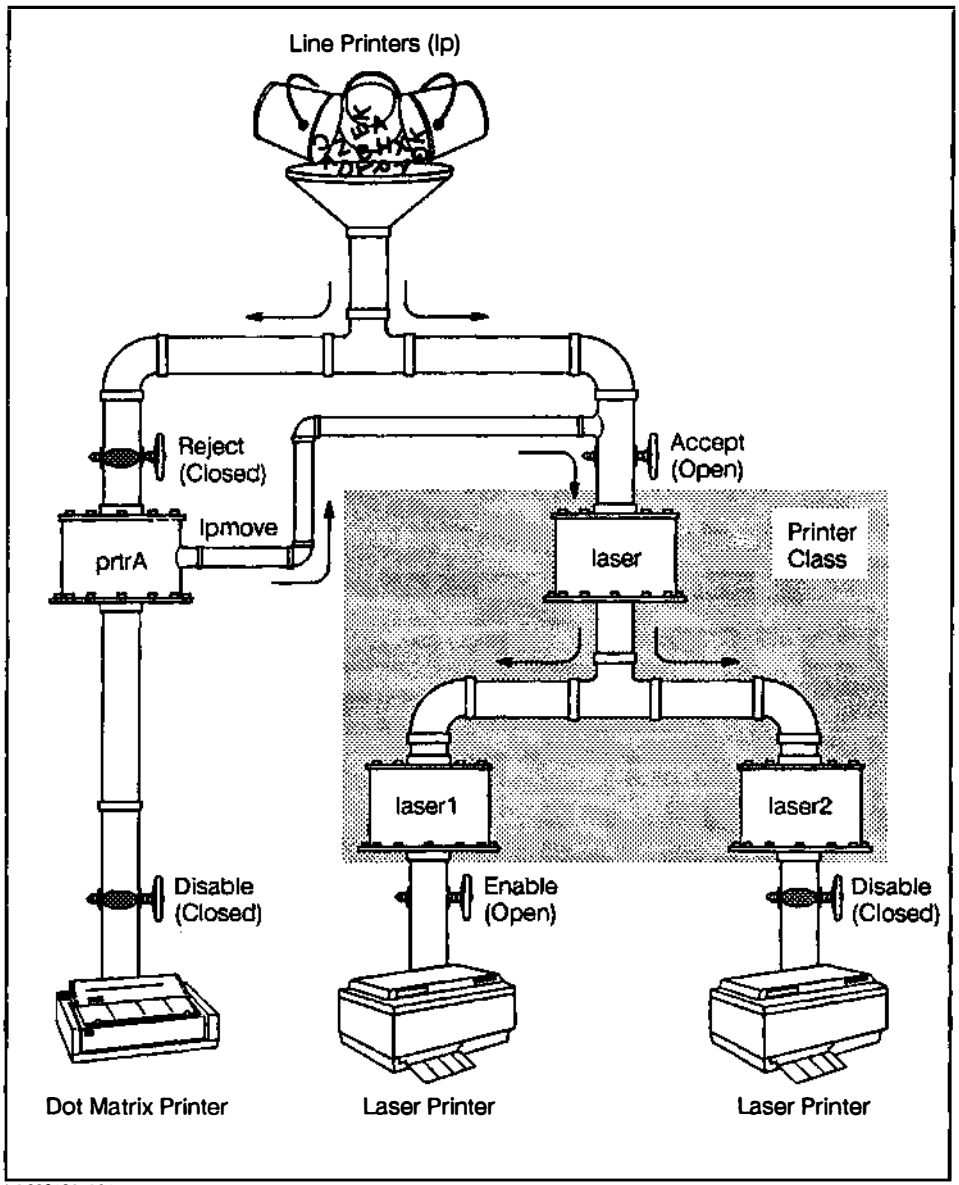
system default printer When a user issues a print request, the user can specify a print destination. A system default printer can be defined so that, if a print destination is not specified, the lp spooler will use the system default printer.

LP Spooler Overview

You can think of the lp spooler as if it were a plumbing system. Figure 10-1 shows how this "plumbing system" might look. The data to be printed represents the "water" in this system. There are various request directories, sometimes referred to as printer queues, which serve as temporary holding tanks for print requests until they are sent to a printer to be printed. The flow of print requests is controlled at the request directory and printer level. The terms *accept* and *reject* refer to controlling the flow of print requests to the request directories while the terms *enable* and *disable* refer to controlling the flow of print requests to the printers. Accepting, rejecting, enabling, and disabling print requests control the data through the lp spooler as valves would control the flow of water in a real plumbing system. Shell scripts (called interface scripts) near the end of the data flow serve as pumps which "pump" an orderly flow of data to the printers.

The line printer scheduler controls the routing of print requests from the request directories to the printers. It functions as an automated flow controller in the "plumbing system" to prevent **intermixed listings** and to provide efficient use of the printers on your system. Intermixed listings are multiple print requests printing on a printer simultaneously that result in printed pages with characters from different print requests mixed together.

You can add a printer to, or remove it from your system. If the "drain gets clogged" for one printer, you can re-route the print requests for that printer to another printer and you can "flush" unwanted print requests from the spooling system. You can also sent a print requests to another computer to printed. Sending print requests to another computer to be printed is called **remote spooling** and the other computer is referred to as a remote system. When you use remote spooling, a special shell script ("pump") is used to send the data to a remote system. A program on the remote system receives the data and directs it into the remote system's LP spooler.



LG200139_001a

Figure 10-1. Line Printer Spooler "Plumbing Diagram."

The Components of the LP Spooler

The components of the lp spooler are:

- printer names
- printer classes
- print destinations
- system default printer
- printer interfaces
- printer models
- device files
- line printer scheduler
- local printer
- remote printer
- print request identification number

Printer Names

When you configure a printer into the lp spooler, you assign it a name that you will use to refer to it when you later submit print requests. This name is referred to as the **printer name**. Printer names can contain up to 14 characters, which can be alpha-numeric or underscores. The name may or may not be the same as the device file name. Some correspondence between the printer name and device file name is suggested. The printer name is the name of the printer that shows up when you request the status of the printer queue with the `lpstat` command.

A hypothetical system "hypo1" has the following printers defined in its lp spooler. The printers have the following names:

- laser1
- laser2
- phred
- letterhead
- invoices
- check_printer

Printer Classes

You can treat a *group of printers* as if they were one printer. A **printer class** is a name that you can use to refer to the *group of printers*. When submitting a print request you can specify a particular printer name or a printer class name. When submitting a print request to a printer class, the print requests will print on the first available printer in the group rather than on a specific printer. Printers that are members of a printer class can still be referenced individually. Creating a printer class is optional.

On the hypothetical system "hypo1," three of the printers are grouped into a printer class called "laser".

printer class: laser
- laser1
- laser2
- phred

- Printer class names can contain up to 14 characters, which can be alpha-numeric or underscores.
- Printer class names and printer names on the same system cannot be the same name. Printer names and class names must all be unique.
- Printer classes cannot include remote printers.
- A printer class must contain at least one printer.
- A printer can only belong to one printer class at a time.
- To remove a printer from a printer class, you must remove the printer from the lp spooler and re-add without specifying a printer class.

LP Spooler Overview

Print Destinations

Several of the commands for the lp spooler require you to specify a print **destination**. A destination is the name of a printer or printer class.

For our example system "hypo1", possible destinations are:

- printer class: laser
- laser1
- laser2
- phred
- invoices
- check_printer
- letterhead

System Default Printer (Destination)

You can appoint one of the print destinations in your lp spooler to be the system default printer. It is not necessary to have a system default printer, but it is recommended. A system default printer receives any print requests that are not sent to a specific print destination. You can have only one *system* default printer.

In addition to, or instead of, a *system* default printer, you can assign each *user* a default printer to use. To do this, simply set the user's LPDEST shell environment variable to the name of the system default printer.

- If LPDEST *is* set and a user does not specify a different printer to use, the printer referenced by LPDEST will be used.
- If LPDEST *is not* set for a user, and the user does not specify a printer, the system default printer (if one is set) will be used.
- If neither LPDEST or the system default printer is set, a user must specify a printer (or printer class).

Printer Interfaces

A printer interface, also known as an **interface script**, is the final stage of the lp spooler. It is the part of the lp spooler that is responsible for sending data to a printer. Each printer that you have defined for use by the lp spooler has its own interface script (shell script) that resides in the `/usr/spool/lp/interface` directory. When printers are added to the lp spooler, an interface script is copied from `/usr/spool/lp/model` to `/usr/spool/lp/interface` and given the printer name.

If we were to list the directory `/usr/spool/lp/interface` on our hypothetical system "hypo1," it would contain the printer interface files `laser1`, `laser2`, `phred`, `letterhead`, `invoices`, and `check_printer`.

The entry for the class name `laser` would be located in the directory `/usr/spool/lp/class`; it would not be found in the interface directory.

Printer Models

There are printer interface script "models" you can choose from that have been created for you in the `/usr/spool/lp/model` directory. Many of them have names that match the model numbers of Hewlett Packard printers and plotters.

When you configure your printer into the lp spooler, you must specify which printer model interface script you want to use. The model will be automatically copied from the `/usr/spool/lp/model` directory into the `/usr/spool/lp/interface` directory and given the name that you specified as your printer name (see "Printer Names").

If you list the `/usr/spool/lp/model` directory, it should look similar to this:

```
HPGL1          draftpro       hp2560         hp2932a        hp7596a
HPGL2          dumb           hp2563a        hp2934a        laserjet
PCL1           dumbplot       hp2564b        hp33440a       laserjetIIISi
PCL2           fonts          hp2565a        hp33447a       paintjet
PCL3           hp2225a        hp2566b        hp3630a        quietjet
PCL4           hp2225d        hp2567b        hp7440a        rmodel
PRINT3K.model  hp2227a        hp2631g        hp7475a        rmttroff
bf_remote      hp2228a        hp2684a        hp7550a        ruggedwriter
colorpro       hp2235a        hp2686a        hp7570a        thinkjet
deskjet        hp2276a        hp2686a.pif    hp7595a
```

LP Spooler Overview

If you have an HP printer, you will probably find a model script that matches its model number or name. Those interface model scripts that match your printers typically do not need to be changed. If you know how to do shell programming, you can customize printer interface model scripts to meet your specific printing needs (see *Shells: User's Guide* for information on shell programming).

Note Use the `PCL4` model script for the HP C2753A Model F100 Printer.

Caution The `update` program described in the *Installing and Updating HP-UX* manual can replace or remove model scripts in the process of updating your system. If you create your own printer interface scripts, keep the file names unique and keep a backup copy somewhere on the system.

If you *do not* have an HP printer, try using the `dumb` interface model. You might have to modify it to be able to use all of the features of your non-HP printer, but "dumb" should work for basic ASCII text printing. If the `dumb` printer interface model script does not work, contact your printer supplier for a UNIX$_{TM}$ line printer spooler interface script or try the script that most closely matches your non-HP printer type.

Device Files

Device files are *not* part of the lp spooler; they are special files that define the necessary device driver and hardware address needed to communicate with a particular physical device (in this case a printer). The printer name referred to by the lp spooler and the name of the device file for a printer are not required to be the same, but a correspondence is recommended.

You can create printer device files using SAM or HP-UX commands when you add a printer to the lp spooler. SAM creates a device file for you. If necessary, SAM can override the default device file naming convention. For information and specific instruction about how to make device files for your printers, see *How HP-UX Works: Concepts for the System Administrator*, HP part number B2355-90029 manual, Chapter 12, "HP-UX Peripherals" or the *Installing Peripherals* manual.

When you configure a printer into your lp spooler, you must supply the name of your printer's device file.

Line Printer Scheduler

The line printer scheduler is the heart of the lp spooler. It is the part of the lp spooler that prevents intermixed listings (output from more than one print request mixed together on a printed page) and controls flow of print requests to the printers. Its duties also include monitoring printer and print request priorities, monitoring/adjusting printer status, and logging lp spooler activities. The `lpsched` command starts the LP spooler. Because of the central role it plays, starting `lpsched` is referred to as "starting the LP spooler", and stopping `lpsched` is often referred to as "stopping the LP spooler." You can use the `lpsched` command directly or through SAM (see "Starting and Stopping the LP Spooler Using HP-UX Commands" and "Starting and Stopping the LP Spooler Using SAM").

Local Printer

A **local printer** is a printer that is physically connected to your system.

LP Spooler Overview

Remote Printer

A **remote printer** is a printer that is not physically connected to your system, but can be accessed by your system through a local area network (LAN). To configure a remote printer into your local lp spooler, you must be able to access the remote system via a LAN. The process of adding a remote printer is similar to that of adding a local printer, though you will need to supply some slightly different information. See "Adding a Remote Printer Using SAM" and "Adding a Remote Printer Using HP-UX Commands".

Network-Based Printer/Plotter

A network-based printer or plotter is connected directly to the local area network (LAN). A network-based printer or plotter is *not* physically connected to any system. This chapter provides instructions for setting up a network-based printer by means of SAM (see "Adding a Network-Based Printer Using SAM"). If you do not prefer to use SAM, consult the instructions shipped with the printer or printer interface card product.

Print Request Identification Number

When you submit a print request by means of the `lp` command, `lp` responds with a **print request identification number** consisting of the name of the printer (or printer class) followed by a number. Here are some examples of typical print request identification numbers:

```
laser-3456
phred-2152
letterhead-1547
```

Remote Spooling

If you have several systems connected to a Local Area Network (LAN) and would like the systems to share the use of a printer, you can set up the lp spoolers of the systems that are not physically connected to the printer to automatically send their print requests (via the LAN) to the lp spooler of the system that does have the printer. The systems without printers act as though they were *a user* on the system with the printer, submitting print requests to that system's lp spooler. This is accomplished by a special program known as the **Remote Spooling Daemon** (`rlpdaemon`).

The `rlpdaemon` program runs in the background (on the system with the printer) monitoring the incoming LAN traffic for any remote print requests from other systems. When these requests arrive, the `rlpdaemon` program submits them to its local lp spooler on behalf of the remote user. In addition to remote print requests, the remote spooling daemon must also handle "cancel" and "status" requests from remote systems.

There are special "interface scripts" on the remote systems that issue cancel and status requests. These special interface scripts have a lot in common with printer interface scripts. They have a model directory that can hold sample versions of these scripts, and they have an interface directory where the scripts currently in use reside. The cancel and status models are copied into their respective interface directories automatically when adding a remote printer.

LP Spooler Overview

The directory `/usr/spool/lp/cmodel` contains a sample interface script, `rcmodel`, that sends a remote cancel command to the system with the printer. When you configure a remote printer into your lp spooler, the **cancel model script** is copied into the `/usr/spool/lp/cinterface` directory and is given the same name as the printer.

The directory `/usr/spool/lp/smodel` contains a sample of an interface script, called `rsmodel`, which sends a remote status command to the system with the printer. When you configure a remote printer into your lp spooler, the **status model script** is copied into the `/usr/spool/lp/sinterface` directory and is given the same name as the printer.

It is unlikely that you will need to customize the remote cancel and status model scripts. If you do customize these "remote control" scripts, you must copy them to a different file name to avoid destroying your changes when updating the system with the `update` utility.

Configuring a remote printer into your lp spooler requires additional information beyond what is needed to configure a local printer. In addition to the information you normally supply when configuring a local printer into your lp spooler, you will need to tell your system:

- The name of the system with the printer
- The interface script to use when it issues a remote cancel request
- The interface script to use when it issues a remote status request
- The name of the printer (as it is defined in the lp spooler of the *remote system*) where you want your printouts to be printed.

See "Adding a Remote Printer Using SAM" and "Adding a Remote Printer Using HP-UX Commands".

Priorities of Printers and Print Requests

To control the order of printed requests, you can assign priority values to printers and to specific print requests. Assigning priorities is *NOT* required.

- Priority values must be in the range of 0 to 7.
- Priority 7 is the highest priority.
- A value assigned to each printer, known as a **printer fence priority**, determines the minimum priority that a print request must have in order to be able to print. A print request having a priority equal to or greater than the fence priority of its printer will print. HP-UX assigns a printer fence priority value of zero (0) when you add a printer to the lp spooler. You can change printer fence priorities dynamically with SAM or the `lpfence` command. See "Changing A Printer Fence Priority Using SAM" or "Setting a Printer Priority Fence Using HP-UX Commands".
- A value assigned to each print request, known as a **print request priority**, is associated with the destination printer. The print request priority for each printer can be determined when each printer is added to the line printer spooling system. If the printer print request priority is changed after a print request has been put in the print queue, the print request's priority does not change.
- Print request priorities *lower* than the printer priority will not print. If a print request's priority is lower than its printer's priority, it will remain in the request directory ("printer queue") for that printer. It will remain there until its priority is raised or its printer's priority is lowered to allow it to print (or until the request is canceled).
- You cannot directly set a printer class priority. See "Printer Classes" for an example of a printer class. The class priority is the same as the highest priority of any printer in the class.
- If multiple print requests are waiting to be printed on a specific printer and all have priorities high enough to print:
  - The lp spooler will print next the print request with the highest priority.
  - If more than one print request has the highest priority, all print requests with that priority will print in the order they were received by the lp spooler.

Using Plotters with the LP Spooler

Because the lp spooler is nothing more than a data routing mechanism, it can be used with other output devices. Apart from printers, the devices most commonly used with the lp spooler are plotters. The following model scripts are supplied so that you can use your lp spooler with Hewlett Packard plotters:

Table 10-1. LP Spooler Models for Plotters

Script Name	Plotters it can be used with
HPGL1:	HP7440A, HP7475A
HPGL2:	HP7550A, HP7595A, HP7596A, HP7570A
colorpro:	HP7440A, HP7475A
draftpro:	HP7550A, HP7595A, HP7596A, HP7570A
dumbplot:	miscellaneous
hp7440a:	HP7440A, HP7475A
hp7475a:	HP7440A, HP7475A
hp7550a:	HP7550A, HP7595A, HP7596A, HP7570A
hp7570a:	HP7550A, HP7595A, HP7596A, HP7570A
hp7595a:	HP7550A, HP7595A, HP7596A, HP7570A
hp7596a:	HP7550A, HP7595A, HP7596A, HP7570A

Controlling Data Flow Through the LP Spooler

There are three points in the lp spooler where you can control the flow of data:

1. You can start or stop the LP spooler. This has a global effect. If you stop the LP spooler, printing for *all printers* stops.
2. You can tell the lp spooler to accept or reject any new print requests for a printer. If you instruct the lp spooler to reject print requests for a printer class, users will be given a message telling them that the printer class that they requested is not accepting requests when they attempt to print something to that destination. Rejecting print requests should be used when a printer or a class of printers is being taken off the system for a extended period of time. Rejecting print requests is not recommended for making the printer unavailable for a short time. For example, rejecting print requests is not recommended for adding paper or changing the toner cartridge. A minor delay due to these short term services is usually acceptable.
3. You can tell the lp spooler to enable or disable a printer for printing. Print requests continue to be accepted for the disabled printer unless you have explicitly rejected print requests. Disable a printer should to make the printer temporarily unavailable for a short time, for example, disabling the printer to add paper or change toner. Do not disable a printer for a long time without also rejecting requests for that printer; otherwise users' print requests will keep accumulating in the print queue and they will not get any notice that their requests will not print. Once you reject print requests for a printer, a user submitting a print request to that printer will get a message stating that the printer is not accepting requests.

   To print, a printer must be accepting and enabled.

Note When you use SAM to "enable" or "disable" a printer, SAM performs *both* the accept/reject operation *and* the enable/disable operation listed above. If you wish to "disable" a printer but still accept requests for that printer (letting them accumulate in the request directory for the printer), you must use the HP-UX commands method.

Logging and Analyzing Printer Activity

Analyzing printer activity can help you determine if there are bottlenecks in your lp spooler. It can also help you determine/justify the need to add additional printers to your lp spooler. There are facilities to help you analyze the flow of data through your lp spooler.

There are two phases to analyzing lp spooler activity: a *data collection* phase and a *data reporting* phase. The data collection phase begins when the lp spooler starts. The `-a` option to the `lpsched` command turns on the data collection processes when you start the LP spooler (see "Starting and Stopping the LP Spooler Using HP-UX Commands"). The data reporting phase can occur any time after the lp spooler has been started. The following statistics are calculated:

- average waiting time from when a print request is submitted to the start of printing
- standard deviation for waiting time
- average printing time from start to end of print request
- standard deviation of printing time
- average number of bytes (characters) printed per request
- standard deviation for number of bytes
- sum of bytes printed for all requests in Kbytes
- total number of requests since logging started

See "Displaying Statistics about Printer Activity Using HP-UX Commands".

Initial LP Spooler Set Up

Initial LP spooler setup consists of the following tasks:

1. Add at least one printer to the lp spooler.
2. Tell the lp spooler to accept print requests for this printer.
3. Tell the lp spooler to enable the printer for printing.
4. Turn on the LP spooler.

When you use SAM to add a printer, SAM:

- tells the lp spooler to accept print requests for the printer.
- enables the printer.
- starts the lp spooler.

If you are not using SAM, you must do these tasks yourself; refer to "Setting Up the LP Spooler Using HP-UX Commands".

Line Printer Spooler Tasks

The two methods of controlling the lp spooler are:

1. The System Administration Manager (SAM)
2. HP-UX commands

SAM allows you to control the lp spooler through its menu-selection and data-entry screens. By combining multiple "manual commands" into single tasks, SAM can save you time and keystrokes. SAM also eliminates the need to know command names and options for the lp spooler.

Note SAM does not support the HP C2753A Model F100 Printer. Use the HP-UX commands method to add this printer to the lp spooler. This printer uses the `PCL4` model script.

Although HP-UX commands require you to learn more details than SAM does, you might need or prefer to use HP-UX commands, for the following reasons:

- HP-UX commands give you a greater degree of control over the lp spooler.
- SAM might not be configured into your system. You *have* to use HP-UX commands to control the lp spooler.
- You might be more comfortable using HP-UX commands.
- You may need to use the data collection facility. If you want to start data collection, you must use the `lpsched -a` command, not SAM, to start the lp spooler. See "Starting and Stopping the LP Spooler Using HP-UX Commands".

See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

Viewing Printers and Print Request Status Using SAM

To view printers:

1. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

2. Highlight `Printers and Plotters` and activate the [Open] control button.

3. Highlight `Printers/Plotters` and activate the [Open] control button.

To view print requests:

1. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

2. Highlight `Printers and Plotters` and activate the [Open] control button.

3. Highlight `Print Requests` and activate the [Open] control button.

You can also view print requests by choosing `Print Requests` from the "List" menu within the "Printer/Plotter Manager" Window.

Additional Task Information

SAM provides an on line help system to assist you when you need additional information.

Activating the Help button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the f1 key gives you context-sensitive information for the object at the location of the cursor.

Note The print request queue can change rapidly. To ensure that you are viewing the current data, choose `Refresh List` from the "Options" menu to view the current state of the print request queue.

The "Printer/Plotter Manager" object list displays the following information about the printers in the lp spooler:

- system default printer
- status of LP spooler (RUNNING or STOPPED)
- the printer name
- printer status (`enabled`, `disabled`, `idle`, `busy`)
- priority for each printer and printer class
- the printer type (`local`, `remote`, or `network`)
- the location of each printer (the device file for local printers; printer name and system for remote printers; no entry for network-based printers)

Adding a Local Printer Using SAM

Note SAM does not support the HP C2753A Model F100 Printer. Use the HP-UX commands method to add this printer to the lp spooler.

To add a local printer:

1. Physically connect the printer(s) to your system. Refer to the instructions shipped with your printer. You should always shut down your system and turn off the power when you are changing hardware configuration of your system. For additional configuration information refer to the *Installing Peripherals* manual.
2. Gather the following information:
   - The name you are giving to this printer or plotter. Printer names can be up to 14 characters in length, and the characters must be alphanumeric (A-Z, a-z, 0-9) or an underscore (_).
   - The name of the device file that the printer or plotter will use. SAM creates the device file for you. SAM uses the default device file named `lp_`*printer-name*. You can override the default device file name by specifying your device file name when filling in the printer information.
   - The model script from the `/usr/spool/lp/model` directory, for example, `laserjetIIISi` for an HP LaserJet IIISi.
   - The print request priority for this printer. The default is zero (0).
   - The class to which the printer or plotter will be added (optional). Printer class names can be up to 14 characters in length, and the characters must be from the set (A-Z, a-z, 0-9). The underscore (_) character is allowed in printer class names.

   In addition, decide whether or not to make this device your system's default printer.

Adding a Local Printer Using SAM

3. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

Note During the process of adding and removing printers, SAM stops the lp spooler. Print requests being printed at the time the lp spooler stopped might not complete successfully. It is best to add a printer when there are no requests currently printing.

---

4. Highlight `Peripheral Devices` and activate the [Open] control button.
5. Highlight `Printers and Plotters` and activate the [Open] control button.
6. Highlight `Printers/Plotters` and activate the [Open] control button.
7. Choose `Add a Local Printer/Plotter >` and the menu item associated with the printer interface type from the "Actions" menu.

Note The printer printer driver must be part of the kernel to add the printer to the lp spooler. If the printer driver is not currently configured into the kernel, SAM prompts you to add the driver(s) and reboot the system.

You should be aware of the effects on other users before rebooting your system. Note especially the following situations:

- If anyone else is logged into your system, rebooting will interrupt their work.
- If your system is an Internet Protocol router, rebooting it affects any IP traffic routed through your system.

---

8. Highlight the interface to which you connected the printer and fill in and additional information (port number or bus address) and activate the [OK] control button.

   If an interface entry is not listed, activate the [Diagnose Missing Card] control button.

9. Fill in the printer interface dialog box fields, choose from the menu button values, and turn on and off check box values.

   Activating the [Help] button from a dialog or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   Pressing the [f1] key gives you context-sensitive information for the object field at the location of the cursor.

10. Activate the [OK] control button.

Additional Task Information

SAM provides an on line help system to assist you when you need additional information.

Activating the [Help] button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the [f1] key gives you context-sensitive information for the object at the location of the cursor.

Adding a Local Printer Using SAM

The printer driver names, major number, and interface types are as follows:

Table 10-2. Series 800 Printer Drivers

Printer Driver Name	Major Number	Printer Driver Description
`lpr2`	26	HP-PB parallel driver
`mux2`	58	HP-PB serial driver for the HP 40299A interface card
`lan1`	51	HP-PB LAN driver
`instr0`	21	CIO HP-IB driver
`lpr0`	26	CIO HP-IB CIPER protocol printers
`lpr1`	26	CIO HP-IB AMIGO protocol printers
`mux0`	1	CIO Serial driver for the HP 98196 interface card
`mux0_16`	1	CIO Serial driver for the HP 98190A 16-channel Multiplexer
`lan0`	50	CIO LAN driver

Refer to the Chapter 2, "Constructing an HP-UX System" chapter of this manual for more details on configuring the kernel using SAM.

Adding a Remote Printer Using SAM

1. Ensure that the remote system has the printer installed and configured into the remote system's line printer spooler system.
2. Gather the following information:
   - The name you are giving to this printer or plotter. See "Printer Names".
   - Whether or not you wish to make this device your system's default printer. See "System Default Printer (Destination)".
   - The name of the remote system to which the printer or plotter is attached.
   - The name of the remote printer or plotter.
   - The "cancel" model on the remote system (optional). See "Remote Spooling".
   - The "status" model on the remote system (optional). See "Remote Spooling".
   - Whether or not you wish to allow any user to cancel any printing request.
   - Whether or not the remote printer is on a system using BSD (Berkeley Software Distribution) UNIX. Using BSD disables any `lp -o`*parm* options. BSD systems do not understand the `-o` option.

Adding a Remote Printer Using SAM

3. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

Note During the process of adding and removing printers, SAM stops the lp spooler Print requests being printed at the time the lp spooler stopped might not complete successfully. It is best to add a printer when there are no requests currently printing.

---

4. Highlight `Peripheral Devices` and activate the (Open) control button.
5. Highlight `Printers and Plotters` and activate the (Open) control button.
6. Highlight `Printers/Plotters` and activate the (Open) control button.
7. Choose `Add a remote printer/plotter >` and the menu item associated with the printer interface type from the "Actions" menu.
8. Fill in the printer interface dialog box fields and turn on off check box values.

   Activating the (Help) button from a dialog or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   Pressing the (f1) key gives you context-sensitive information for the object field at the location of the cursor.
9. Activate the (OK) control button.

Additional Task Information

SAM provides an on line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

To configure a remote printer into your lp spooler, you must be able to access the system with the printer via a local area network (LAN). The process of adding a remote printer is similar to that of adding a local printer, though you will need to supply SAM with some slightly different information.

Remote printers cannot be members of a printer class.

Adding a Network-Based Printer Using SAM

To add a network-based printer or plotter using SAM:

1. Ensure that the printer is connected to the network according to the installation instructions shipped with the network-based printer or the network interface card for the printer.
2. Gather the following information:
   - The name you are giving to this printer or plotter. See "Printer Names".
   - The printer node name.
   - The model or interface that the printer will use. See "Printer Models".
   - The link-level address of the network card installed in the printer.
   - The TCP-IP protocol printer requires an Internet Protocol (IP) address.
   - The priority for this printer. See "Priorities of Printers and Print Requests".
   - The class to which the printer or plotter will be added (optional). See "Printer Classes".

   In addition, decide whether or not you wish to make this device your system's default printer. See "System Default Printer (Destination)".
3. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.
4. Highlight `Peripheral Devices` and activate the [Open] control button.
5. Highlight `Printers and Plotters` and activate the [Open] control button.
6. Highlight `Printers/Plotters` and activate the [Open] control button.
7. Choose `Add a network-based printer` then `Add TCP-IP protocol printer...` or `Add NPX protocol printer...` from the "Actions" menu.

8. Fill in the printer interface dialog box fields and turn on and off check box values.

   Activating the (Help) button from a dialog or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

   Pressing the (f1) key gives you context-sensitive information for the object field at the location of the cursor.

9. Activate the (OK) control button.

Additional Task Information

SAM provides an on line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Ensure that the `NET-PERIPH` file set is loaded on your system. To load this additional software from your operating system media, refer to the *Installing and Updating HP-UX* manual.

Removing a Printer Using SAM

1. Run SAM; type:

   /usr/bin/sam

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

2. Highlight Peripheral Devices and activate the Open control button.

3. Highlight Printers and Plotters and activate the Open control button.

4. Highlight Printers/Plotters and activate the Open control button.

5. Highlight the printer you want to remove in the object list.

6. Choose Remove a printer/plotter > from the "Actions" menu.

Additional Task Information

SAM provides an on line help system to assist you when you need additional information.

Activating the Help button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the f1 key gives you context-sensitive information for the object at the location of the cursor.

Note

- During the process of adding and removing printers, SAM stops the lp spooler. Print requests being printed at the time the lp spooler is stopped might not complete successfully. It is best to stop the lp spooler when there are no requests currently printing.
- SAM cancels all print requests in the request directory for the printer you are removing.

SAM does not remove the device file for the printer removed from the lp spooler.

Starting and Stopping the LP Spooler Using SAM

To *start* the LP spooler:

1. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

2. Highlight `Peripheral Devices` and activate the Open control button.

3. Highlight `Printers and Plotters` and activate the Open control button.

4. Highlight `Printers/Plotters` and activate the Open control button.

5. Choose `Start up printer spooler` from the "Actions" menu.

To *stop* the LP spooler:

1. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

2. Highlight `Peripheral Devices` and activate the Open control button.

3. Highlight `Printers and Plotters` and activate the Open control button.

4. Highlight `Printers/Plotters` and activate the Open control button.

5. Choose `Shut down printer spooler` from the "Actions" menu.

Additional Task Information

SAM provides an on line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Note

- Printing on all printers stops.
- When SAM stops the lp spooler there is no guarantee that print requests being printed at the time will complete successfully. It is best to stop the lp spooler when there are no requests currently printing.

To turn on the data collection processes, refer to "Starting and Stopping the LP Spooler Using HP-UX Commands".

Determining the Status of the LP Spooler Using SAM

To determine the status of the lp spooler:

1. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

2. Highlight `Peripheral Devices` and activate the (Open) control button.

3. Highlight `Printers and Plotters` and activate the (Open) control button.

4. Highlight `Printers/Plotters` and activate the (Open) control button.

The status area of the object list will display the status of the scheduler as "`Scheduler: RUNNING`" or "`Scheduler: STOPPED`".

Additional Task Information

SAM provides an on line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Disabling a Printer Using SAM

1. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

2. Highlight `Peripheral Devices` and activate the (Open) control button.

3. Highlight `Printers and Plotters` and activate the (Open) control button.

4. Highlight `Printers/Plotters` and activate the (Open) control button.

5. Highlight the printer you want to disable in the object list.

6. Choose `Disable printer` from the "Actions" menu.

Additional Task Information

SAM provides an on line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Disabling a Printer Using SAM

- It is best to disable printer when there are no requests currently printing.

Note When you use SAM to "enable" or "disable" a printer, SAM performs *both* the accept/reject operation *and* the enable/disable operation. If you wish to "disable" a printer but still accept requests for that printer (letting them accumulate in the request directory for the printer), you must use the HP-UX commands method to disable the printer (see "Enabling or Disabling a Printer Using HP-UX Commands").

Enabling a Printer Using SAM

To enable a printer using SAM:

1. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.

2. Highlight `Peripheral Devices` and activate the (Open) control button.
3. Highlight `Printers and Plotters` and activate the (Open) control button.
4. Highlight `Printers/Plotters` and activate the (Open) control button.
5. Highlight the printer you want to enable in the object list.
6. Choose `Enable printer` from the "Actions" menu.

Additional Task Information

SAM provides an on line help system to assist you when you need additional information.

Activating the (Help) button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the (f1) key gives you context-sensitive information for the object at the location of the cursor.

Enabling a Printer Using SAM

Note When you use SAM to "enable" or "disable" a printer, SAM performs *both* the accept/reject operation *and* the enable/disable operation. If you wish to "disable" a printer but still accept requests for that printer (letting them accumulate in the request directory for the printer), you must use the HP-UX commands method to disable the printer (see "Enabling or Disabling a Printer Using HP-UX Commands").

Changing A Printer Fence Priority Using SAM

To change a printer priority using SAM:

1. Run SAM; type:

   `/usr/bin/sam`

   See Chapter 1, "Introduction to System Administration" for additional information about using SAM.
2. Highlight `Peripheral Devices` and activate the [Open] control button.
3. Highlight `Printers and Plotters` and activate the [Open] control button.
4. Highlight `Printers/Plotters` and activate the [Open] control button.
5. Highlight the printer for which you want to change the priority.
6. Choose `Modify fence priority` from the "Actions" menu.
7. Choose the new priority value from the `Printer priority` menu button.
8. Activate the [OK] control button.

Additional Task Information

SAM provides an on line help system to assist you when you need additional information.

Activating the [Help] button from the SAM main window, a dialog box, or message box gives you information about the attributes and tasks you can perform from the currently displayed window.

From within a functional area, choosing an item from the "Help" menu gives you information about:

- the current functional area
- keyboard navigation within SAM
- using the SAM help system
- displaying the version of SAM you are currently running

From a dialog box (a window displaying fields to be filled in), pressing the [f1] key gives you context-sensitive information for the object at the location of the cursor.

See "Priorities of Printers and Print Requests" for additional information.

Setting Up the LP Spooler Using HP-UX Commands

1. Add at least one printer to the lp spooler (see "Adding a Local Printer Using HP-UX Commands" or "Adding a Remote Printer Using HP-UX Commands").
2. Tell the lp spooler to accept print requests for this printer (see "Accepting and Rejecting Print Requests for a Printer Using HP-UX Commands").
3. Tell the lp spooler to enable the printer for printing (see "Enabling or Disabling a Printer Using HP-UX Commands").
4. Start the LP spooler (see "Starting and Stopping the LP Spooler Using HP-UX Commands").

Ensuring the Printer Driver is in the Kernel Using HP-UX Commands

Reconfiguring the kernel requires that you reboot your system. Note the following impact on other users *before* you shut down and reboot your system:

- If others are logged into your system, rebooting it interrupts their work. If you have a small number of users on your system, it is best to notify your users in person of the impending system shutdown. It is possible that users can be using an application and not be aware of the message sent by the `shutdown` command.
- If your system is an Internet Protocol router, rebooting it affects any IP traffic routed through your system.

To ensure the printer driver is part of the kernel using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Change your directory to `/etc/conf/gen`:

```
cd /etc/conf/gen
```

Caution You *must* get out of the root directory if you will be creating a new kernel. Otherwise you will overwrite the current kernel.

Ensuring the Printer Driver is in the Kernel Using HP-UX Commands

3. Look at the entries in the kernel configuration description file to determine if the printer driver is part of your current kernel configuration. The S800 is the configuration description file that generally reflects your system.

   If the printer driver *is* part of the current kernel configuration, exit this task module; otherwise, complete the steps detailed in the Chapter 2, "Constructing an HP-UX System" to add the printer driver to the kernel. During this process you will need to refer to Table 10-3.

   The printer driver names are described in Table 10-3.

Table 10-3. Series 800 Printer Drivers

Printer Driver Name	Printer Driver Description
`lpr2`	HP-PB parallel driver
`mux2`	HP-PB serial driver for the HP 40299A interface card
`lan1`	HP-PB LAN driver
`instr0`	CIO HP-IB driver
`lpr0`	CIO HP-IB CIPER protocol printers
`lpr1`	CIO HP-IB AMIGO protocol printers
`mux0`	CIO Serial driver for the HP 98196 interface card
`mux0_16`	CIO Serial driver for the HP 98190A 16-channel Multiplexer
`lan0`	CIO LAN driver

Determining if a Device File Exists for your Printer Using HP-UX Commands

1. Use the *Installing Peripherals* manual to help you determine what the minor numbers for your printer should be (based on the printer's interface and hardware address). The printer driver names, major number, and interface types are as follows:

Table 10-4. Series 800 Printer Drivers

Printer Driver Name	Major Number	Printer Driver Description
`lpr2`	26	HP-PB parallel driver
`mux2`	58	HP-PB serial driver for the HP 40299A interface card
`lan1`	51	HP-PB LAN driver
`instr0`	21	CIO HP-IB driver
`lpr0`	26	CIO HP-IB CIPER protocol printers
`lpr1`	26	CIO HP-IB AMIGO protocol printers
`mux0`	1	CIO Serial driver for the HP 98196 interface card
`mux0_16`	1	CIO Serial driver for the HP 98190A 16-channel Multiplexer
`lan0`	50	CIO LAN driver

2. Use the `ll` command to list the directory `/dev`. Look through the entries for one that matches the major *and* minor numbers. The fifth column of information (immediately to the right of the group ownership) represents the major number for the corresponding device file. The sixth column (immediately to the left of the date) represents the minor number for the device file.

Determining if a Device File Exists for your Printer Using HP-UX Commands

3. If you find one that matches, note its device file name (last column of information in the `ll` listing) for use with the HP-UX commands to add a printer to your spooling system (later in this chapter).

   If you do *not* find one with major *and* minor numbers that match, you will need to create a device file for your printer. See the *Installing Peripherals* for the procedure on how to do this.

Additional Task Information

A device file is the mechanism that HP-UX uses to determine which of the devices attached to your computer it should use for an I/O operation. The major number of the device file tells HP-UX which drivers to use; the minor number tell HP-UX the hardware address of the device. Device files are usually located in the `/dev` directory.

Adding a Local Printer Using HP-UX Commands

To add a local printer using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Ensure the appropriate printer driver is in your kernel (see "Ensuring the Printer Driver is in the Kernel Using HP-UX Commands").
3. Ensure a device file exists for the printer (see "Determining if a Device File Exists for your Printer Using HP-UX Commands").
4. Stop the LP spooler with the `lpshut` command:

   ```
   /usr/lib/lpshut
   ```

Note When the lp spooler is stopped there is no guarantee that print requests currently printing will complete successfully. It is best to stop the lp spooler when there are no requests currently printing.

5. Add the printer to the lp spooler with the `lpadmin` command. There is an example later in this section. The command has the following syntax:

   `/usr/lib/lpadmin -p`*pname* `-v`*devfile* `-m`*model* [`-d`] [`-g`*priority*] [`-c`*class*]

   where:

   pname is the name that you use to refer to this printer when using the various lp spooler commands (required). Printer names can be up to 14 characters in length, and the characters must either be alphanumeric (A-Z, a-z, 0-9) or an underscore (_).

   devfile is the name of the device file to be used to communicate with this printer (required).

model is the script you would like to "model" your printer's interface script after. The `lpadmin` command will make a copy of this model script and place it in the directory `/usr/spool/lp/interface`, with the name you specified in the `-p` option described above (required).

Note For the HP C2753A Model F100 Printer, use the `PCL4` model script.

`-d` specifies that you want this printer to be the system default printer.

priority The minimum priority a print request will need in order to print on this printer (optional). The default value is zero (0), which permits any print request to print on this printer. See "Priorities of Printers and Print Requests".

class The name of the group of printers that this printer is a member (optional). Printer class names can be up to 14 characters in length, and the characters must either be alphanumeric (A-Z, a-z, 0-9) or an underscore (_).

When using the `lpadmin` command, *do not* put any spaces between the options and their respective values. For example:

TYPE THIS:

`-p`*invoices*

NOT THIS:

`-p` *invoices*

6. Allow print requests to enter the request directory for the newly added printer with the `accept` command:

   `/usr/lib/accept` *pname*

   where:

   pname is the name you gave to this printer in the `lpadmin` command.

   See "Accepting and Rejecting Print Requests for a Printer Using HP-UX Commands".

7. Enable the newly added printer to process print requests with the `enable` command:

   `/usr/bin/enable` *pname*

   where:

   pname is the name given to refer to this printer.

   See "Enabling or Disabling a Printer Using HP-UX Commands".

8. Start the line printer scheduler with the `lpsched` command:

   `/usr/lib/lpsched`

   See "Starting and Stopping the LP Spooler Using HP-UX Commands".

Additional Task Information

A **local printer** is a printer that is physically connected to your system.

Adding a printer to the lp spooler is not the same thing as adding a printer to your system. The first involves connecting the printer to your computer and configuring HP-UX to communicate with the printer. The second involves configuring the software subsystem (known as the LP spooler) that manages printer requests.

Because `lpadmin` is constructing and modifying files that are used by the line printer scheduler, it is important that the scheduler is stopped when you use the `lpadmin` command to add a new printer. Therefore, you should be sure that `lpsched` is stopped *before* using the `lpadmin` command.

Examples

To determine the status of the lp spooler:

```
/usr/bin/lpstat -r
scheduler is running
```

To stop the lp spooler:

```
/usr/lib/lpshut
```

To two add printers named `invoices` and `check_printer` to the lp spooler:

```
/usr/lib/lpadmin -pinvoices -v/dev/ivprint -mhp2934a
/usr/lib/lpadmin -pcheck_printer -v/dev/ckprint -mhp2564b -g7
```

To enable the print request directories to accept printer requests:

```
/usr/lib/accept invoices
/usr/lib/accept check_printer
/usr/lib/accept newclass
```

`accept` *applies to printer classes also*

To permit the printers to process print requests:

```
/usr/bin/enable invoices
/usr/bin/enable check_printer
```

To restart the lp spooler:

```
/usr/lib/lpsched
```

Adding a Remote Printer Using HP-UX Commands

To add a remote printer using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Stop the LP spooler with the `lpshut` command:

   ```
   /usr/lib/lpshut
   ```

Note It is best to stop the LP spooler when there are no requests currently printing.

3. Add the printer to the lp spooler using the `lpadmin` command:

   ```
   /usr/lib/lpadmin -ppname -vdevfile -mmodel [-d] [-gpriority] [-ocmcmodel] \
   [-osmsmodel] [-ormremsys] [-orprpname] [-ob3] [-orc]
   ```

   where:

   pname This is the name that you will use to send print requests to this printer. Printer names can be up to 14 characters in length, and the characters must either be alphanumeric (A-Z, a-z, 0-9) or an underscore (_).

   devfile Since the printer is not physically connected to your local system, use the `/dev/null` device file.

   model The remote model script is the `/usr/spool/lp/model/rmodel`. A copy of this file will be put in the `/usr/spool/interface` directory with the name you specified in *pname*.

   `-d` specifies that you want this printer to be the system default printer.

   priority You only need to use this option if you want your printer to have a priority other than zero (optional). See "Changing the Priority of Print Requests Using HP-UX Commands".

Adding a Remote Printer Using HP-UX Commands

cmodel The cancel model script `/usr/spool/lp/cmodel/rcmodel` is used to forward a "cancel" request over to the remote system's lp spooler. The lp spooler copies "/usr/spool/lp/cmodel/rcmodel" to the "/usr/spool/lp/cinterface" directory with the name you specified in *pname*. See "Remote Spooling".

smodel The status model script `/usr/spool/lp/smodel/rsmodel` is used to forward a "status" request over to the remote system's lp spooler. The lp spooler copies "/usr/spool/lp/smodel/rsmodel" to the "/usr/spool/lp/sinterface" directory with the name you specified in *pname*. See "Remote Spooling".

remsys The name of the remote system to which the printer is physically connected. You can get the remote system name by entering the command `hostname` (with no options) on the system with the printer. The name of the remote system must be available to the local system, either from a name server or in the `/etc/hosts` file on the local system (required).

rpname This is the printer name *as it is defined on the remote system.*

When using the `lpadmin` command, *do not* put any spaces between the options and their respective values. `lpadmin` will not interpret your input correctly if you do. For example:

TYPE THIS:

`-p`*invoices*

NOT THIS:

`-p` *invoices*

4. Allow print requests to enter the request directory for the newly added remote printer with the `accept` command:

   `/usr/lib/accept` *pname*

   where:

   pname is the local name given to this remote printer.

   See "Accepting and Rejecting Print Requests for a Printer Using HP-UX Commands".

5. Enable the newly added remote printer to process print requests with the `enable` command:

   `/usr/bin/enable` *pname*

   where:

   pname is the local name given to refer to this remote printer.

   See "Enabling or Disabling a Printer Using HP-UX Commands".

6. Start the line printer scheduler with the `lpsched` command:

   `/usr/lib/lpsched`

   See "Starting and Stopping the LP Spooler Using HP-UX Commands"

Additional Task Information

A **remote printer** is a printer that is not physically connected to your system, but can be accessed by your system through a local area network (LAN). To configure a remote printer into your local lp spooler, you must be able to access the remote system via a LAN. The process of adding a remote printer is similar to that of adding a local printer, though you will need to supply some slightly different information.

Remote printers cannot be members of a printer class.

Adding a printer to the lp spooler is not the same thing as adding a printer to your system. The first involves connecting the printer to your computer and configuring HP-UX to communicate with the printer. The second involves configuring the software subsystem (known as the Line Printer Spooling System) that manages printer output.

Because the `lpadmin` is constructing and modifying files that are used by the line printer scheduler, it is important that the scheduler is stopped when you use the `lpadmin` command to add a new printer.

You only need to use the `-ob3` option if your print request will be printed on or pass through a system that uses the Berkeley Software Distribution (BSD) style lp spooler. BSD systems use three-digit (rather than four-digit) print request-ID numbers (these are the numbers returned when you send something to print). The `-ob3` option disables any `lp -o`*parm* options. BSD systems do not understand the `-o` option to the `lp` command.

Use the `-orc` if you want to restrict users to cancelling only their own print requests.

Examples

To determine the lp spooler status:

```
/usr/bin/lpstat -r
scheduler is stopped
```

To add a remote printer, referred to locally as `letterhead`, physically connected to the system `hypo2` that uses the BSD style print request-ID numbers, and is known on the remote system as "*memos*":

```
/usr/lib/lpadmin -pletterhead -v/dev/null -mrmodel -ocmrcmodel \
   -osmrsmodel -ormhypo2 -ob3 -orpmemos
```

Note Because there are so many options to the commands, these examples use a backslash (\) to represent a line continuation. When you type these commands, you can enter the backslash as shown or you can omit the backslash and type the entire command before pressing Return.

To add a remote printer, referred to locally as *remote_drafts*, physically connect to the system `system13`, known on the remote system as `old_reliable`, and requires a printer priority of 3:

```
/usr/lib/lpadmin -premote_drafts -v/dev/null -mrmodel -ocmrcmodel \
    -osmrsmodel -ormsystem13 -g3 -orpold_reliable
```

To allow print requests to enter the request directory for the newly added remote printers:

```
/usr/lib/accept letterhead
/usr/lib/accept remote_drafts
```

To enable the newly added remote printers:

```
/usr/bin/enable letterhead
/usr/bin/enable remote_drafts
```

To start the line printer scheduler, type:

```
/usr/lib/lpsched
```

Adding a Network-Based Printer Using HP-UX Commands

To add a network-based printer or plotter using HP-UX commands, follow the instructions shipped with the network-based printer or the network interface card for the printer.

Additional Task Information

Ensure that the `NET-PERIPH` file set is loaded on your system. To load this additional software from your operating system media, refer to *Installing and Updating HP-UX*.

Creating a Printer Class Using HP-UX Commands

To create a class of printers, use the `-c` option to the `lpadmin` command when you add a printer to the lp spooler or after you have added several printers to the lp spooler. A printer class must contain at least one printer. See "Adding a Local Printer Using HP-UX Commands" for instructions on creating a printer class as you add a printer to the lp spooler.

To create a printer class after several printers have been added to the lp spooler:

1. Ensure that you have superuser capabilities.
2. Stop the lp spooler (see "Starting and Stopping the LP Spooler Using HP-UX Commands").

Note It is best to stop the LP spooler when there are no requests currently printing.

3. Create the printer class by entering the `lpadmin` command, specifying the `-c` option, for every printer you wish to add to a class of printers. There is an example at the end of this section.
4. Start the lp spooler (see "Starting and Stopping the LP Spooler Using HP-UX Commands").
5. Allow print requests to enter the request directory for the newly added printer class with the `accept` command:

   `/usr/lib/accept` *pname*

   where:

   pname is the name given to this printer class.

   See "Accepting and Rejecting Print Requests for a Printer Using HP-UX Commands".

Additional Task Information

Printer classes cannot include remote printers.

A printer can only belong to one printer class at a time. To remove a printer from a printer class, remove the printer from the lp spooler and re-add the printer omitting the `-c` option of the `lpadmin` command.

It is not necessary to specify the model and device file options because the printers have already been defined for the lp spooler.

Printer class names can be up to 14 characters in length, and the characters must either be alphanumeric (A-Z, a-z, 0-9) or an underscore (_). Note that class names and printer names on the same system cannot be the same name. Class and printer names must be unique. A printer can only belong to one printer class at a time.

Examples

To create a `laser` class of printers consisting of printers *laser1*, *laser2* and *phred*:

```
/usr/lib/lpadmin -plaser1 -claser
/usr/lib/lpadmin -plaser2 -claser
/usr/lib/lpadmin -pphred -claser
```

To remove a printer from a printer class, remove the printer from the lp spooler and re-add the printer without the printer class (see "Adding a Local Printer Using HP-UX Commands" and "Adding a Local Printer Using SAM").

Removing a Printer or Printer Class Using HP-UX Commands

To remove a printer or printer class using HP-UX commands:

1. Ensure that you have superuser capabilities.
2. Deny any further print requests for the printer with the `reject` command:

   `/usr/lib/reject` [`-r` "*message*"] *name* [*name*]

   where:

   message is a message to be displayed when users obtain status information about the printer(s) and/or printer classes.

   name is the name of the printer or printer class.

   See "Accepting and Rejecting Print Requests for a Printer Using HP-UX Commands".

3. *(If you are removing a printer class, skip this step and continue with step 4.)*

   Disable the printer to be removed with the `disable` command:

   `/usr/bin/disable` [`-r` "*message*"] [`-c`] *pname* [*pname*]

   where:

   message is a message to be displayed when users obtain status information about the printer(s).

   pname is the name of the printer to be disabled.

Note When you disable a printer, any print requests waiting to be printed for that printer will remain in the printer's request directory. If you wish to cancel all print requests for a printer at the time you disable it, use the `-c` option with the `disable` command:

```
/usr/bin/disable -c letterhead
```

See "Enabling or Disabling a Printer Using HP-UX Commands".

Removing a Printer or Printer Class Using HP-UX Commands

4. Stop the lp spooler:

   ```
   /usr/lib/lpshut
   ```

   Before you stop the line printer scheduler (spooling system), beware of the following:

   a. *All* printing will stop until you restart the scheduler.

   b. Any print requests that are currently printing will be completely reprinted when you restart the scheduler. This includes the print requests that were printing page 9,999 of a 10,000 page printout.

5. To preserve the print requests in the request directory, move all print requests in the request directory for the printer or printer class to another printer or printer class request directory (see "Moving *All* Requests Using HP-UX Commands").

6. Remove the printer or printer class from lp spooler with the `lpadmin` command:

   `/usr/lib/lpadmin -x`*name*

   where:

   name is the name of the printer or printer class to be removed.

7. *If you have just removed your only printer, omit this step.*

   Start the lp spooler (see "Starting and Stopping the LP Spooler Using HP-UX Commands"):

   ```
   /usr/lib/lpsched
   ```

Additional Task Information

Because the `lpadmin` is deleting and modifying files that are being examined by the line printer scheduler, it is important that the scheduler is stopped when you use the `lpadmin` command to remove the printer from the lp spooler.

When you remove a printer class, the printers in it are *not* removed. You can still use them as individual printers. If the only printer in a printer class is removed, the printer class is removed also.

Examples

To remove the `laser1` printer:

```
/usr/lib/lpadmin -xlaser1
```

To remove the `laser` printer class:

```
/usr/lib/lpadmin -xlaser
```

Accepting and Rejecting Print Requests for a Printer Using HP-UX Commands

To *accept* print requests for a printer or printer class, use the `accept` command:

`/usr/lib/accept` *name* [*name*]

where:

name is the name of the printer class whose request directory is to be enabled to receive print requests.

You can issue individual commands for each printer class or you can combine the printer classes in one command.

To *reject* print requests for a printer or printer class, use the `reject` command:

`/usr/lib/reject [-r"`*message*`"] `*name* `[-r"`*message*`"] [`*name*`]`

where:

message is a message to be displayed when users obtain status information about the printer or printer class.

name is the name of the printer or printer class whose request directory is being prohibited from receiving print requests.

You can issue individual commands for each printer class or you can combine the printer classes in one command separated by spaces. If you combine them, you can also specify different reasons for rejecting printer requests for different printers (and printer classes).

Additional Task Information

Even if all printers that are members of a class are accepting requests, the class can still reject requests. If that were the case, users would need to specify *a specific printer*, not the class, in later print requests.

If all printers in a class are *rejecting* requests but the class itself is accepting requests, the print requests will remain in the request directory until at least one of the printers in the class begins to process print requests.

If you do not specify a reason, the status requests will get the response, "Printer <*printer_name*> is NOT ACCEPTING requests: Reason is unknown."

Examples

To accept print requests for the `laser1`, `laser2`, `phred`, `invoices`, `check_printer` printers and the `laser` printer class:

```
/usr/lib/accept laser1
/usr/lib/accept phred
/usr/lib/accept invoices laser2 check_printer laser
```

To reject print requests for the `laser1`, `laser2`, `phred`, `invoices`, `check_printer` printers and the `laser` printer class:

```
/usr/lib/reject -r"Printer on loan to seismology lab." laser1
/usr/lib/reject -r"Printers being serviced" laser1 check_printer
/usr/lib/reject -r"Invoice forms on order" invoices\
   -r "printers are being serviced" laser1 laser2 phred laser
```

Note A backslash (\) is used to represent a line continuation. When you type these commands, you can enter the backslash as shown or you can omit the backslash and type the entire command before pressing Return.

Enabling or Disabling a Printer Using HP-UX Commands

To *enable* a printer to process print requests, use the `enable` command:

`/usr/bin/enable` *pname* [*pname*]

where:

pname is the name of the printer to be enabled to process print requests.

You can issue individual commands for each printer or you can combine the printers in one command separated by spaces.

To *disable* a printer to process print requests, use the `disable` command:

`/usr/bin/disable [-r"`*message*`"] `*pname* `[-r"`*message*`"] [`*pname*`]`

where:

message is a message to be displayed when users obtain status information about the printer(s).

pname is the name of the printer to be disabled to process print requests.

You can issue individual commands for each printer class or you can combine the printer classes in one command separated by spaces. If you combine them, you can also specify different reasons for disabling printer requests for different printers (and printer classes).

Additional Task Information

Note

When you disable a printer, any print requests waiting to be printed for that printer will remain in the printer's request directory. When the printer is enabled again, the print requests will print. Any print request that are printing at the time the `disable` command is issued will be completely reprinted when the printer is enabled. If you wish to cancel all print requests for a printer at the time you disable it, use the `-c` option with the `disable` command:

```
/usr/bin/disable -c letterhead
```

Examples

To enable the `check_printer`, `laser1`, `laser2`, and `phred` printers:

```
/usr/bin/enable check_printer
/usr/bin/enable laser1 laser2 phred
```

To disable the `check_printer`, `invoices`, `phred`, `letterhead`, and `laser` printers:

```
/usr/bin/disable check_printer
/usr/bin/disable invoices phred letterhead
/usr/bin/disable -r "printer disabled to change paper" laser1
```

Setting a Printer Priority Fence Using HP-UX Commands

To set or change a printer priority:

1. Ensure that you have superuser capabilities.
2. Stop the lp spooler (see "Starting and Stopping the LP Spooler Using HP-UX Commands").

Note It is best to stop the LP spooler when there are no requests currently printing.

3. Use the `lpfence` command to set priority for a particular printer:

   `/usr/lib/lpfence` *pname priority*

   where:

   pname is the printer name.

   priority is the minimum required priority a print request must have in order to be printed on printer *pname*. Fence value range is 0 (lowest) to 7 (highest).

4. Restart the lp spooler (see "Starting and Stopping the LP Spooler Using HP-UX Commands"):

   `/usr/lib/lpsched`

Additional Task Information

When a printer is added to the lp spooler, the default priority is set to 0 (see "Priorities of Printers and Print Requests").

Starting and Stopping the LP Spooler Using HP-UX Commands

To *start* the lp spooler, use the `lpsched` command:

```
/usr/lib/lpsched
```

To *stop* the lp spooler, use the `lpshut` command:

```
/usr/lib/lpshut
```

Additional Task Information

Before you stop the line printer scheduler (spooling system), beware of the following:

- *All* printing will stop until you restart the scheduler.
- Any print requests that are currently printing will be completely reprinted when you restart the scheduler. This includes the print requests that were printing page 9,999 of a 10,000 page printout.

In order to report statistics about data flow through your lp spooler, you must tell the lp spooler that you want it to keep track of these statistics by specifying the `-a` option when starting the lp spooler with the `lpsched` command. The `-a` option tells the lp spooler to log statistical information about its activities to the file "`/usr/spool/lp/lpana.log`", a file which will be used by the `lpana` command to report the statistics.

Examples

To find out lp spooler status:

```
/usr/bin/lpstat -r
scheduler is stopped
```

To collect statistics about the data flow through the lp spooler, start the lp spooler with the `-a` option:

```
/usr/lib/lpsched -a
```

Canceling Print Requests Using HP-UX Commands

To cancel print requests, use the `cancel` command:

`/usr/bin/cancel` *req-ID* [*printer*]

where:

req-ID is the print request identification number.

printer is the printer name.

You can issue individual commands for each print request or you can combine the print requests in one command separated by spaces.

You do not need superuser capabilities to use the `cancel` command.

Additional Task Information

To list print request identification numbers, use the `lpstat` command (see "Viewing the Status of Printers and Print Requests Using HP-UX Commands").

The `cancel` command has several useful options that allow you to do things such as cancel all print requests that *you* have submitted or cancel all requests associated with a particular printer or printer class. Here are a few helpful `cancel` options and their descriptions:

`-a` Remove all requests a user owns on the specified *printer*. The owner is determined by the user's login name and host name on the machine where the `lp` command was invoked

`-e` Empty the spool queue of all requests for the specified *printer*. Only users with superuser capabilities can use the `-e` option

`-i` Cancel only local requests.

`-u`*user* Remove any requests queued belonging to *user*. Multiple `-u` options are allowed. Only users with superuser capabilities can use the `-u` option.

Examples

```
cancel laser-3456
cancel phred-2152
cancel letterhead-1547
```

or

```
cancel laser-3456 phred-2152 letterhead-1547
```

Moving *All* Requests Using HP-UX Commands

To move *all* print requests to another request directory using HP-UX commands:

1. Ensure your have superuser capabilities.
2. Prohibit any further requests from entering the request directory with the `reject` command:

   `/usr/lib/reject` *name* [*name*]

   where:

   name is the name of the printer or printer class request directory to be enabled to receive print requests.

   You can issue individual commands for each printer class or you can combine the printer classes in one command separated by spaces. If you combine them, you can also specify different reasons for rejecting printer requests for different printers (and printer classes). See "Accepting and Rejecting Print Requests for a Printer Using HP-UX Commands".

3. Disable the printer with the `disable` command:

   `/usr/bin/disable [-r"`*message*`"] `*pname* `[-r"`*message*`"] [`*pname*`]`

   where:

   message is a message to be displayed when users obtain status information about the printer(s).

   pname is the name of the printer to be disabled to process print requests.

   You can issue individual commands for each printer class or you can combine the printer classes in one command separated by spaces. If you combine them, you can also specify different reasons for disabling printer requests for different printers (and printer classes). See "Enabling or Disabling a Printer Using HP-UX Commands".

4. Stop the lp spooler with the `lpshut` command:

   `/usr/lib/lpshut`

5. Relocate all of the print requests in the request directory to another request directory with the `lpmove` command:

   `/usr/lib/lpmove` *source dest*

   where:

   source is the printer or printer class request directory that you want to move to the *dest* request directory.

   dest is the printer or printer class request directory to receive the print requests from the *source* request directory.

6. Restart the line printer scheduler with the `lpsched` command:

   `/usr/lib/lpsched`

7. If the *source* printer or printer class is to be made available to receive print requests:

   a. Re-enable the printer(s) to process print requests with the `enable` command:

      `/usr/bin/enable` *pname* [*pname*]

      where:

      pname is the name of the printer to be enabled to process print requests.

      You can issue individual commands for each printer or you can combine the printers in one command separated by spaces.

   b. Re-enable the printer or printer class request directory to accept print requests with the `accept` command:

      `/usr/lib/accept` *name* [*name*]

      where:

      name is the name of the printer or printer class request directory to be enabled to receive print requests.

      You can issue individual commands for each printer class or you can combine the printer classes in one command.

Examples

To move all print requests from `laser1` request directory to `phred` request directory:

```
/usr/lib/reject laser1
/usr/bin/disable laser1
/usr/lib/lpshut
/usr/lib/lpmove laser1 phred
/user/lib/sched
/usr/bin/enable laser1
/usr/lib/accept laser1
```

Moving Selected Print Requests Using HP-UX Commands

To move selected print requests to another request directory using HP-UX commands:

1. Ensure that the lp spooler is running.
2. Move selected print requests using the `lpalt` command:

   `/usr/bin/lpalt` *source* `-d`*dest*

   where:

   source is the identification number of the print request to be moved.

   dest is the printer or printer class request directory to receive the print request specified by *source*.

Additional Task Information

The `lpalt` command cannot be used to alter a print request that is currently printing.

The `lpalt` command will alter a print request from a *remote* printer *only* if the print request is owned by the user who is issuing the `lpalt` command and, again, this alteration will only take place if the print request is not currently printing.

Examples

To move print request laser-6610 to phred request directory:

```
lpalt laser-6610 -dphred
new request id is phred-6613
```

Viewing the Status of Printers and Print Requests Using HP-UX Commands

To view the status of printers and print requests, use the `lpstat` command:

```
/usr/bin/lpstat [-t]
```

If no options are given, `lpstat` displays the status of all requests made by the user. The `-t` option lists the following additional information:

- status of the lp spooler.
- system default printer.
- list of class names and their members.
- list of printers and associated device files.
- status of each print request directory (accepting or rejecting). If a reason was specified when the requests were rejected the reason is displayed.
- status of each printer (enabled or disabled). If a reason was specified when the printer was disabled, the reason is displayed.
- priority for each printer.
- list of print requests for each printer that includes the following attributes for each print request:
  - print request identification number
  - name of user that submitted the print request
  - priority
  - date and time submitted
  - file name
  - size

Additional Task Information

The `-t` option of the `lpstat` command is very detailed. For information on other options of this command, refer to *lpstat*(1) in the *HP-UX Reference Manual*, HP part number B2355-90033.

Examples

To display a summary status of the lp spooler:

```
lpstat -t

scheduler is running
system default destination: laser
members of class laser:
      laser1
      laser2
      phred
device for letterhead: /dev/null
device for check_printer: /dev/null
device for laser1: /dev/lj1
   remote to: shasta on mountian
device for laser2: /dev/lj2
    remote to: hood on mountian
device for phred: /dev/lj3
device for invoices: /dev/invoices
laser1 accepting requests since Apr 18 14:46
laser2 accepting requests since May 13 14:08
phred accepting requests since Apr 18 14:46
laser accepting requests since Apr 18 14:46
letterhead accepting requests since Apr 18 14:46
invoices accepting requests since Apr 18 14:56
check_printer accepting requests since May 3 14:57
printer laser1 now printing laser1-1807.  enabled since Apr 23 13:47
 fence priority : 0
printer laser2 now printing laser2-1809.  enabled since Apr 23 13:47
 fence priority : 0
printer phred is idle.  enabled since Apr 18 14:46
 fence priority : 3
printer letterhead now printing letterhead-1810.  enabled since Apr 23 13:47
 fence priority : 4
printer invoices is idle.  enabled since Apr 19 10:24
 fence priority : 0
printer check_printer is idle.  enabled since Apr 18 14:56
 fence priority : 0
laser1-1808          susanl         priority 0  Jun 14 10:05 on laser1
 disktab                                   5808 bytes
laser1-1809          susanl         priority 0  Jun 14 10:05
 report1                                  17301 bytes
laser2-1810          kimj           priority 0  Jun 14 10:07 on laser2
 memokmj                                    947 bytes
letterhead-1811      johnc          priority 4  Jun 14 10:09 on letterhead
 salaries                                  2999 bytes
```

Changing the Priority of Print Requests Using HP-UX Commands

To change the priority of a print request, use the `lpalt` command:

`/usr/bin/lpalt` *preq-ID* `-p`*new_priority*

where:

preq-ID is the print request identification number for the print request targeted for a new priority.

new_priority is the new priority. Valid values are 0 to 7.

Additional Task Information

There are two primary reasons for changing a print request priority:

1. To move the print request ahead of other requests within the request directory.

   For example, you can change the priority of your print request to be higher than that of the large print request that is ahead of yours. When the line printer scheduler selects the next print request to send to the printer, it will take the one with the highest priority (which is now yours because you changed the priority).

Note Once a print request is *printing*, it will not yield to a print request of higher priority. In this case, you can move your print request to another printer if possible. See "Moving Selected Print Requests Using HP-UX Commands" for details.

2. To match or exceed the printer's priority, enabling the print request to be processed (see "Priorities of Printers and Print Requests").

Unless you tell it otherwise, the `lp` command (used to print things) will assign your print request a priority equal to that of its printer's printer priority setting. If your print request is assigned to a printer class, the highest printer priority setting among all the printers in the class will be used.

Examples

To find the following print request information:

- The *print request-ID* for the print request you want to change
- The *current priority of the print request*
- The *priorities of the other print requests* on the same printer
- The *priority of the printer*

use the `lpstat` command:

```
lpstat
phred-1827      stevenm     priority 0  Jun 14 10:05 on phred
   proglisting                            1708 bytes
phred-1828      paulv       priority 2  Jun 14 10:05
   LONGproglist                           6900714 bytes
phred-1829      chrisn      priority 1  Jun 14 10:05
   urgentmemo                             311 bytes
```

To move print request phred-1829 ahead in the request directory:

`lpalt phred-1829 -p3` *changes phred-1829's priority to 3*

Displaying Statistics about Printer Activity Using HP-UX Commands

Note Prior to displaying statistics about printer activity, the lp spooler must have been started with the `/usr/lib/lpsched -a` command to create a log of activity in the `/usr/spool/lp/lpana.log` file. See "Starting and Stopping the LP Spooler Using HP-UX Commands".

To display statistics about printer activity, use the `lpana` command:

`/usr/lib/lpana` [`-d`*dest*]

where:

dest defines the printer or printer class for which statistics are displayed. By default, `lpana` will report statistics for *all* printers and printer classes (optional).

Additional Task Information

Table 10-5. Interpreting `lpana's` Output

Column Name	How to Interpret the Results
wait [AV]	This is the average length of time (in minutes and seconds) that print requests spent in their printer's queue *before they began to print*. If this number is *low*, that's good! It means that print requests assigned to this printer begin printing quickly. This might be a good printer to "take the load off" of other printers that have longer **wait** times. If this number is *high*, check the following things: ■ Is there a lot of traffic going through this printer? See the columns called **sum KB** and **num_of requests** (later in this table) for information on how to determine this. You might want to route some of the print requests to another printer. ■ Has the printer been taken offline frequently or for an extended period of time? ■ Has this print destination been disabled (but the lp spooler continued to accept requests for it). This can happen if you **disable** a printer but do not also **reject** the print destination.

Table 10-5. Interpreting `lpana's` Output (continued)

Column Name	How to Interpret the Results
wait [SD]	This is the standard deviation from the "average wait time." This number can help you determine how typical the "average wait time" (reported in the **`wait [AV]`** column) is. If this number is *small*, it means that the number in the **`wait [AV]`** column is fairly representative of a "typical" print request. It makes sense to interpret the **`wait [AV]`** as explained in the description of that column, above.
wait [SD] (cont.)	If this number is *large* it indicates that the number in the **`wait [AV]`** column is not representative of a "typical" print request. This could indicate that a small number of print requests had to wait an unusually long (or short) time and their wait times are skewing the data reported as the "average wait time." It might be wise to monitor the statistics for a longer period of time before making changes to your lp spooler as a result of this data.
print [AV]	This is the average amount of time print requests took to actually print (the elapsed time from the time a print request begins to print until it has finished printing). If this number is *small*, print requests for this print destination are not taking long to print. It probably indicates that the "typical" print request for this print destination is small. Fast printers and fast communication lines can also help keep this number down. If this number is *large*, print requests for the corresponding print destination are taking a long time to print. This may be due to their size (perhaps they contain a lot of graphic data) or this may be due to a slow printer or communications line.

Table 10-5. Interpreting `lpana's` Output (continued)

Column Name	How to Interpret the Results
print [SD]	This is the standard deviation from the "average print time." Similar to the standard deviation for the "average wait time" this number lets you know how typical the "print [AV]" number is. If this number is *small*, the "print [AV]" number is representative of the print time for typical print request. Essentially it means that all of the print requests for this print destination take about the same amount of time to print. If this number is *large*, the "print [AV]" time is not too representative of a typical print request. This means that there is a wide variation in how long print requests take to print (for this print destination).
bytes [AV]	This is the average size (in bytes) for print requests going to this print destination.
bytes [SD]	This is the standard deviation from the "average size." It tells you how typical the "bytes [AV]" number is. If this number is *small*, print requests for this print destination do not vary in size much. If this number is *large*, print requests for this print destination vary a lot in size.
sum [KB]	This is the number of kilobytes (# of bytes x 1024) of data sent to this print destination during the reporting period.
num_of requests	This is the number of print requests sent to this print destination during the reporting period.

Examples

To display statistics for all printers:

```
/usr/lib/lpana

performance analysis is done from Jun.22 '90 14:02 through Jun.27 '90 15:29
---printers      ----wait----   ---print---    ---bytes---   -sum- num_of
    /classes--     AV     SD     AV     SD      AV     SD      KB requests
letterhead       0'00      0   0'49      2   59565      0     116        2
phred            0'00      0   0'45     22   14202      0     166       12
check_printer    0'09     31   0'51     73   12378      0     302       25
laser1           0'02      5   0'04      1   36686      0    2400       67
laser2           3'45      0   1'45      0     783      0       1        1
```

To display statistics for the `laser` printer class:

```
/usr/lib/lpana -d laser

performance analysis is done from Jun.22 '90 14:02 through Jun.27 '90 15:29
---printers      ----wait----   ---print---    ---bytes---   -sum- num_of
    /classes--     AV     SD     AV     SD      AV     SD      KB requests
laser1           0'02      5   0'04      1   36686      0    2400       67
laser2           3'45      0   1'45      0     783      0       1        1
```

11

Communicating With the Users on Your System

HP-UX provides you with several utilities to communicate with your system users. These include the following commands, which will be explained in this chapter:

- *news*(1M)
- *write*(1)
- *wall*(1M)
- *uucp*(1)
- *mail*(1), *mailx*(1), *elm*(1).

Taking advantage of these utilities will make your job as system administrator easier, and will enhance communications between you and your users and among individual users.

Terms Used in this Chapter

The following terms are explained in this chapter:

elm An electronic mailer that is more flexible than `mail` or `mailx`.

mail A basic electronic mailer that is the base for both `mailx` and `elm`. It only supports command-line input, and does not provide for editing of outgoing messages. However, you can use `mail` to send a file that was written outside of the mail program.

mailx An enhanced version of `mail`. `mailx` provides commands for saving, deleting, and responding to messages that you have received, and for editing and reviewing messages that you wish to send.

mesg A command, when issued by an individual user, that permits or denies other users from writing to the user's terminal.

motd A file displayed on the terminal whenever a user logs in. This file can contain a "message of the day" that you want your users to read when they log in.

news A command that enables you to leave messages and other news items for your users. Each user can automatically receive the news items upon logging in to the system, or upon invoking the `news` command.

UUCP A utility that allows users on one HP-UX system to send files to users on another HP-UX system.

wall A command that allows you to send, or broadcast, messages to all users, or to users identified in a distribution list.

write A command that allows a user to send a message directly to another user's terminal on the same system.

Displaying Messages to Users Logging into Your System

HP-UX provides two methods by which you can keep your users informed about your system. The first is a file `/etc/motd` (message of the day). The contents of this file is displayed on a user's terminal each time the user logs into the system. The second method uses the HP-UX `news` command.

motd

You can place a message in the `/etc/motd` file, and this file will be read to a user's terminal each time that the user logs into the system. The process of reading the file is handled automatically by `login`. This is an easy way to inform your users about such information as the system backup schedule or scheduled maintenance.

Another `motd` file is automatically read by the printer spooler, which, in turn, prints a message at the top of each user's printout. This file is in `/usr/spool/lp/motd`. This file can be used to keep your user's informed about such information as the printer name and status.

news

The `news` command allows users to read announcements which have been left on the system. Information for system users can be placed in files under the `/usr/news` directory. Any user may create a file in this directory for other users to read. A user can access these files by typing `news`. There are several options:

- `news -a` displays all files in the `/usr/news` directory. This includes both new files and files the user has read.
- `news -n` displays the names of the files in the `/usr/news` directory without displaying their content.
- `news -s` lists the names of the files in the `/usr/news` directory that the user has not yet read.

You can abort the reading of one `news` item and begin the reading of the next `news` item by typing your interrupt key sequence. You can terminate the `news` program by typing a second interrupt sequence within one second of the first. You can determine your interrupt key sequence by typing `stty -a`.

Users can put the command `news` in their `.profile` or `.login` file, and they will automatically receive the system news when they log in.

Communicating With a User Who is Logged into Your System

The *write*(1) command allows one user to send a message directly to another user's terminal, and permits that person to directly respond to the originator. This provides for true two-way communication.

The originator of the session types: `write` *user*, where *user* is the login name of the receiver of the communication. The receiver will see `Message from` *yourname* on the terminal, where *yourname* is the originator's login name.

The receiver may then follow the same procedure to send a message back to the originator, and two-way communication is established. The bell on the originator's terminal will sound twice to indicate that two-way communication is possible.

To end the communication session, type your interrupt key sequence. You can determine your interrupt key sequence by typing the command: `stty -a`.

Permission to write to a user's terminal may by granted or denied by the user typing the `mesg` command. If the user types `mesg y`, (the default) write permission is granted. If the user types `mesg n`, write permission is denied.

Broadcasting a Message to All Users Logged into Your System

The `wall` (write to all) command allows the system administrator to send a message to all users logged into the system. It can be used to notify users of items of immediate concern, such as the need to shut down the system.

If the `wall` command is used without arguments, the standard input from your terminal is read until an end-of-file (ctrl-d) is received. Then the message is sent to all logged-in users preceded by: **Broadcast Message from** . . . .

If the `wall -g`*groupname* command is used, the standard input from your terminal is sent to all logged-in users whose names are listed in the file *groupname* in the directory `/etc/group`. The message is preceded by: **Broadcast Message from ... to group** *groupname*.

If the `wall` command is followed by a file name, the content of the specified file is used for the message, rather than input from your terminal.

When in a HP cluster environment, use the command `cwall` to write to all logged-in users of the cluster.

Communicating With Users on Other Systems Using UUCP

HP-UX provides a utility called UUCP that allows users to send files between different computer systems. The computer systems must be connected directly together (hardwired), or connected to the telephone system by modems. After the hardware is properly connected and the software is configured, users can transfer files using the `uucp` command.

The `uucp` command provides users a number of options in designating the source and destination files for the transfer. Other commands associated with `uucp` are `uulog` and `uuname`.

While UUCP is easy for individual users to use, the process of setting up the hardware and software is complex. This process is described in detail in the manual *Remote Access: User's Guide* (HP part number B2355-90037). Refer to this manual for additional information on UUCP.

Using Electronic Mail Systems

Several utilities are available that permit users to communicate through electronic mail. These are `mail`, `mailx`, and `elm`.

The preferred HP-UX electronic mail program is `elm`, which provides many advantages over `mail` and `mailx`, including an interactive screen-oriented processing system, a command menu, and help. Using `elm`, users can customize their e-mail environment.

Information on using electronic mail is presented in detail in the manual *Mail Systems: User's Guide* (HP part number B1862-90012).

12

System Accounting Concepts and Tasks

Multiuser HP-UX allows concurrent sharing of computer resources among multiple users: several users can be logged in, all sharing disk space, memory, and the CPU. On multiuser systems, HP-UX System Accounting provides the means to:

- Monitor disk space usage for individual users.
- Record connect session data (logins/logouts).
- Collect resource utilization data (such as memory usage and execution times) for individual processes.
- Charge fees to specific users.
- Generate summary files and reports that can be used to analyze system performance and bill users for resource consumption.

Note Much of the material in this chapter assumes greater knowledge of HP-UX than is required of the "average" user. In particular, System Accounting borrows many concepts from from most of the previous chapters. If you are unfamiliar with the concepts and terminology in those chapters, then you should review them.

What Is in This Chapter?

HP-UX System Accounting allows you to accomplish accounting tasks through a number of versatile commands. This chapter illustrates the use of these commands and contains the following sections:

12

- "Installation and Daily Usage" shows the routine daily usage of System Accounting and shows you how to install it.
- "Overview of System Accounting" provides the background information necessary to understand how to use System Accounting.
- "Disk Space Usage Accounting" illustrates the use of the accounting commands that monitor disk space utilization on a per-user basis.
- "Connect Session Accounting" describes the commands that record and report connect session accounting information.
- "Process Accounting" shows how to generate per-process accounting data and reports.
- "Charging Fees to Users" is the section where you learn how to charge fees to users.
- "Summarizing and Reporting Accounting Information" shows how to generate the main daily and monthly accounting reports that are used to monitor system performance and bill users.
- "Updating the Holidays File" describes how to set up the file describing your holidays.
- "Fixing Corrupted Files" is useful when System Accounting files become inconsistent or messed up. This section discusses how to fix these files.
- "Sample Accounting Shell Scripts" provides listings of shell scripts that you might find useful on your system.
- "System Accounting Files" contains brief definitions of the files used by System Accounting.

Installation and Daily Usage

The purpose of this section is to show you:

- What you must do to get System Accounting running on your system.
- How System Accounting automatically creates daily and monthly accounting data and reports.

After reading this section, you should be able to install System Accounting on your system. Once properly installed, System Accounting will automatically generate daily and monthly accounting data and reports.

How to Install System Accounting

Not all users require accounting services on their systems. For this reason, HP-UX System Accounting is provided as an option: if you want to use System Accounting, you must install it yourself. The installation procedure is covered here.

There are three steps in the installation process:

1. Update `/etc/rc`
2. Create `crontab` entries
3. Set PATH for accounting commands

Each of these steps must be carried out to insure that System Accounting automatically creates daily and monthly accounting information. Detailed descriptions of each step follow.

Update /etc/rc

The system initialization shell script `rc` must be updated to automatically start System Accounting when the system is switched into multiuser mode. This requires adding the following entry in the `localrc` section of `/etc/rc`:

```
/bin/su - adm -c /usr/lib/acct/startup
```

in /etc/rc.config.d/acct,
$ START_ACCT=1

Create crontab Entries

To automate the daily and monthly creation of accounting data, you should create a `crontab` file that `cron` can use to automatically run certain accounting commands. This process entails the following steps:

12

1. Log in to System Accounting as the user `adm`.
2. Use an editor to create the `crontab` file containing the accounting commands that are to be run automatically by `cron`. (The actual entries to make in this file are shown after these steps.)
3. Execute the `crontab` command, specifying the file created in step 2 as input. This step insures that the `crontab` file created in step 2 will be scanned by `cron` every minute. After invoking this command, the step 2 file will be stored in the file:

   `/usr/spool/cron/crontabs/adm`

4. At this point, you are finished creating `crontab` entries. If you ever want to change the entries, simply re-edit the file created in step 2 and use the `crontab` command again. Refer to the *crontab(1)* entry in the *HP-UX Reference Manual* for more information.

The following entries, accompanied by a description of each, should be made in the `crontab` file created in above:

```
0 4 * * 1-6 /usr/lib/acct/runacct 2> /usr/adm/acct/nite/fd2log
```

`runacct`, the main accounting shell script, should be executed daily (during non-prime hours) to generate daily accounting reports. The above entry executes `runacct` at 4:00am every Monday through Saturday. Error messages will be redirected to the file `/usr/adm/acct/nite/fd2log`, if any errors occur while `runacct` executes.

```
0 2 * * 4 /usr/lib/acct/dodisk
```

`dodisk` creates total accounting records that summarize disk space usage for individual users. This entry runs `dodisk` at 2:00am every Thursday morning.

```
5 * * * * /usr/lib/acct/ckpacct
```

To insure that the process accounting file, `pacct`, doesn't get too large, the command `ckpacct` should be executed hourly. This entry invokes `ckpacct` at five minutes into every hour.

```
15 5 1 * * /usr/lib/acct/monacct
```


The monthly merging of accounting data is facilitated through the `monacct` command. This entry allows `monacct` to generate a monthly total report and total accounting file. `monacct` will be executed at 5:15am on the first day of every month.

Note The dates and times shown in the `crontab` entries above are only suggestions; you can tailor `crontab` entries to suit your needs. However, if you use different entries than those shown here, be sure that `monacct` is run at such a time as to allow `runacct` sufficient time to finish.

Set PATH for Accounting Commands

Finally, you should set the PATH shell variable in `/usr/adm/.profile` so that System Accounting knows where to look for commands. Path should be set as follows:

```
PATH=/usr/lib/acct:/bin:/usr/bin:/etc:/usr/adm
```

Summary of Daily Operation

The daily operation of System Accounting is summarized by the following steps:

1. When HP-UX is switched into multiuser mode, the system initialization shell script `rc` executes the accounting command `startup`. The purpose of `startup` is to start System Accounting, and it performs the following functions:

   a. Calls `acctwtmp` to add a boot record to `wtmp`. This record is marked by storing "acctg on" in the device name field of the `wtmp` record.

   b. Turns process accounting on via `turnacct on`. `turnacct on` executes `accton` with the filename argument `/usr/adm/pacct`.

   c. Removes work files left in the `sum` directory by `runacct`.

2. A report of the previous day's accounting information can be created by running `prdaily`. Obviously, this step is omitted the first day that System Accounting is installed, because the previous day's accounting information doesn't exist yet. However, after `runacct` has been executed, `prdaily` will generate valid reports.

3. The `ckpacct` command is executed every hour via `cron` to insure that the process accounting file `pacct` doesn't become too large. If `pacct` grows past a set maximum number of blocks, `turnacct switch` is invoked, which creates a new `pacct` file. (Other conditions may also limit the size of the process accounting file or turn process accounting off; for more details, refer to the discussion of `ckpacct` in the "Process Accounting" section of this chapter.) The advantage of having several smaller `pacct` files is that `runacct` can be restarted faster if a failure occurs while processing these records.

4. The `chargefee` program can be used to charge fees to users. It adds records to the file `fee`. These records are processed during the next execution of `runacct` and merged in with total accounting records.

5. `runacct` is executed via `cron` each night. It processes the active fee file and the process, connect session, and disk total accounting files. It produces command and resource-usage summaries by login name.

6. When the system is turned off using `shutdown`, the `shutacct` command is executed. The purpose of `shutacct` is to stop System Accounting, and it performs the following functions:

   a. Writes a termination record to `wtmp` via the command `acctwtmp`. This record is marked by having "acctg off" in the device name field.

   b. Turns process accounting off by calling `turnacct off`.

Overview of System Accounting

In this section, the intrinsics of System Accounting are examined. Key terms are defined, commands are introduced, system data flow is described, and finally, you are shown the login and directory structure of System Accounting.

Definitions

The following terms are specific to System Accounting.

prime/non-prime connect time

Prime time is the time during the day when the computer system is most heavily used—for example, from 9:00am to 5:00pm. Non-prime time is the remaining time during the day when the system is less heavily used—from 5:00pm to 9:00am in this example.

When reporting computer time usage, System Accounting distinguishes between prime and non-prime time usage. You can specify prime and non-prime time on your system by editing the file `/usr/lib/acct/holidays`. (For details on the `holidays` file, refer to the section "Updating the Holidays File" in this chapter.

Note Prime time is in effect only on weekdays (Monday through Friday); non-prime time is in effect during the weekends (Saturdays and Sundays) and on any holidays specified in the `holidays` file.

process accounting records

Once System Accounting is installed and turned on, the following occurs: whenever a process terminates, the kernel writes a process accounting record for the terminating process into the current process accounting file, `/usr/adm/pacct` by default. (You can specify that a file other than `pacct` be used as the process accounting file, if desired.)

A process accounting record contains resource-usage data for a single process; it summarizes *how much* of the various resources the process used during its lifetime. Examples of information contained in process accounting records are:

- the user ID of the process's owner
- the name of the command that spawned the process
- the amount of time it took the process to execute

For greater detail on the contents and format of process accounting records, refer to *acct(4)* in the *HP-UX Reference Manual*.

total accounting records

These records, created by various accounting commands, contain summary accounting information for individual users. These records provide the basic information for many reports generated by System Accounting. Some examples of information contained in these records are:

- the ID and user name of the user for whom the total accounting record was created
- the total number of processes that the user has spawned during the accounting period for which the total accounting record was created
- fees for special services rendered to this user

The exact contents and format of total accounting records can be found in *acct(4)*. In addition, commands covered in later sections of this chapter show how these records are created and used by System Accounting.

Introduction to Commands

System Accounting provides many versatile commands to accomplish numerous, varied tasks. There are commands that create data, commands that display data, commands that remove data, commands that merge data, and commands that summarize and report data. In addition, the output of one command may become the input to other commands.

System Accounting commands can be logically categorized into six basic command groups:

- installation
- disk usage accounting
- connect session accounting
- process accounting
- charging fees
- summarizing and reporting accounting information

Descriptions of these command groups, along with a brief synopsis of each command, follow.

Installation

These commands insure that System Accounting is properly installed. They are used to turn accounting on when HP-UX is powered up and turn accounting off when the system is shut down. They may also do some file cleanups. Two such commands exist:

- `startup`—starts accounting when HP-UX is switched to multiuser mode. `startup` is invoked from `/etc/rc`.
- `shutacct`—turns off accounting when HP-UX is turned off via the `/etc/shutdown` shell.

Disk Space Usage Accounting

In general, these commands produce disk usage accounting information: they show disk space usage (in blocks) for individual users. They also produce total accounting records. There are four commands:

- `acctdusg` and `diskusg`—both commands show how many blocks of disk space users are consuming. They differ in command options, and the manner in which they produce the information—`acctdusg` takes its input from a list of path names created by `find`, and `diskusg` looks at the inodes of the file system to create its output.
- `acctdisk`—produces total accounting records. Its input is supplied (either directly or indirectly) from `acctdusg` or `diskusg`.
- `dodisk`—produces total accounting records by using the `diskusg` and `acctdisk` commands. `dodisk` is normally invoked by `cron`.

Connect Session Accounting

Independently of System Accounting, the programs `login` and `init` record connect sessions by writing records into `/etc/wtmp`. System Accounting commands can display or fix this file, and can produce total accounting records for this file. There are six commands:

- `acctwtmp`—writes records to `wtmp`.
- `fwtmp`—displays the information contained in `wtmp`.
- `wtmpfix`—normalizes connect session records that span date changes (refer to *date(1)*). Also validates login names in connect session records.
- `acctcon1`—summarizes `wtmp` in ASCII readable format, producing one line per connect session.
- `acctcon2`—takes input of the format produced by `acctcon1` and produces total accounting records as output.
- `prctmp`—displays the session record file, normally called:

  `/usr/adm/acct/nite/ctmp`

Process Accounting

When process accounting is turned on, the kernel writes a process accounting record to `pacct` whenever a process terminates. A number of accounting commands exist that summarize and report this accounting information. In addition, certain commands turn process accounting on or off and insure that `pacct` doesn't become too large. The process accounting commands are:

- `accton`—turns process accounting on or off, depending on whether or not a filename argument is supplied with the command. If no filename is given, then process accounting is turned off; the kernel stops writing process accounting records to `pacct`. If a filename is specified, then the kernel starts writing process accounting records to the specified filename.

  `accton` uses the system call `acct` to turn process accounting on or off. Only the superuser can execute `accton`.

- `ckpacct`—checks the size of the process accounting file `pacct`. If `pacct` becomes too large, then a new `pacct` file is created via `turnacct switch`. *If disk space becomes critically short, then process accounting is turned off until sufficient space is available.* This command is normally invoked by `cron`.

- `turnacct on || off || switch`—performs one of three functions, depending on which argument (on, off, or switch) is specified. `turnacct on` turns process accounting on by calling `accton` with the default filename argument `/usr/adm/pacct`; `turnacct off` turns process accounting off by calling `accton` with no filename argument; `turnacct switch` renames the current `pacct` file (so that it is no longer the current process accounting file) and creates a new, empty `pacct` file.

- `acctcom`—displays process accounting records contained in `pacct` (or any specified file).

- `acctcms`—takes `pacct` as input, and produces summary accounting information by command, as opposed to by process.

- `acctprc1`—produces readable process accounting information, mainly for input into `acctprc2`.

- `acctprc2`—takes input of the form produced by `acctprc1` and produces total accounting records.

Charging Fees

Occasionally, you may want to charge a user for something. For example, you might charge fees to users for fixing any damaged files that they have. The `chargefee` command allows you to charge fees to specific users.

Summarizing and Reporting Accounting Information

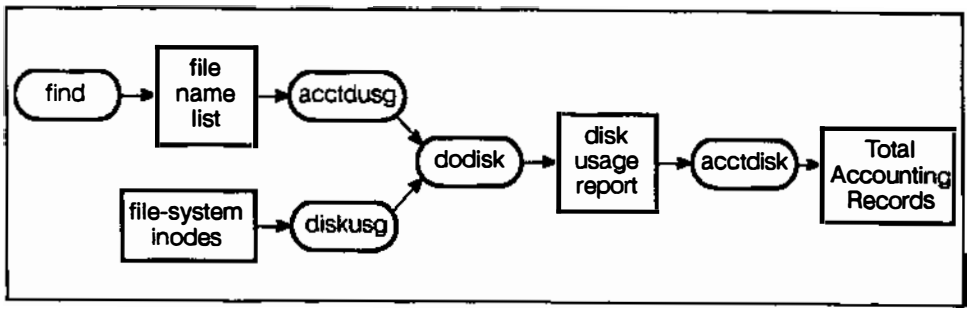
This group of commands summarizes and reports the data created through the command groups described above. These are the commands that are probably used most frequently; they represent the highest level of accounting commands. Five such commands exist:

- `prtacct`—takes as input total accounting records and displays the records in ASCII readable format.
- `acctmerg`—combines the contents of separate total accounting files into a single total accounting file. This command allows the merging of disk, process, and connect session total accounting records.
- `runacct`—is the main accounting shell script. Normally invoked daily by `cron`, this command processes disk, connect session, process, and fee accounting information and produces summary files and reports. It accomplishes its task by proceeding through various states. In each successive state it invokes accounting commands to perform a specific task. For example, in one state, total accounting records for connect sessions are created; in another, disk, connect session, process, and fee total accounting records are merged to create one total accounting file.
- `prdaily`—invoked by `runacct` to format a report of the previous day's accounting data; the report is in the file `/user/adm/acct/sum/rpt`*mmdd* where *mmdd* is the month and day of the report. `runacct` may also be used to display a report of the current day's accounting information.
- `monacct`—invoked once a month (or accounting period), this command summarizes daily accounting files and produces a summary files for the accounting period.

System Data Flow

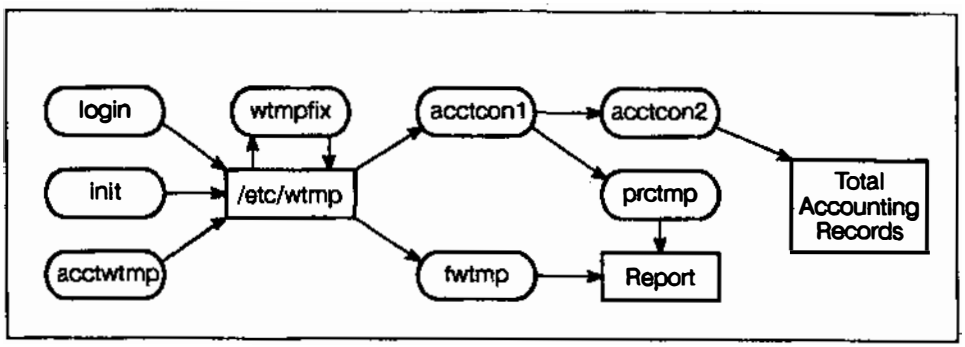
At this point, you have the rudimentary knowledge necessary to understand how System Accounting works; you know some important definitions and should basically know what the various commands do. The purpose of this section is to help you visualize how the different commands work together to create accounting data.

The following figures illustrate, through the use of diagrams, how accounting data is created. Each diagram represents the data flow for a given command group.



LG200172_003

Figure 12-1. Disk Usage Accounting



LG200172_004

Figure 12-2. Connect Session Accounting

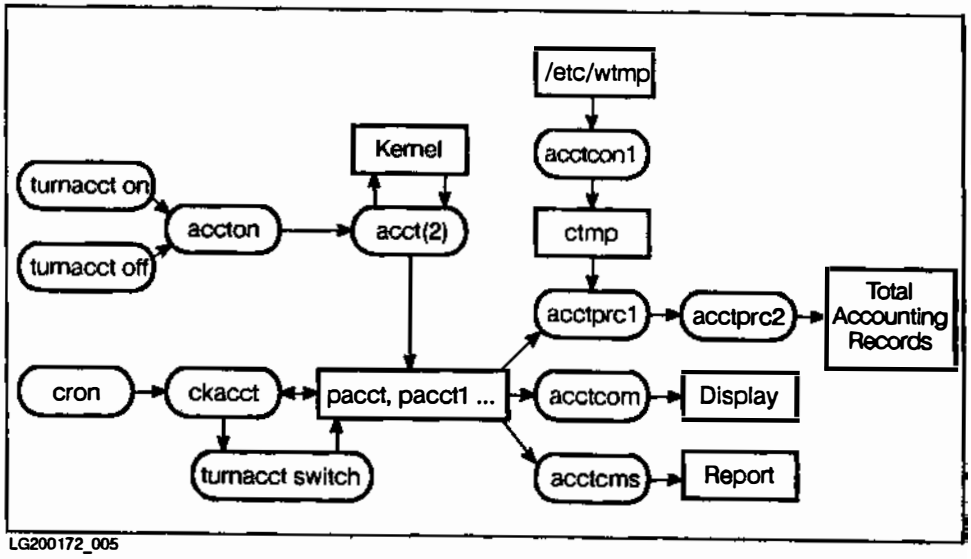


Figure 12-3. Process Accounting

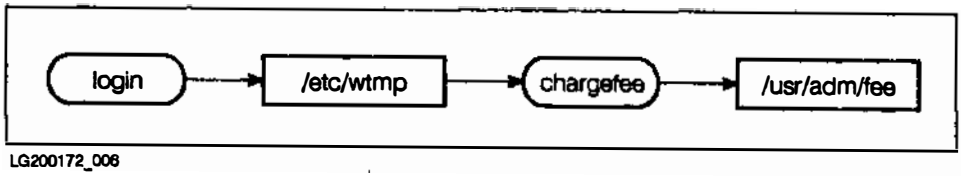


Figure 12-4. Charging Fees

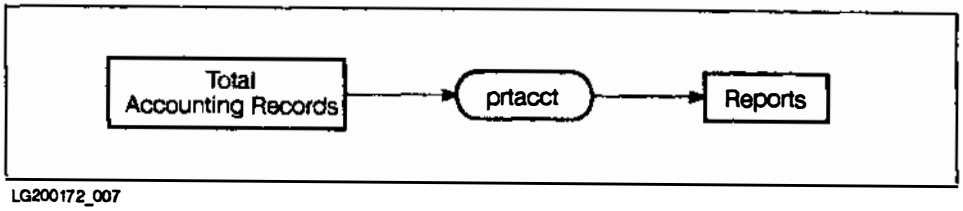


Figure 12-5. Summarizing and Reporting

Login and Directory Structure

You now know the basics, but you still can't begin learning the day-to-day usage of accounting commands until you know where to log in. In addition, you should know the accounting directory structure—where the various commands, directories, and files are located. These two topics are discussed here.

Logging In

The login name for System Accounting is `adm`; the user ID for `adm` is 4. The `adm` login is a member of the *group* `adm`, and the group `adm` has a group ID of 4, also.

The home directory for the `adm` login is `/usr/adm`. You log in to System Accounting the same way you do for any account—simply supply the login name to the HP-UX login prompt:

```
login: adm
```

Note The integrity of accounting data files must be maintained if System Accounting is to generate accurate reports. For this reason, it is highly recommended that a password be used with the `adm` login.

Directory Structure

System Accounting uses a multi-level directory structure to organize its many accounting files. Each directory in this structure stores related groups of files, commands, or other directories. (Refer to the section "System Accounting Files" in this chapter for definitions of the accounting data files.)

Figure 12-6 illustrates this structure, and descriptions of each directory follow:

- `/usr/adm`—contains all active data-collection files, such as `pacct` and `fee`.
- `/usr/adm/acct`—contains the `nite`, `sum`, and `fiscal` directories described below.
- `/usr/adm/acct/nite`—stores data files that are processed daily by `runacct`.

- /usr/adm/acct/sum—cumulative summary files updated by runacct are kept here.
- /usr/adm/acct/fiscal—periodic (monthly) summary files created by monacct are stored here.
- /usr/lib/acct—System Accounting commands reside here.
- /etc—contains wtmp, and shell scripts rc and shutdown.

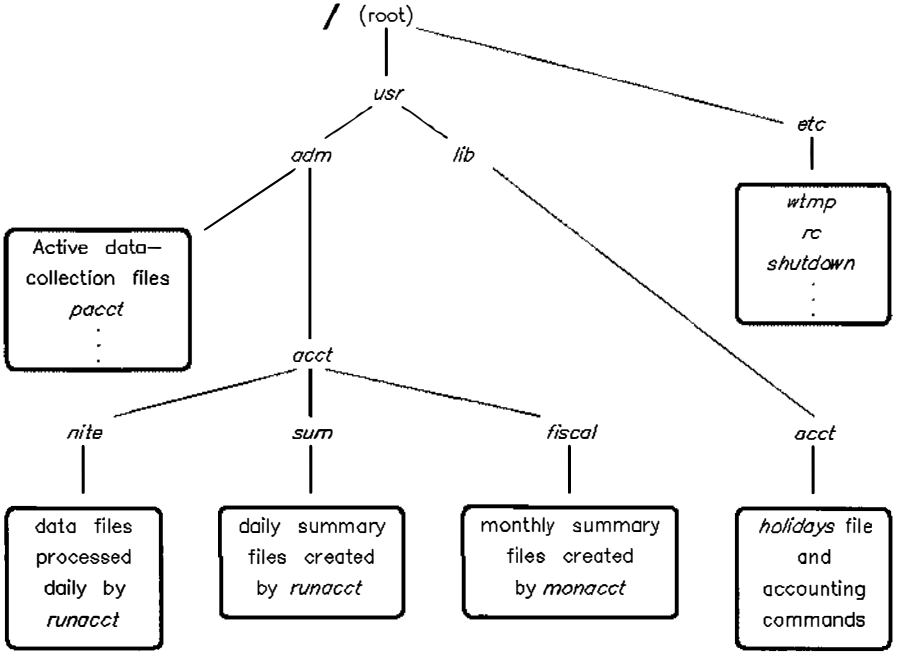


Figure 12-6. System Accounting Directory Structure

Disk Space Usage Accounting

System Accounting provides the means to monitor disk space utilization for individual users. In this section, disk space usage accounting commands are explained. Before reading this discussion, you may want to review the "File System Implementation" chapter of this manual.

Disk usage commands provide two main functions: they report disk usage (in blocks) for individual users and create disk total accounting records (supplied as inputs to commands such as `prtacct` or `runacct`).

Reporting Disk Space Usage

Two commands—`acctdusg` and `diskusg`—report disk usage for individual users; both commands show the number of disk blocks allocated to specific users. However, each command has slightly different options. In addition, each differs in the manner in which it produces accounting information.

acctdusg

`acctdusg` takes from standard input a list of path names, usually created by the `find` command. For each file in the list, `acctdusg` identifies the owner of the file, computes the number of blocks allocated to the file, and adds this amount to a running total for the file's owner. When finished looking through the list, `acctdusg` displays the information accumulated for each user: user ID, user name, and number of blocks used.

This command is useful for reporting disk usage information for specific users or files. For example, suppose you want to know how many blocks of disk space you are using: your user ID is `351`, user name is `bill`, and your home directory is `/users/pseudo/bill`. The following illustrates how you would use the `find` and `acctdusg` commands to show this information.


```
find /users/pseudo/bill  -hidden -print > bills.files
acctdusg < bills.files
00351   bill                30
rm bills.files
```

In the above example, `bill` is using 30 blocks of disk space. The series of commands shown could easily have been combined into one line, such as:

```
find $HOME  -hidden -print | acctdusg
00351   bill                30
```

The next example shows how to use `acctdusg` to generate disk usage information for all files in the system:

```
find /  -hidden -print | acctdusg
00350   fred                11
00351   bill                30
00352   mike                17
00353   sarah               13
00354   molly               18
00000   root                 3
00004   adm                 36
00001   bin               2434
```

Two options are included with `acctdusg`:

`-u` *no_owners* — If `-u` is given, then path names of the files for which no owner is found are written into the file *no_owners*. This option could potentially find users who are trying to avoid disk charges.

`-p` *p_file* — The password file `/etc/passwd` is the default file used by `acctdusg` to determine ownership of files. If the `-p` option is used, then `acctdusg` will use *p_file* instead. This option is not needed if your password file is `/etc/passwd`.

The shell script `grpdusg`, provided in the section "Sample Accounting Shell Scripts" later in this chapter, displays disk accounting information for users in a given group. It illustrates the use of the `-u` option with `acctdusg`.

diskusg

This command reports disk usage information in the same format as `acctdusg`—user ID, user name, and total disk blocks used. However, `diskusg` generates disk accounting information by looking through the *inodes* of a specified special file. (Refer to *inode(4)* and Chapter 8, "HFS File System" in the manual *How HP-UX Works: Concepts for the System Administrator* for more information on inodes and special files.) Therefore, `diskusg` is faster and more accurate than `acctdusg`.

The syntax of the `diskusg` command is:

`diskusg` [*options*] [*files*]

It generates a disk usage report from data in *files*, if specified; otherwise standard input is used. `diskusg` is normally invoked with the *files* argument. When specified, *files* are the special file names of the devices containing the inode information used by `diskusg` to generate its report. *files* is normally a special file from the `/dev` directory.

The following options may be used with `diskusg`:

`-s` This tells `diskusg` that: (1) input is in `diskusg` output format, and (2) that all lines for a single user should be combined into a single line. This option is used to merge data from separate files, each containing the output from using `diskusg` on different devices.

`-v` This option is useful for finding users who are trying to avoid disk space accounting charges. When this option is specified, `diskusg` writes records to `stderr` (standard error output) showing the special file name, inode number, and user ID of files that apparently have no owner.

`-i` *fnmlist* Causes `diskusg` to ignore the data on those file systems whose file system name is in *fnmlist*. *fnmlist* is a list of file systems separated by commas or enclosed within quotes.

`-p` *p_file* This is the same as the `-p` option of `acctdusg`.

`-u` *u_file* This option produces *exactly* the same output as the `-v` option. The difference between the two options is that `-v` writes its output to `stderr`; this option writes its output to the file *u_file*.

The output of `diskusg` is normally used by `acctdisk` to create disk total accounting records. In addition, `diskusg` is normally called by `dodisk`.

The following example creates disk usage information for all users whose files reside on the disk whose device file is `/dev/rdsk/c2d0s7`. *Note that the file system used in this example is the same as was used in the previous* `acctdusg` *example.*

```
diskusg /dev/rdsk/c2d0s7
0       root      10616
1       bin       778
4       adm       96
350     fred      14
351     bill      32
352     mike      20
353     sarah     16
354     molly     22
355     julie     2
501     guest     2
```


The differences between `diskusg` and `acctdusg` are best illustrated by comparing their outputs. Note that:

1. `acctdusg` places leading zeros on user IDs; `diskusg` doesn't.
2. `acctdusg` counts files *only under each users* $HOME *directory*. Files that users own in directories other than their home directory (for example, files in the `/tmp` directory) are counted as files with no owner.
3. Two extra users—`julie` and `guest`—show up in the output of `diskusg` when compared with the output from `acctdusg`. This occurred because the directories of these two users were empty; therefore, no disk usage totals were generated by `acctdusg`. However, `diskusg` looked at inodes and saw that `julie` and `guest` were actually using two blocks for the directories themselves.
4. If two or more users have links to a particular file, then `acctdusg` will prorate disk space usage for the file between each user. For example, if three users had a link to a 300-block file called `skurbnich.dat`, each user would be charged for 100 blocks of this file.

Creating Total Accounting Records

Two commands are used to create total accounting records: `acctdisk`, and `dodisk`.

acctdisk

`acctdisk` uses standard input records of the format produced by `acctdusg` and `diskusg`. From these records, `acctdisk` produces disk total accounting records that may be inputs to `prtacct` or `runacct`.

The following would write disk total accounting records to the file `disktacct` for all users in the group `pseudo`:

```
find / -group pseudo -print | acctdusg | acctdisk > disktacct
```

The next example would generate disk total accounting records for all users who have files on the disk section `/dev/rdsk/c2d0s7`. The total accounting records are written to the file `disktacct`.

```
diskusg /dev/rdsk/c2d0s7 | acctdisk > disktacct
```

`acctdisk` has no options and is normally invoked by `dodisk`.

dodisk

`dodisk` is normally invoked by `cron` to create disk total accounting records for daily usage by System Accounting. The syntax for `dodisk` is:

`dodisk` [`-o`] [*files* ...]

In the default case, `dodisk` creates disk total accounting records on the special files whose names are stored in `/etc/checklist`; the special file names are supplied as input to `diskusg`, which pipes its output to `acctdisk`, which in turn creates total accounting records.

If the `-o` option is used, `dodisk` creates total accounting records more slowly by using `acctdusg` instead of `diskusg`.

If *files* are used, disk accounting will be done on these file systems only. When the `-o` option is used, *files* should be mount points of mounted file systems; if omitted, *files* should be the special file names of mountable file systems.

Note Refer to the "Installation and Daily Usage" section of this chapter for more information on how `dodisk` should be invoked by `cron`.

It is possible for malicious users to defeat disk space accounting by giving their files away to other users with *chown(2)* or *chown(1)* (by default, all users can execute them). To avoid this, take away the ability to use these commands from some or all users with the *setprivgrp(1M)* command. To let only the superuser use the change-ownership abilities, add the following line to `/etc/rc`:

```
setprivgrp -n CHOWN
```

To let one or more groups of users use the change-ownership abilities, add a line for each group to `/etc/rc`, similar to the following:

`setprivgrp` *group_name* `CHOWN`

Note Taking away the change-ownership ability may cause problems when running some commands or applications.

Connect Session Accounting

Whenever a user logs in or out of HP-UX, the program `login` records the connect session in the file `/etc/wtmp`. Records in `wtmp` contain the following information:

- the terminal name on which the connect session occurred
- the login name of the user
- the current time/date at login or logout
- other status information (refer to *utmp(4)* for details)

System Accounting provides commands that allow you to write records to `wtmp`, to display and manipulate `wtmp`, and to create total accounting records from `wtmp`. These commands are covered in this section.

Writing Records to wtmp - acctwtmp

The command `acctwtmp` allows you to write records to `wtmp` for whatever reason you might have. `acctwtmp` is normally invoked by `startup` and `shutacct` to record when System Accounting was turned on and off, respectively. The format of the command is:

```
acctwtmp "reason"
```

where *reason* is a string describing the reason for writing the record to `wtmp`. Note that `acctwtmp` does not directly write records to `wtmp`: it writes a record containing the terminal name, current time, and reason string to standard output. To actually write the record to `wtmp` you must append the output from `acctwtmp` to the `wtmp` file as follows:

```
acctwtmp "reason" >>/etc/wtmp
```

The *reason* string may be any combination of letters, numbers, spaces, and the dollar sign ($), but may not exceed 11 characters in length. (*reason* must be enclosed in double quotes as shown.)

Displaying Connect Session Records - fwtmp

To display the contents of `wtmp`, you can use the command `fwtmp`. When no options are used, `fwtmp` uses standard input records of the format contained in `wtmp`; it writes to standard output the ASCII readable equivalent of the input records. Two alternatives exist for the output from this command:

- The output of this command can be edited, via an HP-UX editor such as `vi`, and then rewritten to `wtmp` using special `fwtmp` options described below.
- The output can be supplied as input to commands which convert the information to total accounting records.

The syntax of `fwtmp` is:

```
fwtmp [-ic]
```

If no option is specified for the `fwtmp` command, then input is in binary format and is to be converted to ASCII readable format. The various combinations of the options *i* and *C* provide other combinations of input and output formats. The possible options are described below:

Option	Description
`-ic`	Input is in ASCII readable form and is to be converted to binary form. This is essentially the opposite of using `fwtmp` without any options.
`-i`	Both input and output are in ASCII readable format. This is the same as performing an ASCII to ASCII copy.
`-c`	Both input and output are in binary format—a binary to binary copy.

The following example shows the output produced by **fwtmp**. It is followed by a description of each column in the report:

```
fwtmp < /etc/wtmp
                 system boot    0  2 0000 0000 479472540 Mar 12 03:49:00 1985
root          co console       0  7 0000 0000 479475173 Mar 12 04:32:53 1985
                 acctg on      0  9 0000 0000 479493135 Mar 12 09:32:15 1985
mike          a1 ttya1       352  7 0000 0000 479493590 Mar 12 09:40:00 1985
mike          a1 ttya1       352  8 0011 0000 479496000 Mar 12 10:20:00 1985
sarah         07 tty07       353  7 0000 0000 479518335 Mar 12 16:32:15 1985
bill          10 tty10       351  7 0000 0000 479521475 Mar 12 17:24:35 1985
sarah         07 tty07       353  8 0011 0000 479522478 Mar 12 17:41:18 1985
bill          10 tty10       351  8 0011 0000 479526487 Mar 12 18:48:07 1985
              co console       0  8 0011 0000 479526488 Mar 12 18:48:08 1985
                 acctg off     0  9 0000 0000 479526493 Mar 12 18:48:13 1985
                 system boot   0  2 0000 0000 479389800 Mar 12 05:00:00 1985
```

Column	Description
1	The login name of the user who logged in or out.
2	**/etc/inittab** ID (this is usually the number of the line on which the connect session took place).
3	The name of the device on which the connect session occurred.
4	Process ID of the user who logged in or out.
5	Entry type. This field contains information on the type of record—for example, it shows whether the record is a login record (entry type=7), logout record (entry type=8), or if the record was written by **acctwtmp** (entry type=9). Refer to *utmp(4)* for more details on this field.
6-7	Exit status for connect session. Refer to *login(1)* and *utmp(4)* for details.
8	Time that entry was made (in elapsed seconds since January 1, 1970).
9-12	The equivalent of column 8 in date/time format showing month, day, time of day (in 24-hour format), and year.

Fixing wtmp Errors - wtmpfix

When a user logs into HP-UX, the `login` program stores the value seven (7) in the entry type field of the connect session record. When the same user logs out, an entry type of eight (8) is recorded. You can see this by examining the sample output created by `fwtmp` in the previous section. Note that in the example, login records precede their corresponding logout records in chronological order.

Occasionally, this time-stamped ordering becomes inconsistent: logout records might precede login records. (This occurs when the date and time are reset while users are still logged in.) When this happens, the commands that create connect session total accounting records will not work properly.

Fortunately, the command `wtmpfix` fixes corrupted `wtmp` files. `wtmpfix` takes `wtmp` binary records as input and corrects the time/date stamps to be consistent; its standard output is also binary `wtmp` records. The syntax for `wtmpfix` us:

`wtmpfix` [*files*]

If *files* is given, then input is taken from *files*. A dash (-) can be used in place of *files* to indicate standard input. *Note that if you specify* `wtmp` *as both input to and output from this command,* `wtmp` *will be destroyed*. Therefore, take care not to destroy `wtmp`. The following shows how to properly fix `wtmp` using `wtmpfix`:

```
wtmpfix /etc/wtmp > wtmp.temp
fwtmp -c < wtmp.temp > /etc/wtmp
rm wtmp.temp
```

Creating Total Accounting Records

This final set of connect session accounting commands is used to create connect session total accounting records. Before reading any further, you may want to review Figure 12-5 (in the "System Data Flow" section of this chapter).

acctcon1

`acctcon1` converts a sequence of login/logoff records (of the format contained in `wtmp`) read from its standard input to a sequence of records, one per login

session. Its input is normally redirected from `wtmp`; its output is columnar ASCII and can be supplied as input to `prctmp` or `acctcon2`.

The use of `acctcon1` is illustrated below by first displaying the contents of `wtmp` with `fwtmp`, and then using `acctcon1` to create a connect session summary file. `acctcon1`'s columnar data `acctcon1` is described following the report:

12

```
fwtmp < /etc/wtmp
                system boot    0  2 0000 0000 479472540 Mar 12 03:49:00 1985
root         co console       0  7 0000 0000 479475173 Mar 12 04:32:53 1985
                acctg on      0  9 0000 0000 479493135 Mar 12 09:32:15 1985
mike         a1 ttya1       352  7 0000 0000 479493590 Mar 12 09:40:00 1985
mike         a1 ttya1       352  8 0011 0000 479496000 Mar 12 10:20:00 1985
sarah        07 tty07       353  7 0000 0000 479518335 Mar 12 16:32:15 1985
bill         10 tty10       351  7 0000 0000 479521475 Mar 12 17:24:35 1985
sarah        07 tty07       353  8 0011 0000 479522478 Mar 12 17:41:18 1985
bill         10 tty10       351  8 0011 0000 479526487 Mar 12 18:48:07 1985
             co console       0  8 0011 0000 479526488 Mar 12 18:48:08 1985
                acctg off     0  9 0000 0000 479526493 Mar 12 18:48:13 1985
acctcon1 < /etc/wtmp
20095488    353    sarah   1665   2478   479518335  Tue Mar 12 16:32:15 1985
521012224   352    mike                  479493590  Tue Mar 12 09:40:00 1985
520095488   351    bill    0      5012   479521475  Tue Mar 12 17:24:35 1985
521011712   0      root    41047  6488   479475173  Tue Mar 12 04:32:53 1985
```

Column	Description
1	Shows the device address (in decimal equivalent of major/minor device address) at which the connect session occurred.
2	Gives the user ID for the connect session record.
3	Displays the login name for the user.
4	Shows the number of prime connect time seconds that were used during the connect session.
5	Shows non-prime connect seconds.
6	Displays the connect session starting time (in seconds elapsed since January 1, 1970).
7-11	Shows the conversion of column six to date/time format showing month, day time of day (in 24-hour format), and year.

In addition to its normal usage, `acctcon1` has four options:

Option	Description
`-p`	This option tells `acctcon1` not to produce one record per connect session. Instead, `acctcon1` simply echoes its input—one line per `wtmp` record—showing line name, login name, and time (in both seconds and day/time format). Using this option is similar to using `fwtmp`, except that this option doesn't show status information, whereas `fwtmp` does.
`-t`	`acctcon1` maintains a list of lines on which users are logged in. When it reaches the end of its input, it emits a session record for each line that still appears to be active. It normally assumes that its input is a current file, so that it uses the current time as the ending time for each session in progress. The `-t` flag causes it to use, instead, the last time found in its input, thus assuring reasonable and repeatable numbers for non-current files.
`-l` *file*	This option causes a line usage summary report to be placed in *file*. This report shows each line's name, number of minutes used, percentage of total elapsed time used, number of sessions charged, number of logins, and number of logins and logoffs. This report can be used to keep track of line usage, identify bad lines, and find software/hardware oddities. *Note that hang-up, termination of* `login`*, and termination of the login shell each generate logoff records; therefore, the number of logoffs is often three to four times the number of connect sessions.*
`-o` *file*	Using the `-o` option (for example, `acctcon1 -o f_overall`) causes *file* to be filled with an overall record for the accounting period, giving starting time, ending time, number of reboots, and number of date changes.

The following example of the line use file (`line_use`) is created from the same `wtmp` file used in the previous `acctcon1` example; the standard output of `acctcon1` has been redirected into the file `ctmp`:

```
acctcon1 -t -l line_use < /etc/wtmp > ctmp
cat line_use
TOTAL DURATION IS 899 MINUTES
LINE     MINUTES PERCENT # SESS  # ON    # OFF
console  856     95      1       1       1
tty07    69      8       1       1       1
ttya1    40      4       1       1       1
tty10    84      9       1       1       1
TOTALS   1049    --      4       4       4
```

prctmp

The `prctmp` command is simple. Its only function is to put headings on the output created by `acctcon1`. `prctmp` makes a readable report from the output of `acctcon1`.

`prctmp` takes its input from standard input; therefore, to create a `prctmp` report from `acctcon1` information, you can simply pipe the output from `acctcon1` into `prctmp` as follows:

```
acctcon1 < /etc/wtmp | prctmp
```

`prctmp` will respond by generating a report with appropriate headings over the columns of output from `acctcon1`.

acctcon2

`acctcon2` creates connect session total accounting records from standard input of the format created by `acctcon1`. In other words, to create connect session total accounting records, simply send the output from `acctcon1` into the input of `acctcon2`.

The total accounting records created by `acctcon2` are sent to standard output. So if you want to store these records, you must redirect standard output. The following command line shows how to write total accounting records from the connect session record file (`wtmp`) into the file `ctacct`:

```
acctcon1 < /etc/wtmp | acctcon2 > ctacct
```

Process Accounting

Process accounting commands provide the means to accumulate execution statistics—such as memory usage, CPU time, number of input/output transfers—for individual processes. This section describes how to:

12

1. Turn process accounting on.
2. Turn process accounting off.
3. Make sure that the process accounting file (`pacct`) doesn't become too large.
4. Display process accounting records.
5. Generate a command summary report.
6. Create total accounting records from the process accounting file.

You might find it helpful to look at the System Data Flow Diagram Figure 12-3 before reading this section.

Turning Process Accounting On

Before System Accounting can generate process accounting data, process accounting must be turned on. Two commands can be used to accomplish this task: `turnacct on` and `accton`. After process accounting has been turned on, the kernel will write a process accounting record for every terminating process. The record will be written into the current process accounting file (`pacct` by default).

Note

The `startup` command, placed in the system initialization shell script `/etc/rc`, automatically turns process accounting on. Therefore, if you have updated `/etc/rc` for System Accounting (as described in the section "How to Install System Accounting" in this chapter), process accounting will automatically be activated, and you should seldom need to use the commands described here.

These commands are described for your benefit in case you ever need to manually turn process accounting on or off.

turnacct on

The command used most often to activate accounting is `turnacct on`; only the superuser and the `adm` login can execute this command. `turnacct on` assumes that the process accounting file is the default file `pacct`. The action of `turnacct on` can be summarized as follows:

1. Check to see if the process accounting file `pacct` exists.
2. If `pacct` doesn't exist, then create a new `pacct` file.
3. Turn process accounting on by invoking `accton` with the filename argument `pacct`.

To execute this command, simply enter `turnacct on` at the HP-UX prompt.

accton

Again, only the superuser and the `adm` login can execute `accton`. When invoked with a filename argument, `accton` turns on process accounting and makes the specified filename the current process accounting file. For example,

```
accton /usr/adm/pacct
```

tells the kernel to start writing process accounting records to the file called `/usr/adm/pacct`. The next example would activate process accounting and make the current process accounting file `/usr/adm/XX107`:

```
accton /usr/adm/XX107
```

Note The filename you specify must be an existing file; otherwise, `accton` will fail.

Note that in the Figure 12-3, `accton` is shown calling another routine, `acct`. `acct` is the system call that actually tells the kernel to start writing process accounting records. Refer to the *HP-UX Reference Manual* for more details on *acct(2)*.

Turning Process Accounting Off

Two commands are used to turn process accounting off: `turnacct off` and `accton` (with no filename argument). These commands tell the kernel to stop writing records to the current process accounting file.

12

Note If you have updated the `/etc/shutdown` shell script as described in the section "How to Install System Accounting" in this chapter, you will seldom, if ever, use these commands. The reason is that the `shutacct` command, added to `/etc/shutdown`, automatically turns process accounting off.

---

turnacct off

`turnacct off` can be executed only by the superuser and the `adm` login. `turnacct off` turns process accounting off by invoking the `accton` command without the optional filename argument. You execute this command by typing:

```
turnacct off
```

accton

When `accton` is invoked without the optional filename argument, process accounting is turned off. You would enter this command as:

```
accton
```

As shown in Figure 12-3, `accton` tells the kernel to stop writing process accounting records by using the system call `acct`.

Checking the Size of pacct

On a multiuser system, many processes can execute during a single hour. Therefore, process accounting files have the potential to become quite large. System Accounting has built-in mechanisms that insure that the default process accounting file `pacct` doesn't become too large. The two commands used for this purpose are: `turnacct switch` and `ckpacct`.

Note The commands described here work only on the default process accounting file, `pacct`.

ckpacct

The command `ckpacct` is normally invoked by `cron` every hour to insure that the current process accounting file `pacct` hasn't become to large. The format of `ckpacct` is:

`ckpacct` [*blocks*]

If the size of `pacct` exceeds the *blocks* argument, 1 000 by default if *blocks* is not specified, then `turnacct switch` is executed. `turnacct switch` renames the current `pacct` file and creates a new `pacct` file.

Note If the amount of free space falls below a certain threshold, `ckpacct` will automatically turn off process accounting via `turnacct off`. With 5.0 HP-UX, accounting was turned off when free space fell below 2%, and back on when it went above 4%. Beginning with 5.1 HP-UX, these percentages are configured using the `acctresume` and `acctsuspend` system parameters (Refer to "Operating System Parameters"). These percentages are in addition to the `minfree` attribute. When free space goes over the specified percentage, process accounting will be reactivated.

The kernel may also enforce a size limit on the size of `pacct`. This will take precedence over the limit set by `ckpacct`. Refer to *acctsh(1M)* and *acct(2)* in the *HP-UX Reference Manual* for more details.

turnacct switch

`turnacct switch` is used to create a new `pacct` file when the current `pacct` file is too large. The action of `turnacct switch` can be summarized as follows:

12

1. Process accounting is temporarily turned off.
2. The current `pacct` file is renamed to `pacct`*incr*, where *incr* is a number starting at *1* and incrementing by one for each additional `pacct` file that is created via `turnacct switch`.
3. After the old `pacct` file is renamed to `pacctincr`, a new, current `pacct` file is created.
4. Process accounting is restarted; the kernel starts writing records to the newly created `pacct` file.

The example below illustrates the effect of using the `turnacct switch` command. In the example, `turnacct switch` is executed from the `adm` home directory `/usr/adm`. Comment lines begin with a cross-hatch (#) and are included in the example only as explanatory material:


```
$ #
$ # First, list all the process accounting files
$ # (at this point, there is only one).
$ #
ll pacct*
-rw-rw-r--    1 adm     adm      2196 Mar 21 12:44 pacct
$ #
$ # Now execute turnacct switch, which will rename the current
$ # pacct file to pacct1 and will create a new pacct file.
$ #
turnacct switch
$ #
$ # Now verify this by listing all process accounting
$ # files again.
$ #
ll pacct*
-rw-rw-r--    1 adm     adm        72 Mar 21 12:46 pacct
-rw-rw-r--    1 adm     adm      2196 Mar 21 12:44 pacct1
$ #
$ # The current process accounting file is pacct.  The previous
$ # process accounting file is now named pacct1.
$ #
```

Displaying Process Accounting Records - acctcom

The `acctcom` command allows you to display records from any file containing process accounting records. Normally you would use this command to display records from the `pacct` files (`pacct`, `pacct1`, `pacct2` ...).

`acctcom` is a very versatile command; its syntax follows:

`acctcom` [[*options*] [*file*]] ...

If no *file* is specified, `acctcom` uses the current `pacct` file as input. Input can also be taken from standard input. Some of `acctcom`'s options allow you to select only the records that you want to see; other options control the format of the report.

The information contained in this section is organized as follows:

- First, definitions are given for the columnar data produced by `acctcom`.
- Command options that control the format of the report are discussed.
- Options that allow you to select particular records are described.
- Finally, to help you understand how to use `acctcom`'s options, sample `acctcom` reports are shown.

Definitions of Information Produced by acctcom

`acctcom` generates a columnar report with descriptive headings on each column. Each line of the report represents the execution statistics that a particular process accumulated during its lifetime. The standard columns in the report—that is, the columns that are displayed when none of `acctcom`'s options are specified—are shown below:

Column Header	Definition
`COMMAND NAME`	The name of the command or program that spawned the process is shown here. Whenever you enter a command, you are spawning a process. For example, if you enter the command `ll /usr/lib/acct` you are creating a process with the command name `ll`. If a command requiring superuser privileges is executed, a # appears before the command name.
`USER`	The login name of the user who created the process is displayed here.
`TTYNAME`	This is the name of the terminal from which the process was executed. If the process was not executed from a known terminal (for example, if it was executed via `cron`), then a question mark(*?*) appears in this column.

Column Header	Definition
`START TIME`	The time that the process began executing (in *hh*:*mm*:*ss* format) is displayed here.
`END TIME`	This is the time (*hh*:*mm*:*ss*) that the process finished executing.
`REAL (SECS)`	The number of seconds that elapsed from `START TIME` to `END TIME` is shown in this column.
`CPU (SECS)`	This column shows how much of the CPU's time a process used during its execution.
`MEAN SIZE(K)`	This is a rough estimate (in kilobytes) of the amount of memory that a process used during execution. This estimate is determined from the current process's memory usage at each system clock interrupt. It is, therefore, subject to statistical sampling errors. Only the memory resident pages of a process are counted; no pages in the swap space are counted. Shared code and data is divided among the processes using it. The size is divided by the number of processes sharing the code or data.

Listed below are the columns that are not displayed on the standard report, but which can be displayed by using `acctcom` options:

Column Header	Definition
F	For a process created by **fork** which does not do an **exec**, this column takes the value *1*; commands which require superuser privileges show a *2*; superuser commands which do a **fork** without an **exec** show a *3*; otherwise, this column shows a *0*.
STAT	This column displays the system exit status. (This is *not* the status returned by **exit** to a parent process during **wait**). When a process terminates normally, this field shows a *0*. If a command terminates abnormally, then a value other than zero is shown. For example, if you interrupt a command with the DEL key, this column will contain a *2*.
HOG FACTOR	The hog factor is computed as the **CPU** time divided by **REAL** time; it provides a relative measure of the available CPU time used by the process during its execution. For example, a hog factor of less than 0.50 indicates that the process spent less than half of its time using the CPU. A hog factor of 0.75 indicates that a process spent 75% of its time using the CPU.
KCORE MIN	This calculation provides a combined measurement of the amount of memory used (in kilobytes) and the length of time it was used (in minutes). It is computed as follows: `KCORE MIN = CPU (SECS) * MEAN SIZE(K) / 60`
CPU SYS	This is the portion of total CPU time that was spent executing operating system code, such as system calls (for example, writing to disk).
USER (SECS)	This is the remaining portion of CPU time. User CPU time is the amount of time actually spent executing a process's code (rather than system code).

Column Header	Definition
CPU FACTOR	Whenever you execute a command, the CPU spends part of its time actually executing the command's code (user CPU time) and spends the rest of its time performing system functions, such as writing to the disk or terminal (system CPU time). That is, total CPU time is comprised of both CPU SYS and USER CPU time: CPU (SECS) = CPU SYS + USER (SECS) The CPU factor shows the ratio of user CPU time to total CPU time: CPU FACTOR = USER (SECS) / (CPU SYS + USER (SECS)) For example, if a command has a CPU factor of 0.35, that means that the CPU spent 35% of its time executing user code and 65% performing system functions.
CHARS TRNSFD	The number of characters (bytes) read and/or written by the command is displayed in this column.
BLOCKS R/W	This column shows the number of file system blocks read and/or written as a result of executing this command. *This number is not directly related to* CHARS TRNSFD *and may vary each time the command is executed*, because BLOCKS R/W is affected by directory searches made before opening files, other processes accessing the same files, and general file system activity.

Report Format Options

When no report format options are specified, `acctcom` will produce a report containing only the default information. Optional information can be displayed only by using the report format options. Definitions of the report format options follow:

12

Option	Description
`-a`	Cause average statistics to be displayed at the end of the report. The following information is shown: total number of commands processed (`cmds=xxx`) ■ average real time per process (`Real=x.xx`) ■ average CPU time per process (`CPU=x.xx`) ■ average USER CPU time per process (`USER=x.xx`) ■ average SYS CPU time per process (`SYS=x.xx`) ■ average characters transferred (`CHARS=x.xx`) ■ average blocks transferred (`BLK=x.xx`) ■ average CPU factor (`USR/TOT=x.xx`) ■ average HOG factor (`HOG=x.xx`)
`-b`	Display the process records in reverse order: most recently executed commands will be shown first.
`-f`	Print the `fork/exec` flag (`F` column) and process exit status (`STAT` column) on the report.
`-h`	Cause the optional `HOG FACTOR` column to be displayed, instead of the standard mean memory size column `MEAN SIZE(K)`.
`-i`	Replace the standard `MEAN SIZE(K)` column in the report with the optional I/O counts—`CHARS TRNSFD` and `BLOCKS R/W`.
`-k`	Replace the standard `MEAN SIZE(K)` column with `KCORE MIN`.

Option	Description
`-m`	Show the default column `MEAN SIZE(K)` on the report. This option is used to include `MEAN SIZE(K)` when it has been bumped off by another option. The following example: produces a report showing both `KCORE MIN` and `MEAN SIZE(K)`. `acctcom -km`
`-r`	Include the optional `CPU FACTOR` column in the report.
`-t`	Show separate system and user CPU times (`CPU SYS` and `USER (SECS)`, respectively).
`-v`	Suppress the printing of column headings at the top of the report.
`-q`	This option is the same as the `-a` option, except that individual process accounting records are not displayed—only the averages are displayed.
`-o` *ofile*	Copy the input process accounting records to *ofile*.

Record Selection Options

The options described here allow you to select the records that are included in the report produced by `acctcom`. For each option, descriptions and examples are provided:

Option	Description
`-l` *line*	Display only the processes that were executed from the user terminal `/dev/`*line*. For example: `acctcom console` would display records only for the processes that were created from the terminal `console`.

`-u` *user*	Show only the processes belonging to *user*. *user* can be any of the following:
	■ a user ID (for example, `acctcom -u 355`)
	■ user name (`acctcom -u julie`)
	■ a cross-hatch # (`acctcom -u#`)
	■ a question mark ? (`acctcom -u?`)
	If # is specified as the user name, then only the commands that require superuser privileges will be displayed by acctcom. If ? is given as the user, then only the processes with unknown process IDs will be displayed. As an example, the following two commands are equivalent:
	`acctcom -u 0`
	`acctcom -u root`

`-g` *group*	Show only the processes belonging to *group*. *group* may be specified as either a group name or group ID. For example, suppose the group `pseudo` with group ID `300` is defined in `/etc/group`; then the following two commands are equivalent:
	`acctcom -g 300`
	`acctcom -g pseudo`

Option	Description
`-s` *time*	Select processes existing *at or after time*. Time is given in 24-hour format—*hr*[:*min*[:*sec*]]. The following example would display all the processes that existed at or after 3:30pm:

```
acctcom -s 15:30
```

`-e` *time*	Select processes that existed *at or before time*. Time is supplied in 24-hour format *hr*[:*min*[:*sec*]]. The next example would display all the processes that existed between midnight and 12:15am:

```
acctcom -e 0:15
```

`-S` *time*	Select processes *starting* at or after time where *time* is in 24-hour format. The following example would display all the processes that *started* at 1:30:42pm or after:

```
acctcom -S 13:30:42
```

`-E` *time*	Display only the processes that *terminated* at or before *time*, where time is in 24-hour format *hr*:[*min*[:*sec*]]. Note both the `-S` and `-E` options with the same *time* argument will cause `acctcom` to display only the processes that existed at the specified *time*. For example, to see all the processes that existed at exactly 30 minutes past noon, you would enter:

```
acctcom -S 12:30 -E 12:30
```

`-n` *pattern*	Show only the commands matching *pattern*. *pattern* can be a regular expression as described in *ed(1)*, except that + means one or more occurrences. For example, to display all processes that were created by executing the `ls` command, you would enter:

```
acctcom -n ls
```

To display all the commands that start with `acct`, enter:

```
acctcom -n acct
```

To see all the commands that contain the letter *m* in their spelling you can use the wild card character `*`. Type:

```
acctcom -n .*m.*
```

Option	Description
`-H` *factor*	Display only those processes whose hog factor exceeds *factor*. For example, `acctcom -H 0.85` would display all the processes that spent over 85% of their execution time in the CPU. You can use this option to find greedy processes—processes that are hogging the CPU.
`-O` *time*	Show only those processes whose system CPU time exceeds *time*, specified in seconds. The following example would be used to determine which processes took more than 8.25 seconds of operating system CPU time to execute: `acctcom -O 8.25` This option could be used to determine which processes are making heavy use of the operating system calls.
`-C` *sec*	Show only the processes whose total CPU time (`SYS + USER`) exceeds *sec* seconds. The next example would display all the processes that used over 5.28 seconds of CPU time to execute: `acctcom -C 5.28`
`-I` *chars*	Display only the processes transferring more characters than the limit given by *chars*. For example, `acctcom -I 10240` will display all the processes that transferred over ten kilobytes of characters (10 240 = 10 × 1 024 bytes).

Sample Reports

The following sample report illustrates the use of `acctcom` without any options. The report generated is the standard report produced when no options are specified:


```
acctcom
ACCOUNTING RECORDS FROM:  Thu Mar 21 12:52:26 1985
COMMAND                          START    END          REAL     CPU    MEAN
NAME       USER     TTYNAME      TIME     TIME       (SECS)  (SECS) SIZE(K)
#accton    root     console    12:52:26 12:52:26      0.12    0.10   19.00
ls         sarah    tty07      14:04:08 14:04:08      0.28    0.23   16.50
ckpacct    adm      ?          14:30:00 14:30:05      5.13    1.45   24.00
pwd        bill     tty10      15:09:07 15:09:07      0.48    0.22   22.50
find       sarah    tty07      18:51:37 18:51:39      2.73    0.15   26.50
tabs       root     console    19:10:18 19:10:18      0.92    0.13   23.50
stty       root     console    19:10:19 19:10:19      0.88    0.08   26.00
mail       bill     tty10      19:10:21 19:10:22      1.78    0.23   28.50
news       root     console    19:10:23 19:10:23      0.73    0.12   23.00
acctcom    adm      ttya0      19:53:16 19:53:38     22.58    2.55   28.50
```

The next example displays all the processes created between 7:00pm and 7:30pm by the user `root`. In addition, the optional CPU factor and average statistics are included in the output:

```
acctcom -S 19:00 -E 19:30 -u root -ah
START AFT: Thu Mar 21 19:00:00 1985
END BEFOR: Thu Mar 21 19:30:00 1985
COMMAND                          START    END          REAL     CPU     HOG
NAME       USER     TTYNAME      TIME     TIME       (SECS)  (SECS)  FACTOR
tabs       root     console    19:10:18 19:10:18      0.92    0.13    0.14
stty       root     console    19:10:19 19:10:19      0.88    0.08    0.09
news       root     console    19:10:23 19:10:23      0.73    0.12    0.16
cmds=3 Real=0.84   CPU=0.11   USER=0.02   SYS=0.09   CHAR=26.12   BLK=11.50
USR/TOT=0.19 HOG=0.13
```

Sample reports are helpful, but the best way to learn the various `acctcom` options is to use them. Take a few minutes to experiment with this command; it is very powerful and can provide you with much useful information if used properly.

Command Summary Report - acctcms

The `acctcms` command takes process accounting records as input; but instead of reporting on the individual processes, `acctcms` generates a report on the commands that generated the process accounting records. The action of `acctcms` can be summarized as follows:

1. `acctcms` looks through the input process accounting records and accumulates execution statistics for each unique command name. This information is stored in internal summary format—one record per command name.
2. Depending on the `acctcms` options used, the command summary records created in step 1 are sorted.
3. The command summary records are written to standard output in the internal summary format mentioned in step 1. This format is not readable.

The syntax of the `acctcms` command is:

`acctcms` [*options*] `files`

where *files* is a list of the input process accounting files for which the command summary report is to be generated. The *options* are discussed in the following sections.

Producing a Readable Report - the -a option

By default, the output of `acctcms` is in internal summary record format; if you display it to your terminal, all you see is gibberish. To get a ASCII, readable report, use the `-a` option.

The `-a` option causes `acctcms` to produce a report with descriptive column headings. Total and average (mean) execution statistics for each command are displayed—one line per command—along with total and average statistics over all commands in the report. Descriptions of the columnar data produced by `acctcms` are shown in the following table.

Column Header	Description
COMMAND NAME	The name of the command for which execution statistics are summarized. Unfortunately, all shell procedures are lumped together under the name **sh**, because only object modules are reported by the process accounting system.
NUMBER CMDS	The total number of times that the command was invoked.
TOTAL KCOREMIN	The total amount of kcore minutes accumulated for the command. (Refer to the section "Definitions of Information Produced by acctcom" in this chapter for a more complete description of kcore minutes.)
TOTAL CPU-MIN	The total CPU time that the named command has accumulated.
TOTAL REAL-MIN	Total accumulated real time minutes are displayed in this column.
MEAN SIZE-K	The average amount of memory (in kilobytes) consumed by the command.
MEAN CPU-MIN	The average CPU time consumed per command invocation is shown here; the following equation shows how it is computed: `MEAN CPU-MIN = TOTAL CPU-MIN / NUMBER CMDS`
HOG FACTOR	The average hog factor over all invocations of the command. It is computed as: `HOG FACTOR = TOTAL CPU-MIN / TOTAL REAL-MIN`
CHARS TRNSFD	The total number of characters transferred by the command. Note that this number may sometimes be negative.
BLOCKS READ	A total count of the physical blocks read and written by the given command. (Refer to the section "Displaying Process Accounting Records—acctcom" in this chapter for details on the significance of this total.)

Note When only the `-a` option is specified, the report is sorted in descending order on the `TOTAL KCOREMIN` column: commands using more `TOTAL KCOREMIN` are shown before those using fewer `TOTAL KCOREMIN`. This report gives a relative measure of the amount of memory used over time by the various commands: commands toward the start of the report are making more use of memory resources than are commands toward the end of the report.

Other Options

In addition to the `-a` option, several other options can be used to control the format of the report generated by `acctcms`. Some options specify which field to sort the report on; other options control the printing of prime/non-prime time usage. The options and a description of their use follow:

Option	Description
`-c`	Sort the commands in descending order on `TOTAL CPU-MIN`, rather than the default `TOTAL KCOREMIN`. This report can be used to determine which commands are using most of the computer's CPU time.
`-n`	Cause the report to be sorted in descending order on the column named `NUMBER CMDS`. Commands toward the start of this report are the ones used most frequently; commands toward the end are used least often.
`-j`	Combine all commands invoked only once on one line of the report. This line is denoted by having "`***other`" in the `COMMAND NAME` column. This option is useful for shortening a report that has many one-invocation commands.
`-o`	*Used only with the* `-a` *option*, `-o` causes the report to be generated only for commands that were executed during non-prime time (as specified in the `holidays` file). You can use this option to get a non-prime time command summary report.
`-p`	*Also used only with the* `-a` *option*, `-p` elicits a report generated only for commands that were executed during prime time (as specified in `holidays`). This option is used to get a prime time command summary report.

Option	Description
-apo	When the options -o and -p are used together with -a, a combination prime and non-prime time report is produced. The output of this report is same as that produced by -a alone, except that the NUMBER CMDS, TOTAL CPU-MIN, and TOTAL REAL-MIN columns are divided into two columns—one for prime time totals, the other for non-prime time. (Prime time columns have a (P) header, while non-prime time columns are headed by (NP).)
-s [*files*]	Specifies that any named input *files* following the -s on the command line are already in internal summary format. This option is useful for merging previous acctcms reports with current reports. The following example uses -s to create a command summary report from previous process accounting files (pacct?) and the current process accounting file (pacct). The final ASCII report is stored in the file ascii_summary.

```
acctcms pacct? > old_summary
acctcms pacct > new_summary
acctcms -as old_summary new_summary > ascii_summary
```


Sample Report

The ASCII reports produced by acctcms contain more than 80 characters per line. When these reports are displayed at an 80-column terminal, the lines wrap around on the screen. In addition, if the report is printed on an 80-column printer, some of the rightmost columns will be lost. Therefore, be sure to do one of the following:

- Use a printer with compressed print capabilities, so that all of the report will fit on standard computer paper.
- Use a printer with enough columns to display all the information—for example, a 132-column printer.

The following example generates a command summary report for the current process accounting file (no file is specified, so the current `pacct` file is assumed). By giving the `-j` option, all the commands that were executed only once are grouped under the command name `***other`. Note also that total execution statistics for all commands are grouped under the command name `TOTALS`.

```
acctcms -aj
                                TOTAL COMMAND SUMMARY
COMMAND NUMBER   TOTAL    TOTAL     TOTAL  MEAN     MEAN     HOG    CHARS BLOCKS
NAME      CMDS KCOREMIN  CPU-MIN REAL-MIN SIZE-K  CPU-MIN  FACTOR TRNSFD  READ

TOTALS      61    17.63    0.38    164.49  46.25     0.01    0.00 104553   1027

acctcms     17    12.13    0.16      0.35  76.72     0.01    0.45  49192    306
sh           8     2.43    0.09    152.86  26.79     0.01    0.00   9043    163
more         3     0.73    0.02     10.50  31.00     0.01    0.00  21618     83
ll           6     0.61    0.04      0.11  16.50     0.01    0.33   5715     95
acctcom      4     0.58    0.02      0.07  28.50     0.01    0.30  15319     42
***other     9     0.54    0.02      0.14  25.26     0.00    0.16    459    161
cat          4     0.19    0.01      0.35  22.97     0.00    0.02   3112     52
rm           2     0.11    0.00      0.02  22.22     0.00    0.29      0     29
chmod        2     0.10    0.00      0.01  22.00     0.00    0.35      0     15
accton       2     0.08    0.00      0.02  19.00     0.00    0.29      0     22
sed          2     0.08    0.01      0.04  14.50     0.00    0.13     73     38
echo         2     0.05    0.00      0.02  20.00     0.00    0.16     22     21
```

Creating Total Accounting Records

Two commands—`acctprc1` and `acctprc2`—are used to create total accounting records from the process accounting files. The output from `acctprc1` is supplied as input to `acctprc2` which produces the total accounting records. These commands are normally invoked by `runacct` to produce daily accounting information.

acctprc1

This command reads process accounting records from standard input, adds login names corresponding to the user ID of each record, and then writes for each process an ASCII line showing:

- the ID of the user that created the process
- the user's login name
- prime CPU time in ticks (a "tick" is one fiftieth of a second)
- non-prime CPU time, also in ticks
- mean memory size (in pages—4 Kbytes per page)

The format of `acctprc1` is:

`acctprc1` [*ctmp*]

where `ctmp` contains a list of login sessions of the form created by `acctcon1`, sorted by user ID and login name.

Note The number of sessions should be 1000 or less. If there are more than 1000 sessions, the accounting system "hangs"—i.e., suspends indefinitely—and must be killed manually (via the `kill` command) and restarted.

To use `acctprc1`, input must be redirected from a process accounting file. The following example creates a file, `ascii_ptacct`, containing ASCII process accounting information that can be used to create process total accounting records. This file is created from the current process accounting file `pacct`.

```
acctprc1 <pacct >ascii_ptacct
```

Normally, `acctprc1` gets login names from the password file `/etc/passwd`, which is sufficient on systems where each user has a unique user ID. However, on systems where different users share the same user ID, the `ctmp` file should be specified; it helps `acctprc1` distinguish different login names that share the same user ID.

acctprc2

This command reads from standard input records of the form created by `acctprc1`; it then summarizes the records by user ID and name, and writes the sorted summaries to standard output as total accounting records. The following example creates total accounting records for all processes in the current process accounting file `pacct`; the total accounting records are stored in the file `ptacct`.

```
acctprc1 <pacct |acctprc2 >ptacct
```

Charging Fees to Users – chargefee

System Accounting provides the capability to charge fees to specific users; the `chargefee` command is used to accomplish this task. `chargefee` allows you to charge generic *units* to a specific login name. The syntax of this command is:

`chargefee` *login_name number*

where *number* is the number of units to be charged to a particular user, and *login_name* is the login name of the user to whom *number* units are to be charged.

Note *number* can be any whole number in the range -32 768 to 32 767; when charging fees, keep in mind that the sum of each user's fees must also be within this range.

`chargefee` accumulates fee charge records in the file `/usr/adm/fee`. These records are then merged with other accounting records via `runacct`.

The following example charges 25 units to the user whose login name is `julie`:

```
chargefee julie 25
```

Suppose you inadvertently charged 247 units to the user named `zimblits`, and you want to return his charges to their original value. You would enter the following:

```
chargefee zimblits -247
```

Summarizing and Reporting Accounting Information

This final group of commands summarizes and reports accounting information. Certain commands display and merge total accounting files, while others generate the daily and monthly reports used to analyze system performance and bill users for resource usage. The following commands are discussed in this section:

- `prtacct`—displays total accounting records
- `acctmerg`—merges total accounting files
- `runacct`—generates daily summary files and reports
- `prdaily`—displays the daily summary files and reports created by `runacct`
- `monacct`—creates monthly summary files and reports

Displaying Total Accounting Records - prtacct

The `prtacct` command allows you to display the contents of a process accounting file. Its format is

```
prtacct file "heading"
```

where:

- *file* is the name of the total accounting file to be displayed.
- "*heading*" is a comment to be included in the standard report header produced by `prtacct`.

The format of the `prtacct` report is described next and is followed by an example.

Report Format

`prtacct` produces a columnar report with one line per total accounting record. Descriptive column headings are included in the report. Definitions of each column follow:

Column Header	Description
`UID`	This column shows the user ID of the owner of the total accounting record—that is, the ID of the user for whom the total accounting record was created.
`LOGIN NAME`	The login name of the owner of the total accounting record is displayed here.
`CPU (MINS)`	This column shows the total amount of CPU time (in minutes) that the user has consumed. This column is divided into prime and non-prime columns (`PRIME` and `NPRIME`, respectively). Information in these columns is created through process accounting commands.
`KCORE-MINS`	This represents a cumulative measure of memory and CPU time that a user consumed (refer to the section "Definitions of Information Produced by acctcom" in this chapter for a more precise definition). Information in this column is also divided into `PRIME` and `NPRIME` columns. This information is created through process accounting commands.
`CONNECT (MINS)`	This identifies the real time used (in minutes). In essence, what this column identifies is the amount of time that the user was logged in to the system. This column is also subdivided into `PRIME` and `NPRIME` columns. The connect session accounting commands are the source of this information.
`DISK BLOCKS`	The total number of disk blocks allocated to the user is shown here. This information is created via disk space accounting commands.
`# OF PROCS`	The total number of processes spawned by the user is displayed here. This information is created via the process accounting commands.

Column Header	Description
`# OF SESS`	This column shows how many times the user logged in. Connect session accounting commands create this data.
`# DISK SAMPLES`	This column indicates how many times the disk accounting was run to obtain the average number of disk blocks listed in the `DISK BLOCKS` column.
`FEE`	The number of fee units charged via `chargefee` is displayed here.

Example

The following example displays disk total accounting records. First, the total accounting records are created via disk space accounting commands; then, they are displayed using `prtacct`. When examining this report, take note of the following:

- There are many similarities between this and the sample report produced by `diskusg` (refer to the section "Disk Space Usage Accounting" in this chapter).
- Only the columns relating to disk space usage have non-zero values, because the total accounting records were created only from disk space usage accounting commands.

The example report produced by `prtacct` follows:

```
for file_system in `cat /etc/checklist`
> do
> diskusg $file_system >dtmp.`basename $file_system`
> done
diskusg -s dtmp.* |sort +0n +1 |acctdisk >disktacct
prtacct disktacct "DISC TOTAL ACCOUNTING RECORDS"

Mar 26 17:01 1985  DISC TOTAL ACCOUNTING RECORDS Page 1

    LOGIN   CPU (MINS)   KCORE-MINS  CONNECT (MINS) DISK   # OF  # OF # DISK FEE
UID NAME   PRIME NPRIME PRIME NPRIME PRIME   NPRIME BLOCKS PROCS SESS SAMPLES
0   TOTAL  0     0      0     0      0       0      11598  0     0    10      0
0   root   0     0      0     0      0       0      10616  0     0    1       0
1   bin    0     0      0     0      0       0      778    0     0    1       0
4   adm    0     0      0     0      0       0      96     0     0    1       0
350 fred   0     0      0     0      0       0      14     0     0    1       0
351 bill   0     0      0     0      0       0      32     0     0    1       0
352 mike   0     0      0     0      0       0      20     0     0    1       0
353 sarah  0     0      0     0      0       0      16     0     0    1       0
354 molly  0     0      0     0      0       0      22     0     0    1       0
355 julie  0     0      0     0      0       0      2      0     0    1       0
501 guest  0     0      0     0      0       0      2      0     0    1       0
```

Merging Total Accounting Files - acctmerg

Normally executed by `runacct`, the `acctmerg` command merges separate total accounting files into a single total accounting file. All the total accounting records for a particular user name and ID are merged together to form one total accounting record for the given user name and ID. This command is useful for merging disk, connect session, and process total accounting files together to form a single, comprehensive total accounting file.

`acctmerg` reads standard input and up to nine additional files, all in total accounting record format. Its syntax is:

`acctmerg` [*options*] [*file*] ...

where:

- *options* control the report format and the manner in which records are merged.
- *file* is one of up to nine files (in addition to standard input) that are to be merged into a single total accounting file, written to standard output.

Command Options

The following options may be used with `acctmerg` to control the report format and the manner in which the total accounting records are merged:

Option	Description
`-a`	`acctmerg` normally produces output as total accounting records. The `-a` option causes `acctmerg` to produce output in ASCII. Note that the output generated by using this option is the same as the report produced by `prtacct`, except that no report headings or totals are displayed—only the columnar data is shown.
`-i`	In the default case, `acctmerg` assumes that its input files contain total accounting records. If `-i` is specified, then `acctmerg` will expect input files to be in the ASCII format created by the `-a` option.
`-p`	This option simply echoes input records—no merging or processing is done. The output is displayed in the format produced by the `-a` option.
`-t`	This option produces a single total accounting record that summarizes all input records. To see the ASCII version of this record, you must use the `-t` and `-a` options together: `acctmerg -t -a <tot_acct_recs` *Note that* `-t` *and* `-a` *can be specified in any order, but they must be specified separately as shown.*
`-u`	Normally, `acctmerg` merges records that have the same user ID and user name. Using `-u` causes `acctmerg` to merge records on the basis of same user ID only—that is, the user name is disregarded as a key on which to merge records.
`-v`	This option causes `acctmerg` to produce output in verbose ASCII format. The same report is produced as the `-a` option, except that floating point numbers are displayed in more precise notation: *<mantissa>e<exponent>*

Use the -a, -v, and -i options to edit total accounting records. For example, if you created a total accounting file (ptacct) containing process total accounting records, and you want to make some adjustments to these records, use the following sequence to "repair" this file:

12

```
acctmerg -v -a <ptacct >ptacct.ascii      edit ptacct.ascii as desired ...
acctmerg -i <ptacct.ascii >ptacct         then copy the changes back to
                                          ptacct
```

Example

The following example creates disk, process, and connect session total accounting records, merges them together, and stores the merged file in the file merged_file:

```
for fs in `cat /etc/checklist`
> do
> diskusg $fs >dtmp.`basename $fs`
> done
diskusg -s dtmp.* |sort +0n +1 |acctdisk >dtacct
First, create disk space usage total accounting records (dtacct)
acctcon1 </etc/wtmp |acctcon2 >ctacct
Now create the connect session total accounting records (ctacct)
ptacct
for p_file in pacct*
> do
> acctprc1 <$p_file |acctprc2 >>ptacct
> done
Create process total accounting records (ptacct)
acctmerg dtacct ctacct <ptacct >tacct
$
```

Now merge all the total accounting files into a single total accounting file (tacct)

Creating Daily Accounting Information - runacct

runacct is the main daily accounting shell procedure. Start runacct via cron during non-prime hours, when users are logged off. This is because it does not correctly log time for users that log on before running runacct.

runacct processes disk, connect session, process, and fee accounting files. It prepares cumulative summary files for use by prdaily and for billing purposes. This section discusses the following aspects of runacct:

- files processed by runacct
- the states that runacct progresses through while executing
- recovery from runacct failure
- restarting runacct
- reports produced by runacct

Files Processed by runacct

The following files, processed by runacct, are of particular interest to the reader. (Filenames are given relative to the directory /usr/adm/acct.)

- nite/lineuse contains usage statistics for each terminal line on the system. This report is especially useful for detecting bad lines. If the ratio of logoffs to logins on a particular line exceeds 3 to 1, then there is a good possibility that the line is failing.
- nite/daytacct contains total accounting records from the previous day.
- sum/tacct contains accumulated total accounting records for each day's total accounting records (nite/daytacct) and can be used for billing purposes. It is restarted each month or fiscal period by the monacct shell script.
- sum/daycms is produced by acctcms. It contains the daily command summary. The ASCII version of this file is in nite/daycms.
- sum/cms holds the accumulation of each day's command summaries (sum/daycms). A new sum/cms file is created each month by monacct. The ASCII version of this file is in nite/cms.
- sum/loginlog maintains a record of the last time each user logged in.

- `sum/rprt`*mmdd* is the main daily accounting report created by `runacct`. The name for this report is created automatically by the system with *mm* being the month and *dd* the day of the report. This report can be printed via `prdaily`.

12

`runacct` takes care not to damage files in the event of errors. A series of protection mechanisms are used that attempt to recognize errors, provide intelligent diagnostics, and terminate processing in such a way that `runacct` can be restarted with minimal intervention. To accomplish these goals, the following actions are performed by `runacct`:

- `runacct`'s progress is recorded by writing descriptive messages to the `nite/active` file.
- All diagnostics output during the execution of `runacct` are redirected to the file `nite/fd2log`.
- If the files `lock` and `lock1` exist when `runacct` is invoked, an error message will be displayed, and execution will terminate.
- The `lastdate` file contains the month and day that `runacct` was last run and is used to prevent more than one execution per day.
- If `runacct` detects an error, a message is written to `/dev/console`, mail is sent to `root` and `adm`, locks are removed, diagnostics files are saved, and execution is terminated.

The States of runacct

In order to allow `runacct` to be restartable, processing is broken down into separate re-entrant *states*. As `runacct` executes, it records its progress by writing the name of the most recently completed state into the file called `/usr/adm/statefile`. After processing for a state is complete, `runacct` examines `statefile` to determine which state to enter next. When `runacct` reaches the final state (`CLEANUP`), the `lock` and `lock1` files are removed, and execution terminates. `runacct`'s states are described in the next table.

State	Action
SETUP	The command `turnacct switch` is executed. The process accounting files, `pacct?`, are moved to `Spacct?.`*mmdd*. The `/etc/wtmp` file is moved to `nite/wtmp.`*mmdd* with the current time added on the end.
WTMPFIX	`nite/wtmp.`*mmdd* is checked for correctness by `wtmpfix`. Some date changes will cause `acctcon1` to fail, so `wtmpfix` attempts to adjust the time stamps in the `nite/wtmp.`*mmdd* file if a date change record appears.
CONNECT1	Connect session records are written to `ctmp`. The `lineuse` file is created, and the `reboots` file, showing all of the boot records found in `nite/wtmp.`*mmdd*, is created.
CONNECT2	`ctmp` is converted to connect session total accounting records in the file `ctacct..`*mmdd*.
PROCESS	The `acctprc1` and `acctprc2` programs are used to convert the process accounting files, `Spacct?.`*mmdd*, to the total accounting records in `ptacct?.`*mmdd*. The `Spacct` and `ptacct` files are correlated by number so that if `runacct` fails, the unnecessary reprocessing of `Spacct` files will not occur. One precaution should be noted: *when restarting* `runacct` *in this state, remove the last* `ptacct` *file; if you don't,* `runacct` *will not finish.*
MERGE	Merge the process and connect session total accounting records to form `nite/daytacct`.
FEES	Merge in any ASCII `tacct` records from the file `fee` into `nite/daytacct`.
DISK	On the day after the `dodisk` shell script runs, merge `nite/disktacct` with `nite/daytacct`.
MERGETACCT	Merge `nite/daytacct` with `sum/tacct`, the cumulative total accounting file. Each day, `nite/daytacct` is saved in `sum/tacct`*mmdd*, so that `sum/tacct` can be recreated in the event it becomes corrupted or lost.
CMS	Merge in today's command summary with the cumulative summary file `sum/cms`. Produce ASCII and internal format command summary files.

State	Action
USEREXIT	Any installation-dependent (local) accounting programs can be run in this state. For example, you might want to execute commands that generate daily billing data for individual users (the shell script acct_bill in the section "Sample System Accounting Shell Scripts" could be used for this purpose). To have local accounting programs executed by runacct, simply enter the commands in runacct in the code for the USEREXIT state of runacct.
CLEANUP	Clean up the temporary files, run prdaily and save its output in the file sum/rprt*mmdd*, remove the locks, then exit.

Recovering from Failure

It is possible that runacct might fail and terminate abnormally. The primary reasons for runacct failure are:

- a system "crash"
- not enough disk space remaining in /usr
- a corrupted wtmp file

If the nite/active*mmdd* file exists, check it first for error messages. If the nite/active file and lock files exist, check fd2log for any mysterious messages. The following are error messages produced by runacct and the recommended recovery actions:

ERROR: locks found, run aborted

The files lock and lock1 were found. These files must be removed before runacct can be restarted.

ERROR: acctg already run for *date*: check /usr/adm/acct/nite/lastdate

The date in lastdate and today's date are the same. Remove lastdate before restarting runacct.

ERROR: turnacct switch returned rc=*?*

Check the integrity of turnacct and accton. The accton program must be owned by root and have the setuid bit set.

ERROR: Spacct?.*mmdd* already exists

File setups have probably already run. Check the status of files, then run setups manually.

`ERROR: /usr/adm/acct/nite/wtmp.`*mmdd* `already exists, run setup manually`

You must perform the `SETUP` step manually, because the daily `wtmp` file already exists.


```
ERROR: wtmpfix errors see /usr/adm/acct/nite/wtmperror
```

`wtmpfix` detected a corrupted `wtmp` file. Refer to the section "Fixing Corrupted Files" in this chapter for details on fixing `wtmp` errors.

```
ERROR: connect acctg failed: check /usr/adm/acct/nite/log
```

`acctcon1` encountered a bad `wtmp` file. Again, refer to the section "Fixing Corrupted Files" in this chapter for information on how to fix the file.

```
ERROR: Invalid state, check /usr/adm/acct/nite/active
```

the file `statefile` is probably corrupted. Check `statefile` and read `active` before restarting.

Restarting runacct

`runacct` is normally run via `cron` only once per day. However, if an error occurs while executing `runacct` (as described above), it may be necessary to restart `runacct`. `runacct` has the following syntax:

`runacct` [*state*]

`runacct` assumes that it is being invoked for the first time on the current day. The entry point for processing is based on the contents of `statefile`. To override `statefile`, include the desired entry *state* on the command line.

For example, to start `runacct`, you would enter:

```
nohup runacct 2> /usr/adm/acct/nite/fd2log &
```

To restart `runacct` at state `WTMPFIX`:

```
nohup runacct WTMPFIX 2>/usr/adm/acct/nite/fd2log &
```

All the above examples were run in the background (`&`) and use the `nohup` command so the process continues running even though you may log out.

Daily Reports

`runacct` generates five basic reports upon each invocation. Brief descriptions of each report follow. Detailed descriptions of the reports are found in the following section, "Displaying runacct Reports—prdaily."

- **Daily Line Usage Report**—summarizes connect session accounting since the last invocation of `runacct`. It provides a log of system reboots, power failure recoveries, and any other records dumped into `/etc/wtmp` via `acctwtmp`. In addition, it provides a breakdown of line utilization.
- **Daily Resource Usage Report**—gives a summary of resource usage per individual user: it basically merges all the total accounting records for individual users and displays the records, one per user.
- **Daily Command Summary**—summarizes resource usage data for individual commands since the last invocation of `runacct`. The data included in this report is useful in determining the most heavily used commands; you can use these commands' characteristics of resource utilization when "tuning" your system.

  This report is sorted by `TOTAL KCOREMIN`, an arbitrary but useful yardstick for calculating "drain" on a system.
- **Monthly Total Command Summary**—This report is exactly the same as the Daily Command Summary, except that the Daily Command Summary contains command summary information accumulated only since the last invocation of `runacct`, while the Monthly Total Command Summary summarizes commands from the start of the fiscal period to the current date. In other words, the monthly report reflects the data accumulated since the last invocation of `monacct`.
- **Last Login**—simply gives the date each user last logged in to the system. This could be a good source for finding likely candidates for the archives, or getting rid of unused login directories.

Displaying runacct Reports - prdaily

As `runacct` finishes executing, it deposits a report of the current day's accounting in the file `/usr/adm/acct/sum/rptmmdd`, where *mmdd* is the month and day that the report was generated. The `prdaily` command is used to display the contents of any daily report file created by `runacct`. Its syntax is:

`prdaily` [`-l`][`-c`][*mmdd*]

where:

- *mmdd* is an optional report date. If no date is specified, `prdaily` produces a report of the current day's accounting information. Previous days' accounting reports can be displayed by using the *mmdd* option and specifying the exact report date desired.
- The `-l` option prints a report of exceptional usage by login name for the specified date. This option is used to determine which users are consuming excessive amounts of system resources. The limits for exceptional usage are kept in the file `/usr/lib/acct/ptelus.awk` and can be edited to reflect your installation's requirements.
- Valid only for the current day's accounting, the `-c` option is used to get a report of exceptional resource usage by command. This option is used to determine which commands are using excessive amounts of system resources. The limits for exceptional usage are maintained in the file `/usr/lib/acct/ptecms.awk` and can be edited to reflect your system's needs.

The reports produced by `runacct` were described briefly in the previous sub-section. Now the reports are discussed in more detail.

Daily Line Usage Report

In the first part of this report, the FROM/TO banner should alert you to which period is being reported. The times are the date-time that the last report was generated by runacct, and the date-time that the current report was generated. It is followed by a log of system reboots, shutdowns, power failure recoveries, and any other records dumped into wtmp by the acctwtmp command.

The second part of the report is a breakdown of line utilization. The TOTAL DURATION shows how long the system was in a multiuser state. The columns of the report are defined as follows:

Column	Description
LINE	The terminal line or access port being reported on.
MINUTES	The total number of minutes that the line was in use during the accounting period.
PERCENT	The percentage of TOTAL DURATION that the line was in use: PERCENT = (MINUTES / TOTAL DURATION) * 100
# SESS	The number of times that this port was accessed for a login session.
# ON	Historically, this column displayed the number of times that the port was used to log a user on; but since login can no longer be executed explicitly to log in a new user, this column should be identical to # SESS.
# OFF	This column reflects not only the number of times a user logged off, but also any interrupts that occurred on the line. Interrupts occur on a port when getty is first invoked. A getty is invoked when the system is brought to run-level 2. This column comes into play when # OFF exceeds # ON by a large factor. This usually indicates that the multiplexer, modem, or cable is going bad, or that there is a bad connection somewhere. The most common cause of this is an unconnected cable dangling from the multiplexer.

During real time, wtmp should be monitored as this is the file from which connect session accounting is taken. If it grows rapidly, execute acctcon1 to determine which line is the noisiest. If the interrupting is occurring at a high rate, general system performance will be affected.

Daily Resource Usage Report

This report gives a by-user breakdown of system resource usage. The format of this report is the same as that produced by the `prtacct` command. (For definitions of the data and format of this report, refer to the discussion of `prtacct` in the "Displaying Total Account Records – prtacct" section of this chapter.)

Daily and Monthly Command Summary

These two reports are the same, except that the Daily Command Summary reports information only for commands executed since the last invocation of `runacct`; the Monthly Command Summary contains information on commands executed since the last invocation of `monacct`.

The output of this report is identical to that produced by `acctcms`. For definitions of the data found in this report, refer to the discussion of `acctcms` in the "Process Accounting" section of this chapter.

Last Login

This report simply shows the last date and time that each user logged in. The longer it has been since a particular user logged in, the more likely it is that the user's files could be archived, or maybe even that the user could be removed from the system.

Creating Monthly Accounting Reports - monacct

monacct creates monthly summary files and reports; the resulting output is stored in the directory /usr/adm/acct/fiscal. After creating its monthly reports, it removes the old daily accounting files from the directory /usr/adm/acct/sum and replaces them with new summary accounting files.

monacct should be invoked once each month or accounting period. Its syntax is

monacct *number*

where *number* indicates which month or period it is (01=January, 12=December). If *number* is not specified, monacct assumes that it is being invoked for the current month; this default is useful if monacct is executed via cron on the first day of each month (as described in the "Daily Usage and Installation" section of this chapter).

Descriptions of the files created in the acct/fiscal directory follow:

- cms*?*—contains the total command summary file for the accounting period denoted by *?*. The file is stored in internal summary format. Therefore, to display this file, you must use the acctcms command. The following example shows how to display this file for the month of June:

```
acctcms -a -s /usr/adm/acct/nite/fiscal/cms06
```

- fiscrpt*?*—contains a report similar to that produced by prdaily. The report shows line and resource usage for the month represented by *?*. The following would display the fiscal accounting file for the month of November:

```
cat /usr/adm/acct/nite/fiscal/fiscrpt11
```

- tacct*?*—is the total accounting file for the month represented by *?*. To display this file, you must use the prtacct command. The following would display the total accounting summary file for the month of January:

```
prtacct /usr/adm/acct/fiscal/tacct01 "JANUARY TOTAL ACCOUNTING"
```

Updating the Holidays File

The file `/usr/lib/acct/holidays` contains the information that System Accounting needs to distinguish between prime and non-prime time. It contains the following information:

- **Comment Lines**—Comment lines are entered by placing an asterisk (*) as the first character in the line; they may appear anywhere in the file.
- **Year Designation Line**—This line should be the first non-comment line in the file and must appear only once. The line consists of three four-digit numbers (leading blanks and tabs are ignored). The first number designates the year; the second denotes the time (in 24-hour format) that prime time starts; the third gives the time that prime time ends and non-prime time starts.

  For example, to specify the year as 1985, prime time at 9:00 a.m., and non-prime time at 4:30 p.m., the following entry would be appropriate:

  ```
  1985    0900    1630
  ```

  A special condition allowed for in the time field is that `2400` is automatically converted to `0000`.

- **Company Holiday Lines**—These entries follow the year designation line. Company holidays are days when few people should be using the computer. Therefore, System Accounting assumes that non-prime time is in effect during the entire 24 hours of a specified holiday.

  Company holiday lines have the following format:

  day_of_year Month Day Description of Holiday

  The *day_of_year* field is a number in the range 1 through 366, corresponding to the day of the year for the particular holiday (leading blanks and tabs are ignored). The remaining fields are simply commentary and are not used by other programs.

Note As delivered, the `holidays` file contains valid entries for Hewlett-Packard's prime/non-prime time, and holidays. You should check this file and edit it as necessary to reflect your organization's requirements.

Fixing Corrupted Files

System Accounting files may become corrupted or lost. Some of these files can simply be ignored or restored from the files saved through backup procedures. However, certain files must be fixed in order to maintain the integrity of System Accounting. Two of the files that must be fixed are `/etc/wtmp` and `/usr/adm/acct/sum/tacct`.

Fixing wtmp Errors

The `wtmp` files seem to cause the most problems in the daily operation of System Accounting. When the date is changed and HP-UX is switched into multiuser mode, a set of date change records is written into `/etc/wtmp`. The `wtmpfix` command is designed to adjust the time stamps in the `wtmp` records when a date change is encountered. However, some combinations of date changes and reboots won't be caught by `wtmpfix` and cause `acctcon1` to fail. The following steps show how to "patch" a damaged `wtmp` file.

```
cd /usr/adm/acct/nite
fwtmp <wtmp.mmdd>wtmp.temp
        Using an editor, delete corrupted records or
        delete all records from beginning up to the date change
fwtmp -ic <wtmp.temp >wtmp.mmdd
rm wtmp.temp
```

If the `wtmp` file is beyond repair, create a null `wtmp` file. This will prevent any charging of connect time. `acctprc1` will not be able to determine which login owned a particular process, but it will be charged to the login that is first in the password file for that user ID.

Fixing tacct Errors

If your installation is using System Accounting to charge users for system resource usage, the integrity of `sum/tacct` is quite important. If `sum/tacct` ever becomes corrupted, then check the contents of `sum/tacctprev` with the command `prtacct`. If it looks correct, then the latest `sum/tacct.`*mmdd* should be patched up, and `sum/tacct` should then be recreated. A simple patch procedure would be:

```
cd /usr/adm/acct/sum
acctmerg -a -v <tacct.mmdd>tacct.temp
        Using an editor, remove the bad records and
        write duplicate UID records to another file
acctmerg -i <tacct.temp >tacct.mmdd
acctmerg tacctprev <tacct.mmdd>tacct
rm tacct.temp
```

Remember that `monacct` removes all the `tacct.`*mmdd* files; therefore, `sum/tacct` can be recreated by merging these files together.

Sample Accounting Shell Scripts

grpdusg

This shell script displays disk space usage totals for the users who are members of a specified group. The syntax of this command is:

```
grpdusg group_name
```

where *group_name* is the name of the group for which disk space accounting information is to be generated.

For example,

```
grpdusg pseudo
```

generates disk space usage information for all the users in the group `pseudo`.

The Shell Script

```
# Check for the group-name parameter.
#
if      [ $# -ne 1 ]
then    echo "\nUsage:  grpdusg group-name\n"
        exit 1
fi
echo    "\nOne moment please...\n"
#
# Use the find command to find all the files whose owners are members of
# group-name.  Pipe the output from find into acctdusg which will accumulate
# disk space usage information for the users in group-name.
# NOTE:
#       - accounting data is temporarily stored in _${1}_tmp
#       - error messages are stored temporarily in _${1}_err
#       - if files exist that have no owners, then the names of
#         these files are stored in _no_owners
#
fn=_${1}_
find / -group $1 - hidden -print 2>${fn}err |acctdusg -u _no_owners >${fn}tmp
#
# Remove the _no_owners file if its size is not greater than zero.
#
if      [ -s _no_owners ]
then    echo "\nFiles having no owners exist--check _no_owners\n"
else    rm _no_owners
        echo "\nAll files have owners-- _no_owners not created\n"
fi
#
# Use echo and awk to display disk usage totals for this group.
#
echo "\nDisk space usage information (group is ${1}):\n"
awk 'BEGIN {print "\n_UID____USER
NAME________BLOCKS"}
     { sum += $3 ;              # add up total disk blocks used
       print $0                 # display information for user
     }
     END { print "\nTOTAL DISC SPACE USAGE= ", sum, "blocks" }' ${fn}tmp
#
# Remove temporary files, then exit.
#
rm ${fn}*
```

acct_bill

`acct_bill` takes as input a total accounting file and produces as output billing totals for all users found in the input file. The syntax of `acct_bill` is:

12

`acct_bill` [*mmdd*]

If the optional *mmdd* is not specified, then `acct_bill` takes as input the current day's total accounting file (`acct/nite/daytacct`); if *mmdd* is given, then input is taken from the total accounting file for the date specified by *mmdd* (`acct/sum/tacct`*mmdd*). Output is written to the file `bills`*mmdd*, where *mmdd* is the date given with the command, or the current date if *mmdd* was not specified with the command.

Examples

To generate billing information for the current day, simply enter:

```
acct_bill
```

and the billing information will be stored in the file `acct/sum/bills`*mmdd*, where *mmdd* is the current date.

To create billing information for January 23rd, you would enter:

```
acct_bill 0123
```

after which the billing information would be stored in the file named `acct/sum/bills0123`.

To automatically generate daily billing totals for all users, you should call `acct_bill` without the date argument from the `USEREXIT` state of `runacct`.

Output Produced by acct_bill

The output of `acct_bill` contains one line per user and has the following format:

user_ID *user_name* *billing_amount*

where *user_ID* and *user_name* identify the user who is being billed, and *billing_amount* shows the total amount that the user is to be charged.

billing_amount is computed by multiplying *accounting coefficients* (found in the shell script) by columns of the report generated by `prtacct`. Assuming that billing amounts are in dollars, the coefficients (as they are shown in the shell script that follows) produce the following billing amounts:

- ten cents for every minute of prime CPU time consumed
- five cents for every minute of non-prime CPU time consumed
- a half cent for every prime kcore minute used
- two-tenths of a cent for every non-prime kcore minute
- a half cent for every prime connect time minute
- two-tenths of a cent for every non-prime connect minute
- two-and-a-half cents for every block of disk space used
- two-and-a-half cents for every process spawned by the user
- ten cents for every connect session
- each fee unit charged via `chargefee` counts as one cent

You should experiment with this command by altering the coefficients to see how *billing_amount* is affected. After gaining confidence with this shell script, you can alter the coefficients to suit your installation's needs.

The Shell Script

```
_date=`date +%m%d`
_outfile=/usr/adm/acct/sum/bills
_infile=/usr/adm/acct
#
# Set _infile and _outfile, based on whether or not mmdd was given
#
if      [ $# -eq 0 ]
then                    # Generate billing data for current day.
        _infile=${_infile}/nite/daytacct
        _outfile=${_outfile}${_date}
else                    # Create billing data for date given (mmdd).
        _infile=${_infile}/sum/tacct${1}
        _outfile=${_outfile}${1}
```

```
fi
#
# Create a file containing the ASCII equivalent of the input total
# accounting file (tacct_ASC.tmp_).  The file can then be supplied as input
# to awk, which will generate billing data for each user.
#
acctmerg -a -t <$_infile >tacct_ASC.tmp  # output TOTAL amount first
acctmerg -a <$_infile >>tacct_ASC.tmp # append users' total accounting records
#
# Using awk, compute billing totals for each user in the total accounting file.
#
awk 'BEGIN {
            # ***************************************************
            #    A C C O U N T I N G   C O E F F I C I E N T S
            # ***************************************************
            cpu_P =0.10  # 0.10 monetary units per minute of prime CPU time
            cpu_NP=0.05  # 0.05 monetary units per non-prime CPU minute used
            kcm_P =0.005 # for prime kcore minutes consumed
            kcm_NP=0.002 # for non-prime kcore minutes used
            con_P =0.005 # prime connect (real) time
            con_NP=0.002 # non-prime connect time used
            blk =  0.025 # number of blocks used
            prc =  0.025 # number of processes spawned
            ses =  0.10  # number of connect sessions
            fee =  0.01  # 100 charge units per monetary unit
            # ***************************************************
          }
     # Start computing billing amounts for each user.
     { _sum = cpu_P*$3 + kcm_P*$5 + con_P*$7        # compute prime usage
       _sum+= cpu_NP*$4+ kcm_NP*$6+ con_NP*$8       # add non-prime usage
       _sum+= blk*$9 + prc*$10 + ses*$11 + fee*$13  # add remaining amounts
       printf "%-8s %-10s %10.3f\n", $1, $2, _sum   # display user total
     }' tacct_ASC.tmp_ >$_outfile  # write output from awk to appropriate file
rm tacct_ASC.tmp_                  # remove the temporary ASCII file
```

System Accounting Files

This section contains descriptions of the different files processed by HP-UX System Accounting. The files are grouped according to the directory in which they are found.

Files in the /usr/adm Directory

Filename	Contents
`diskdiag`	Diagnostic output from the execution of disk space accounting commands.
`dtmp`	Output from the `acctdusg` program.
`fee`	Output from the `fchargefee` command (ASCII total accounting records).
`pacct`	The current active process accounting file.
`pacct?`	Process accounting files switched via `turnacct switch`.

Files in the /usr/adm/acct/nite Directory

Filename	Contents
`active`	Used by `runacct` to record progress. It contains warning and error messages. `active`*mmdd* is the same as `active` after `runacct` detects an error.
`ctacct.`*mmdd*	Total accounting records created from connect session accounting where *mmdd* is the month and day the file was created.
`ctmp`	Output of `acctcon1`—connect session records.
`daycms`	ASCII daily command summary used by `prdaily`.
`daytacct`	Total accounting records for current day.
`disktacct`	Total accounting records created by the `dodisk` command.
`fd2log`	Diagnostic output from the execution of `runacct` (refer to `crontab` entry).
`lastdate`	The last day that `runacct` was executed, in `date|#+`*%m%d* format. (Refer to *date(1)* for a description of *+%m%d* date format.)
`lock` and `lock1`	Used to control serial use of `runacct`.
`lineuse`	Terminal (`tty`) line usage report used by `prdaily`.
`log`	Diagnostics output from `acctcon1`.
`log`*mmdd*	Same as `log` after `runacct` detects an error.
`reboots`	Contains beginning and ending dates from `wtmp`, and a listing of reboots.
`statefile`	Used to record the current state being executed by `runacct`.
`tmpwtmp`	`wtmp` file, corrected by `wtmpfix`.
`wtmperror`	Error messages, if any, from `wtmpfix`.
`wtmperror`*mmdd*	Same as `wtmperror` after `runacct` detects an error.
`wtmp.`*mmdd*	The previous day's `wtmp` file.

Files in the /usr/adm/acct/sum Directory

Filename	Contents
`cms`	Total command summary file for current month in internal summary format.
`cmsprev`	Command summary file without latest update.
`daycms`	Command summary file for previous day in internal summary format.
`loginlog`	Shows the last login date for each user.
`rpt`*mmdd*	Daily accounting report for date *mmdd*.
`tacct`	Cumulative total accounting file for current month.
`tacctprev`	Same as `tacct` without latest update.
`tacct`*mmdd*	Total accounting file for date *mmdd*.
`wtmp.`*mmdd*	Saved copy of wtmp file for date *mmdd*. Removed after reboot.

Files in the /usr/adm/acct/fiscal Directory

Filename	Contents
`cms`*?*	Total command summary for month *?* in internal summary format.
`fiscrpt`*?*	Report similar to `prdaily` for the month *?*.
`tacct`*?*	Total accounting file for the month *?*.

A

System Parameters

This appendix describes the tunable system parameters and also shows their Backus Naur Forms. The information in this appendix is Series 800 specific.

If necessary, you can change the value of these tunable parameters with SAM or the **uxgen** command to customize the HP-UX kernel. The **uxgen** command is described in Chapter 2, "Constructing an HP-UX System."

Logical Groups of Parameters

This reference is organized alphabetically. However, many parameters tend to fall into logical groups. Table A-1 illustrates these logical groups.

Table A-1.

Group	Parameters
Accounting	`acctresume`, `acctsuspend`, `timeslice`
File System	`bufpages`, `fs_async`, `maxfiles`, `maxfiles_lim`, `nbuf`, `nfile`, `nflocks`, `ninode`
Hardware	`iomemsize`
Message	`msgmap`, `msgmax`, `msgmnb`, `msgmni`, `msgseg`, `msgssz`, `msgtql`
Miscellaneous	`dst`, `maxusers`, `ncallout`, `npty`, `nstlbe`, `scroll_lines`, `timezone`, `unlockable_mem`
Networking	`netisr_priority`
Obsolete[1]	`BX25BUFS BX25BYTES BX25HLPROT BX25LINKS BX25SBUFS BX25SBYTES BX25SMAX dskless_fsbufs`, `dskless_node`, `maxswapchunks`, `minswapchunks`, `ngcsp`, `num_cnodes`, `selftest_period`, `server_node`, `serving_array_size`, `using_array_size`, `ntext`, `scroll_lines`
Process	`maxdsiz`, `maxssiz`, `maxtsiz`, `maxuprc`, `nproc`
Semaphore	`semaem`, `semmap`, `semmni`, `semmns`, `semmnu`, `semume`, `semvmx`
Shared Memory	`shmmax`, `shmmni`, `shmseg`
Streams[2]	`NSTREVENT NSTRPUSH STRCTLSZ STRMSGSZ`
Swap	`nswapdev`, `nswapfs`, `swchunk`

1 These parameters are no longer used by any supported funcionality within the current release of HP-UX. Do not modify these parameters.

2 Consult the documentation for the optional Streams product to learn more about these parameters

Using SAM to Check or Change HP-UX Kernel Parameters

To check HP-UX kernel parameters with SAM:

1. Log on as `root`.
2. Run SAM:

   ```
   # /usr/bin/sam
   ```

3. Highlight `Kernel Configuration->` and activate [Open].
4. Highlight `Configurable Parameters` and activate [Open].
5. If you are presented with a window entitled "Kernel Configuration," go on to the next step.

   If you are presented with a window entitled "Select New Template File," activate [Template path name...]. A window entitled "Template Files on the System" appears. Choose `/hp-ux` from the list in this window and activate [OK].

6. Look in the upper-left corner of the "Kernel Configuration" window. If you see a line that reads `Template file: /hp-ux`, continue. If not, choose `Select New Template File...` from the "Actions" menu and go to the previous step.

   Examine the list of drivers in the "Kernel Configuration" window. (You may have to scroll through the list to see them all.) When you find the parameter your wish to change, examine its entry in the column labled `Current Value`. If it is *not* the value you need, change it with the next procedure. If it is the value you need, exit the "Kernel Configuration" window, then exit SAM.

To change HP-UX kernel parameters with SAM:

1. If you are not currently logged onto the system, log on as `root`.
2. Run SAM:

   ```
   # /usr/bin/sam
   ```

A

3. Highlight `Kernel Configuration->` and activate (Open).

4. Highlight `Configurable Parameters` and activate (Open).

5. Highlight the parameter you wish to change. (You may have to scroll through the list to find it.)

Note If you wish to change more than one parameter, you may do so by highlighting several of them.

A

6. From the "Actions" menu, choose `Modify Configurable Parameter...`.

7. On the "Modify Configurable Parameter" screen, highlight and turn on one of the methods for parameter modification, then fill in the associated fields and activate (OK).

   You will receive a series of messages indicating the progress of the task. If the task is successful, the entry under the column labeled `Pending Value` will change to the new value you have chosen.

8. From the "Actions" menu, choose `Create a New Kernel`.

Note Creation of a new kernel requires that the system be rebooted.

---

9. You will be presented with a confirmation message. Take one of the following actions:

   - If you want to create a new kernel now, activate (Yes). After the new kernel is built the system will reboot. You will be given an opportunity to take one of three actions:

     - Move the new kernel into place and reboot the system.
     - Move the new kernel into place *without* rebooting the system.
     - Exit without moving the new kernel into place.

     Turn on the radio button for the action you wish to take and activate (OK). If you chose the option to reboot, the system will reboot itself.

   - If you do *not* want to create a new kernel now, activate (No). You may create the new kernel at any time. Exit the "Kernel Configuration" window. You will be given an opportunity to take one of three actions:

     - Create a new kernel.
     - Defer creation of a new kernel.
     - Cancel the kernel modifications you have specified.

     Turn on the radio button for the action you wish to take and activate (OK).

10. Exit SAM by returning to the "System Administration Manager" window and activating (Exit SAM).

A

Parameter Descriptions

Note You can damage your system by changing these parameters improperly. Be sure you know the implications before you change them. Never set system parameters outside the given range. These parameters interact and should be changed in a balanced way.

A

Parameter descriptions include:

Name	The parameter name.
Range	The range of the parameter. Due to interactions with other parameters, sometimes this range cannot be attained.
Default	The default value of the parameter. This value may be specified as a formula which depends on other system parameters. If these other system parameters change, the default value changes correspondingly.
Use	A description of how the system uses the parameter.
Space Utilization	A formula for the allocation of dependent space. Not every parameter description has this field. Usually, dependent space is not an issue for most systems. If you have to calculate dependent space, use the `size(1)` command (refer to the *HP-UX Reference*).
Dependencies	The interaction of the parameter with other system parameters, or how changing it affects system performance, or a formula showing relationships with other parameters.
BNF Format	The Backus Naur Form (BNF) format for the parameter. The BNF formats for the tunable system parameters is:

parameter (integer `or formula`*);*
"anychars";

acctresume

Name `acctresume` - resume accounting due to disk usage

Range -100 -> 101

Default 4

Use

The system disables process accounting when the available space on the file system where the accounting file resides falls below a certain threshold. The threshold is described under acctsuspend. The system re-enables process accounting when sufficient space becomes available. The parameter, acctresume, is the threshold (percentage of free space) that the system must have to re-enable process accounting. This percentage is added to minimum free percentage (minfree) for the file system.

A value of zero re-enables accounting when the free space reaches minfree. A value less than zero allows process accounting to use the space which is reserved for superuser use. A value greater than 100 prevents the the system from re-enabling process accounting when space is available.

When accounting is re-enabled in this way, the system issues this message:

```
Accounting resumed
```

Parameter acctresume is relevant only to systems that use process accounting.

Dependencies (interactions with other system values)

acctsuspend < acctresume

BNF Format

```
acctresume 4;

BNF:

    <acctresume-stmt>  ::= acctresume <integer>;
                        |  acctresume "<anychars>";
```

acctsuspend

Name acctsuspend - suspend accounting due to disk usage

Range -100 -> 100

Default 2

Use

The system disables process accounting when the available space on the file system where the accounting file resides falls below a certain threshold. The parameter, acctsuspend (specified as a percentage of free space), is the threshold. This percentage is added to minimum free percentage (minfree) for the file system.

A value of zero disables process accounting when the free space falls below minfree. A value less than zero allows process accounting to use the space which is reserved for superuser use. If the sum of acctsuspend and minfree is less than zero, process accounting will remain enabled.

When accounting is disabled in this way, the system will issue this message:

```
Accounting suspended
```

Parameter acctsuspend is only relevant to systems that use process accounting.

Dependencies (interactions with other system values)

acctsuspend < acctresume

BNF Format

```
acctsuspend 2;

 BNF:

     <acctsuspend-stmt> ::= acctsuspend <integer>;
                         |  acctsuspend "<anychars>";
```

bufpages

Name `bufpages` - number of buffer pages

Range 0, 64 -> memory limited

Default 0 (configured dynamically)

If bufpages is equal to zero when the system boots, then two pages are allocated for every buffer header defined by nbuf. If nbuf is also zero, then 10 percent of available memory is used.

Use

Parameter bufpages defines the number of pages in the file system buffer cache. Each page is allocated 2048 bytes of memory.

These buffers are used for all file system I/O operations, plus all other block I/O operations in the system (exec, mount, inode reading and some device drivers.)

Note If you set nbuf to a number less than 64 or greater than the maximum supported by the system, the number will be increased or decreased as appropriate, and a message is printed at boot time.

Dependencies (interactions with other system values)

This variable may override the value specified for nbuf.

The maximum memory allocated to the buffer pool will be limited based on the memory allocated to the system for other purposes. Thus modifying parameters that affect system memory may affect the maximum memory allocatable to the buffer pool.

BNF Format

A

```
bufpages 0;

BNF:

<bufpages-stmt>  ::= bufpages <integer>;
                  |  bufpages "<anychars>";
```

dst

Name dst - daylight savings time

Range 0 -> 5

Default 1

Use

This parameter specifies whether to convert to daylight savings time.

These definitions of the parameter values are from the file **/usr/include/sys/time.h:**

```
#define DST_NONE        0        /* not on dst */
#define DST_USA         1        /* USA style dst */
#define DST_AUST        2        /* Australian style dst */
#define DST_WET         3        /* Western European dst */
#define DST_MET         4        /* Middle European dst */
#define DST_EET         5        /* Eastern European dst */
```

Dependencies (interactions with other system values)

It is used with time zone.

BNF Format

```
dst 0;
dst 1;

BNF:

    <dst-stmt> ::= dst <integer>;
```

A

fs_async

Name `fs_async` - select asynchronous writes to disk

Range 0, 1

Default 0 (disk writes are synchronous)

Use

A non-zero value specifies asynchronous writes to disk. If no value is specified for **fs_async**, writes of the file system data structures to disk are performed synchronously.

Synchronous writes to disk help insure system integrity.

Asynchronous writes to disk can improve file system performance significantly. However, with asynchronous writes, file system data structures might be left in an inconsistent state in the event of a system crash; system administrator intervention to run `fsck` might be required.

If asynchronous writes are selected, HP-UX file system semantics for NFS and HP-UX cluster environments are preserved. In addition, files opened with the `O_SYNC` flag (synchronous writing) will continue to be written synchronously when the asynchronous writes feature has been configured into the kernel.

Dependencies (interactions with other system values)

None.

BNF Format

```
fs_async 0;
fs_async 1;

BNF:

     <fs_async-stmt> ::= fs_async <integer>;
```

iomemsize

Name iomemsize - amount of memory

Range None.

Default 10 * NBPG

Use

This value is entered in bytes. Parameter iomemsize defines the maximum number of bytes that are available to the I/O system.

The default value should provide enough memory for most systems. If iomemsize is too small, the system will panic and issue this message:

```
panic: io_get_pages: out of memory
```

Dependencies (interactions with other system values)

None.

BNF Format

```
iomemsize "(10*NBPG)";

BNF:

    <iomemsize-stmt>  ::= iomemsize <integer>;
                       |  iomemsize "<anychars>";
```

A

maxdsiz

Name `maxdsiz` - maximum data size

Range 0x400000 -> 0x27FFFFFF

Default 0x4000000

Use

This value is entered in pages (2048 bytes per page).

A

Parameter `maxdsiz` defines the maximum size of the data segment of an executing process.

The default value is large enough for most processes. The `maxdsiz` parameter should be increased only if you have one or more processes that use large amounts of data.

Each time the system loads a process, or an executing process attempts to expand its data segment, the system checks the size of the process's data segment.

If `maxdsiz` is exceeded, the system will terminate the process or issue an error message.

Dependencies (interactions with other system values)

BNF Format

```
maxdsiz 0x8000;

BNF:

    <maxdsize-stmt>  ::= maxdsize <integer>;
                      |  maxdsize "<anychars>";
```

maxfiles

Name `maxfiles` - soft limit to the number of files a process can have open

Range 30 - 2048

Default 60

Use

The parameter represents the system default soft limit to the number of open files a process may have. It is possible for a process to increase its soft limit and therefore open more than `maxfiles` files.

Non-superuser processes can increase their soft limit until their hard limit (`maxfiles_lim`) is reached.

Dependencies (interactions with other system values)

`maxfiles` is limited by `nfile` and the inode table.

BNF Format

```
maxfiles  1024;

BNF:

<maxfiles-stmt>  ::= maxfiles <integer>;
                 |  maxfiles "<anychars>";
```

maxfiles_lim

Name `maxfiles_lim` - hard limit for the number of files a process can have open

Range 30 - 2048

Default 1024

Use

A

The parameter represents the system default hard limit to the number of open files a process may have. It is possible for a non-superuser process to increase its soft limit (`maxfiles`) up to the hard limit.

Dependencies (interactions with other system values)

`maxfiles_lim` is limited by `nfile`, `ninode`.

BNF Format

```
maxfiles_lim  2048;

BNF:

<maxfiles_lim-stmt>  ::= maxfiles <integer>;
                     |  maxfiles_lim "<anychars>";
```

maxssiz

Name `maxssiz` - maximum stack size

Range 0x4000 -> 0x27FFFFFF

Default 0x800000

Use

This value is entered in bytes. Parameter `maxsize` defines the maximum size of the stack segment of an executing process.

The default is large enough for the stack of most processes. Parameter `maxssiz` should only be increased if you have one or more processes that need a large stack.

The stack grows dynamically. As it grows, the system checks the size of the process's stack segment. If `maxssiz` is exceeded, the process is killed.

Dependencies (interactions with other system values)

BNF Format

```
maxssiz 0x1000;

BNF:

    <maxssize-stmt>  ::= maxssize <integer>;
                      |  maxssize "<anychars>";
```

maxtsiz

Name `maxtsiz` - maximum text size

Range 0x40000 -> 0x27FFFFFF

Default 0x4000000

Use

This value is entered in bytes. Parameter `maxtsiz` defines the maximum size of the shared text segment of an executing process.

The current default accommodates the text segments of most processes. Unless you plan to execute a process with a text segment larger than 64 Mb, `maxtsiz` should not be modified.

Dependencies (interactions with other system values)

Each time the system loads a process with shared text, the system checks the size of its shared text segment. The system issues an error message and aborts the process if the text segment is larger than `maxtsiz`.

BNF Format

```
maxtsiz 0x8000;

BNF:

    <maxtsize-stmt>  ::= maxtsize <integer>;
                      |  maxtsize "<anychars>";
```

maxuprc

Name maxuprc - maximum number of user processes

Range 3 -> nproc - 4

Default 25 processes

Use

Parameter maxuprc defines (for each user) the maximum number of simultaneous processes. A user is identified by the user ID number, not by the number of login instances. Each user will need at least one process for the shell, and other processes for work purposes (the default is usually adequate).

The superuser is exempt from this limit.

Pipelines need at least one simultaneous process for each side of a | (pipe). Some commands, such as cc, fc, and pc, use more than one process per invocation.

If a user tries to start a new process when the total number of processes for the user is larger than maxuprc, the system will issue this message to the user:

```
no more processes
```

If a user is doing a *fork*(2) system call to create a new process and the total number of processes for the user exceeds maxuprc, *fork*(2) will return -1 and set the errno to EAGAIN.

Dependencies (interactions with other system values)

If maxuprc is set to a value greater than or equal to nproc (maximum number of processes in the system) then maxuprc is no longer a limit, a single user could monopolize system resources.

BNF Format

```
maxuprc 50;

BNF:

    <maxuprc-stmt>  ::= maxuprc <integer>;
                     |  maxuprc "<anychars>";
```

maxusers

Name `maxusers` - limiter for system resource allocation

Range 0 -> memory-limited

Default 32

Use

Parameter `maxusers` limits system resource allocation, not the actual number of users on the system. By itself, `maxusers` does not determine the size of any structures in the system. The default value of other global system parameters depend on `maxusers`. If you tune the parameters that use `maxusers`, then the effect of `maxusers` on kernel size is proportionately smaller.

The `maxusers` parameter defines the macro MAXUSERS (for example, `#define MAXUSERS 8`). It determines the size of system tables. The actual limit of the number of users depends on the license version of HP-UX purchased. You can examine the license version using the *uname*(util) command.

Rather than varying each configurable parameter individually, it is easier to specify certain parameters using a formula based on the maximum number of expected users (for example, `nproc` "20 + 8 * MAXUSERS;"). Thus, if you increase the maximum number of users on your system, you only need to change the maxusers statement.

This statement must appear before any other statement that uses the MAXUSER macro. For example, nproc "20 + 8 * MAXUSERS;" must follow "maxusers 8;".

Dependencies (interactions with other system values)

The default values of `nproc`, `ncallout`, `ninode`, and `nfile` depend on `maxusers`.

BNF Format

```
maxusers 8;
maxusers 32;

BNF:

    <maxusers> ::= maxusers <integer>;
```

A

msgmap

Name `msgmap` - message map

Range 3 -> memory limited

Default `msgtql` + 2

Use

Each set of messages allocated per identifier occupies one or more contiguous slots in the `msg` array. As messages are allocated and deallocated the msg array may become fragmented.

Parameter `msgmap` dimensions the resource map used to allocate the buffer space for messages. This map shows the free holes in the `msg` array. An entry in the map is used to point to each set of contiguous unallocated slots, and it consists of a pointer to the set, plus the size of the set.

If a message set request cannot be accommodated, the system issues the message:

```
DANGER: mfree map overflow
```

If you get this error message, regenerate the kernel with a larger value for `msgmap`.

There is less fragmentation of the msg array if all message identifiers have the same number of messages. Parameter **msgmap** can then be smaller.

Dependencies (interactions with other system values)

msgmap <= (msgtql + 2)

msgmap <= (msgseg + 2)

If **msgmap** is greater than msgtql + 2, then part of the space allocated for **msgmap** will not be used.

BNF Format

```
msgmap 100;

BNF:

    <msgmap-stmt>  ::= msgmap <integer>;
                    |  msgmap "<anychars>";
```

A

msgmax

Name msgmax - message maximum size

Range 0 -> 65536

Default 32768 bytes

Use

Parameter msgmax limits the size, in bytes, of a single message.

A

Increase the value of msgmax only if you plan to execute applications that require larger messages. This parameter keeps malicious or poorly written programs from using all of the message buffer space.

A *msgsnd*(2) system call which attempts to send a message larger than msgmax bytes returns an error.

Dependencies (interactions with other system values)

msgmax <= msgmnb

BNF Format

```
msgmax 8192;

BNF:

    <msgmax-stmt>  ::= msgmax <integer>;
                    |  msgmax "<anychars>";
```

msgmnb

Name `msgmnb` - maximum number of bytes on the message queue

Range 0 -> 65536 bytes

Default 32768 bytes

Use

Parameter `msgmnb` is the maximum total size, in bytes, of all messages that can be queued on a message queue at the same time.

A *msgsnd*(2) system call which attempts to exceed this limit returns either:

- an EAGAIN error if IPC_NOWAIT is set.
- an EINTR error if IPC_NOWAIT is not set.

Dependencies (interactions with other system values)

msgmnb >= msgmax

BNF Format

```
msgmnb 16384;

BNF:

    <msgmnb-stmt>  ::= msgmnb <integer>;
                     |  msgmnb "<anychars>";
```

A

msgmni

Name `msgmni` - number of message queue identifiers

Range 1 -> memory limited

Default 50

Use

Parameter `msgmni` dimensions the array of message queue identifiers.

A message queue identifier is needed for each message queue in the system.

An attempt to allocate a new message queue with the *msgget*(2) system call when `msgmni` message queues already exist returns a ENOSPC error.

If a message queue remains allocated, it is on the system even after the process(es) using it have stopped. Deallocate message queues using *ipcrm*(1).

Users should deallocate messages when processes are complete.

Dependencies (interactions with other system values) None.

BNF Format

```
msgmni 50;

BNF:

    <msgmni-stmt>  ::= msgmni <integer>;
                    |  msgmni "<anychars>";
```

msgseg

Name msgseg - message segments

Range 1 -> 32768 bytes

Default 7168bufpages bytes

Use

Parameter **msgseg**, together with **msgssz**, determines the size of the buffer available for queuing messages.

Parameter **msgssz** determines the size, in bytes, of the units in which messages are allocated space. When a message is allocated, its size is rounded o the nearest multiple of **msgssz**.

Parameter **msgseg** is the number of these units available.

In most cases the product of **msgseg** * **msgssz** is of most interest since it determines the total amount of space available for messages. Different **msgseg:msgssz** ratios which yield the same product will just cause this space to be fragmented differently for the same usage.

Space Utilization msgseg * msgssz bytes

Dependencies (interactions with other system values)

(msgseg * msgssz) >= msgmax

BNF Format

```
msgseg 1024;

BNF:

    <msgseg-stmt>  ::= msgseg <integer>;
                    |  msgseg "<anychars>";
```

msgssz

Name msgssz - message segment size

Range 1 -> memory limited

Default 8 bytes

Use

Parameter **msgssz**, together with **msgseg**, determines the size of the buffer available for queuing messages.

A

Parameter **msgssz** determines the size, in bytes, of the units in which messages are allocated space. When a message is allocated, its size is rounded up to the nearest multiple of **msgssz**.

Parameter **msgseg** is the number of these units available.

In most cases, the product of **msgseg** * **msgssz** is of most interest since it determines the total amount of space available for messages. Different **msgseg**:**msgssz** ratios which yield the same product will just cause this space to be fragmented differently for the same usage.

Space Utilization msgseg * msgssz bytes

Dependencies (interactions with other system values)

(msgseg * msgssz) >= msgmax

BNF Format

```
msgssz 8;

BNF:

    <msgssz-stmt>  ::= msgssz <integer>;
                    |  msgssz "<anychars>";
```

msgtql

Name `msgtql` - number of message headers

Range 1 -> memory limited

Default 256

Use

Parameter `msgtql` dimensions an array of message headers. A message header is used for each message queued in the system.

A *msgsnd*(2) system call which attempts to exceed this limit either:

- blocks waiting for a free header or,
- returns EAGAIN error

depending on whether the IPC_NOWAIT flag is set with the call.

Dependencies (interactions with other system values)

msgmap <= msgtql + 2

If `msgmap` is greater than msgtql + 2, then some allocated space will be wasted.

BNF Format

```
msgtql 40;

BNF:

    <msgtql-stmt>  ::= msgtql <integer>;
                    |  msgtql "<anychars>";
```

A

nbuf

Name nbuf - number of buffer headers

Range 0, 16 -> memory limited

Default 0 (configured dynamically)

If at boot time nbuf is = 0, then one buffer header is allocated for every two pages of buffer memory defined by the bufpages parameter. If bufpages is also zero, 10% of available memory is used.

A

Use

Parameter nbuf defines the number of file system buffer-cache buffer headers. Each buffer is allocated 4096 bytes of memory unless overridden by a conflicting value for bufpages.

These buffers are used for all file system I/O operations, plus all other block I/O operations in the system (such as exec, mount, inode reading, and some device drivers).

While nbuf is available for compatibility with previous releases, it is recommended that the size of the buffer pool be configured with the bufpages parameter.

Note If you set nbuf to a number less than 16, greater than the maximum supported by the system, or to a value that is inconsistent with the value of bufpages, the number will be increased or decreased as appropriate, and a message printed at boot time.

Dependencies (interactions with other system values)

bufpages controls the actual memory allocated to the buffer pool. If both bufpages and nbuf are set and the values conflict so that it is impossible to configure a system using both of them, bufpages overrides.

BNF Format

```
nbuf 0;

BNF:

    <nbuf-stmt>  ::= nbuf <integer>;
                  |  nbuf "<anychars>";
```

A

ncallout

Name `ncallout` - number of timeouts

Range 6 -> memory limited

Default (64+NPROC)

Use

Parameter `ncallout` is the maximum number of timeouts that can be scheduled by the kernel at any one time. Timeouts are used by:

- alarm (system call)
- setitimer (system call)
- select (system call)
- Drivers
- uucp processes
- Process scheduling

When the system runs out of timeouts, it prints the following fatal error to the

```
panic: timeout table overflow
```

Dependencies (interactions with other system values)

The larger `nproc` is, the larger `ncallout` should be. A guideline of 1 callout per process should be used unless you have processes that use many of the callouts.

BNF Format

```
ncallout "(64 + NPROC)";

BNF:

    <ncallout-stmt>  ::= ncallout <integer>;
                      |  ncallout "<anychars>";
```

netisr_priority

Name `netisr_priority` - a realtime process priority for networking

Range -1, 0 -> 127

Default -1

Use

The `netisr_priority` parameter specifies the realtime process priority of the netisr process. The `netisr` daemon executes on systems with networking—ARPA/BSD, X.25/9000, and HP-UX clusters—processing the packets for these networking services.

A value of -1 specifies that networking packets are handled on an interrupt basis. This yields the fastest possible packet handling rate.

A value between 0 and 127 (zero being highest priority) specifies the priority at which the process scheduler handles networking packets.

Refer to *rtprio*(1) in *HP-UX Reference Manual*; also refer to the *HP-UX Real-Time Programming Manual* for more information.

Dependencies (interactions with other system values).

None.

BNF Format

```
netisr_priority 0;

BNF:

    <netisr_priority-stmt>  ::= netisr_priority  <integer>;
```

nfile

Name `nfile` - number of files

Range 14 -> memory-limited

Default (16 * (NPROC + 16 + MAXUSERS)/10 + 32 + 2 * NPTY)

Use

Parameter `nfile` defines the maximum number of open files at any one time in the system.

A

It is the number of slots in the file descriptor table. Be generous with this number since the cost is low, and not having enough slots would cut down on the amount of work that can be done simultaneously in the system.

Dependencies (interactions with other system values)

Processes At least three file descriptors per process (stdin, stdout, stderr).

Pipes 2 per pipe (1 per side).

BNF Format

```
nfile "(16 * (NPROC + 16 + MAXUSERS)/10 + 32 + 2 * NPTY)";

BNF:

    <nfile-stmt>  ::= nfile <integer>;
                   |  nfile "<anychars>";
```

nflocks

Name `nflocks` - number of file locks

Range 2 -> memory-limited

Default 200

Use

Parameter `nflocks` gives the possible number of file/record locks in the system. When choosing this number, note that one file may have several locks and databases may need an exceptionally large number of locks (if they use `lockf`).

Dependencies None.

BNF Format

```
nflocks 200;

BNF:

    <nflocks-stmt>  ::= nflocks <integer>;
                     |  nflocks "<anychars>";
```

ninode

Name `ninode` - number of inodes

Range 14 -> memory-limited

Default (NPROC + 16 + MAXUSERS) + 32

Use

Parameter `ninode` defines the maximum number of open inodes which can be in-core.

A

It is the number of slots in the inode table. The inode table is used as a cache memory. For efficiency reasons, the last ninode (number of) open inodes is kept in main memory. The table is hashed.

Dependencies (interactions with other system values)

Each unique open file has an open inode associated with it. Therefore, the larger the number of unique open files, the larger `ninode` should be.

The default value of `ninode` depends on `nproc` and `maxusers`.

BNF Format

```
ninode "(NPROC + 16 + MAXUSERS + 32)";

BNF:

    <ninode-stmt>  ::= ninode <integer>;
                    |  ninode "<anychars>";
```

nproc

Name nproc - number of processes

Range 10 -> memory limited

Default (20+8 * MAXUSERS)

Use

Parameter nproc specifies the maximum total number of processes that can exist simultaneously in the system.

There are at least four system overhead processes at all times, and one entry is always reserved for the superuser.

When the total number of processes in the system is larger than nproc, the system issues these messages:

At the system console:

```
proc: table is full
```

Also, if a user tries to start a new process from a shell, the following message prints on their terminal:

```
no more processes
```

If a user is executing *fork*(2) to create a new process, *fork*(2) will return -1 and set the errno to EAGAIN.

Dependencies (interactions with other system values)

The default values of ninode, nfile and ncallout depend on nproc.

maxuprc <= (nproc - 4)

BNF Format

```
nproc "(20 + 86 * MAXUSERS)";

BNF:

    <nproc-stmt>  ::= nproc <integer>;
                   |  nproc "<anychars>";
```

npty

Name npty - number of pseudo-teletypes

Range 1 -> memory limited

Default 60

Use

Parameter **npty** limits the number of the following structures that can be used by the pseudo-teletype driver:

A

struct tty	pt_tty[npty];
struct tty	*pt_line[npty];
struct pty_info	pty_info[npty];

Dependencies (interactions with other system values) None.

BNF Format

```
npty 60;

BNF:

    <npty-stmt>  ::= npty <integer>;
                  |  npty "<anychars>";
```

nstlbe

Name `nstlbe` - number of software TLB entries requested

Range 256 (MINSTLBE) -> memory limited

Default 0

Use

The overhead of a Translation Lookaside Buffer (TLB) miss may adversely affect the performance of Precision Architecture implementations that use small and/or single-set TLBs. The software TLB is an intermediate data structure that provides buffering between the hardware TLB and the software page directory. The two-level mechanism that uses the Software TLB can decrease the overall TLB miss ratio and can also significantly reduce the number of cycles necessary to resolve a TLB miss. However, the software TLB is no substitute for a larger hardware TLB. The software TLB is two-way set associative and equally split, in its organization, between instruction translations and data translations.

Parameter `nstlbe` enables or disables use of the two-level software TLB mechanism by specifying the number of software TLB entries requested. A zero value disables this software TLB entirely. A nonzero value specifies the number of software TLB entries requested. Each entry contains an address translation, the associated protection and a tag, and occupies 16 bytes of physical memory. The system adjusts any nonzero `nstlbe` request to meet the following constraints:

- The minimum number of software TLB entries (MINSTLBE) that will be allocated is 256.
- At least twice as many software TLB entries as there are physical hardware TLB entries, including set associativity, will be allocated to make the software TLB effective.
- The number of software TLB entries allocated must be a power of two.

The actual number of software TLB entries allocated is displayed on the console with the kernel configuration messages. For example,

```
using 1024 software TLB entries occupying 16384 bytes of memory
```

The software TLB is always disabled on multiprocessor configurations because broadcast purge of the software TLB is not implemented.

Dependencies (interactions with other system values)

None.

BNF Format

```
nstlbe 1024;

BNF:

    <nstlbe-stmt>  ::= nstlbe <integer>;
                    |  nstlbe "<anychars>";
```

A

nswapdev

Name **nswapdev** - number of file systems that you can enable for dynamic swap

Range 1-25

Default 10

Use

Parameter **nswapdev** defines the maximum number of devices that you can use for device swap.

Dependencies (interactions with other system values)

None.

BNF Format

```
nswapdev;

BNF:

    <nswapdev-stmt>  ::= nswapdev <integer>;
                     |  nswapdev "<anychars>";
```

A

nswapfs

Name nswapfs - number of file systems that you can enable for dynamic swap

Range 0-25

Default 10

Use

A

Parameter nswapfs defines the maximum number of file systems that you can use for dynamic swap.

Dependencies (interactions with other system values)

None.

BNF Format

```
nswapfs;

BNF:

    <nswapfs-stmt>  ::= nswapfs <integer>;
                     |  nswapfs "<anychars>";
```

semaem

Name semaem - "adjust on exit" maximum value for semaphores

Range 0 -> min (semvmx, 32767)

Default 16384

Use

An undo is an optional flag in a semaphore operation which causes that operation to be undone if the process which invoked it dies.

Parameter semaem is the maximum value by which a semaphore can be undone.

This value is cumulative per process, so if one process has more than one undo operation on a semaphore, the value of each undo operation is added up in the variable *semadj*. Parameter semadj is the number by which the semaphore will be incremented or decremented if the process dies.

Read the manual page for *semop*(2) for more detailed information on semaphore undos.

Any *semop*(2) calls which attempts to set |semadj | > semaem result in an ERANGE error.

Dependencies (interactions with other system values)

semaem <= semvmx

BNF Format

```
semaem 16384;

BNF:

    <semaem-stmt>  ::= semaem <integer>;
                    |  semaem "<anychars>";
```

semmap

Name `semmap` - semaphore map

Range 4 -> memory limited

Default (`semmni` + 2)

Use

A

Each set of semaphores allocated per identifier occupies 1 or more contiguous slots in the `sem` array. As semaphores are allocated and deallocated the `sem` array may become fragmented.

Parameter `semmap` dimensions the resource map which shows the free holes in the sem array. An entry in this map is used to point to each set of contiguous unallocated slots, and it consists of a pointer to the set, plus the size of the set.

If semaphore usage is heavy and a request for a semaphore set cannot be accommodated, the following message appears:

```
danger: mfree map overflow
```

You should then configure a new kernel with a larger `semmap`.

Fragmentation of the `sem` array is reduced if in usage of the system all semaphore identifiers have the same number of semaphores; `semmap` can then be somewhat smaller.

Four is the lower limit: 1 slot is overhead for the map and the second slot is always needed at system initialization to show that the `sem` array is free.

Dependencies (interactions with other system values)

(semmap-2) = maximum # of contiguous unallocated pieces of the **sem** array.
semmap<=(semmni+2)

If **semmap** is greater, then some allocated space will be wasted.

BNF Format

```
semmap 10;

BNF:

    <semmap-stmt>  ::= semmap <integer>;
                    |  semmap "<anychars>";
```

A

semmni

Name `semmni` - number of semaphore identifiers

Range 2 -> `semns`

Default 64

Use

Parameter `semmni` defines the number of sets (identifiers) of semaphores available to the users.

A

When the system runs out of semaphore sets, the *semget*(2) system call will return a ENOSPC error message.

Dependencies (interactions with other system values)

semmni <= semmns
semmns <= (semmni * semmsl)
semmap <= (semmni+2)

`semmsl` is the value of the maximum number of semaphores that can be associated with a semaphore ID. The value of `semmsl` is set at 500 and is *not* tunable.

BNF Format

```
semmni 64;

BNF:

    <semmni-stmt>  ::= semmni <integer>;
                    |  semmni "<anychars>";
```

semmns

Name semmns - total number of semaphores in system

Range 2 -> memory limited

Default 128

Use

Parameter semmns defines the total number of semaphores available to the users of the system.

When the system does not have enough contiguous semaphores in the sem array to satisfy a semget request, the call returns a ENOSPC error. This error may occur even though there may be enough free semaphores, but they are not contiguous.

Dependencies (interactions with other system values)

semmni <= semmns
semmns <= (semmni * semmsl)

semmsl is the value of the maximum number of semaphores that can be associated with a semaphore ID. The value of semmsl is set at 500 and is *not* tunable.

BNF Format

```
semmns 128;

BNF:

    <semmns-stmt>  ::= semmns <integer>;
                    |  semmns "<anychars>";
```

semmnu

Name semmnu - number of semaphore **undo** structures

Range 1 -> nproc - 4

Default 30

Use

An undo is a special, optional, flag in a semaphore operation which causes that operation to be undone if the process which invoked it terminates.

A

Parameter semmnu is the number of processes which can have undo's pending on a given semaphore. It determines the size of the sem_undo structure.

Refer to *semop*(2) in the *HP-UX Reference Manual* for more information.

You should increase semume if the user gets ENOSPC errors on *semop*(2) calls using the SEM_UNDO flag.

Dependencies (interactions with other system values)

- semmnu determines the size of the structure sem_undo, which in turn contains the substructure dimensioned by semume.
- There is no point in having semmnu = (nproc -4) since it is the largest number of processes in the system which could use semaphores simultaneously.

BNF Format

```
semmnu 30;

BNF:

    <semmnu-stmt>  ::= semmnu <integer>;
                    |  semmnu "<anychars>";
```

semume

Name semume - semaphore **undo** entries per process

Range 1 -> semmns

Default 10

Use

An undo is an optional flag in a semaphore operation which cause that operation to be undone if the process which invoked it dies.

Parameter semume limits the number of semaphores that each process can have undos pending on.

Read the manual page for *semop*(2) for a more detailed explanation of undo.

semop is the value of the maximum number of semaphores you can change with one system call. Check the file /usr/include/sys/sem.h for this value.

When the user gets EINVAL errors on *semop*(2) calls with the SEM_UNDO flag, then you should increase the value of semume.

Dependencies (interactions with other system values)

semume <= semmns

Parameter semume is the size of the substructure undo, which is part of the sem_undo structure. The size of sem_undo is determined by semume.

BNF Format

```
semume 10;

BNF:

    <semume-stmt>  ::= semume <integer>;
                    |  semume "<anychars>";
```

semvmx

Name semvmx - semaphore maximum value

Range 1 -> 65536

Default 32768

Use

Parameter semvmx is the maximum value that a semaphore is allowed to reach. This limit is imposed by the largest number that can be stored in a 16-bit unsigned integer (65536).

A *semop*(2) system call which tries to increment a semaphore value greater than semvmx will result in an ERANGE error. If semvmx > 65536, then semaphore values can overflow and these errors will not be caught.

semop is the value of the maximum number of semaphores you can change with one system call. Check the file /usr/include/sys/sem.h for this value.

Dependencies (interactions with other system values)

semaem <= semvmx

BNF Format

```
semvmx 32767;

BNF:

    <semvmx-stmt>  ::= semvmx <integer>;
                    |  semvmx "<anychars>";
```

shmmax

Name shmmax - shared memory maximum

Range 2 Kbytes -> 767 Mbytes

This is a system-wide limit.

Default 0x04000000 (64 Mbytes)

Use

Parameter shmmax defines the maximum shared memory segment size in bytes.

Dependencies (interactions with other system values)

None.

BNF Format

```
shmmax 0x4000000;

BNF:

    <shmmax-stmt>  ::= shmmax <integer>;
                    |  shmmax "<anychars>";
```

A

shmmni

Name shmmni - shared memory maximum number of identifiers

Range 3 -> 1024

Default 100 identifiers

Use

Parameter shmmni defines the maximum number of shared memory segments systemwide. Make it large enough to hold as many shared memory segments as will be used simultaneously.

Dependencies (interactions with other system values) None.

BNF Format

```
shmmni 100;

BNF:

    <shmmni-stmt>  ::= shmmni <integer>;
                    |  shmmni "<anychars>";
```

shmseg

Name shmseg - shared memory segments

Range 1 -> shmmni

Default 12

Use

Parameter shmseg defines the maximum number of shared memory segments that can be attached to a process at any given time.

Dependencies (interactions with other system values) None.

BNF Format

```
shmseg 12;

BNF:

    <shmseg-stmt>  ::= shmseg <integer>;
                    |  shmseg "<anychars>";
```

A

swchunk

Name swchunk - chunk size for swap

Range 1 -> 0x4000

Default 0x800

Use

This parameter defines the chunk size for swap. This value must be a power of two. This parameter cannot be configured with SAM.

A

Dependencies (interactions with other system values)

Total swap space is limited by:

```
swchunk * maxswapchunk * DEV_BSIZE
```

where DEV_BSIZE is 1024 bytes.

Total swap space cannot exceed 2 gigabytes.

BNF Format

```
swchunk 2048;
swchunk 4096;

BNF:

<swchunk-stmt>  ::= swchunk<integer>;
        |  swchunk "<anychars>";
```

timeslice

Name `timeslice` - scheduling timeslice interval

Range -1 -> 2147483647

Default HZ/10

Use

The timeslice interval is the amount of time one process is allowed to run before the CPU is given to the next process at the same priority.

The value of `timeslice` is specified in units of (10 millisecond) clock ticks. There are two special values:

0 Use the system default value (currently 10 ticks, or 100 milliseconds)

-1 Disable round-robin scheduling completely

Impact on System

This parameter will cause a process to check for pending signals when its timeslice expires. This guarantees that a process which does not make any system calls (including a runaway process in an infinite loop) can be terminated. Thus setting timeslice to a very large value, or to -1, can prevent such processes from getting signals.

Change this parameter only on systems dedicated to applications with specific realtime needs.

No memory allocation relates to this parameter. Some CPU time is spent at each timeslice interval, but this time has not been precisely measured.

Dependencies (interactions with other system values) None.

BNF Format

```
timeslice  "HZ/10";

BNF:

    <timeslice-stmt>  ::= timeslice <integer>;
                       |  timeslice "<anychars>";
```

timezone

Name timezone - minutes west of the Greenwich meridian

Range 0 -> 1440

Default 420

Use

The timezone parameter indicates the minutes west of the Greenwich meridian:

A

```
struct  timezone tz = { TIMEZONE, DST };
struct timezone {
int tz_minuteswest; /* minutes west of Greenwich */
int tz_dsttime;     /* type of dst correction */
};
#define DST_NONE      0     /* not on dst */
#define DST_USA       1     /* USA style dst */
#define DST_AUST      2     /* Australian style dst */
#define DST_WET       3     /* Western European dst */
#define DST_MET       4     /* Middle European dst */
#define DST_EET       5     /* Eastern European dst */
```

Dependencies (interactions with other system values)

It is used with dst (daylight savings time). This should be made consistent with the TZ environment variable (see *environ*(MISC) and *login*(1) in the *HP-UX Reference Manual*).

BNF Format

```
timezone 480;

BNF:

    <timezone-stmt>  ::= timezone <integer>;
```

unlockable_mem

Name `unlockable_mem` - unlockable memory

Range 0 -> (the available memory indicated at power-up)

Default 0

Use

The parameter `unlockable_mem` defines the minimum amount of memory that will always be available for virtual memory and/or system overhead.

It limits the amount of memory that can be locked (lockable memory) to `unlockable_mem` (the available memory indicated at power up).

If `unlockable_mem` is greater than available memory, the system sets `unlockable_mem` to available memory.

Lockable memory is used for:

- Process images and overhead locked with *plock*(2)
- Shared memory segments locked with the SHM_LOC command of the *shmctl*(2) system call
- Miscellaneous dynamic kernel data structures used by the shared memory system and some drivers.

Any call that needs lockable memory may fail if the value is too small. Note that lockable memory limits the amount of memory that can be locked, but that this memory is available for virtual memory except when it is locked.

Dependencies (interactions with other system values)

unlockable_mem <= physical memory

BNF Format

```
unlockable_mem 0;

BNF:

    <unlockable_mem-stmt>  ::= unlockable_mem <integer>;
                            |  unlockable_mem "<anychars>";
```

B

Swap Space Computation

Swap Space Computation

A. SUM (all shared code sizes) of all running processes as shown by `ps -el`.

   Do not count the page daemon, swapper, or statdaemon processes. The `file` command will show you if a file contains shared text, and you can find the size of the text by means of the `size` command.

B. SUM (all data and stack sizes) of all processes. By using the `size` command, you can calculate the size of initialized data and uninitialized data (BSS). This represents only part of the total swap space requirements of the process. In addition, you must calculate the amount of dynamic heap and stack space that the program might require. If you are familiar with the program's runtime logic, you might be able to calculate this by looking at requests made to `sbrk` or `malloc`.

   You can approximate by running the program with a typical input stream, and determine the total virtual memory size in number of pages. The virtual memory size can be obtained by running `ps -el` and looking in the SZ (size in 512-byte blocks) field for the program you are interested in.

   Subtract the code size calculated in step A from this to get the total data and stack size.

C. SUM (sticky code sizes) for all **sticky code** files that will be executed, but are not currently being used by any processes. Typically editors fall into this category. (See the discussion under "The Sticky Bit" in *How HP-UX Works: Concepts for the System Administrator*.)

D. SUM (all existing shared memory segment sizes) for shared memory segments created by users via `shmget`. `Ipcs` can be used to show active shared memory segments.

E. Size of the scratch area used by **exec** to hold arguments. The default size of this area is 256 Kbytes; it can be changed by using **uxgen**.

F. Fragmentation and overhead.

   Fragmentation is the difference between the swap space needed at any given time and the actual amount allocated. The parameter **swchunk** controls swap space allocation—see Appendix A for details.

   Overhead is additional disk space needed to store system-related information when a process is swapped out.

   There is no easy way to figure out an an accurate value for fragmentation and overhead. We suggest you take an arbitrary value such as 6Mb.

Swap space = A + B + C + D + E + F.

Swap Space Computation Worksheet

A. For shared code, fill out the code space needed by the process.

Process ID	Code size.
---	---
---	---
---	---
---	---
---	---

B. For each shared process listed above, fill out the data and stack space needed by the process. For each nonshared process, add the process's code size to its data size and enter the amount (that is, total from executing size).

Process ID	Data size. (minimum data space = dmmin x 1Kb block default = 32Kb)
---	---
---	---
---	---
---	---
---	---

B

C. For each sticky bit file that was executed since power-up, but not currently used, fill out the code space.

Process ID	Code size.
---	---
---	---
---	---
---	---
---	---

B

D. For each shared memory segment, give the shared segment size.

Shared Memory	Segment Size.
---	---
---	---
---	---
---	---
---	---

E. Scratch area used for arguments during **exec**. Default is 256 Kb:

```
-----------------

-----------------
```

F. Fragmentation and overhead.

We suggest using an estimate such as 6 Mb.

```
-----------------------------

  TOTAL AMOUNT OF SWAP SPACE NEEDED IS:

  A   +  B +  C +  D +  E +  F =  Total swap space

  ----+----+----+----+----+----=  ------------------------------
```

B

Example

Let's assume that you are going to run four FORTRAN compiles with optimization. This example shows how to calculate the additional swap space that might be required to handle this. Let's start with one compile and gather some statistics.

Note The following example may not be indicative of how much swap space your compiles would use, because the amount of swap space required by HP-UX compilers depends on the size of the program being compiled. Further, compiling with optimization may consume almost twice as much swap space.

```
% fc -o test.f >& out &
[1] 3663
ps\ -l)
F S   UID   PID  PPID  C PRI NI   ADDR  SZ    WCHAN TTY      TIME COMD
1 S   867 22055     1  1 168 20 a0a308 121  1515308 ttyd3p4  0:07 csh
1 S   867  3663 22055  0 158 20 5f3900  28   4c8024 ttyd3p4  0:00 fc
1 R   867  3664  3663204 229 20 d2ae306963          ttyd3p4  0:25 f77comp
1 R   867  3680 22055 49 190 20 f9d260 182          ttyd3p4  0:00 ps

% file /usr/lib/f77comp
/usr/lib/f77comp:       S800 shared executable

% size /usr/lib/f77comp
5720($MILLICODE$) + 141600($LIT$) + 1726120($CODE$) + 8($CODE20$) +
576($UNWIND$MILLICODE$) + 47152($UNWIND$) + 1824($UNWIND$) + 16($UNWIND20$) +
132($RECOVER$) + 164192($GLOBAL$) + 93032($DATA$) + 9424($DATA$) +
8($PFACOUNTER$) + 24400($BSS$) = 2214204
```

From the above information, you can determine that

- The code size for /usr/lib/f77comp is 5720+1726120+8= 1731848 bytes
- The data size for /usr/bin/f77comp is 93032+9424= 102456 bytes
- The BSS size for /usr/bin/f77comp is 24400
- (BSS and initialized data = 102456 + 24400 = 126856 bytes)

The **ps** command tells you that this FORTRAN compile uses 6963 pages or 14260224 bytes. Subtracting the code size from this and leaves 12528376 bytes for the total data, heap, and stack. The difference between this data size and the fixed (obtained from **size** above), is the amount of dynamic space used by the FORTRAN compiler for this program.

A.)Process ID Code Size

3664 1731848

B.)Process ID Data size

3664 12528376

Therefore, you need 1.7 Mb of swap space for the compiler text (of which there will be only one copy) and 12.5 Mbyte of swap space for the per process data. This would mean that the system would need approximately 1.7+4 * 12.5 (or 51.7 Mb) of additional swap space for the four compiles to execute in parallel.

C

Federal Information Processing Standard

The U.S. Government has published the Federal Information Processing Standard (FIPS 151-1, hereafter called FIPS). Based on the POSIX standard IEEE Std 1003.1-1988, FIPS specifies the behavior of a system in areas where the POSIX standard permits divergent behavior. In three areas, HP-UX permits a wider range of behavior than allowed by FIPS. These areas are:

- Changing file ownership
- Group ID of new files
- Truncation of filenames

Based on the announcement of the FIPS published in the *Federal Register, Volume 54, no. 70, April 13, 1989*, the following sections explain the configuration of HP-UX so it conforms to the FIPS, pending its approval.

Restricting the chown(1) Function

The POSIX standard permits an implementation to allow users to change the ownership of their own files (as does System V) or restricts this action to privileged users (as does 4.3BSD). You can control this with the privileged group facility (see *getprivgrp*(2) in the *HP-UX Reference*). The FIPS makes the following statement, which requires the 4.3BSD behavior:

```
The implementation shall support the option
_POSIX_CHOWN_RESTRICTED.
```

Configure HP-UX to behave this way by removing users from groups with the privilege PRIV_CHOWN. By default, HP-UX grants this privilege to all users, so you must revoke the privilege each time you start up the system, or add this line to the file `/etc/rc`:

```
setprivgrp -n CHOWN
```

Note The use of some commands—those associated with backup and recovery operations, for example—may be affected by restricting users' access privileges.

Controlling the Group ID of New Files

The POSIX standard permits an implementation either to set the group ID of a newly created file either to the effective group ID of the creating process (as does System V) or to the group ID of the parent directory of the new file (as does 4.3BSD). HP-UX follows the 4.3BSD semantics if the set-group-ID bit of the parent directory is set, and follows System V semantics otherwise. The FIPS makes the following statement, which effectively requires the 4.3BSD behavior:

C

```
The implementation shall support the setting of the
group ID of a file (when it is created) to that of its
parent directory.
```

An HP-UX system can be configured to behave this way by setting the set-group-ID bit of all directories in the system. When HP-UX is installed, this bit is not set on directories, so this requires setting it once for all directories where the FIPS behavior is desired. Executing the following command as superuser will do this for the entire system; the command should be executed with no NFS mounts or RFA netunams in effect:

```
find / -type d -exec chmod g+s \;
```

You may choose to leave the setgid bit off for some directories which are not associated with any group, because the effective group ID of the creating process may be more meaningful for files in those directories. Examples of such directories include `/tmp` and `/usr/tmp`. This can be done by constructing a more complex `find` command, or by turning the set-group-ID bit off for those files after the `find` command, with a command such as:

```
chmod g-s /tmp /usr/tmp
```

This practice may not conform strictly to the FIPS.

Truncating Filenames During Pathname Resolution

When a filename specified by a user is longer than the maximum supported by the file system, the POSIX standard permits an implementation either to truncate the name to the supported maximum (as does System V) or to give an `ENAMETOOLONG` error (as does 4.3BSD). HP-UX follows the 4.3BSD semantics for file systems that support long filenames, and follows System V semantics for other file systems. The FIPS makes the following statement, which effectively requires the 4.3BSD behavior:

```
The implementation shall support the functionality
associated with the feature {_POSIX_NO_TRUNC}.
```

To conform with the FIPS, convert all file systems to support long filenames (see Chapter 6, "Managing the File System").

Index

A

access control lists, 4-11
  creating entries, 4-43
  removing entries, 4-47, 4-49, 4-63, 4-66, 4-77, 4-79
  replacing entries, 4-71
accessing **setuid** and **setgid** system calls, 4-8, 4-81
access permissions, 4-2
  logical volumes, 8-57
accounting, 2-19. *See also* system accounting
acctresume parameter, A-7
acctsuspend parameter, A-8
ACLs (access control lists), 4-11
activating a volume group, 8-96, 8-97
  options, 8-99
  to make disk available, 8-97
adding
  a local printer, 10-25, 10-49
  an automated backup schedule using HP-UX commands, 9-73–78
  an automated backup schedule using SAM, 9-31–41
  a network-based printer, 10-32, 10-58
  a remote printer, 10-29, 10-53
  a user group, 4-30, 4-58
  device drivers to kernel using SAM, 2-36
  disks, 6-40
  disks to a volume group, 8-51
  disks to a volume group using SAM, 8-34
  file system to **/etc/checklist**, 6-56
  peripherals, 2-6, 5-4
  subsystems, 2-45
  subsystems using SAM, 2-47
  users, 4-16, 4-34
  users and groups, 2-14
  users to groups, 4-33, 4-72
adding device swap
  in disk sections, 7-24
  in logical volumes, 7-24
  using **swapon**, 7-24
adding file systems
  CD-ROM file system, 6-57
  NFS file system, 6-64
  using **mount**, 6-55
  using SAM, 6-51, 6-52
adding file system swap
  using **swapon**, 7-30
after installing HP-UX, 2-7
alias
  setting up, 2-26
allocation
  contiguous, 8-57, 8-69
  strict, 8-69
assessing users' needs, 2-1, 2-2
assigning special group privileges, 4-80
assigning swap priority, 7-12
AUTO
  changing, 3-9
  creating with **mkboot -a**, 3-10

showing contents, 3-9
updating with **mkboot**, 8-111
autofile, 3-10
autofile (**AUTO**)
creating, 8-119
automating backups, 9-20
using HP-UX commands, 9-73–78
using SAM, 9-31–41

B

backing up file system
mirrored logical volume, example, 8-90
backup
LVM configuration file, 8-103
backup index file
viewing using **frecover**, 9-87
viewing using SAM, 9-54, 9-56, 9-58
backup kernel
booting, 2-44
creating, 2-33
backup levels, 9-2, 9-9–12
backup media capacities, 9-14–19
backups
automating, 9-20, 9-31–41, 9-73–78
automating using HP-UX commands, 9-73–78
automating using SAM, 9-31–41
cartridge tape data blocking, 9-61, 9-65
cartridge tapes, 9-66, 9-81
choosing storage device, 9-13–14
determining how often, 9-8
determining which data, 9-4
excluded files, 9-2, 9-6, 9-7
fbackup, 9-21, 9-22, 9-60–67
file extraction, 9-2
full, 9-2, 9-9–12, 9-64, 9-66, 9-70
graph file, 9-3, 9-7
included files, 9-3, 9-5, 9-7
incremental, 9-3, 9-8, 9-9–12, 9-65
index files, 9-55, 9-57, 9-62, 9-66, 9-87
initial backup, 2-30
magnetic tapes, 9-64, 9-65
media capacity, 9-14–19
other utilities, 9-88
over the network using **fbackup**, 9-61, 9-65
over the network using SAM, 9-27
recovery, 9-3
restoring data, 9-21, 9-44, 9-50, 9-79
system, 9-3
tcio, 9-66, 9-81
to a hard disk, 9-70
using HP-UX commands, 9-22, 9-59–78
using SAM, 9-23–59
utilities, 9-22
backup storage devices, 9-14
bad block relocation, 8-30
guidelines, 8-30
bdf, 6-75, 6-76, 7-11
BDRA
updating with **lvlnboot**, 8-120
updating with **lvlnboot -r**, 8-110
block device file
definition, 6-2
logical volumes, 8-4
block size file system parameter, 6-45
boot, 3-2
bootable LVM disk
creating with **pvcreate -b**, 8-119
boot disks
logical volumes, 8-111
booting the backup kernel using ISL, 2-44
boot-up
customizing system startup scripts, 2-22
bootup problems, 4-86
boot utilities
installing with **mkboot**, 8-119
Bourne shell

login files, 2-21, 2-26
bufpages parameter, A-9
bytes-per-inode file system parameter, 6-47

C

cancel model script (printer), 10-4
cartridge tape backup example, 9-66
cartridge tape data blocking (**tcio**), 9-61, 9-65, 9-81
cartridge tape (HP format) capacity, 9-16
cartridge tape (QIC format) capacity, 9-19
catman, 2-12
CD-ROM file system, 6-10
adding using **mount**, 6-57
changing file ownership, 4-8, 4-81
changing run-levels, 4-87
character device file
definition, 6-2
logical volumes, 8-4
required by **diskinfo**, 6-29
required by **newfs**, 6-29, 6-30
charging fees, 12-12, 12-55
checking
file system consistency, 6-74
checking **quotas** consistencies, **quotacheck**, 6-90
command
lpana, 10-82, 10-83
accept, 10-6, 10-52, 10-64, 10-73
bdf, 6-76, 7-11, 9-70, 9-71
cancel, 10-70
catman, 2-12
chacl, 4-11, 4-43, 4-47, 4-49, 4-63, 4-66, 4-71, 4-77, 4-79
chfn, 4-42, 4-57
chgrp, 4-40, 4-45
chmod, 4-45
chown, 4-40, 4-45, 4-49, 4-65
convertfs, 6-71
cpio, 9-88
cpset, 6-82
crontab, 9-74, 9-76, 9-78
csh, 4-38
cwall, 11-5
date, 2-10, 2-11
dd, 9-70, 9-85
df, 6-76
disable, 10-6, 10-61, 10-66, 10-72
du, 6-76
edquota, 6-83
elm, 11-6
enable, 10-6, 10-52, 10-66, 10-73
exportfs, 9-22
extendfs, 8-63
fbackup, 9-21, 9-22, 9-60–67
find, 4-47, 4-49, 4-63, 4-65, 4-68, 4-71, 4-77, 6-77
finger, 4-42, 4-57
frecover, 9-21, 9-22, 9-78–84
fsck, 6-74
ftio, 9-88
getprivgrp, 4-80
grep, 4-50
grpck, 4-50, 4-60, 4-72
hpux, 3-5
init, 4-13, 4-88
ksh, 4-38
lifcp, 5-43
lifinit, 6-108
lifls, 6-107
lpadmin, 10-49, 10-53, 10-57, 10-60, 10-62
lpalt, 10-75, 10-78
lpana, 10-69, 10-80, 10-84
lpmove, 10-73
lpsched, 10-13, 10-36, 10-69, 10-73
lpshut, 10-49, 10-53, 10-62, 10-69
lpstat, 10-52, 10-57, 10-69, 10-70, 10-76

lvchange, 8-60
lvcreate, 8-55
lvdisplay, 8-59
lvextend, 8-61
lvmerge, 8-90
lvreduce, 8-92
lvremove, 8-66
lvsplit, 8-90
lvsync, 8-94
mediainit, 6-40, 7-38
mkboot, 5-43, 8-119
mkdir, 4-44
mkfs, 6-42
motd, 11-3
mount, 6-55
newfs, 6-42
news, 2-17, 11-3
pvcreate, 8-47
pvdisplay, 8-54
pwck, 4-50, 4-52, 4-54, 4-66
quot, 6-88
quota, 6-88
quotacheck, 6-86, 6-90
quotaoff, 6-87
quotaon, 6-86
regen, 2-35
reject, 10-6, 10-61, 10-64, 10-72
repquota, 6-87
rlpdaemon, 10-15
rsh, 4-38
setprivgrp, 4-8, 4-81
sh, 4-38
shutdown, 3-12, 4-88, 4-89
swapinfo, 7-6
swapon, 7-25
tar, 9-89
tcio, 9-66, 9-81, 9-82, 9-83
tic, 2-29
tset, 2-20, 2-21
umount, 6-58
untic, 2-29
uucp, 11-5
vgcfgbackup, 8-103
vgcfgrestore, 8-103
vgchange, 8-96
vgcreate, 8-48
vgdisplay, 8-52
vgexport, 8-101
vgextend, 8-51
vgimport, 8-101
vgsync, 8-95
vipw, 4-5
wall, 11-4
write, 11-4
computing swap space, B-1
concepts for the system administrator, 1-8, 1-9
configurable parameters, 2-39. *See also* system parameters
configuring
  kernel, 2-30
conforming to FIPS, C-1
connect session accounting, 12-10, 12-24
console, 5-32
constructing a system, 2-1
contiguous allocation, 8-69
convertfs, 6-71
cpset, 6-82
crash recovery
  mirror consistency, 8-72
  retoring files, 9-21
creating
  an automated backup schedule using HP-UX commands, 9-73–78
  an automated backup schedule using SAM, 9-31–41
  a printer class using **lpadmin**, 10-59
  file systems, 2-11, 6-13, 6-21
  file systems in disk section, 6-33
  file systems using **newfs**, 6-42
  kernel, 2-30
  LIF volume, 6-107

logical volumes for file systems, 6-24, 6-26
message of the day, 2-25
mirrored logical volume for a file system, 6-28
physical volumes, 8-47
terminfo file, 2-29
volume group:example, 8-50
volume groups, 8-47
cron
problems caused by changing system clock, 2-11
terminating to set date and time, 2-10
crontab file, 9-74–78
C shell
login files, 2-21, 2-26
csh.login, 2-21
.cshrc, 2-21
editing, 2-26
customizing
/etc/inittab, 2-23
/etc/issue, 2-25
/etc/motd, 2-25
/etc/rc, 2-23
message of the day, 2-25
printer model scripts, 10-12
SAM, 1-15, 4-21
SAM interface colors, 1-25
SAM interface fonts, 1-25
shutdown, 3-17
system startup, 2-22
the login prompt, 2-25
users' editing environments, 2-26
users' login environments, 2-26
cwall, 11-5

D

database partitions, 8-28
organizing in logical volumes, 8-28
DataPair/800
migrating to MirrorDisk/UX, 8-139
date, 2-10, 2-11
format, 2-10
setting, 2-10, 2-11
DDS format (DAT) tape capacity, 9-17
deactivating a volume group, 8-96
deactivating users account, 4-24, 4-51
default group, 4-2. *See also* primary group
default printer, 10-5
detail log file (SAM), 1-45
DEV_BSIZE, 7-8
device drivers
adding to kernel using SAM, 2-36
printers, 10-28, 10-46, 10-47
removing from kernel using SAM, 2-36
device files, 10-13
block, 7-24
definition, 6-2, 8-4
logical volumes, 8-20
logical volumes, block, 8-22
logical volumes, character, 8-22
naming of block device files, 6-3, 8-4
naming of character device files, 6-3, 8-4
physical volumes, block, 8-22
physical volumes, character, 8-22
devices. *See* peripherals
device swap, 7-4
adding using **swapon**, 7-24
allocating, 7-10
enabling on boot, 7-36
removing, 7-23, 7-27
df, 6-76
diagnostic bit, 6-40, 7-38
directory
/etc/filesets, 2-50
/system, 2-50
disabling device swap, 7-23
disabling printers, 10-39, 10-66

disk drive, 5-40
  moving using HP-UX commands, 5-36
  moving using SAM, 5-13
  removing using HP-UX commands, 5-50
  removing using SAM, 5-24
diskinfo, 6-29
disk layout, 6-11
  sections, 6-11, 6-35
disk product ID
  using **diskinfo** to find, 6-29
disk quotas
  block limits, 6-80
  described, 6-79
  file limits, 6-80
  hard limits, 6-80
  individual users, 6-84
  planning, 6-80
  prototype user, 6-83
  reaching hard limits, 6-89
  setting up, 6-80
  soft limits, 6-80
  starting, 6-85
  turning off, 6-87
disk sections, 6-11, 6-35
  block device files, 6-3
  character device files, 6-3
  selecting, 6-35
disk space
  manpages, 2-12
  using **vgdisplay**, 6-25
disk space usage accounting, 12-17
disk spanning
  guidelines, 8-27
  logical volumes, 8-1
disk type, 6-43
  using **diskinfo** to find, 6-29
disk usage
  accounting, 12-10
  bdf, 6-76
  df, 6-76
  du, 6-76
  limiting with disk quotas, 6-79
  quot, 6-88
  quota, 6-88
  repquota, 6-87
displaying logical volume information with **lvdisplay**, 8-59
displaying physical volume information, 8-54
displaying user account information, 4-28, 4-55
displaying volume group information
  vgdisplay, 6-25
displaying volume group information with **vgdisplay**, 8-52
documentation
  on-line (manpages), 2-12
dst parameter, A-11
du, 6-76
dumps
  logical volumes, 8-133
  moving, 5-43
dumps area
  defined, 7-2
dumps logical volumes
  creating, 8-117
  updating BDRA using **lvlnboot -d**, 8-117

E

editing **/etc/passwd**, 4-5
editing system files
  /etc/csh.login, 2-26
  /etc/inittab, 2-23
  /etc/issue, 2-25
  /etc/motd, 2-25
  /etc/profile, 2-26
  /etc/rc, 2-23
editing user files
  .exrc, 2-26

.profile, .cshrc, etc., 2-26
EDITOR environment variable, 4-44, 4-50, 4-52, 4-54, 4-65, 4-71
edquota, 6-83
-p, 6-83
-t, 6-84
effective group, 4-2
electronic mail, 11-6
elm, 2-15
mail, mailx, 2-15
setting up, 2-15
elm, 11-2, 11-6
setting up, 2-15
email (*see* electronic mail), 2-15
enabling device swap
on boot, 7-36
using swapon, 7-25
enabling file system swap
on boot, 7-37
using swapon, 7-33
enabling printers, 10-41, 10-66
ENV
setting for .kshrc, 2-21
.environ, 2-21
environment file
.cshrc, 2-21
.environ, 2-21
/etc/csh.login, 2-21
/etc/group, 2-20
/etc/inittab, 2-20
/etc/issue, 2-21
/etc/motd, 2-21
/etc/passwd, 2-20
/etc/profile, 2-21
/etc/rc, 2-20
/etc/ttytype, 2-20, 2-21
.exrc, 2-21
.kshrc, 2-21
local files, 2-21
.login, 2-21
.profile, 2-21
environment file /etc/inittab, 4-86
environment variable
EDITOR, 4-5, 4-44, 4-50, 4-52, 4-54, 4-65, 4-71
ENV, 2-21
LPDEST, 10-10
TERM, 2-27
TZ, 2-8
/etc/bcheckrc, 6-66
/etc/checklist, 6-66
options for mounting, 6-66
quota option, 6-81
/etc/disktab, 6-43, 7-10
/etc/dmesg, 7-41
/etc/exports, 6-64
/etc/filesets directory, 2-50
/etc/hosts, 6-64
/etc/rc, 6-66
.exrc, 2-21, 2-26
extendfs, 8-63
extending a file system extendfs, 8-63
extending a volume group, 8-35
extents, 8-19
logical, 8-19
physical, 8-19
stale, 8-72

F

fbackup
active files, 9-23, 9-27, 9-60
backup levels, 9-9
crossing NFS mount points, 9-22
excluded files, 9-7
graph file, 9-7
included files, 9-7
multiple devices, 9-61, 9-67
syntax, 9-60
fbckup
examples, 9-63
Federal Information Processing Standard (FIPS), C-1

file
/usr/adm/fbackupfiles/dates, 9-62
crontab, 12-4
csh.login, 4-38
.cshrc, 2-26, 4-38
d.profile, 4-39
/etc/bcheckrc, 6-66, 6-91
/etc/checklist, 6-66, 7-36
/etc/csh.login, 2-8, 2-26
/etc/disktab, 6-11, 6-35, 6-43
/etc/d.profile, 2-26
/etc/group, 2-20, 4-6, 4-36, 4-50, 4-58, 4-61, 4-68, 4-72, 4-74, 4-75, 4-78
/etc/hosts, 2-25
/etc/inittab, 2-20, 2-23, 4-12, 4-13, 4-85, 4-86
/etc/issue, 2-21, 2-25
/etc/logingroup, 4-6, 4-36, 4-41, 4-50, 4-58, 4-61, 4-68, 4-72, 4-74, 4-75, 4-78
/etc/motd, 2-21, 2-25
/etc/passwd, 2-20, 4-5, 4-18, 4-34, 4-42, 4-48, 4-51, 4-53, 4-55, 4-61, 4-65, 4-67
/etc/profile, 2-8, 2-26
/etc/ptmp, 4-5
/etc/rc, 2-8, 2-20, 2-23, 6-66, 12-3, 12-32
/etc/ttytype, 2-20
.exrc, 2-21, 2-26
.kshrc, 2-26
.login, 2-26, 4-38
.profile, 2-26, 4-38
profile, 4-38
quotas, 6-82
/SYSBCKUP, 2-33
ttytype, 2-21
/usr/adm/fbackupfiles/dates, 9-9, 9-10
/usr/bin/x11start, 2-18
/usr/spool/lp/cmodel/rcmodel, 10-16, 10-53
/usr/spool/lp/lpana.log, 10-69, 10-80
/usr/spool/lp/smodel/rsmodel, 10-16, 10-53
file access permissions, 4-2
file group ownership, 4-2
file name length, 6-69
file ownership, 4-2
files
access control lists, 4-11
access permissions, 4-4, 4-9, 4-13
creating access control list entries, 4-43
locating by access control list entry, 4-47, 4-65, 4-71
locating by both user and group ownership, 4-65, 4-71
locating by file ownership, 4-47, 4-49, 4-63, 4-65, 4-68, 4-77
locating by file type, 4-49, 4-65
locating by group ownership, 4-63, 4-65, 4-68, 4-77
ownership, 4-4, 4-9, 4-39
removing access control list entries, 4-47, 4-49, 4-63, 4-66, 4-77, 4-79
replacing access control list entries, 4-71
system, 2-47, 2-50
fileset
NET-PERIPH, 10-33
NONHPTERM, 2-27
SAM, 1-21
X11-RUN, 1-21
fileset partitions, 2-50
filesets, 2-50
file system
block size, 6-45
bytes per inode, 6-47
checking consistency, 6-74

creating, HP-UX commands, 6-23
creating with **newfs**, 6-30
fragment size, 6-45
parameters, 6-42, 6-45, 6-47
reserved area, 6-46
file systems, 8-39, 8-40
capacity, 6-75
CD-ROM, 6-10
creating, 2-11, 6-21
HFS, 6-10
in disk sections, 6-11
in logical volumes, 6-11
listing in logical volumes, 8-20
listing with **bdf**, 6-75
mirroring, 6-28
modifying file name length, 6-69
mountable, 6-7
mounting, 6-51
mounting at bootup, 6-66
moving, 6-92
NFS, 6-10
on-line backup of mirrored file systems, 8-90
setting up, 2-11
types, 6-10
unmounting at shutdown, 6-60
unmounting using **umount**, 6-58
usage, 6-75
file system swap, 7-4
adding using **swapon**, 7-30
allocating, 7-10
enabling on boot, 7-37
guidelines, 7-10
removing, 7-35
selecting file system, 7-10
find, 6-77
fragment size file system parameter, 6-45
frecover
active files, 9-79
crossing NFS mount points, 9-22
examples, 9-81–84, 9-87
syntax, 9-79, 9-87
viewing index file on device, 9-87
fs_async parameter, A-12
fsck, 6-74
fsck command, 6-74
full backups, 9-2, 9-9–12, 9-70

G

gecos information, 4-42
getty
entry for each terminal in **/etc/inittab**, 2-23
GID, 4-2, 4-7, 4-36, 4-58, 4-67
graph file (backups), 9-7
group, 2-20
group character device file, 8-47
creating with **mknod**, 8-47
group-ID, 4-7, 4-36, 4-58, 4-67
group ID, 4-2
group membership, 4-6, 4-58, 4-67
group ownership, 4-2, 4-63, 4-68, 4-77
groups, 4-5
adding, 2-14, 4-30, 4-58
adding users to, 4-33, 4-41, 4-72
defining membership, 4-36, 4-50
maximum membership, 4-5
passwords, 4-7
removing, 4-31, 4-61
removing users from, 4-33, 4-50, 4-75
special privileges, 4-8, 4-80

H

HFS file systems, 6-10
host name
obtained from **/etc/hosts**, 2-25
how to use **rmfn**, 2-51
HP-FL interface
logical volumes, 8-29
HP format cartridge tape capacity, 9-16
HP-IB

disks require separate volume group, 8-46
HP-IB interface
logical volumes, 8-29
HP-UX
constructing a system, 2-1
installing, 2-4
kernel, 2-32
shutdown, 3-12
updating, 2-4
hpux boot command, 3-5
HP-UX cluster administration, 1-8, 1-9
HP-UX configuration
printer drivers, 10-28, 10-46, 10-47
HP-UX Reference
setting up on-line, 2-12
HP-UX Reference Manual, 1-9

I

increasing swap space, 7-5, 7-9
incremental backups, 9-3, 9-9–12
index files (backup), 9-66
creating using **fbackup**, 9-62
creating using SAM, 9-26, 9-30, 9-37
viewing using **frecover**, 9-87
viewing using SAM, 9-54, 9-56, 9-58
inittab, 2-20
editing, 2-23
installing
HP-UX, 2-4
optional software, 2-5
peripherals, 2-6
Installing and Updating HP-UX, 2-4
Installing Peripherals, 2-6
interface scripts (printer), 10-3, 10-6
interleaved swapping, 7-9
I/O interface
types supported by logical volumes, 8-29
iomemsize parameter, A-13
ISL, 3-5
issue
editing, 2-25

K

kernel, 3-5
backup copy (**/SYSBCKUP**), 2-33
backup kernel, 2-33
configuring, 2-30
pseudo-drivers listed, 2-46
reconfiguring to add/remove device drivers using SAM, 2-36
reconfiguring to modify special devices using SAM, 2-41
reconfiguring to modify system parameters using SAM, 2-39
reconfiguring using HP-UX commands, 2-32
when to reconfigure, 2-31
kernel devices. *See* special devices
kernel parameters. *See* system parameters
Korn shell
login files, 2-21, 2-26
.kshrc, 2-21
editing, 2-26

L

lifcp, 5-43
lifinit, 6-108
lifls, 6-107
LIF volumes, 6-107
initialization, 6-107
limiting disk usage, disk quotas, 6-79
line-printer spooler
setting up, 2-16
line printer spooling system, 10-1
local printer, 10-3, 10-13, 10-25, 10-49
locking files for read only, 4-8, 4-81
locking processes into memory, 4-8, 4-81
log files (SAM)

detail, 1-45
summary, 1-44
verbose, 1-45
logical extents, 8-19
logical volume naming convention, 8-23
logical volumes, 6-11
access permissions, 8-57
bad block relocation, 8-30
block device file, 8-23, 8-55
block device files, 8-4
boot disks, 8-111
changing characteristics with **lvchange**, 8-60
character device file, 8-23, 8-55
character device files, 8-4
configuring with SAM, 8-42
contiguous allocation, 8-57
creating mirrored logical volumes, 8-83
creating using SAM, 8-37
creating volume groups, 8-47
creating with **lvcreate**, 8-55
database partitions, 8-28
decreasing size using SAM, 8-41
device files, 8-20
disk spanning, 8-1
displaying information with **lvdisplay**, 8-59
dumps, 8-133
estimating space for file system, 8-24
extending a logical volume's file system, 8-63
file systems, 8-24
group character device file, 8-47
HP-FL interface, 8-29
HP-IB interface, 8-29
increasing extents, 8-61
increasing extents with **lvextend**, 8-61
I/O interface types,guidelines, 8-29
kernel configuration, 8-131
managing with SAM, 8-32
mirror allocation, 8-69
mirroring, 8-19, 8-74
mirroring, guidelines, 8-68, 8-74
mirror write cache, 8-72
moving to another disk, 8-66
naming, 8-57
naming for raw data storage, 8-21
primary swap, 8-133
raw data storage, 8-28
reducing mirror copies with **lvreduce**, 8-92
reducing size with **lvreduce**, 8-64
removing with **lvremove**, 8-66
root file system, 8-133
scheduling mirror writes, 8-71
SCSI interface, 8-29
spanning disks, 8-27
spanning disks, guidelines, 8-27
swap:guidelines, 8-27
unmirroring, 8-92
.login, 2-21
editing, 2-26
log in, 4-2
login
customizing the prompt, 2-25
customizing users' login files, 2-26
message of the day, 2-25
login directory, 4-36, 4-37
long file names, 6-44, 6-69
disabling, 6-73
enabling using **convertfs**, 6-71
enabling using SAM, 6-70
LPDEST environment variable, 10-10
lp spooler
accepting print requests, 10-19, 10-64
adding a local printer, 10-25, 10-49
adding a network-based printer, 10-32, 10-58
adding a remote printer, 10-29, 10-53
adding printers, 10-25–33, 10-49–58

canceling print requests, 10-70
changing a printer fence priority, 10-43, 10-68
checking status, 10-38, 10-69
collecting printer activity statistics, 10-20, 10-69, 10-80
commands, 10-6
components of, 10-8
controlling with HP-UX Commands, 10-22, 10-44–84
controlling with SAM, 10-22, 10-23–43
creating a printer class, 10-59
device files, 10-13, 10-53
disabling printers, 10-19, 10-39, 10-66
displaying printer activity statistics, 10-80–84
enabling printers, 10-19, 10-41, 10-66
interface scripts, 10-6, 10-11
LPDEST environment variable, 10-10
moving all print requests, 10-72
moving selected print requests, 10-75
network-based printer, 10-14
plotters, 10-18
print destinations, 10-10
printer classes, 10-9
printer models, 10-11, 10-53
printer names, 10-8, 10-53
printer priorities, 10-17, 10-53
printer queues, 10-6
print request, 10-4
print request identification number, 10-14
print request priorities, 10-17, 10-78
print requests, 10-6
priorities, 10-17
rejecting print requests, 10-19, 10-61, 10-64
remote print requests, 10-15
remote spooling, 10-6, 10-15
removing a printer, 10-34, 10-61
removing a printer class, 10-61
removing printers, 10-61, 10-62
request directories, 10-6
scheduler, 10-13
setting up, 10-21, 10-44
starting scheduler, 10-36, 10-62, 10-69
stopping scheduler, 10-36, 10-37, 10-62, 10-69
system default printer, 10-10, 10-25, 10-29, 10-50, 10-53
viewing printer request status, 10-23, 10-76
viewing printer status, 10-23, 10-76
lvchange, 8-60
lvcreate, 6-27
description, 8-55
example, 8-58
lvdisplay, 8-59
lvextend, 8-61
increasing number of extents, 8-61
increasing number of mirror copies, 8-85
mirroring specific physical volumes, 8-86
lvlnboot
-p, 8-120
-r, 8-120
lvlnboot -r
when adding disks to root volume group, 8-110
lvlnboot -v, 8-109
LVM
checking for, 8-9
commands, mirroring, summary, 8-44
commands, summary, 8-43
migrating root disk, 8-12
migrating to, 8-12
overview, 8-13
LVM bad block relocation, 8-30
lvmerge
example, 8-91
lvmrc, 8-98

lvmrc script, 8-94
lvm subsystem
  S800 file, 8-131
lv pseudo driver
  S800 file, 8-131
lvreduce, 8-92
lvremove, 8-66
lvsplit, 8-90
lvsync, 8-94

M

magnetic tape capacity, 9-16
mail, 11-2
  setting up, 2-15
mailx, 11-2
  setting up, 2-15
major number
  LVM pseudo driver, 8-47
make
  problems caused by changing system clock, 2-11
managing peripherals, 5-1
manpages
  disk space, 2-12
  removing source files, 2-13
  setting up, 2-12
manuals related to system administration, 1-6
maxdsiz parameter, A-14
maxfiles_lim parameter, A-16
maxfiles parameter, A-15
maxssiz parameter, A-17
maxswapchunks parameter, 7-8
maxtsiz parameter, A-18
maxuprc parameter, A-19
maxusers parameter, A-20
media capacity
  DDS format (DAT) tape, 9-17
  HP format cartridge tape, 9-16
  magnetic tape, 9-16
  optical disk, 9-18
  optical disk sections, 9-18
  QIC format cartridge tape, 9-19
mediainit, 6-40, 7-38
  device file to use, 6-40, 7-38
memory
  virtual, B-1
merging mirrored logical volumes with **lvmerge**, 8-90
mesg, 11-2
message of the day
  posting, 2-25
migration
  DataPair/800 to MirrorDisk/UX, 8-139
minfree file system parameter, 6-46
mirror allocation
  guidelines, 8-69
mirror consistency recovery, 8-72
MirrorDisk/UX, 8-19
mirrored logical volumes
  creating, 8-83
  merging with **lvmerge**, 8-90
  splitting with **lvsplit**, 8-90
mirroring
  a file system, 6-28
  contiguous allocation, 8-69
  double mirroring, 8-68
  guidelines, 8-74
  logical volume for a file system, 6-28
  logical volumes, 6-25, 8-19, 8-68
  mirror write cache, 8-72
  number of copies, 8-68
  single mirroring, 8-68
  strict allocation, 8-69
mirror write
  scheduling, 8-71
mirror write cache (MWC), 8-72
mkboot
  example, 5-43, 8-119
mkfs, 6-42
  file system parameters, 6-45

model script (printer), 10-4
modem
  moving using HP-UX commands, 5-31
  moving using SAM, 5-11
  removing using HP-UX commands, 5-49
  removing using SAM, 5-22
modifying
  special devices using SAM, 2-41
  system parameters using SAM, 2-39
  the login prompt, 2-25
motd, 11-2, 11-3
  editing, 2-25
mount, 6-55
  -a, 6-66
  /etc/rc, 6-66
  options, 6-66
  quotas option, 6-85
mountable file systems, 6-7
mount directory, 6-7, 6-51
  creating with **mkdir**, 6-30
mounting file systems, 6-7, 6-51
  at bootup, 6-66
mounting local file systems
  using SAM, 6-52
mounting NFS file systems
  using HP-UX commands, 6-64
  using SAM, 6-61
moving file systems, 6-92
moving peripherals
  disk drive, 5-13, 5-36
  local printer or plotter, 5-5, 5-27
  root disk drive, 5-40
  tape drive, 5-8, 5-29
  terminal or modem, 5-11, 5-31
  the system console, 5-32
msgmap parameter, A-22
msgmax parameter, A-24
msgmnb parameter, A-25
msgmni parameter, A-26
msgseg parameter, A-27
msgssz parameter, A-28
msgtql parameter, A-29
multiple devices (backup), 9-61
multi-user mode, 4-2

N

naming conventions
  logical volumes, 8-22
nbuf parameter, A-30
ncallout parameter, A-32
netisr_priority parameter, A-33
NET-PERIPH fileset, 10-33
network-based backups, 9-61, 9-65
network-based printer, 10-4, 10-14, 10-32, 10-58
networking
  documentation, 2-14
  setting up, 2-14
newfs, 6-42
news, 11-2, 11-3
  setting up, 2-17
nfile parameter, A-34
nflocks parameter, A-35
NFS file system, 6-10
  adding using **mount**, 6-64
  crossing mount points for backups using **fbackup**, 9-22
  crossing mount points for backups using SAM, 9-26, 9-30, 9-32, 9-37
  crossing mount points for recovery using **frecover**, 9-22
  crossing mount points for recovery using SAM, 9-41, 9-44, 9-47, 9-50
  mounting with SAM, 6-61
ninode parameter, A-36
NONHPTERM fileset, 2-27
non-standard terminals
  setting up, 2-27
nproc parameter, A-37
npty parameter, A-38

nstlbe parameter, A-39
nswapdev parameter, A-41
nswapdev parameter, 7-9
nswapfs parameter, A-42
nswapfs parameter, 7-11

O

on-line documentation
  setting up manpages, 2-12
operating system
  installing, 2-4
  parameters, A-1
  updating, 2-4
optical disk capacity, 9-18
optical disk section capacity, 9-18
optional software
  installing and updating, 2-5

P

parameters, file system
  block size, 6-45
  bytes per inode, 6-47
  fragment size, 6-45
  reserved area, 6-46
parameters, system, A-1
partition
  moving root, 6-106
partitions (fileset groups), 2-50
passwd, 2-20
passwords, 4-34, 4-35
PATH, 12-5
path name
  logical volume, 8-23
peripherals
  adding, 2-6, 5-4
  adding a local printer, 10-25, 10-49
  adding a network-based printer, 10-32, 10-58
  adding a remote printer, 10-29, 10-53
  installing, 2-6
  Installing Peripherals, 2-6
  moving, 2-6
  moving using HP-UX commands, 5-27–46
  moving using SAM, 5-5–17
  not supported by SAM, 5-2
  removing, 2-6
  removing a printer, 10-61
  removing using HP-UX commands, 5-47–51
  removing using SAM, 5-18–26
physical extents, 8-19
  synchronizing with **lvsync**, 8-94
physical volume
  naming conventions, 8-22
physical volume group
  naming convention, 8-23
physical volume groups, 8-87
  creating, 8-87
physical volumes
  creating with a boot section, 8-47
  creating with **pvcreate**, 8-47
plotter, 10-18. *See also* printer
posting a message of the day, 2-25
post-installation tasks, 2-7
pre-installed system, 8-34, 8-46
prerequisites (for the system administrator), 1-6
primary boot path, 3-2
primary group, 4-2, 4-67
primary swap
  logical volumes, 8-133
primary swap logical volume
  updating BDRA using **lvlnboot -s**, 8-115
primary swap logical volumes, 8-115
prime time, 12-7
print destinations, 10-3, 10-10
printer
  cancel model script, 10-4
  classes, 10-9, 10-59
  device drivers, 10-28, 10-46, 10-47

environment variable **LPDEST**, 10-10
interface scripts, 10-3, 10-11
local, 10-3
logical, 10-3
model scripts, 10-4, 10-11
moving using HP-UX commands, 5-27
moving using SAM, 5-5
names, 10-8
network-based, 10-4
queues, 10-6
remote, 10-5
removing using HP-UX commands, 10-61
removing using SAM, 5-18, 10-34
setting up the line-printer spooler, 2-16
statistics, 10-20, 10-69, 10-80
status model script, 10-4
system default, 10-5, 10-10
printer class, 10-4, 10-61
printer fence, 10-43, 10-68
printer interface scripts, 10-6
printer name, 10-4
printing
spooled (*see* line-printer spooler), 2-16
print priority, 10-5
print queues, 10-4
print request, 10-4
print request identification number, 10-4, 10-14, 10-70
print requests, 10-6, 10-14, 10-70
print routing (*see* line-printer spooler), 2-16
priorities (printers and print requests), 10-17
process accounting, 12-8, 12-11, 12-32, 12-37
.profile, 2-21
editing, 2-26
program
/etc/init, 2-23
/etc/update, 2-31
/usr/bin/x11start, 2-18
prompt
login, 2-25
pseudo-driver
defined, 2-45
list of subsystems and drivers, 2-46
pvcreate, 8-47
-B, 8-119
pvdisplay, 8-54
pvmove, 8-66

Q

QIC format cartridge tape capacity, 9-19
quorum
recovering, 8-106
quot, 6-88
quota, 6-88
quotacheck, 6-86, 6-90
running interactively, 6-91
quotaoff, 6-87
quotaon, 6-86

R

raw data storage
logical volumes, 8-28
naming logical volumes, 8-21
raw device file
See *character* device file, 6-2
rc, 2-20
editing, 2-23
reactivating an active volume group, 8-97
reactivating users account, 4-26, 4-53
real-time priorities, 4-8, 4-81
reconfiguring the kernel, 2-30
recovering data. *See* restoring data
regen, 2-35

remote printer, 10-5, 10-14, 10-16, 10-29, 10-53
remote spooling, 10-5
remote spooling daemon, 10-5, 10-15
remote system administration, 1-42
removing
  access control list entries, 4-47, 4-49, 4-63, 4-66, 4-77, 4-79
  a user group, 4-31, 4-61
  device drivers from kernel using SAM, 2-36
  peripherals, 2-6
  printer class, 10-61
  printers, 10-34, 10-61, 10-62
  subsystems, 2-45, 2-47
  subsystems using SAM, 2-47
  system files, 2-47, 2-50
  users, 4-19, 4-47
  users from groups, 4-33, 4-50, 4-75
removing device swap, 7-23, 7-27
removing file system swap, 7-35
removing peripherals
  disk drive, 5-24, 5-50
  printer or plotter, 5-18, 5-47
  tape drive, 5-20, 5-48
  terminal or modem, 5-22, 5-49
reporting quotas status
  quot, 6-88
  quota, 6-88
  repquota, 6-87
repquota, 6-87
request directories, 10-6
responsibilities of a system administrator, 1-7
restoring data
  all files, 9-21, 9-41, 9-47, 9-79, 9-83
  cartridge tape data blocking, 9-81
  cartridge tapes, 9-81, 9-83, 9-89
  DDS format (DAT) tape, 9-83, 9-88
  frecover, 9-21, 9-78, 9-79–84
  frecover examples, 9-81–84
  frecover syntax, 9-79
  from a hard disk using **dd**, 9-85
  index files, 9-55, 9-57, 9-87
  individual files, 9-21, 9-44, 9-50, 9-78, 9-83
  LVM configuration file, 8-103
  magnetic tape, 9-83, 9-88
  other utilities, 9-88, 9-89
  retrieving index from tape, 9-87
  to a new location, 9-82
  using HP-UX commands, 9-78–86
  using SAM, 9-41–46, 9-47–53
rmfn
  how to use, 2-51
  removing dependent filesets, 2-50
  removing system files, 2-50–52
  removing unneeded filesets, 2-50
  symbolic links, 2-50
root disk, 5-40
root file system, 6-6
  moving to larger logical volume, 8-121
  moving to logical volumes from disk section, 8-121
root logical volume
  creating, 8-112
  mirroring, 8-113
  updating BDRA with **lvlnboot -r**, 8-114
root volume group, 8-109
  displaying contents, 8-109
  extending, 8-109
  special nature, 8-107
  updating BDRA, 8-109
run-level, 4-3, 4-12
  2, 4-12, 4-86
  changing, 4-87, 4-88
  creating new, 4-12, 4-85
  s, 4-3, 4-12, 4-86, 4-89–90

S

S800 file
  LVM requirements, 8-131
SAM
  access to help, 1-13
  adding and removing subsystems, 2-47
  adding entry for terminal to /etc/inittab, 2-23
  customizing the adding and removing a user procedure, 4-21
  customizing the object list, 1-15
  customizing the X window interface, 1-25
  entering, 1-23, 1-26
  exiting, 1-24
  getting into the online help system, 1-12
  logging actions, 1-44
  main menu, 1-23
  overview, 1-11
  primary components, 1-11
  reconfiguring the kernel to add/remove device drivers, 2-36
  reconfiguring the kernel to modify special devices, 2-41
  reconfiguring the kernel to modify system parameters, 2-39
  remote system administration, 1-42
  returning to main menu, 1-24
  tasks, 1-10
  terminal interface, 1-26
  X window interface set up, 1-21
SAM fileset, 1-21
SAM logging
  detail level, 1-45
  summary level, 1-44
  turning on, 1-44
  verbose level, 1-45
SAM terminal interface, 1-26
  changing windows, 1-40
  checkboxes, 1-38
  choosing a menu item, 1-37
  control buttons, 1-29
  entering SAM, 1-26
  exiting SAM, 1-32
  functional area navigation, 1-32
  function keys, 1-30
  menubar, 1-36
  menu buttons, 1-37
  navigation with keys, 1-39
  opening a menu, 1-36
  radio buttons, 1-38
  selecting an object, 1-34
  softkeys, 1-30
  unavailable menu items, 1-37
scheduling mirror writes, 8-71
SCSI interface
  logical volumes, 8-29
section
  moving primary swap, 7-14
  moving root, 6-101, 6-106
security, 1-8, 1-9
semaem parameter, A-43
semmap parameter, A-44
semmni parameter, A-46
semmns parameter, A-47
semmnu parameter, A-48
semume parameter, A-49
semvmx parameter, A-50
sendmail
  setting up, 2-15
setting
  date, 2-11
  system clock, 2-8
  system date, 2-10
  system time, 2-10
  time zone (TZ), 2-8
setting date, 3-11
setting internet protocol address, 3-11
setting system name, 3-11
setting time, 3-11

setting time zone, 3-11
setting up
  disk quotas, 6-80
  electronic mail, 2-15
  elm, 2-15
  file systems, 2-11
  line-printer spooler, 2-16
  mail, **mailx**, 2-15
  manpages, 2-12
  networking, 2-14
  news, 2-17
  on-line *HP-UX Reference* (manpages), 2-12
  sendmail, 2-15
  system accounting, 2-19
  users and groups, 2-14
  X-Windows, 2-18
shell
  Bourne, 2-26
  C, 2-26
  Korn, 2-26
shell initialization files, 4-38
shmmax parameter, A-51
shmmni parameter, A-52
shmseg parameter, A-53
short file names, 6-44, 6-69
shutdown, 3-12
  authorization, 3-17
  customizing, 3-17
shutting down the system, 3-12
single-user mode, 4-3, 4-89–90
software
  installing and updating, 2-5
spanning disks
  performance, 6-43
special devices
  modifying using SAM, 2-41
special group privileges, 4-80
special privilege groups, 4-8
special privileges
  accessing **setuid** and **setgid** system calls, 4-8, 4-81
  changing file ownership, 4-8, 4-81
  locking files for read only, 4-8, 4-81
  locking processes into memory, 4-8, 4-81
  real-time priorities, 4-8, 4-81
splitting mirrored logical volumes with **lvsplit**, 8-90
spooled printing (*see* line-printer spooler), 2-16
spooler (*see* line-printer spooler), 2-16
stale physical extents
  synchronizing with **lvsync**, 8-94
startup
  csh, 4-38
  customizing, 2-22
  files, 4-38
  file system checking, 3-10
  ISL boot, 3-2
  keysh, 4-38
  ksh, 4-38
  peripherals, 3-1
  program, 4-36, 4-38
  rksh, 4-38
  rsh, 4-38
  sh, 4-38
  system selection, 3-5
status model script, 10-4
subsystem
  adding, 2-45
  adding using HP-UX commands, 2-45
  adding using SAM, 2-47
  pseudo-drivers for subsystems, 2-46
  removinging, 2-45
  removing using HP-UX commands, 2-47
  removing using SAM, 2-47
summarizing and reporting accounting information, 12-12, 12-56
summary log file (SAM), 1-44

survey
  assessing users' needs, 2-2
swap
  defined, 7-1
  logical volumes:guidelines, 8-27
swap area
  primary, 7-14
swap device
  moving, 5-43
swapinfo, 7-6, 7-11
swapon
  -a option, 7-26
  parameters, 7-32
  -p option, 7-25
swap priority, 7-12
swap space
  adding with SAM, 7-19
  adding with SAM, disk section, 7-21
  adding with SAM, logical volume, 7-20
  assigning priority, 7-12
  calculating requirements, 7-5
  computing, B-1, B-3
  creating logical volume for, 7-20
  creating logical volume for using SAM, 8-42
  determining current amount, 7-6
  device, 7-2, 7-4
  estimating, 7-7
  file system, 7-4
  increasing, 7-5, 7-9
  listing in logical volumes, 8-20
  listing with **swapinfo**, 7-6
swchunk parameter, A-54
swchunk parameter, 7-8
synchronizing extents with **lvsync**, 8-94
synchronizing logical volumes in volume group, 8-95
/SYSBCKUP, 2-33
system
  constructing, 2-1
  files, 2-50
  name and host name, 2-25
  parameters, 2-39
  setting clock, 2-8
  setting date and time, 2-10
  setting up accounting, 2-19
  shutdown, 3-12
system accounting
  corrupted files, 12-74
  crontab entries, 12-4
  daily usage, 12-5, 12-63
  directory structure, 12-15
  errors, 12-66
  files, 12-81
  holidays, 12-73
  installation, 12-3, 12-9
  logging in, 12-15
  monthly usage, 12-72
  overview, 12-1, 12-2, 12-7
  shell scripts, 12-76
  system data flow, 12-13
system administration
  concepts, 1-8, 1-9
  HP-UX cluster, 1-8, 1-9
  installing and updating HP-UX, 1-8, 1-9
  installing peripherals, 1-8, 1-9
  manual page references, 1-9
  prerequisites, 1-6
  related manuals, 1-6
  responsibilities, 1-7
  system security, 1-8, 1-9
system backups, 9-3
system clock
  problems you can cause by changing, 2-11
  setting, 2-8
system console
  moving using HP-UX commands, 5-32
  S800 file definition, 5-33

system date and time
  setting, 2-10, 2-11
system default printer, 10-5, 10-10, 10-25, 10-29, 10-50, 10-53
/system directory, 2-50
system files
  described, 6-95
  removing, 2-47
system parameters, A-1
system security, 1-8, 1-9

T

tape drive
  moving using HP-UX commands, 5-29
  moving using SAM, 5-8
  removing using HP-UX commands, 5-48
  removing using SAM, 5-20
tasks
  after installing HP-UX, 2-7
TERM
  setting, 2-27
terminal interface (SAM), 1-26
terminals
  adding entry to **/etc/inittab**, 2-23
  creating a **terminfo** file, 2-29
  moving using HP-UX commands, 5-31
  moving using SAM, 5-11
  non-HP, 2-27
  non-standard, 2-27
  removing using HP-UX commands, 5-49
  removing using SAM, 5-22
  /usr/lib/terminfo/*, 2-27
terminfo
  creating a new file, 2-29
  files under **/usr/lib/terminfo**, 2-27
time
  Canadian Standard Time, 2-9
  format, 2-10
  setting, 2-10
timeslice parameter, A-55
timezone parameter, A-56
time zone (**TZ**)
  examples, 2-9
  form of declaration, 2-8
  setting, 2-8
  tzset library routine, 2-8
total accounting records, 12-8, 12-22, 12-27, 12-52, 12-56
tset
  and **/etc/ttytype**, 2-20, 2-21
ttytype, 2-21

U

UID, 4-3, 4-18, 4-35, 4-42
umount, 6-58, 8-63
  -a, 6-60
unlockable_mem parameter, A-57
unmounting file systems
  using **umount**, 6-58
update
  using to add subsystems, 2-45
  using to load HP subsystems, 2-31
updating
  HP-UX, 2-4
  optional software, 2-5
updating HP-UX (from a previous release), 1-8, 1-9
user account information, 4-28, 4-55
user_ID, 4-3, 4-18, 4-35, 4-42
user IDs, 6-82
users
  account information, 4-28, 4-55
  adding, 2-14, 4-16, 4-34
  adding a group, 4-58
  adding users to groups, 4-72
  assessing needs, 2-1
  assigning IDs, 6-82

changing primary group membership, 4-67
deactivating account, 4-24, 4-51
defining passwords, 4-35
group membership, 4-6
login directory, 4-37
passwords, 4-34
reactivating account, 4-26, 4-53
removing, 4-19, 4-47
removing from groups, 4-75
startup program, 4-36, 4-38
survey, 2-2
utility
cron, 2-10, 2-11
elm, 2-15
mail, 2-15
mailx, 2-15
make, 2-11
news, 2-17
rmfn, 2-50–52
sendmail, 2-15
update, 2-4, 2-5, 2-31, 2-45
uucp, 11-2, 11-5

V

verbose log file (SAM), 1-45
vgcfgbackup, 8-103
routine use, 8-103
vgcfgrestore, 8-103
vgchange, 8-96
vgchange -a y, 8-97
vgcreate, 8-48
vgdisplay, 8-48, 8-52
vgexport, 8-101
vgextend, 8-51
vgimport, 8-101
vgscan, 8-105
-p to preview, 8-105
running to update **lvmtab**, 8-105
vgsync, 8-95
vipw editor, 4-5
virtual memory, B-1
volume group
major number, 8-47
naming convention, 8-22
volume group directory, 8-47
volume groups
activating, 8-96
activation with **vgchange -a y**, 8-97
adding disks to using SAM, 8-35
adding physical volumes, 8-51
creating, 8-47, 8-48
creating:specifying bad blocks, 8-47
creating using SAM, 8-36
creating volume group directory, 8-47
deactivating, 8-96
displaying information with **vgdisplay**, 8-52
extending using SAM, 8-35
extending:with **vgextend**, 8-51
group file, 8-47
making available with **vgchange**, 8-96
moving disk between, **vgimport**, **vgimport**, 8-100
organization, 8-15
reducing:with **vgreduce**, 8-51
removing physical volumes, 8-51

W

wall, 11-2, 11-4
windows, 2-18. *See also* X Windows
worksheet, computing swap space, B-3
write, 11-2, 11-4

X

X11-RUN fileset, 1-21
.x11start, 2-18
x11start, 2-18
X window interface (SAM), 1-21
X Windows
setting up, 2-18
/usr/bin/x11start, 2-18

.x11start, 2-18