HP-UX System and Network Administration I

H3064S E.00

HP Training

Student guide

invent

Copyright 2004 Hewlett-Packard Development Company, L.P.

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

This is an HP copyrighted work that may not be reproduced without the written permission of HP. You may not use these materials to deliver training to any person outside of your organization without the written permission of HP.

OSF, OSF1, OSF/Motif, Motif, and Open Software Foundation are trademarks of the Open Software Foundation in the U.S. and other countries.

UNIX® is a registered trademark of The Open Group.

X/Open is a trademark of X/Open Company Limited in the UK and other countries.

All other product names mentioned herein may be trademarks of their respective companies.

Printed in the USA

HP-UX System and Network Administration I
Student guide
June 2004

Contents

Module 1 — Introduction to HP-UX System Administration

1–1. SLIDE: Course Audience 1-2
1–2. SLIDE: Course Agenda 1-3
1–3. SLIDE: HP-UX System Administration Resources 1-4
1–4. SLIDE: HP-UX System Administration via SAM 1-6
1–5. SLIDE: Using the SAM GUI 1-8
1–6. SLIDE: Using the SAM TUI 1-10
1–7. LAB: SAM 1-12

Module 2 — Creating and Managing User Accounts

2–1. SLIDE: Users and Groups — Access to System and Data 2-2
2–2. SLIDE: What Defines a User Account? 2-3
2–3. SLIDE: The `/etc/passwd` File 2-4
2–4. SLIDE: The `/etc/shadow` File 2-7
2–5. SLIDE: The `/etc/group` File 2-10
2–6. SLIDE: Creating User Accounts with SAM 2-12
2–7. SLIDE: Creating User Templates in SAM 2-15
2–8. SLIDE: Deactivating and Removing Users with SAM 2-17
2–9. SLIDE: Managing Group Membership with SAM 2-20
2–10. SLIDE: Managing Users and Groups from the Command Line 2-21
2–11. SLIDE: Managing Passwords from the Command Line 2-23
2–12. LAB: Hands-On Adding Users 2-26

Module 3 — Customizing User Accounts

3–1. SLIDE: Why Customize a User Account? 3-2
3–2. SLIDE: Some Sample Customizations 3-3
3–3. SLIDE: Customizing ASCII Logins 3-6
3–4. SLIDE: Customizing CDE Logins 3-8
3–5. SLIDE: The `/etc/skel` Directory 3-10
3–6. LAB: Customizing User Accounts 3-12

Module 4 — Guided Tour of the HP-UX File Hierarchy

4–1. SLIDE: Introducing the File System Paradigm 4-2
4–2. SLIDE: System Directories 4-4
4–3. SLIDE: Application Directories 4-8
4–4. SLIDE: Commands to Help You Navigate 4-9
4–5. LAB: HP-UX File System Hierarchy 4-11

© 2004 Hewlett-Packard Development Company, L.P.

Module 5 — Configuring Hardware

5–1. SLIDE: HP-UX Processor Roadmap 5-2
5–2. SLIDE: HP-UX Operating System Roadmap 5-6
5–3. SLIDE: HP-UX Hardware Product Overview 5-8
5–4. SLIDE: Hardware Component Overview 5-10
5–5. SLIDE: CPU, Memory, and Cell Boards 5-11
5–6. SLIDE: SBAs, LBAs, and PCI Expansion Buses 5-13
5–7. SLIDE: Device Adapter Cards 5-15
5–8. SLIDE: Core I/O Cards 5-16
5–9. SLIDE: Hardware Addressing 5-17
5–10. SLIDE: HP-UX Hardware Addresses 5-19
5–11. SLIDE: Viewing HP-UX Hardware Addresses(**ioscan**) 5-22
5–12. SLIDE: Slot Addresses 5-25
5–13. SLIDE: Viewing Slot Addresses (**olrad** and **rad**) 5-28
5–14. SLIDE: Viewing Slot Addresses (**pdweb** and **sam**) 5-30
5–15. SLIDE: EFI Hardware Addresses 5-34
5–16. SLIDE: Viewing EFI Hardware Addresses (**ioscan**) 5-37
5–17. SLIDE: SCSI Device Concepts and Addresses 5-39
5–18. SLIDE: SCSI LUN Addressing 5-44
5–19. SLIDE: SCSI LUN Addressing with Alternate Paths 5-46
5–20. SLIDE: Node Partition (nPar) Concepts 5-49
5–21. SLIDE: Virtual Partition (vPar) Concepts 5-51
5–22. SLIDE: nPar and vPar Hardware Addresses 5-53
5–23. SLIDE: Installing Interface Cards without OLA/R 5-54
5–24. SLIDE: Installing Interface Cards with OLA/R (11i v1) 5-56
5–25. SLIDE: Installing Interface Cards with OLA/R (11i v2) 5-59
5–26. SLIDE: Installing New Devices 5-65
5–27. LAB: Viewing the Hardware Configuration with ioscan 5-67

Module 6 — Configuring Device Files

6–1. SLIDE: HP-UX Device File Overview 6-2
6–2. SLIDE: Listing Device Files with **ll** 6-4
6–3. SLIDE: Listing Device Files with **ioscan** 6-6
6–4. SLIDE: Listing Device Files with **lssf** 6-7
6–5. SLIDE: Device File Directories 6-9
6–6. SLIDE: Device File Naming Convention 6-10
6–7. SLIDE: Disk Device File Names 6-12
6–8. SLIDE: Tape Device File Names 6-14
6–9. SLIDE: Terminal and Modem Device File Names 6-16
6–10. SLIDE: Creating Device Files 6-18
6–11. SLIDE: Creating Device Files with Autoconfiguration 6-19
6–12. SLIDE: Creating Device Files with **insf** 6-21
6–13. SLIDE: Creating Device Files with SAM 6-23
6–14. SLIDE: Creating Device Files with **mksf** 6-25
6–15. SLIDE: Creating Device Files with **mknod** 6-27
6–16. SLIDE: Removing Device Files with **rmsf** 6-29
6–17. LAB: Device Files 6-30

© 2004 Hewlett-Packard Development Company, L.P.

Module 7 — Configuring Disk Devices

7–1. SLIDE: Disk Partitioning 7-2
7–2. SLIDE: Whole Disk Partitioning 7-4
7–3. SLIDE: LVM Disk Partitioning 7-6
7–4. SLIDE: LVM Device Files 7-8
7–5. SLIDE: LVM Extents 7-10
7–6. SLIDE: Creating Physical Volumes 7-12
7–7. SLIDE: Creating Volume Groups 7-15
7–8. SLIDE: Creating Logical Volumes 7-18
7–9. SLIDE: Disk Space Management Tool Comparison 7-22
7–10. LAB: Logical Volume Manager 7-25

Module 8 — File System Concepts

8–1. SLIDE: What Is a File System? 8-2
8–2. SLIDE: File System Types 8-4
8–3. SLIDE: What's in a File System? 8-6
8–4. SLIDE: Accessing a File System 8-8
8–5. SLIDE: HP-UX Hard Links 8-10
8–6. SLIDE: HP-UX Symbolic Links 8-12
8–7. SLIDE: HFS Structural Overview 8-14
8–8. CHALK TALK: HFS File System Updates 8-16
8–9. SLIDE: JFS Structural Overview 8-20
8–10. CHALK TALK: JFS File System Updates 8-22
8–11. SLIDE: JFS Blocks and Extents 8-28
8–12. SLIDE: File System Comparison 8-29

Module 9 — Creating and Mounting File Systems

9–1. SLIDE: File System Creation Overview 9-2
9–2. SLIDE: File System Creation Commands 9-3
9–3. SLIDE: Creating the File System 9-4
9–4. SLIDE: Mounting the File System 9-8
9–5. SLIDE: Unmounting the File System 9-12
9–6. SLIDE: Automatically Mounting File Systems 9-14
9–7. SLIDE: Mounting CDFS File Systems 9-16
9–8. SLIDE: Mounting LOFS File Systems 9-19
9–9. LAB: Creating File Systems 9-21

Module 10 — Maintaining File Systems

10–1. SLIDE: Monitoring File System Free Space 10-2
10–2. SLIDE: Reclaiming Wasted File System Space 10-4
10–3. SLIDE: Extending a Volume Group 10-7
10–4. SLIDE: Extending a Logical Volume 10-10
10–5. SLIDE: Extending a File System 10-12
10–6. SLIDE: Defragmenting a File System 10-14
10–7. SLIDE: Repairing a Damaged File System 10-17
10–8. LAB: File System Management 10-21

© 2004 Hewlett-Packard Development Company, L.P.

Module 11 — File System Backup

11–1. SLIDE: Why Back Up? 11-2
11–2. SLIDE: What Do You Back Up? 11-4
11–3. SLIDE: How Often Do You Back Up? 11-6
11–4. SLIDE: How Do You Perform the Backup? 11-9
11–5. SLIDE: Data Backup with **fbackup** 11-13
11–6. SLIDE: Data Recovery with **frecover** 11-17
11–7. SLIDE: Backing up the Boot Disk with **make_tape_recovery** 11-20
11–8. SLIDE: Creating the **make_tape_recovery** Tape 11-22
11–9. SLIDE: Using the **make_tape_recovery** Tape 11-24
11–10. TEXT PAGE: System Recovery Checklist 11-29
11–11. LAB: Backup and Recovery 11-30

Module 12 — Scheduling `cron` Jobs

12–1. SLIDE: The cron Daemon 12-2
12–2. SLIDE: Understanding the **cron** File 12-4
12–3. SLIDE: Managing cron Files with **crontab** 12-6
12–4. SLIDE: What Happens When a Job Is Scheduled? 12-7
12–5. LAB: **cron** 12-9

Module 13 — Managing Swap Space

13–1. SLIDE: HP-UX Memory Concepts 13-2
13–2. SLIDE: HP-UX Swap Concepts 13-4
13–3. SLIDE: HP-UX Swap Types 13-6
13–4. SLIDE: HP-UX Pseudoswap 13-8
13–5. SLIDE: Enabling Swap from the Command Line 13-10
13–6. SLIDE: Enabling Swap via **/etc/fstab** 13-13
13–7. SLIDE: Monitoring Swap Space 13-15
13–8. SLIDE: Guidelines for Selecting Device Swap Areas 13-18
13–9. SLIDE: Guidelines for Selecting File System Swap Areas 13-20
13–10. LAB: Swap 13-22

Module 14 — Managing Printers

14–1. SLIDE: LP Spooler Concepts 14-2
14–2. SLIDE: LP Printer Types 14-4
14–3. SLIDE: Adding a Local Printer via SAM 14-6
14–4. SLIDE: Adding a Remote Printer via SAM 14-9
14–5. SLIDE: Adding a Network Printer with **hppi** 14-12
14–6. SLIDE: LP Spooler Configuration Files 14-15
14–7. SLIDE: What Happens When a File Is Submitted with **lp**? 14-17
14–8. SLIDE: Managing Print Queues 14-19
14–9. SLIDE: Managing Printer Classes 14-22
14–10. SLIDE: Managing Priorities and Fences 14-23
14–11. SLIDE: Troubleshooting the Spooler 14-25
14–12. LAB: Hands-On Adding Printers 14-28

© 2004 Hewlett-Packard Development Company, L.P.

Module 15 — Accessing the System Console via the Management Processor

15–1. SLIDE: Introducing the MP 15-2
15–2. SLIDE: Viewing the MP Backplane 15-5
15–3. SLIDE: Connecting the MP Serial Port 15-7
15–4. SLIDE: Connecting the MP LAN Port 15-8
15–5. SLIDE: MP User Accounts 15-9
15–6. SLIDE: Accessing the MP LAN Interface 15-11
15–7. SLIDE: Navigating the MP Core Menus (Older) 15-12
15–8. SLIDE: Navigating the MP Core Menus (Newer) 15-14
15–9. SLIDE: Accessing the MP nPar Console Interface 15-16
15–10. SLIDE: Accessing the MP vPar Console Interface 15-18
15–11. SLIDE: Accessing the MP Virtual Front Panel 15-19
15–12. SLIDE: Accessing the MP Console Logs 15-21
15–13. SLIDE: Accessing the MP Chassis Logs 15-23
15–14. SLIDE: Accessing the MP Help Menu 15-26
15–15. SLIDE: Accessing the MP Command Menu 15-28
15–16. SLIDE: Managing the MP LAN Interface 15-30
15–17. SLIDE: Managing the MP Serial Port 15-33
15–18. SLIDE: Managing MP Logins 15-36
15–19. SLIDE: Configuring MP User Accounts 15-38
15–20. SLIDE: Rebooting via the MP 15-41
15–21. SLIDE: For Further Study 15-45
15–22. LAB: Interacting with the MP 15-46

Module 16 — Booting PA-RISC Systems

16–1. SLIDE: HP-UX Shutdown and Reboot 16-2
16–2. SLIDE: PA-RISC Boot Process Major Players 16-6
16–3. SLIDE: PA-RISC Boot Disk Structure 16-8
16–4. SLIDE: PA-RISC Boot Process Overview 16-10
16–5. SLIDE: Autoboot versus Manual Boot 16-12
16–6. SLIDE: Interacting with the BCH 16-13
16–6. SLIDE: Interacting with the ISL/IPL 16-17
16–7. LAB: Shutting Down and Rebooting Your System 16-21

© 2004 Hewlett-Packard Development Company, L.P.

Module 17 — Booting Integrity Systems

17–1. SLIDE: HP-UX Shutdown and Reboot 17-2
17–2. SLIDE: Integrity Boot Process Major Players 17-6
17–3. SLIDE: Integrity Boot Disk Partitions 17-8
17–4. SLIDE: Integrity Boot Disk System Partition Structure 17-10
17–5. SLIDE: Integrity Boot Disk OS Partition Structure 17-12
17–6. SLIDE: Integrity Boot Disk HPSP Structure 17-13
17–7. SLIDE: PARISC/Integrity Boot Process Comparison 17-14
17–8. SLIDE: Autoboot versus Manual Boot 17-16
17–9. SLIDE: Interacting with the EFI Boot Manager 17-18
17–10. SLIDE: Interacting with the EFI Shell 17-19
17–11. SLIDE: Viewing System Information via the EFI Shell 17-22
17–12. SLIDE: Managing EFI File Systems via the EFI 17-26
17–13. SLIDE: Managing the EFI Boot Menu via the EFI 17-30
17–14. SLIDE: Booting an Arbitrary Device via the EFI Shell 17-32
17–15. SLIDE: Interacting with the hpux.efi OS Loader 17-34
17–16. SLIDE: For Further Study 17-36
17–17. LAB: Booting Integrity Servers 17-37

Module 18 — Configuring the HP-UX 11i v1 Kernel

18–1. SLIDE: Why Reconfigure the Kernel? 18-2
18–2. SLIDE: Static Kernel Modules 18-4
18–3. SLIDE: Dynamically Loadable Kernel Modules 18-5
18–4. SLIDE: Configuring Kernel Drivers and Subsystems 18-6
18–5. SLIDE: Configuring Kernel Tunable Parameters 18-9
18–6. SLIDE: Moving the New Kernel into Place 18-11
18–7. SLIDE: What If the New Kernel Won't Boot? 18-13
18–8. LAB: Kernel Configuration 18-15

Module 19 — Configuring the HP-UX 11i v2 Kernel

19–1. SLIDE: Why Reconfigure the Kernel? 19-2
19–2. SLIDE: Kernel Configuration Command Overview 19-4
19–3. SLIDE: **kcweb** Overview 19-5
19–4. SLIDE: **kcweb** Architecture 19-6
19–5. SLIDE: Using **kcweb** via a Local Browser 19-7
19–6. SLIDE: Using **kcweb** via a Remote Browser 19-9
19–7. SLIDE: Kernel Modules 19-11
19–8. SLIDE: Kernel Module States 19-13
19–9. SLIDE: Kernel Module State Changes 19-16
19–10. SLIDE: Viewing Module States via **kcweb** 19-17
19–11. SLIDE: Managing Module States via **kcweb** 19-19
19–12. SLIDE: Kernel Tunables 19-22
19–13. SLIDE: Types of Tunables 19-23
19–14. SLIDE: Viewing Tunables via **kcweb** 19-25
19–15. SLIDE: Managing Tunables via **kcweb** 19-27
19–16. SLIDE: Kernel Configurations 19-32
19–17. SLIDE: Special Kernel Configurations 19-33
19–18. SLIDE: Kernel Configuration Command Overview 19-36
19–19. SLIDE: Example: Creating a New Configuration 19-38
19–20. SLIDE: Example: Copying a Config to Another Host 19-40
19–21. SLIDE: Example: Loading a Configuration 19-42

© 2004 Hewlett-Packard Development Company, L.P.

19–22. SLIDE: Kernel Troubleshooting Overview 19-44
19–23. SLIDE: Viewing the Kernel Change Log 19-45
19–24. SLIDE: Viewing the **kcweb** Logs 19-47
19–25. SLIDE: Booting from an Alternate Kernel 19-48
19–26. SLIDE: Booting to Tunable Maintenance Mode 19-50
19–27. SLIDE: For Further Study 19-51
19–28. LAB: Configuring the HP-UX 11i v2 Kernel 19-52

Module 20 — Managing Software with SD-UX

20–1. SLIDE: Introducing SD-UX 20-2
20–2. SLIDE: SD-UX Software Structure 20-4
20–3. SLIDE: SD-UX Software Depots 20-6
20–4. SLIDE: SD-UX IPD 20-8
20–5. SLIDE: SD-UX Daemons and Agents 20-9
20–6. SLIDE: Listing Software 20-10
20–7. SLIDE: Installing and Updating Software 20-12
20–8. SLIDE: Removing Software 20-16
20–9. LAB: Hands-On, Using the Software Distributor 20-18

Module 21 — Managing Patches with SD-UX

21–1. SLIDE: Why Install Patches? 21-2
21–2. SLIDE: Patch Naming Convention 21-4
21–3. SLIDE: Patch Supersession 21-5
21–4. SLIDE: Patch Ratings 21-6
21–5. SLIDE: Patch Sources 21-7
21–6. SLIDE: Patch Tools 21-10
21–7. SLIDE: Downloading Patches from the ITRC (1 of 4) 21-12
21–8. SLIDE: Downloading Patches from the ITRC (2 of 4) 21-14
21–9. SLIDE: Downloading Patches from the ITRC (3 of 4) 21-16
21–10. SLIDE: Downloading Patches from the ITRC (4 of 4) 21-17
21–11. SLIDE: Installing a Single Patch from the ITRC 21-19
21–12. SLIDE: Installing Multiple Patches from the ITRC 21-24
21–13. SLIDE: Installing Patches from CD 21-27
21–14. SLIDE: Installing Patches from Tape 21-30
21–15. SLIDE: Installing Patches from a Depot Server 21-32
21–16. SLIDE: Listing Patches with **swlist** 21-34
21–17. SLIDE: Removing Patches with **swremove** 21-36
21–18. LAB: Installing Patches with SD-UX 21-38

© 2004 Hewlett-Packard Development Company, L.P.

Module 22 — Installing the OS with Ignite-UX

22–1. SLIDE: Why Install? 22-2
22–2. SLIDE: Install Sources 22-4
22–3. SLIDE: Planning the Install 22-6
22–4. SLIDE: Choosing an Operating Environment 22-8
22–5. SLIDE: Choosing an Install-Time Security Bundle 22-11
22–6. SLIDE: Locating the Source Media 22-13
22–7. SLIDE: Initiating a PARISC Install 22-15
22–8. SLIDE: Initiating an Integrity Install 22-17
22–9. SLIDE: Navigating the Ignite Menus 22-19
22–10. SLIDE: Verifying the Install 22-24
22–11. SLIDE: Completing Post-Install Configuration Tasks 22-26
22–12. SLIDE: For Further Study 22-29
22–13. LAB: Installing HP-UX 22-30

Appendix A — Connecting to a Network

A–1. SLIDE: Setting an IP Address and Subnet Mask A-2
A–2. SLIDE: Setting a Default Route A-4
A–3. SLIDE: Setting a System Hostname A-6
A–4. SLIDE: Resolving Hostnames to IP Addresses A-8
A–5. SLIDE: Configuring /etc/hosts A-9
A–6. SLIDE: Configuring a DNS Client A-11
A–7. SLIDE: Configuring an NIS Client A-13
A–8. SLIDE: Choosing a Look-Up Service A-15
A–9. SLIDE: Troubleshooting Tools A-18
A–10. LAB: Connecting to the Network A-20

Solutions

© 2004 Hewlett-Packard Development Company, L.P.

Module 1 — Introduction to HP-UX System Administration

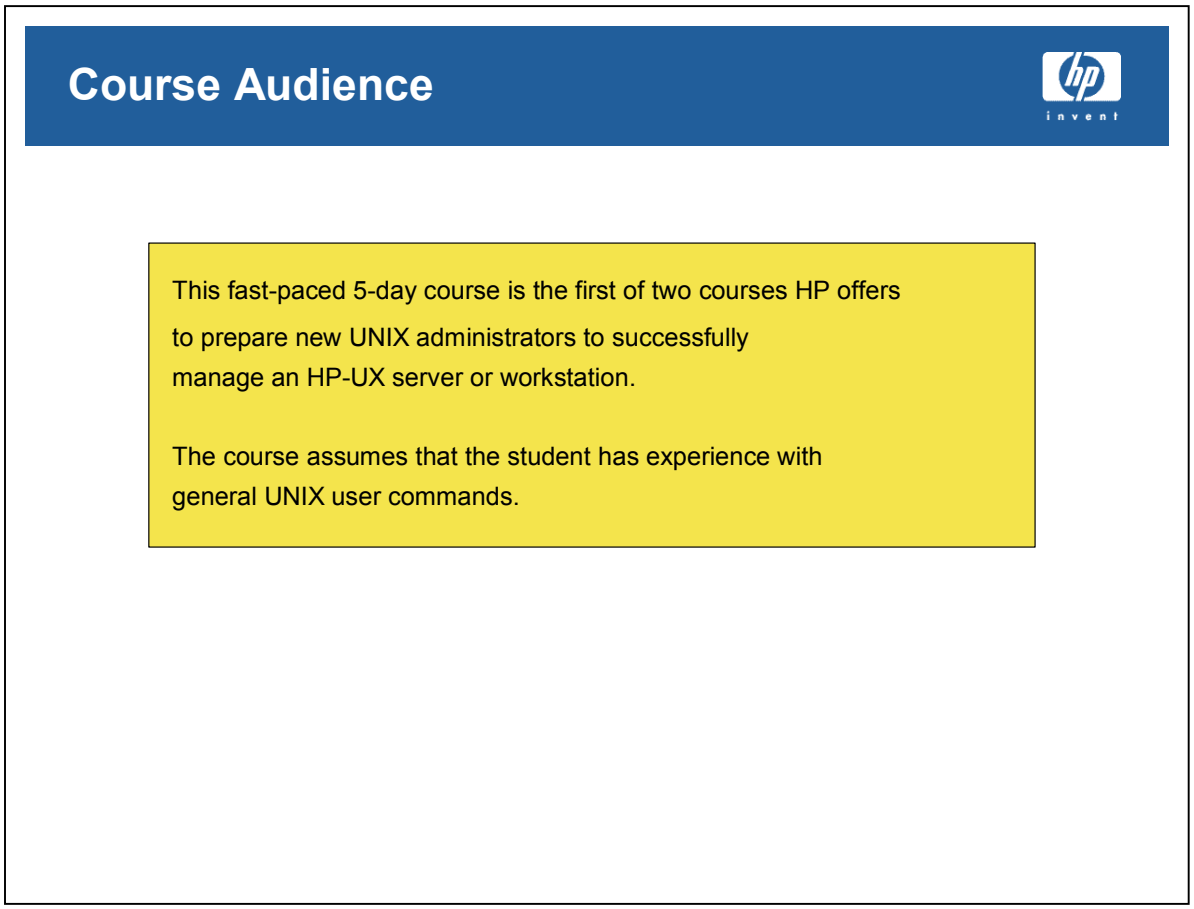
Objectives

Upon completion of this module, you will be able to do the following:

- Describe the target audience for this course.
- List the topics covered in this course.
- List some common reference sources used by HP-UX system administrators.
- Perform basic system administration tasks using the SAM GUI and TUI interfaces.

© 2004 Hewlett-Packard Development Company, L.P.

1–1. SLIDE: Course Audience



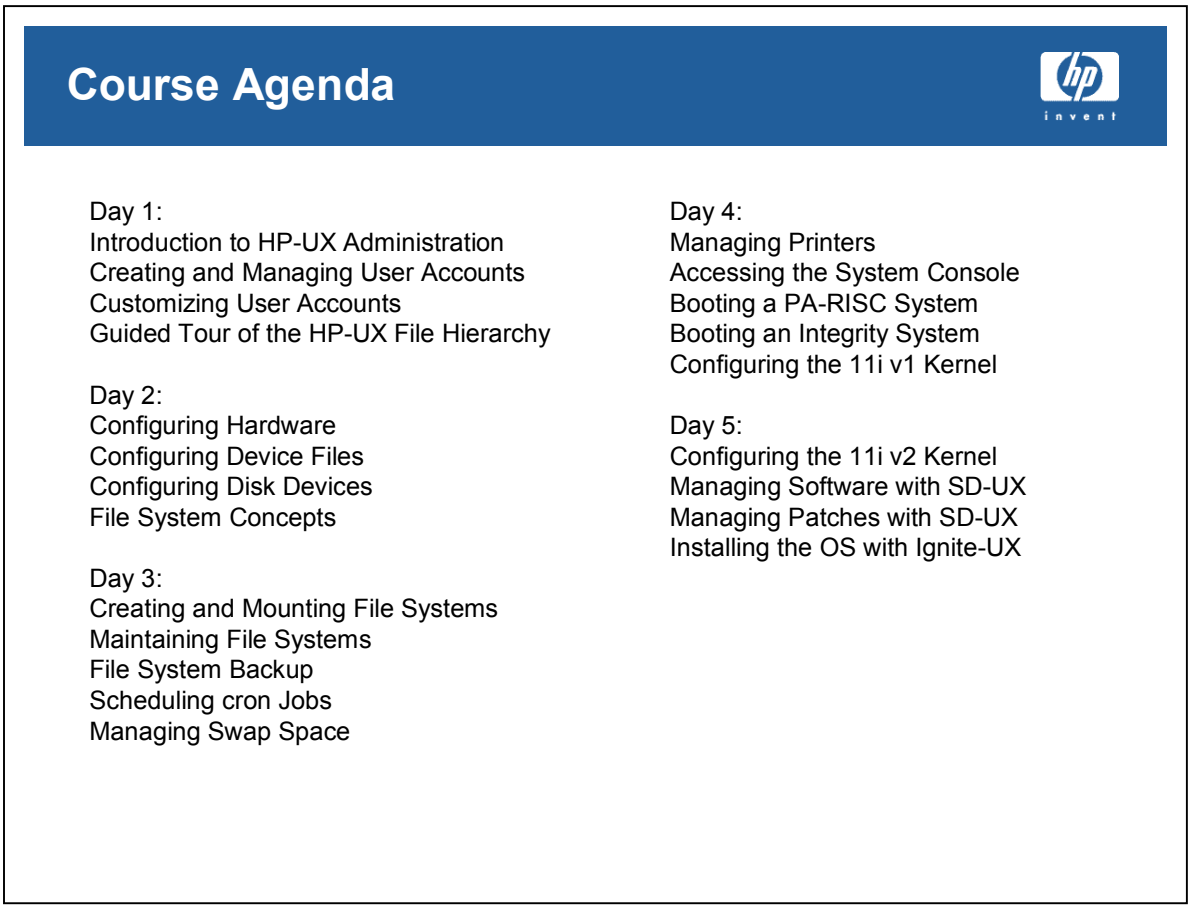
Student Notes

This fast-paced 5-day course is the first of two courses HP offers to prepare new UNIX administrators to successfully manage an HP-UX server or workstation.

The course assumes that the student has experience with general UNIX user commands.

© 2004 Hewlett-Packard Development Company, L.P.

1–2. SLIDE: Course Agenda



Student Notes

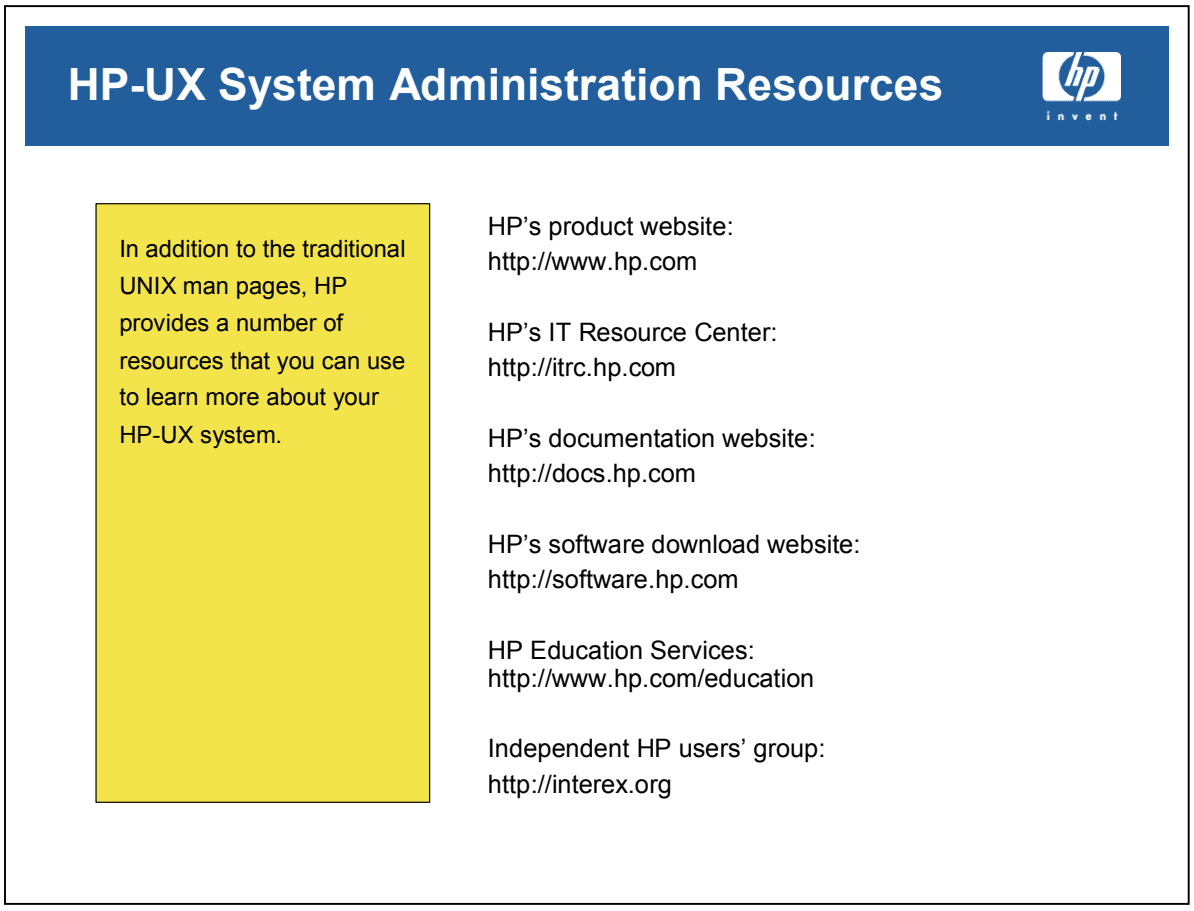
HP-UX System Administrators often serve a number of roles – from configuring peripherals, to managing user accounts, to installing software and patches. Over the span of approximately twenty-two modules and five days, this course covers the core skills required by all HP-UX system administrators.

HP recommends that students attend the follow-on to this course, HP-UX System and Network Administration 2 (H3065S), to complete the course sequence for new HP-UX administrators.

HP Education also offers courses covering numerous advanced HP-UX system and network administration topics. See our website, **`http://www.hp.com`** for more information.

© 2004 Hewlett-Packard Development Company, L.P.

1–3. SLIDE: HP-UX System Administration Resources



Student Notes

Beyond this course, there are a wealth of resources available to assist new HP-UX system administrators.

`http://www.hp.com` — The HP' corporate/product website describes all of HP's current hardware, software, and service offerings.

`http://itrc.hp.com` — HP's IT Resource Center provides a wealth of cookbooks, white papers, FAQ lists, patches, user forums, and an online response center that you can use to research HP-UX features and problems. The ITRC user forums are particularly helpful. Portions of the ITRC content are only available to customers with support contracts.

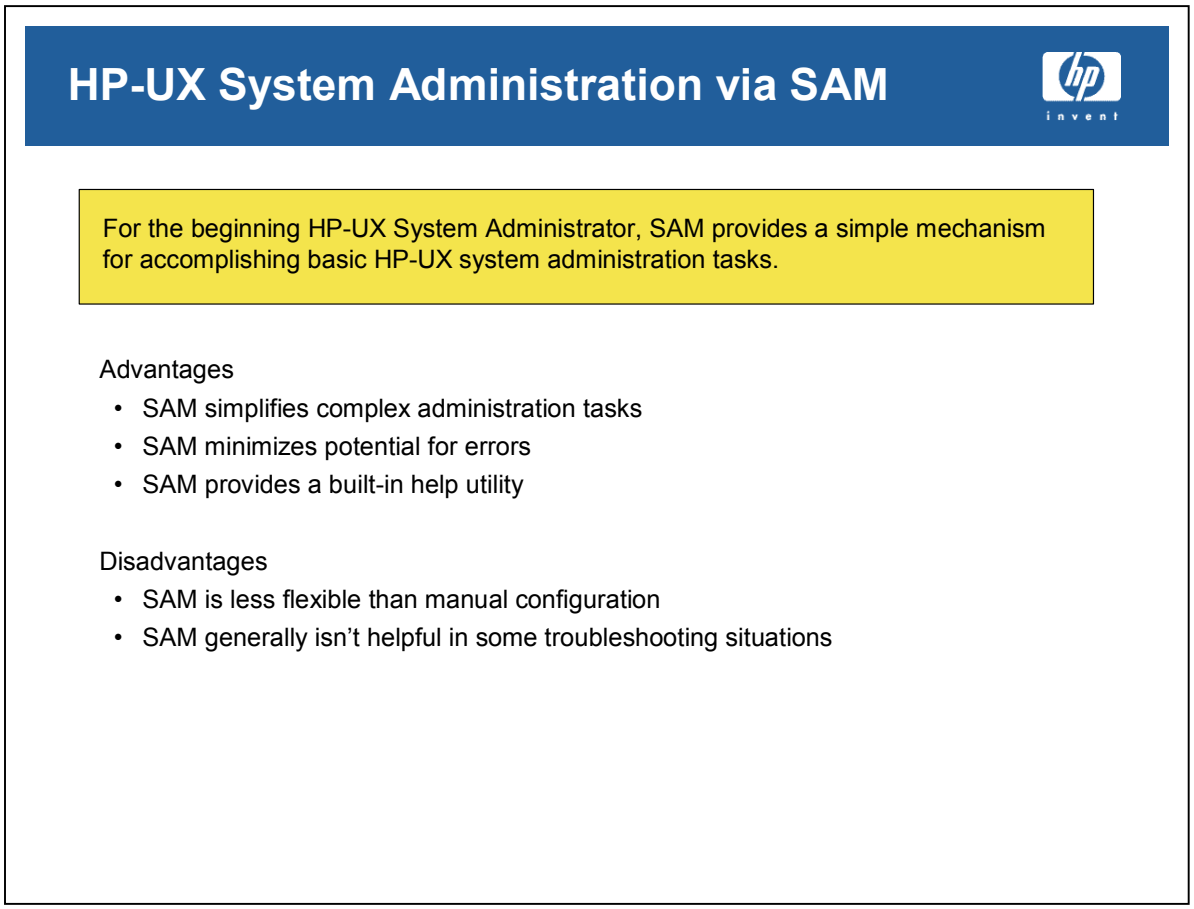
`http://docs.hp.com` — HP's documentation website provides an online, searchable library containing all of HP's HP-UX manuals. If your site doesn't have Internet access,

© 2004 Hewlett-Packard Development Company, L.P.

	you can purchase a CDROM version of the HP-UX documentation called HP Instant Information.
`http://software.hp.com`	Visit HP's software download website to download and purchase HP-UX software products and updates.
`http://www.hp.com/education`	HP Education Services offers a wide variety of courses on HP-UX and other HP products. Visit our website regularly to stay abreast of the latest course offerings.
`http://interex.org`	Consider joining the independent HP users' group, Interex. Interex represents users' interests to HP, publishes a variety of publications, and sponsors the annual HP World users' conference to provide users an opportunity to learn from HP-UX experts.

© 2004 Hewlett-Packard Development Company, L.P.

1–4. SLIDE: HP-UX System Administration via SAM



Student Notes

Administrators who are new to HP-UX often find that the System Administration Manager (SAM) is an easy way to accomplish many of the most common system administration tasks. SAM is a menu-driven tool designed to perform typical system administration tasks without using the underlying HP-UX commands.

Advantages

- SAM simplifies complex administration tasks
- SAM minimizes potential for errors
- SAM provides a built-in help utility

Disadvantages

- SAM is less flexible than manual configuration
- SAM generally isn't helpful in some troubleshooting situations

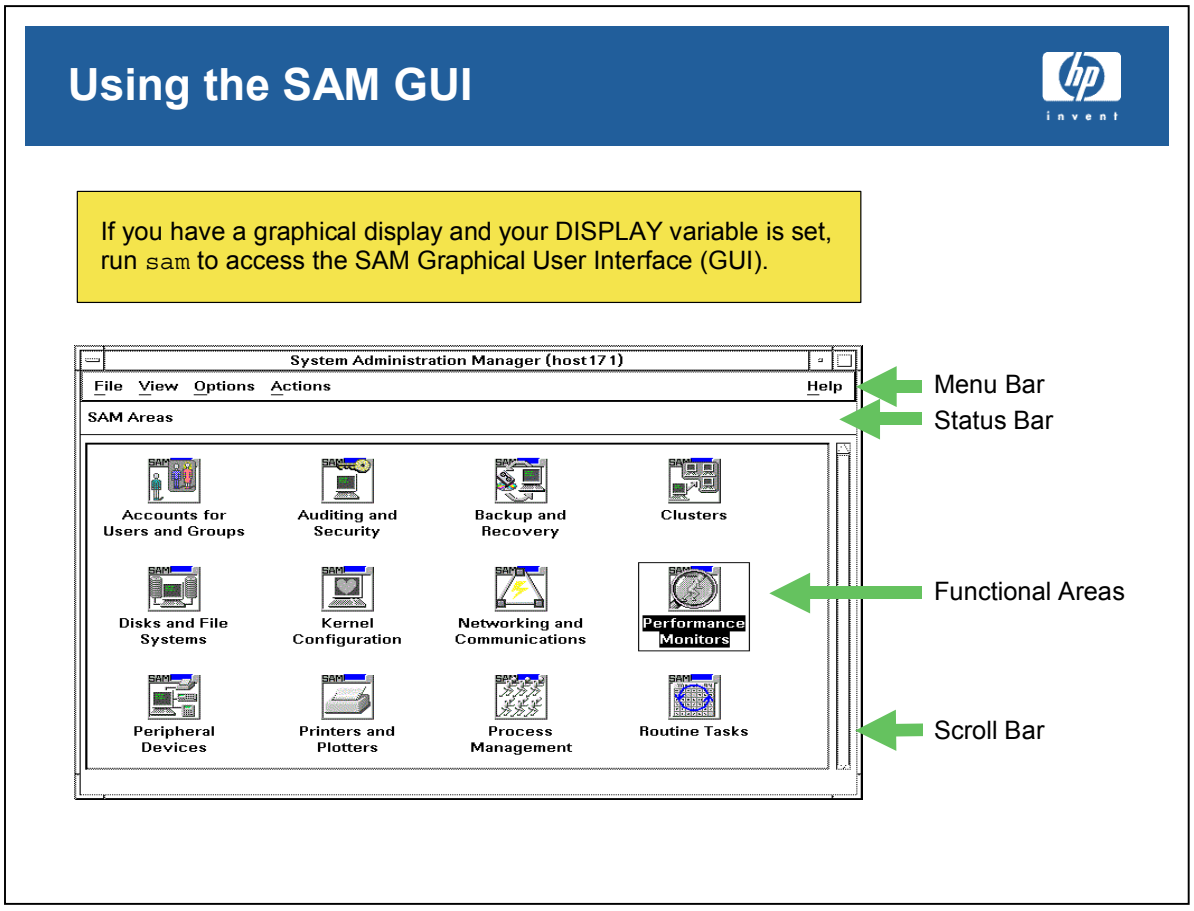
© 2004 Hewlett-Packard Development Company, L.P.

While SAM may make it easier to perform certain tasks, the HP-UX commands can be more flexible and more powerful. As a System Administrator, it is very important that you understand the "manual" way of doing things so that when you need to do something that SAM cannot do, you have the skills. The following items suggest a strategy for using SAM:

Remember, administering a system requires problem-solving skills. The more you understand about your system, the better equipped you will be to solve the problems.

© 2004 Hewlett-Packard Development Company, L.P.

1–5. SLIDE: Using the SAM GUI



Student Notes

Administrators who have access to a terminal running the X Window system can use SAM's Graphical User Interface (GUI).

To use SAM on the X Window System, the DISPLAY environment variable must be set correctly to reflect the display on which you want SAM to appear. The value of the **DISPLAY** variable should be ***hostname*** **:0.0**, where hostname is the name returned when you type the **/usr/bin/hostname** command. The **DISPLAY** environment variable is typically set and exported in the user's environment file at login.

To view your current environment variable values, type **env**. This is how to set the correct values for the **DISPLAY** variable, depending on your shell:

Shell	Environment Variable	Environment File
POSIX, Korn or Bourne shells	**export DISPLAY=** *hostname* **:0.0**	**.profile** or **.dtprofile**
C shell	**export DISPLAY=** *hostname* **:0.0**	**.login**

© 2004 Hewlett-Packard Development Company, L.P.

After your DISPLAY variable has been set, you can run SAM simply by typing: **`sam`**

The opening screen encountered when running SAM's GUI contains several components:

- The Menu Bar at the top of the SAM screen provides a series of pull-down menus. The pull-down menus provided vary from screen to screen within SAM. Single-click on any menu to view the available menu options.

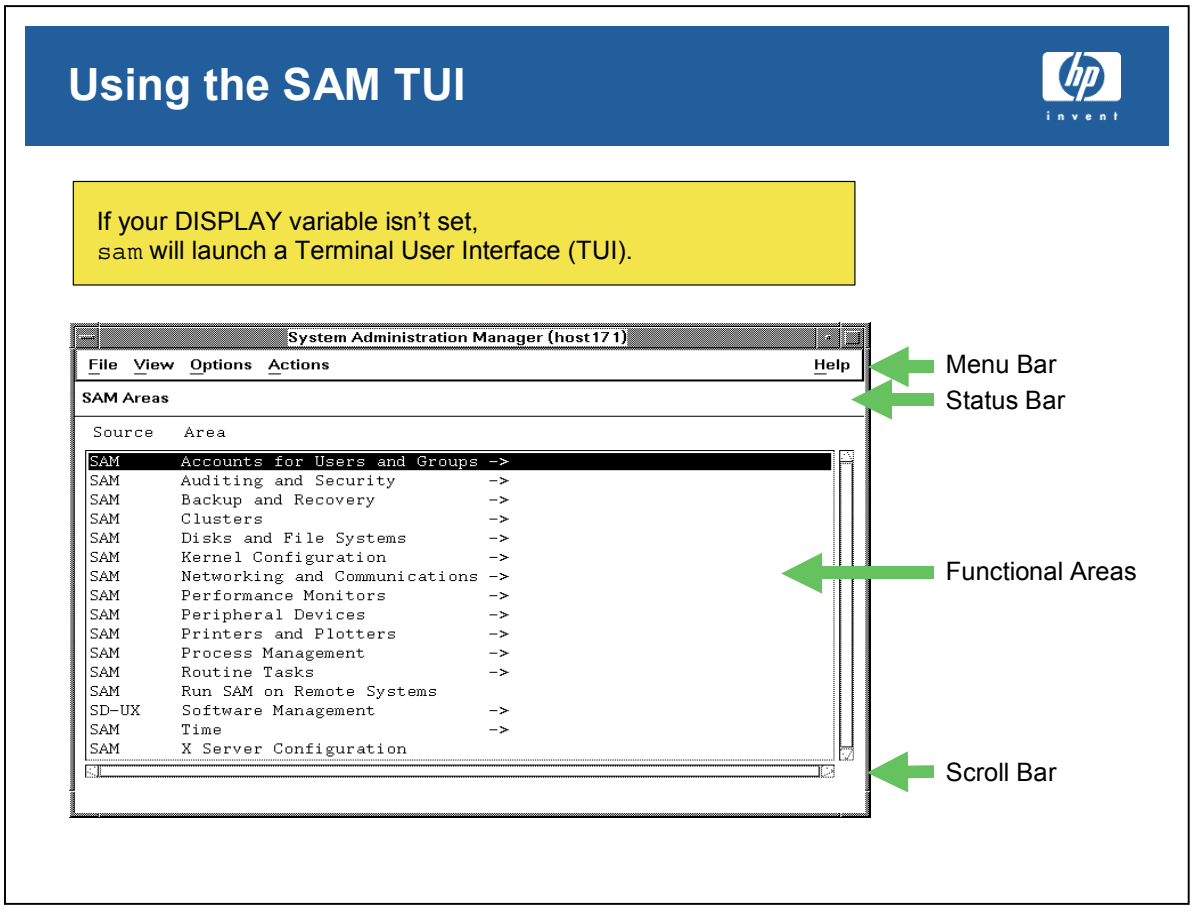
- Icons in the middle portion of your SAM window list the available functional areas within SAM. SAM provides tools for most common system administration tasks. To manage user accounts, double-click on the **`Accounts for Users and Groups`** functional area icon. To add a peripheral to your system, double-click the **`Peripheral Devices`** functional area icon.

- The Status Bar at the top of each SAM window identifies your current functional area. In some functional areas, SAM will provide additional status and configuration information in the Status Bar.

Some SAM windows may also provide horizontal and vertical scroll bars.

© 2004 Hewlett-Packard Development Company, L.P.

1–6. SLIDE: Using the SAM TUI



Student Notes

SAM has a special Terminal User Interface (TUI) for use on character terminals. You use specific keys (or combinations of keys) to move from one part of the screen to another and to move among screens. The structure of SAM is identical for both character and X Window system interfaces; instead of the point-and-click approach, you will use the keyboard to control SAM's actions.

The slide shows how the Control window looks on a character terminal.

Use the up and down arrow keys to highlight different items on the list.

Use Tab and Shift + Tab to move forward and backward to different control buttons.

Activate a highlighted control button (execute that control button function) by pressing the Spacebar. You can also use the mnemonic that is underlined on the control button. For example, notice that the **O** on the **Options** is underlined. You can activate the **Options** control button by typing **o** on the keyboard.

© 2004 Hewlett-Packard Development Company, L.P.

To turn a checkbox on or off, use the Tab key to move to the checkbox, then press the Spacebar. An **X** in the checkbox indicates the "on" state. In the "off" state, the checkbox is empty. The space bar toggles the state of the checkbox.

To move to the Menu Bar:

1. Press **F4** (or the Tab key).

2. Use the left arrow and right arrow keys to move to the menu you wish to open, then press the Spacebar.

3. Use the up arrow and the down arrow keys to move the highlight to the desired menu item, then press Spacebar or type the mnemonic.

In addition, many terminals have function keys that can be used to perform some of these maneuvers.

What the Function Keys Mean

Help	Context-sensitive help
Alt	Modifier key (press first, but don't hold down)
Select	Select or deselect the item, press a button, etc.
Menubar	Activate or deactivate the menu bar
OK	Accelerator for the **OK** button in a dialog
Apply	Accelerator for the Apply button in a dialog
Shell	Suspends the user interface and provides a shell
Cancel	Accelerator for the Cancel button in a dialog
Close	Accelerator for the Close button in a dialog
Exit	Accelerator for Exit in an object-list screen or control window

© 2004 Hewlett-Packard Development Company, L.P.

1–7. LAB: SAM

Part 1: Exploring SAM's Functional Areas

Our goal in this first portion of the lab exercise is to become more familiar with the SAM interface, and to explore some of SAM's functional areas. Typically when using SAM you will use the following procedure:

1. Run SAM.
2. Navigate functional areas to the proper object list.
3. Sort and filter the object list to your liking.
4. Select an object or objects with the mouse or space bar.
5. Choose an action from the **`Actions`** menu to perform on the selected object(s).

Let's try it!

1. Log onto your workstation or server and launch the SAM GUI or TUI. The first screen in SAM lists a number of functional areas.

2. Which functional area would you choose to view a list of user accounts on your system? Select the appropriate area, and determine the number of accounts on your system. (Note: In some cases, a functional area may contain one or more sub-areas. You may have to drill down through several menus to get to the information you are looking for.) Explore some of the other SAM functional areas to find answers to the questions that follow.

3. Without selecting a user from the object list, pull down the **`Actions`** menu. Can any actions be performed here without selecting a user?

© 2004 Hewlett-Packard Development Company, L.P.

4. The **Add** action allows you to create a new user account. Select **Actions –>Add**. This should bring up a dialog screen requesting information about the new account. Set **operator** as the user name for the new account, then click the **OK** button so SAM will take the defaults for the rest of the screen. After SAM prompts you for a password for the new account, check the object list to see what happened.

5. Choose **Actions –> Add** again and enter user name **dbmgr**. Instead of pressing **OK**, try the **Apply** button. What is the difference between the **OK** and **Apply** buttons in a dialog box? Can you think of a situation where **Apply** might be a more efficient choice than **OK**?

6. Now try performing an action on an existing object. Using the mouse (GUI) or space bar (TUI), select **user1** from the object list. Now pull down the Actions menu. Now that you have selected a user, there should be many more actions to choose from. Choose **Modify user's password** and answer the questions that follow.

7. In some cases, the dialog boxes that result from SAM actions may reference terms or concepts you haven't encountered before. Select **user1** from the object list again, then choose **Actions –> Modify**. This should open the **Modify a User** dialog box. The dialog box allows you to change the user's Primary group and Start-up Program. For more information about terms and concepts you encounter in any SAM window, look for a Help button or Help menu on the menu bar. Click on the **Help** button at the bottom of the dialog box window. Skim the text, and try clicking on a couple of the underlined phrases. (In the TUI, select one of the "See also" topics instead.) Any underlined phrase in a SAM help window is a link to additional information. When you have finished with the Help window, click **Close** to return to the **Modify a user** dialog box. We didn't really want to modify **user1**'s account after all, so click on **Cancel**.

© 2004 Hewlett-Packard Development Company, L.P.

8. Occasionally, SAM will complain about an action you request. Now select the root user account. Choose **`Actions -> Remove`**. Can you explain SAM's response to this action? Do you consider this to be a *feature* or a *bug*?

9. In other situations, SAM simply provides a warning message when you attempt a risky action. Try another experiment: From the Users screen, select the root user account, then choose **`Actions -> Modify`**. You will get a warning box, to which the most prudent response at this point is **`No`**; do not modify root's account.

10. SAM keeps a record of all actions that you perform. You can view this log via the SAM log viewer. Access the SAM log by selecting **`View SAM Log`** from the **`Options`** menu.

11. Experiment with some of the buttons in the dialog box that appears. Which Message level provides the most detail? When does your SAM log begin?

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Using the Restricted SAM Builder

By default, SAM may only be run by user root. Administrators in large shops often have the luxury of a staff of operators to assist with system administration tasks. So how can we allow multiple operators, and perhaps even regular users, to access some of the functionality in SAM? Sharing the root password probably isn't the best solution. The **`restricted SAM builder`** makes it possible to grant non-root users access to selected functional areas in SAM. Our goal in the exercise that follows is to allow the operator user that you created earlier in the lab to manage processes via SAM and run automated backups, while denying them access to all other areas of SAM.

1. Choose **`File -> Exit`** to exit your current SAM session. Then type **`sam -r`** to start the restricted SAM builder. (Note: you must be logged in as root to run the restricted SAM builder.)

2. The SAM builder should display a list of user names and templates. Just accept the defaults for now and click **`OK`**.

3. Next, you will see a window listing each of the SAM functional areas. Note that some functional areas are marked to be enabled (indicated by a green icon), some are disabled (indicated by a red icon), and some are partially enabled (indicated by a yellow icon). In the TUI interface, the status of a functional area is displayed in text form (enabled, disabled, partially enabled). Start by choosing **`Actions -> Disable all`**. What happens to the functional area icons?

4. Next, select the Process management icon with a single-click (GUI) or the space bar (TUI). Choose **`Actions -> Enable`**. What happens to the Process Management icon?

© 2004 Hewlett-Packard Development Company, L.P.

5. Next, we need to enable automated backups. Go to the **`Backup and recovery`** functional area, and select the Automated backup icon. Again go to the Actions menu and choose: **`Actions -> Enable`**. Go back up to the main functional area window. What happened to the Backup and recovery icon? Why is this icon marked as partially enabled?

6. You can enable and disable as many functional areas as you wish in the restricted SAM builder. Once you have enabled all the desired icons, choose **`Actions -> Save privileges`**.

7. The "Save privileges screen defines which user(s) will have access to the functional areas you have selected. Select the **`operator`** user from the list, and click **`OK`**. (If you had multiple operators, you could select multiple user names from the list while holding the shift key. If the names are not together in the list, hold the control key and select the names.) SAM saves the privileges you selected, then returns you to the icon window. Exit the restricted SAM builder, then log off your workstation or server.

8. Log in as operator and try running SAM by typing:**`/usr/sbin/sam.`** Which functional areas are available for use by your operator user?

© 2004 Hewlett-Packard Development Company, L.P.

9. Exit SAM, then try running SAM again by just typing **`sam`**. You will probably get an error message, **`sh: sam: not found`**. Try typing **`/usr/sbin/sam`**. Can you explain why SAM wouldn't start in the first case, but would start in the second?

10. Log out as user operator, then log back in again as user root.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Customizing SAM (Optional)

Although SAM has a fair amount of built-in functionality, you may have some database utilities or custom shell scripts that you would like to be able to execute directly from SAM. SAM allows you to do this. In the exercise that follows, you will create a custom SAM icon that could be used. In the exercise that follows, you will create a custom SAM icon that could be used to execute the **/usr/sbin/whodo** command to view a description of who is doing what on your system.

1. From the functional area launcher, choose `Actions -> Add Custom Application`.

2. Choose a name for your icon in the Label field.

3. Enter the full path name for the **whodo** command, **/usr/sbin/whodo**, in the **Command** field.

4. **whodo** is not a graphical utility, so choose **Terminal Environments** as the user interface.

5. Skip the optional fields and click **OK**.

© 2004 Hewlett-Packard Development Company, L.P.

6. Try your new icon.

7. Navigate to SAM's **`Time`** functional area. Create another custom SAM application that automatically runs the **`/usr/bin/cal`** program for you to display a calendar of the current month.

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 2 — Creating and Managing User Accounts

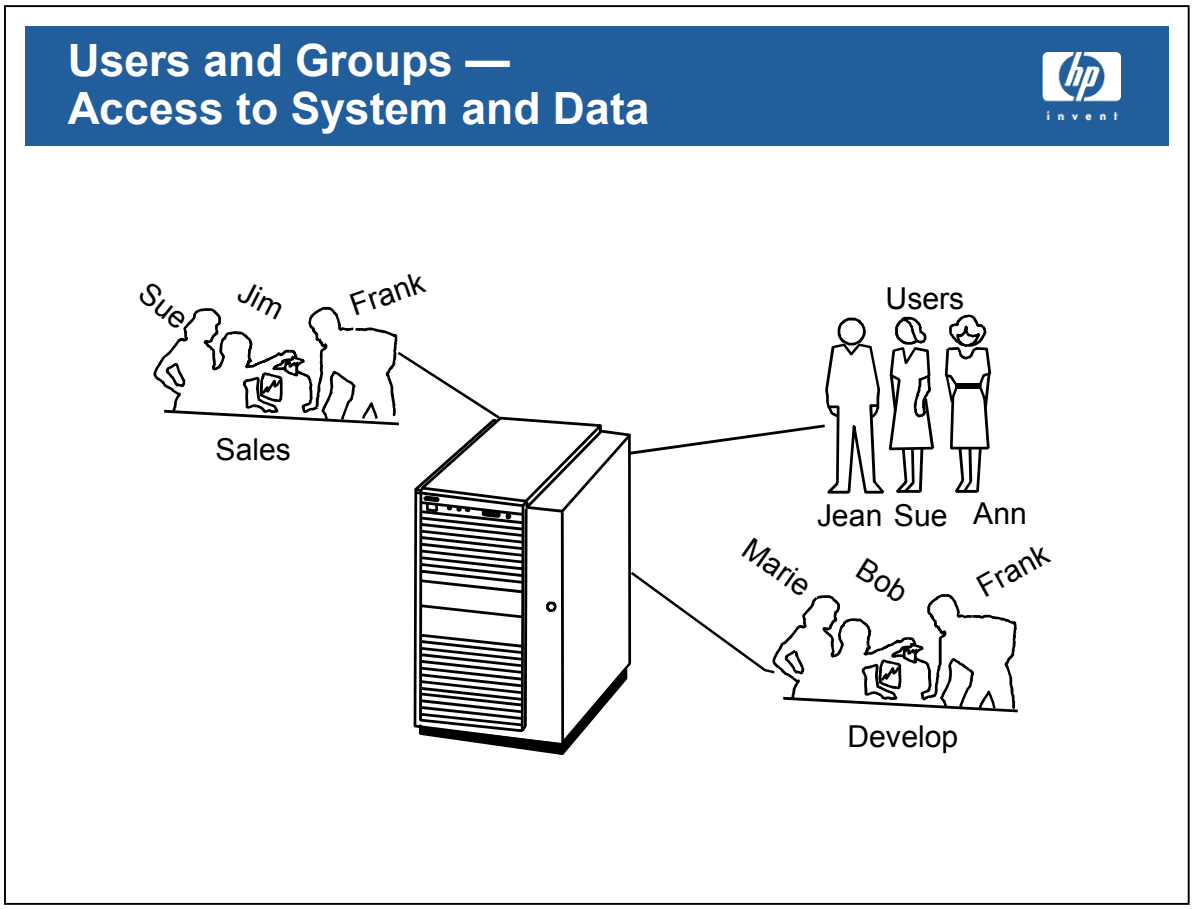
Objectives

Upon completion of this module, you will be able to do the following:

- List the minimum requirements for a user account.
- Identify each field in the **`/etc/passwd`** file.
- Identify each field in the **`/etc/shadow`** file.
- Identify each field in the **`/etc/group`** file.
- Create, modify, and delete a user account from SAM or the command line.
- Convert to a shadowed password system.
- Deactivate and reactivate a user account from SAM or the command line.

© 2004 Hewlett-Packard Development Company, L.P.

2–1. SLIDE: Users and Groups — Access to System and Data



Student Notes

In order to gain access to the system and its resources, users are required to log in. By controlling access to your system, you can prevent unauthorized users from running programs that consume resources, as well as control access to the data stored on your system.

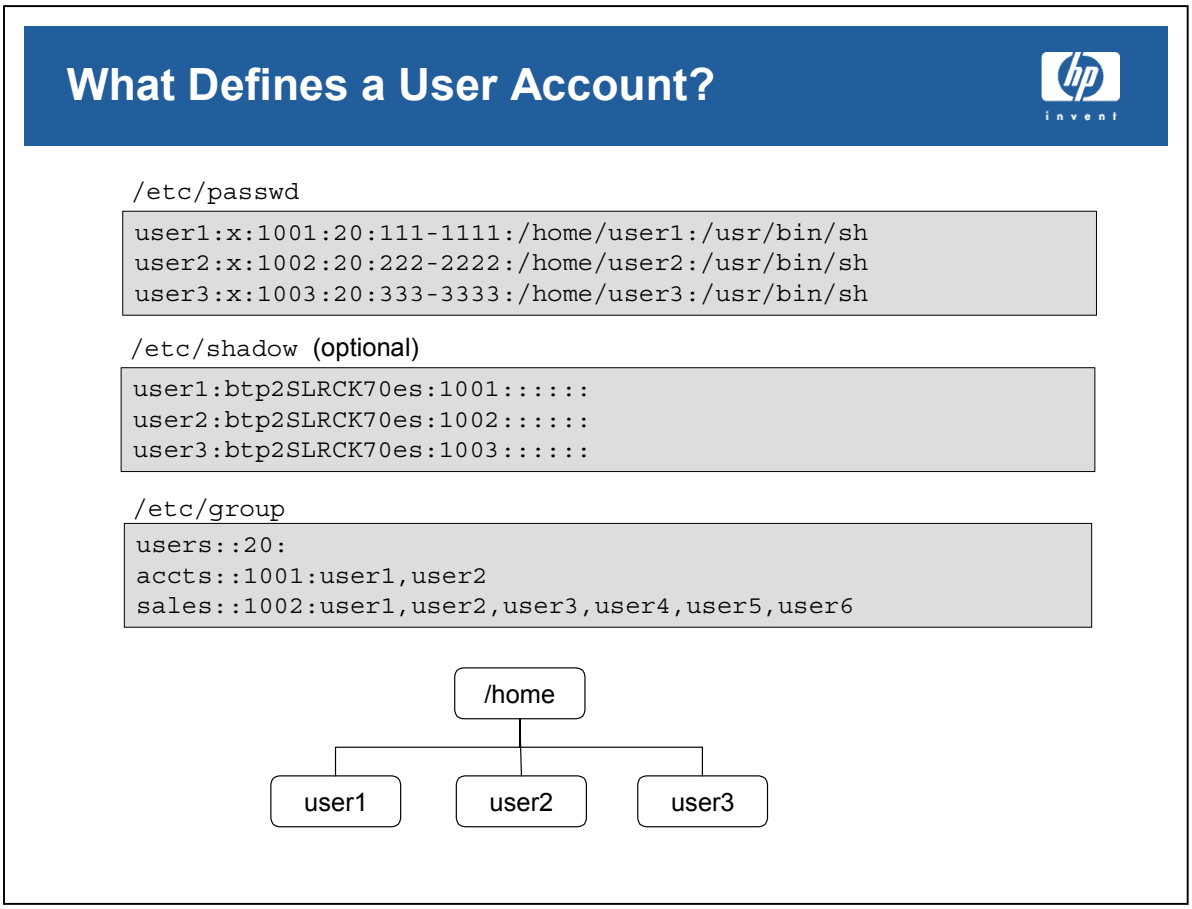
Users on the system are assigned to one or more groups. Files can typically be accessed by members of the owner's group, yet they are protected from access by users outside the owner's group. Each user is granted access to files and directories based on the groups to which the user belongs.

You choose a unique user name (or login name) for each person who will be accessing the system. You also choose group names and decide which user names will be assigned to which groups. You can perform these tasks either by using HP-UX commands, or by using SAM. SAM automatically updates system files and calls appropriate commands for you.

Users can determine the userid and the group membership you have assigned for them by typing **`/usr/bin/id`** without any arguments.

© 2004 Hewlett-Packard Development Company, L.P.

2–2. SLIDE: What Defines a User Account?



Student Notes

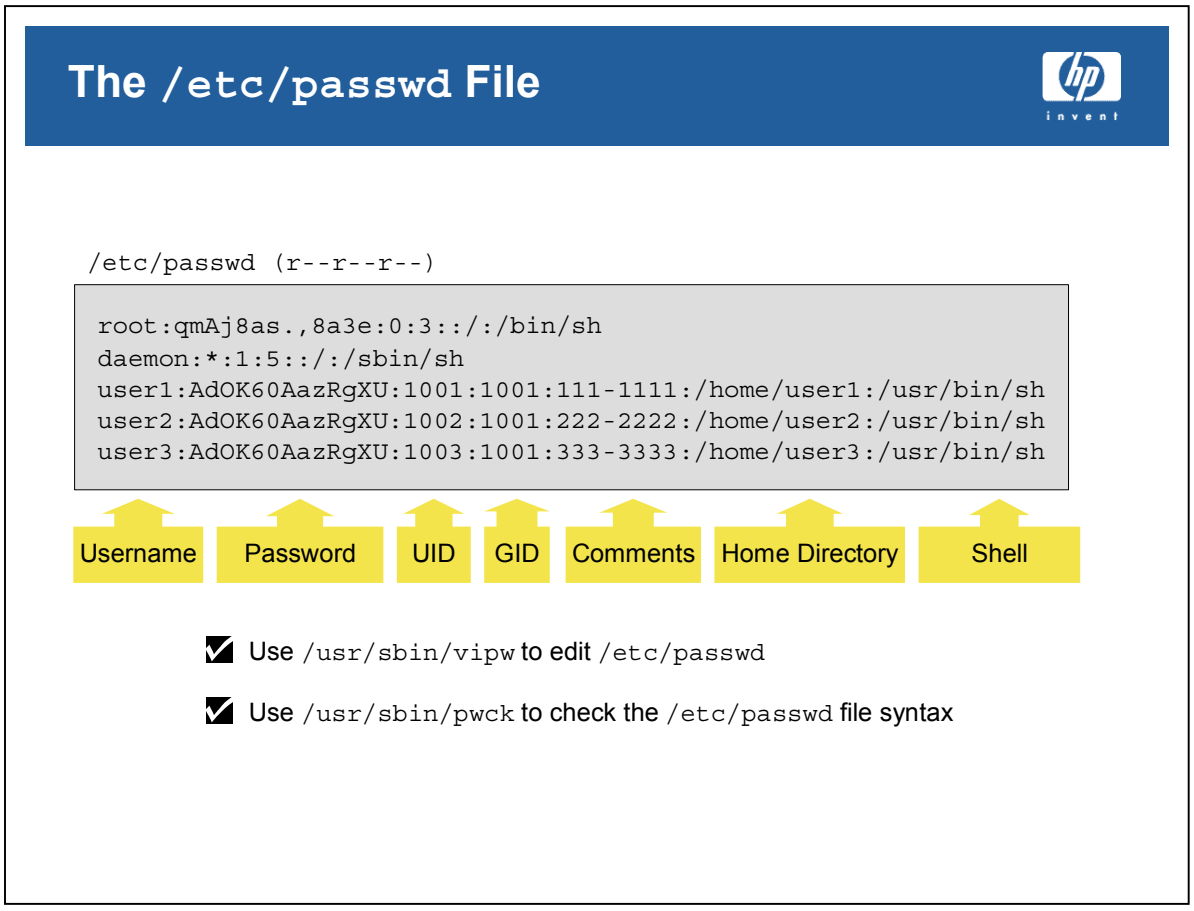
For users to successfully log in, they must have a valid user name, user ID, password, and other account information defined in the **/etc/passwd** file. The user may be granted access to additional groups on the system via the **/etc/group** file. Finally, most users have a home directory under **/home**, beneath which they can store their personal files and directories.

This chapter will discuss the structure and purpose of each of these, then consider several approaches for managing user accounts:

- via SAM
- from the command line with **useradd**, **userdel**, and **usermod**
- manually, via the **vi** editor

© 2004 Hewlett-Packard Development Company, L.P.

2–3. SLIDE: The `/etc/passwd` File



Student Notes

The **`/etc/passwd`** file contains essential information required during login. It contains one entry per line for each valid user of the system. All fields are delimited by a colon (:).

User name — The user name that is used when a user logs in. It should be between one and eight characters in length and the first character should be alphabetic. If the name contains more than eight characters, only the first eight are significant.

Password — The encrypted password. It is encrypted by the system when the user sets the password using the **`passwd`** command. The password should be six to eight characters, one of which is a number or a special character. If the password field is empty, no password is associated with the login name. Never leave the password field empty. Leaving it empty makes it very easy to break into a system.

An asterisk (*) in the password field deactivates an account. Nothing you can type will encrypt to an asterisk, so, no one can log in using the associated login name.

© 2004 Hewlett-Packard Development Company, L.P.

User ID Each user must be assigned a user ID. User ID 0 is reserved for root, and UIDs 1-99 are reserved for other predefined accounts required by the system. Version 10.20 of HP-UX introduced support for User IDs as large as 2,147,483,646. Prior to HP-UX 10.20, UIDs greater than 60,000 were not supported. When users are added using SAM, SAM begins assigning UIDs starting with UID 101. You can choose blocks of UIDs for particular groups:

- 100-199 – marketing
- 200-299 – engineering
- 300-399 – managers

CAUTION: There may be incompatibilities when sharing files owned by large UIDs with systems that don't support large UIDs.

Group ID The group ID (GID). This number corresponds with an entry in the **/etc/group** file.

ID string The comment field. It allows you to add extra information about the users, such as the user's full name, telephone extension, organization, or building number. This field is used by the line printer spooler system and by the **finger** command.

Home directory The absolute path to the directory the user will be in when they log in. If this directory does not exist or is invalid, then the user's home directory becomes **/.**

Command The absolute path of a command to be executed when the user logs in. Typically, this is a shell. The shells that are usually used are **/usr/bin/sh**, **/usr/bin/ksh**, and **/usr/bin/csh**. For system UIDs the shell is **/sbin/sh**, which is a special (POSIX) shell for the superuser. It should not be changed to another shell. If the field is empty, the default is **/usr/bin/sh**.

The *command* entry does not have to be a shell. For example, you can create the following entry in **/etc/passwd**:

```
date:rc70x.4.hGJdc:20:1::/:/usr/bin/date
```

The command is **/usr/bin/date**. If you type `date` at the login prompt, then type the appropriate password, the system will run the **/usr/bin/date** command, and then log you out.

NOTE: The permissions on the **passwd** file should be read only (**r--r--r--**) and the owner must be **root**.

© 2004 Hewlett-Packard Development Company, L.P.

Required Entries in `/etc/passwd`

Several entries are required in **`/etc/passwd`** to support various system daemons and processes. The list below lists the most critical required user accounts; other may be required, too, to support your system's applications.

```
root:rZ1lps2JYh3iA:0:3::/:/sbin/sh
daemon:*:1:5::/:/sbin/sh
bin:*:2:2::/usr/bin:/sbin/sh
sys:*:3:3::/:
adm:*:4:4::/var/adm:/sbin/sh
uucp:*:5:3::/var/spool/uucppublic:/usr/lbin/uucp/uucico
lp:*:9:7::/var/spool/lp:/sbin/sh
nuucp:*:11:11::/var/spool/uucppublic:/usr/lbin/uucp/uucico
hpdb:*:27:1:ALLBASE:/:/sbin/sh
nobody:*:-2:60001::/:
```

Editing `/etc/passwd`

If you are using **`vi`** to edit **`/etc/passwd`** and a user attempts to change a password while you are editing, the user's change will not be entered into the file. To prevent this situation, use **`vipw`** when editing **`/etc/passwd`**.

```
# vipw
```

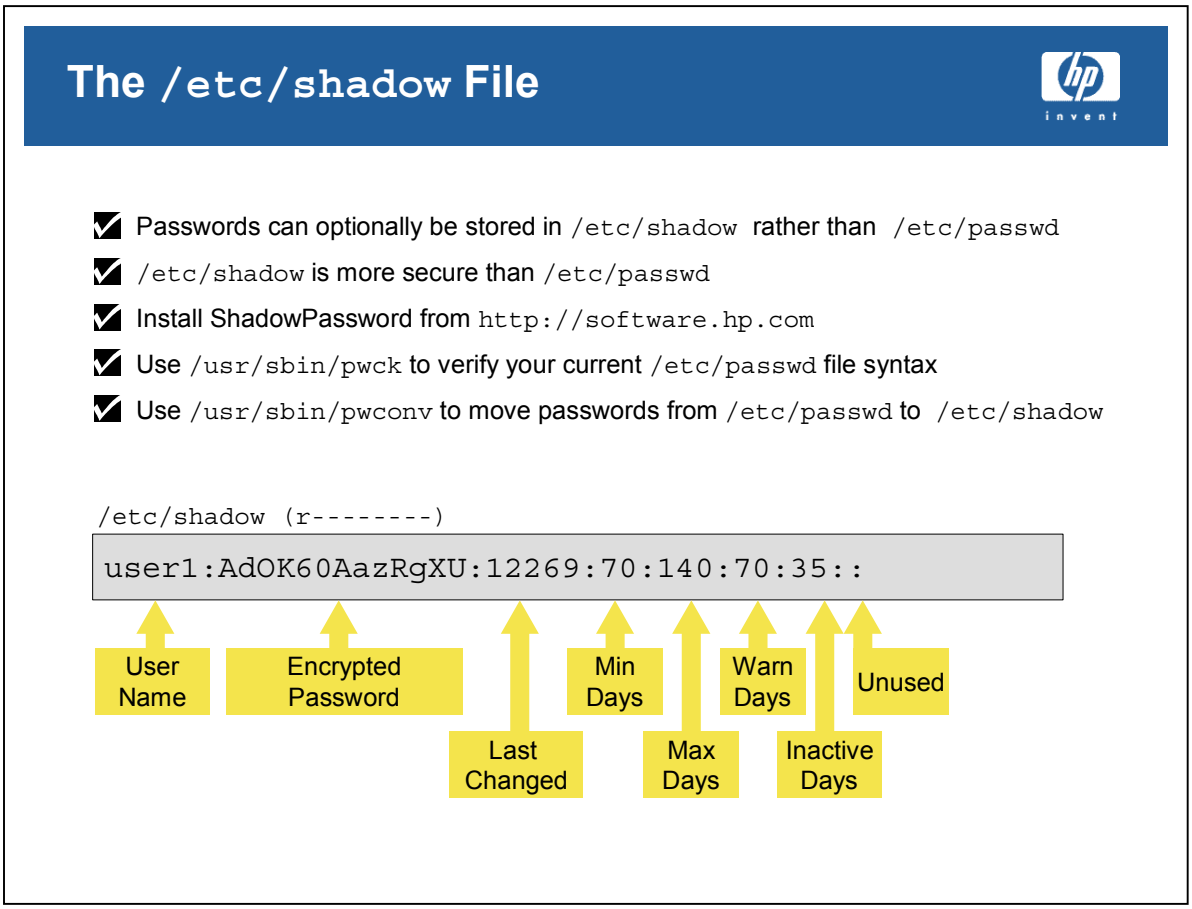
This command puts a lock on the **`/etc/passwd`** file by copying **`/etc/passwd`** to **`/etc/ptmp`**. If a user attempts to change a password, he or she will be told that the **`passwd`** file is busy. When you leave **`vipw`**, some automatic checks are done, and if your changes are correct, the temporary file is moved to **`/etc/passwd`**. Otherwise, **`/etc/passwd`** will remain unchanged.

Checking the `/etc/passwd` File

The consistency of the **`/etc/passwd`** file can be checked with the **`/usr/sbin/pwck`** command. It will check for the number of fields in each entry, and whether login directory and optional program name exist, and validate the number of fields, login name, user ID and group ID.

© 2004 Hewlett-Packard Development Company, L.P.

2–4. SLIDE: The `/etc/shadow` File



Student Notes

The default permissions on the **`/etc/passwd`** file are **`r--r--r--`**. Since the file is world-readable, anyone with a valid login can view the file and view encrypted passwords. Hackers sometimes exploit this fact to extract a list of encrypted passwords, run a password cracking utility, and potentially gain access to other users' accounts.

Unfortunately, removing world-read permission on the **`/etc/passwd`** file isn't a viable solution to this problem. Many commands, from **`login`**, to **`ps`**, to **`ll`** use the **`/etc/passwd`** file to convert UIDs to usernames, and vice versa. Changing the **`/etc/passwd`** file permissions to 400 would cause these commands to fail.

HP's shadow password product addresses this problem by moving encrypted passwords and other password information to the **`/etc/shadow`** file, which has 400 permissions to ensure that it is only readable by root. Other user account information (UIDs, GIDs, home directory paths, and startup shells) remain in the **`/etc/passwd`** file to ensure that **`login`**, **`ps`**, **`ll`**, and other commands can still convert UIDs to usernames.

© 2004 Hewlett-Packard Development Company, L.P.

Configuring Shadow Passwords

By default, the **/etc/shadow** file doesn't exist. Use the cookbook below to convert to a shadow password system:

1. If you are running 11i v2, Shadow Passwords are included in the Base OS. If you are running 11i v1, download and install the ShadowPassword product from **http://software.hp.com**. You can use the **swlist** command to determine if the product has already been installed on your system.

   ```
   # swlist ShadowPassword
   ```

2. Run **pwck** to verify that there aren't any syntax errors in your existing **/etc/passwd** file.

   ```
   # pwck
   ```

3. Use the **pwconv** command to move your passwords to the **/etc/shadow** file.

   ```
   # pwconv
   ```

4. Verify that the conversion succeeded. The **/etc/passwd** file should remain world-readable, but the **/etc/shadow** file should only be readable by root. The encrypted passwords in **/etc/passwd** should have been replaced by "**x**"s.

   ```
   # ll /etc/passwd /etc/shadow
    -r--r--r--   1 root   sys   914 May 18 14:35 /etc/passwd
    -r--------   1 root   sys   562 May 18 14:35 /etc/shadow
   ```

All the standard password commands, including **passwd**, **useradd**, **usermod**, **userdel**, and **pwck** are shadow password aware.

You can revert to the traditional non-shadowed password functionality at any time via the **pwunconv** command.

```
# pwunconv
```

Fields in /etc/shadow

The **/etc/shadow** file is an ASCII file consisting of any number of user entries separated by newlines. Each user entry line consists of the following fields separated by colons:

login name Each login name must match a login name in **/etc/passwd**.

password When you convert to a shadowed system, each password in **/etc/passwd** is replaced with an "**x**", and the encrypted passwords are copied to the second field in **/etc/shadow**. If the **/etc/shadow** password field is null, then there is no password and no password is demanded on login. Login can be prevented by entering a "*****" in the **/etc/shadow** password field.

© 2004 Hewlett-Packard Development Company, L.P.

last changed The number of days since January 1, 1970 that the password was last modified. This field is used by the password aging mechanism, which will be described later in the chapter.

min weeks The minimum number of weeks that a user must retain a password before it can be changed. This field is used by the password aging mechanism, which will be described later in the chapter.

max weeks The maximum number of weeks for which a password is valid. A user who attempts to login after his password has expired is forced to supply a new one. If min weeks and max weeks are both zero, the user is forced to change his password the next time he logs in. If min days is greater than max weeks, then the password cannot be changed. These restrictions do not apply to the superuser. This field is used by the password aging mechanism, which will be described later in the chapter.

warn weeks The number of weeks the user is warned before his password expires. This field is used by the password aging mechanism, which will be described later in the chapter.

inactivity The maximum number of days of inactivity allowed after a password has expired. The account is locked if the password is not changed within the specified number of days after the password expires. If this field is set to zero, then the user is required to change his password. This field is only used by HP-UX trusted systems, which aren't discussed in this course.

expiration The absolute number of days since Jan 1, 1970 after which the account is no longer valid. A value of zero in this field indicates that the account is locked. This field is only used by HP-UX trusted systems, which aren't discussed in this course.

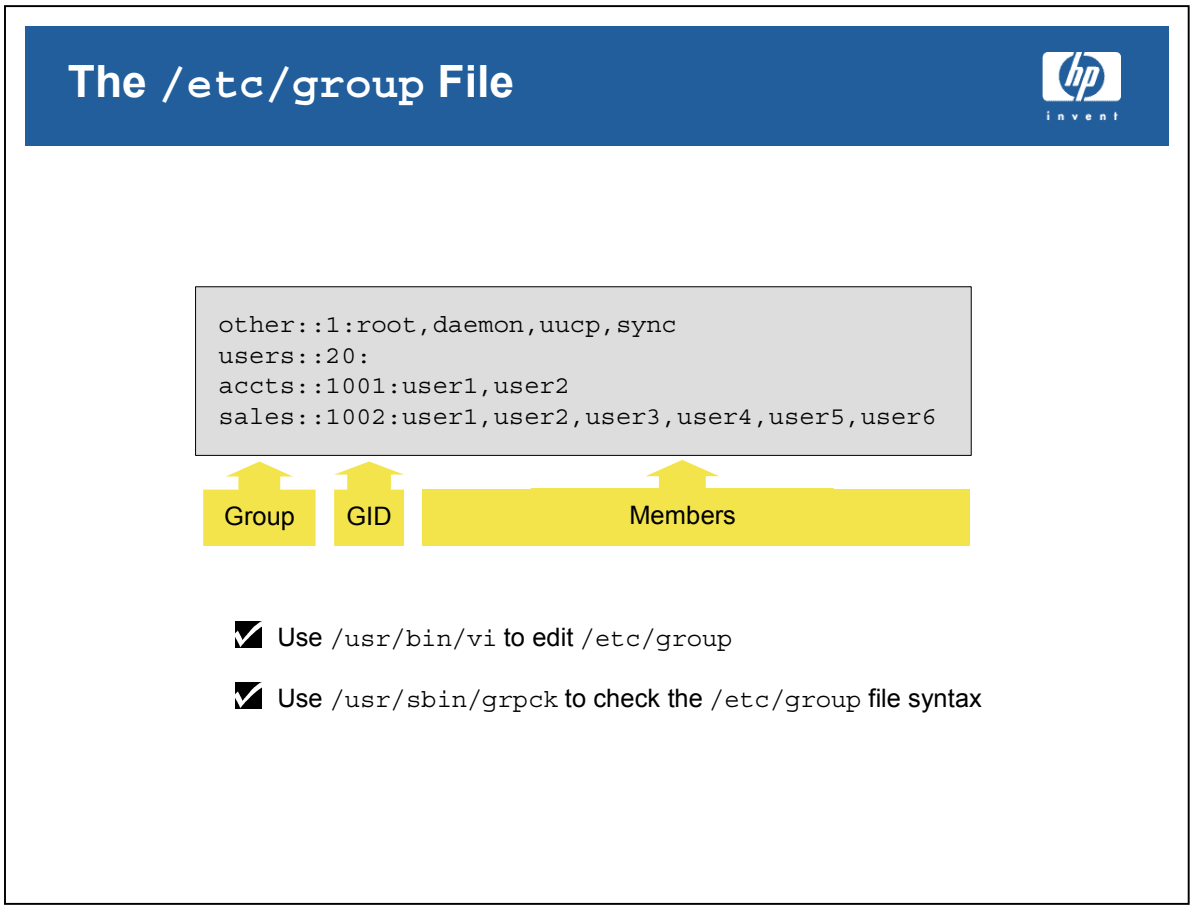
reserved The reserved field is always null, and is reserved for future use.

Editing `/etc/shadow`

Manually editing the **`/etc/shadow`** file isn't recommended. On a shadow password system, you should use the **`useradd`**, **`usermod`**, **`userdel`**, and **`passwd`** commands to manage user accounts in both **`/etc/passwd`** and **`/etc/shadow`**. These commands will be described in detail later in the chapter.

© 2004 Hewlett-Packard Development Company, L.P.

2–5. SLIDE: The `/etc/group` File



Student Notes

The **`/etc/group`** file is used to define user groups. A colon delimits the fields (**`:`**).

group_name is the mnemonic name associated with the group. If you do an **`ll`** on a file, you will see this name printed in the group field.

Password is typically not used, so it is blank. It can contain an encrypted group-level password if you implement privileged groups.

group_id is the group ID (GID). This is the number that should be placed in the **`/etc/passwd`** file in the *group_id* field. All group members share this number. It is recommended that the system administrator create groups with IDs of greater than 100 to ensure they do not collide with current or future system needs.

group_list is a list of user names of users who are members of the group. In HP-UX 11.0, not all members of a given group are listed in the **`/etc/group`** file. A user's initial login group is defined in the fourth field of **`/etc/passwd`**, not in the **`/etc/group`** file.

© 2004 Hewlett-Packard Development Company, L.P.

When a user logs in on HP-UX system, HP-UX checks the primary group field in the user's **/etc/passwd** entry to determine which group the user should be placed in. The user's group membership help determine which files and other resources the user will be able to access.

On HP-UX 10.x systems, if a user wants to access a file that is associated with another group, the user must change his or her effective group membership via the **newgrp** command:

```
$ newgrp group_name
```

HP-UX checks the **/etc/group** file to determine if the user has access to the requested group, then changes the user's effective group id accordingly. Executing the **newgrp** command with no parameters returns the user to the group to which he is assigned in the **/etc/passwd** file.

Some administrators choose to configure a file called **/etc/logingroup** to save users the tedium of repeatedly executing the **newgrp** command. The **/etc/logingroup** file associates each user with multiple group id's simultaneously at login. Typically, **/etc/logingroup** is simply configured as a link to **/etc/group**:

```
# ln /etc/group /etc/logingroup
```

At HP-UX 11.x, the **/etc/logingroup** file is no longer required; all users are granted access to all of their secondary groups at login regardless of the state of the **/etc/logingroup** file.

Required Entries in /etc/group

```
root::0:root
other::1:root,hpdb
bin::2:root,bin
sys::3:root,uucp
adm::4:root,adm
daemon::5:root,daemon
mail::6:root
lp::7:root,lp
tty::10:
nuucp::11:nuucp
nogroup:*:-2:
```

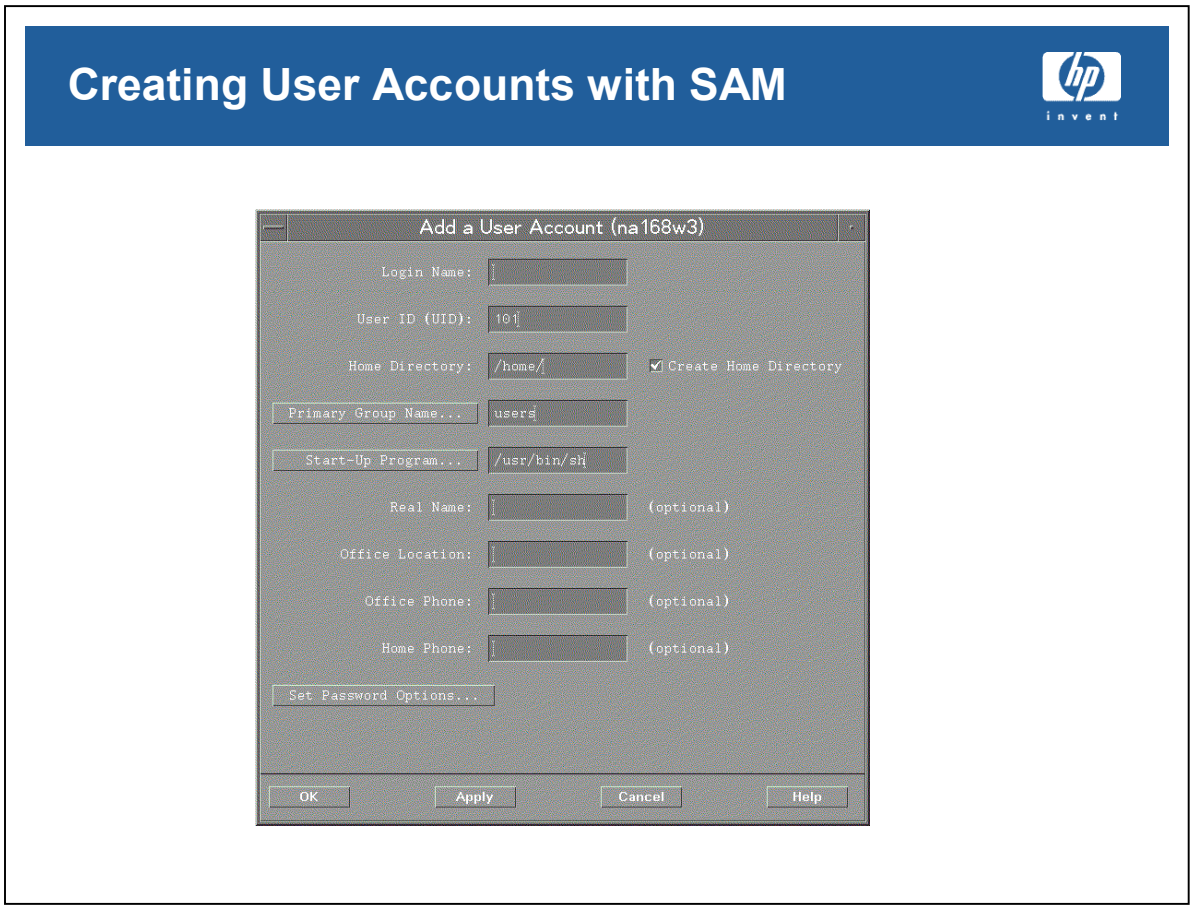
For more information on the **/etc/group** file, see **group(4)** in the *HP-UX Reference* manual.

Checking the /etc/group File

The consistency of the **/etc/group** file can be checked with the **/usr/sbin/grpck** command. It will check for the number of fields in each entry, and whether all login names appear in the password file.

© 2004 Hewlett-Packard Development Company, L.P.

2–6. SLIDE: Creating User Accounts with SAM



Student Notes

SAM provides a menu driven interface for adding user accounts. In order to add an account using SAM, choose **`Accounts for Users and Groups`** from the SAM control window.

Choose Users from the next menu. SAM will display a list of users. If there are more than 500 users on your system, you will be asked to select a subset of users.

Choose **`Add..`** from the **`Actions`** menu. SAM will display a form to be completed in order to add a user.

You will need to fill in the following information:

Login Name — The user name that is used when a user logs in. It should be between one and eight characters in length. The first character should be alphabetic. If the name contains more than eight characters, only the first eight are significant.

© 2004 Hewlett-Packard Development Company, L.P.

User ID	The user ID (UID). UID zero (0) is reserved for root. Typically, values 1-99 are reserved for the system. User IDs as large as 2,147,483,646 can be used with HFS file systems and with JFS Version 3 file systems. JFS Version 3 is installed by default with HP-UX 10.20; file systems created with earlier versions of JFS must be upgraded to Version 3 to support large user IDS (IDs larger than 60,000). SAM will use UIDs starting from 101. You can choose override the UID SAM has assigned. You may want to do this to ensure unique UIDs in a network.
Home Directory	The absolute path to the directory the user is in when logging in. If this directory does not exist or is invalid, the user is unable to log in. By default SAM will append the Login Name to **`/home`**. There is a check box to determine whether the home directory is to be created. In the case where users will be sharing a home directory you may want to turn this off.
Primary Group Name	This field defines the primary group that is assigned to the user at login. The default is users. To choose another group, type over the word **`users`**. Or, to choose from a list of groups, highlight the **`Primary Group Name`** and SAM will display a list for you to choose from.
Start-Up Program	The absolute path of a command to be executed when the user logs in. Typically, this is a shell. The shells that are usually used are **`/usr/bin/sh`**, **`/usr/bin/ksh`**, **`/usr/bin/keysh`**, and **`/usr/bin/csh`**. Other choices for restricted shells include **`/usr/bin/rsh`** and **`/usr/bin/rksh`**. It is not required that the start-up program be a shell. You can enter the full path name of any valid command here. When the user logs in, the command will be executed. When the command terminates, the user is logged off.
Set Password Options…	Selecting this push-button opens a new dialog window for defining which of the following password behavior or aging characteristics are in force for the selected user:

No restrictions	The user account being added or modified can have the same password indefinitely or for the duration of the account without changing it.
Force password change at next login	A user must modify his or her password the next time he or she logs on to the system. Then the password aging reverts to No Restrictions.
Allow only superuser to change at next login	Only the superuser (user root) can change a user password. Individual users are not allowed to select their own passwords.

© 2004 Hewlett-Packard Development Company, L.P.

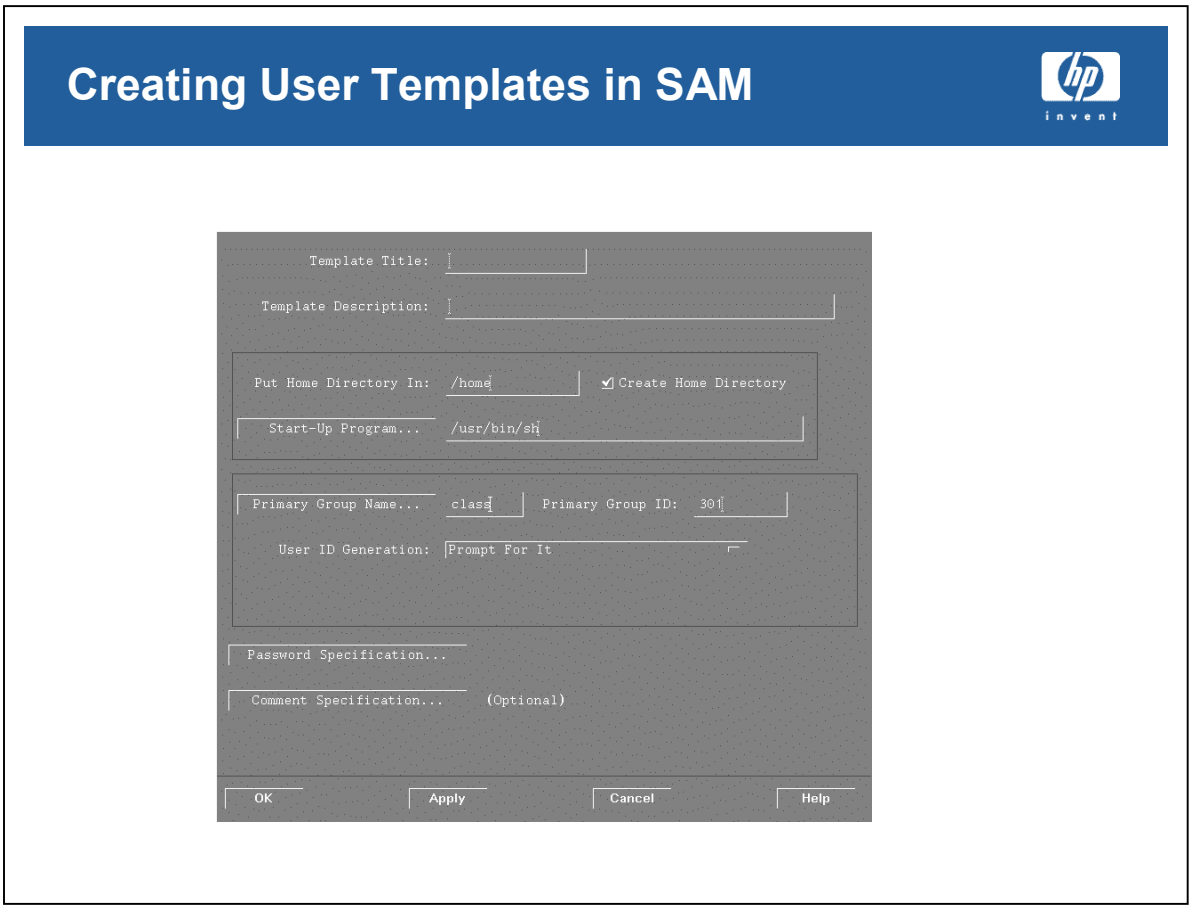
Set up periodic password aging	Set up password expiration with a minimum required time between password changes. This option leads to additional fields for setting up the time intervals for expiration and minimum time between changes.

When you have completed filling out the menu, press **OK** if you do not want to add additional users, or **APPLY** if you do.

You will be prompted for a password for the new user. SAM will create the **/etc/passwd** entry, update **/etc/group**, and create the home directory if requested.

© 2004 Hewlett-Packard Development Company, L.P.

2–7. SLIDE: Creating User Templates in SAM



Student Notes

If you plan to add many users with the same requirements you can simplify the process by creating a template.

From the **`User Accounts`** screen select **`User Templates`** from the **`Actions`** menu.

When you select **`Create`**, SAM will display the menu shown.

You must supply a template name. Names can be up to 16 characters in length and any combination of letters, numbers, and the underscore.

Most fields are the same as those on the Add User menu. You may choose to handle user ID generation in one of four ways:

- first available
- first available greater than one
- first available within range
- prompt for it

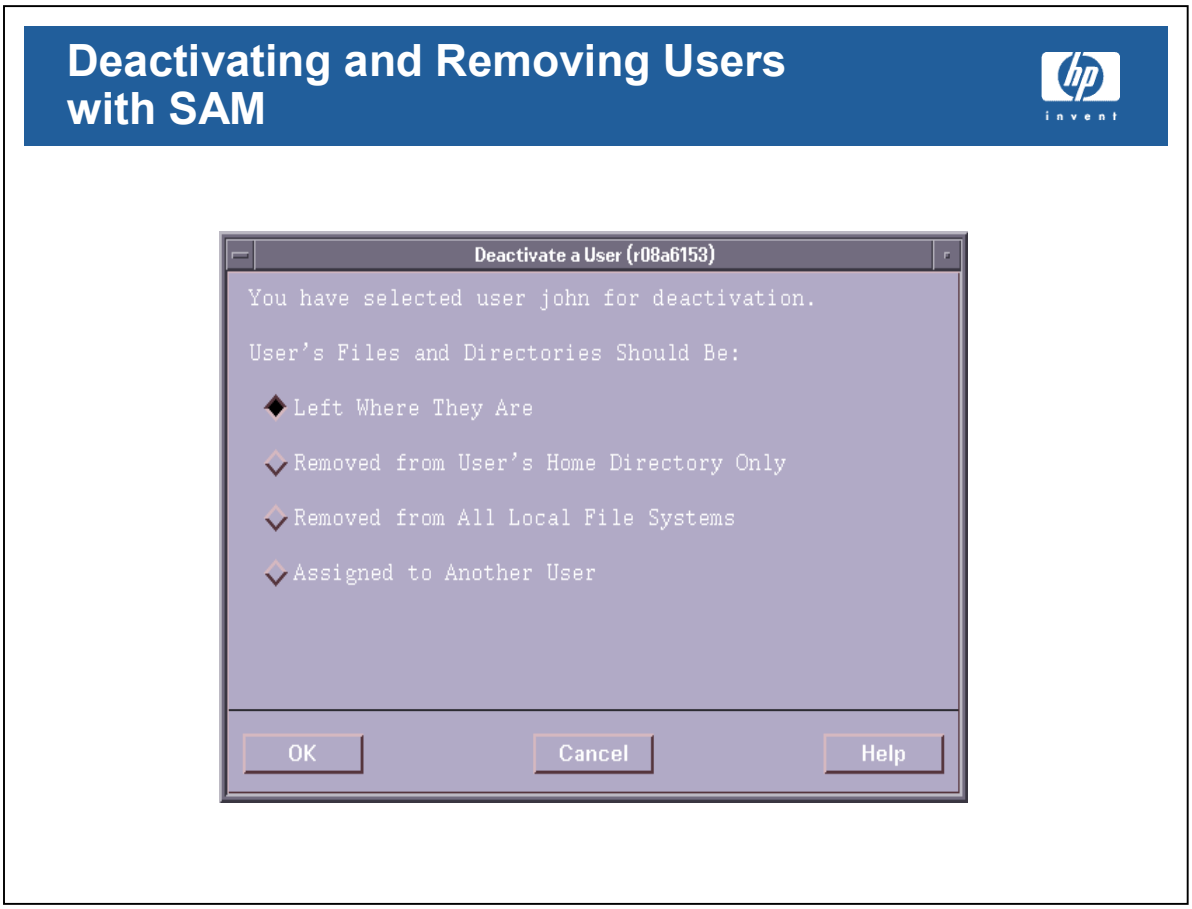
© 2004 Hewlett-Packard Development Company, L.P.

When you have completed the menu press **OK** if you do not wish to create additional templates or **Apply** if you do.

To use a template when adding users select **User Templates** from the **Actions** menu of the User Accounts screen. When you choose **Select** SAM will display a list of templates. Highlight one and press **OK**. The selected template will be in effect until it is unselected. The template will be used for all new users added until you **Unselect** the template.

© 2004 Hewlett-Packard Development Company, L.P.

2–8. SLIDE: Deactivating and Removing Users with SAM



Student Notes

If you no longer want a user to have access to the system, you can either remove the user or deactivate the account. Removing the user deletes the user's **`/etc/passwd`** and **`/etc/group`** entries. Deactivating the user causes SAM to place an * in the password field of **`/etc/passwd`**. You can either use SAM to reactivate the account, or simply change the user's password using the **`passwd`** command.

To use SAM to deactivate a user, choose the **`Accounts for Users and Groups`** icon on the SAM opening screen.

Choose **`Users`** from the next menu. SAM will display a list of users. If there are more than 500 users on your system, you will be asked to select a subset of users.

Choose either **`Remove`** or **`Deactivate...`** from the **`Actions`** menu. SAM will display a list of choices for handling the user's files.

The reason you are removing a user will guide you in determining whether to keep that user's files on the system, reassign them to someone else, or remove them entirely.

© 2004 Hewlett-Packard Development Company, L.P.

SAM provides two features to help protect against inadvertent removal of users or files when removing users:

- Editable list of users to exclude from removal.

  When prompting for the name of a user to remove from the system, SAM checks the name given against a list of names specified in the file **/etc/sam/rmuser.excl**. If the name matches one within the file, SAM does not remove the user.

- Editable list of files to exclude from removal when a user is removed from the system.

  When SAM removes a user, all files (or a subset thereof) for that user are also removed, unless the ownership is given to another user. Before removing a file belonging to the user, SAM checks to see if the file resides in a path that has been excluded from removal. SAM uses the file **/etc/sam/rmfiles.excl** to determine which paths have been excluded from removal. So, for example, if the path **/home/joe/test** is named in the file, SAM will not remove any files residing beneath that directory. SAM logs a list of all files it removes in the file **/var/tmp/sam_remove.log**.

NOTE:	You can edit the files **/etc/sam/rmuser.excl** and **/etc/sam/rmfiles.excl** to contain users and directories that you want to exclude from removal from SAM.

Here is a sample **/etc/sam/rmuser.excl** file:

```
root
daemon
bin
sys
adm
uucp
lp
nuucp
hpdb
nobody
```

Here is a sample **/etc/sam/rmfiles.excl** file:

```
/dev
/etc
/export
/mnt
/opt
/sbin
/stand
/usr
/var
```

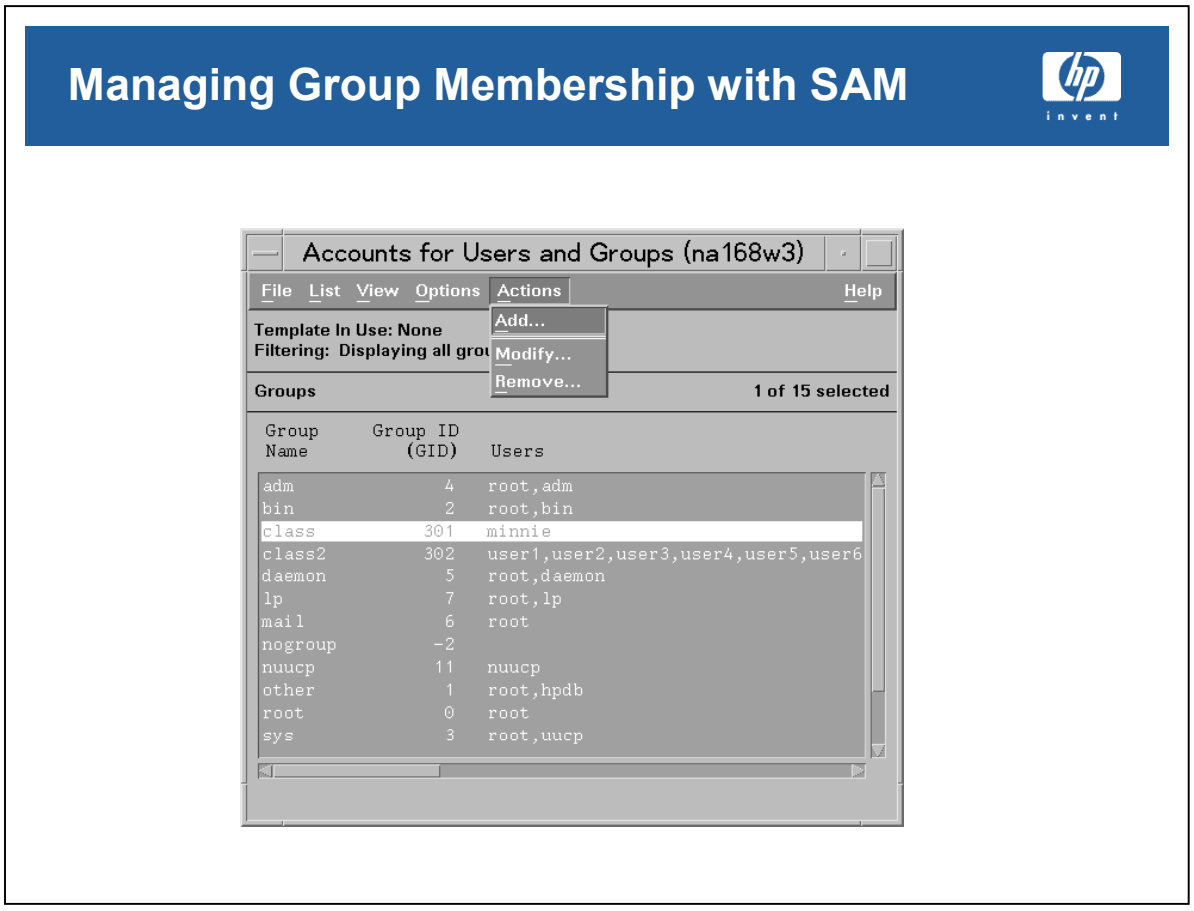

© 2004 Hewlett-Packard Development Company, L.P.

Here is a sample **/etc/sam/rmgroup.excl** file:

```
root
other
bin
sys
admin
daemon
mail
lp
nogroup
```


© 2004 Hewlett-Packard Development Company, L.P.

2–9. SLIDE: Managing Group Membership with SAM



Student Notes

SAM provides the most intuitive method for managing group membership. After selecting any group from the object list, go to the Actions menu and choose:

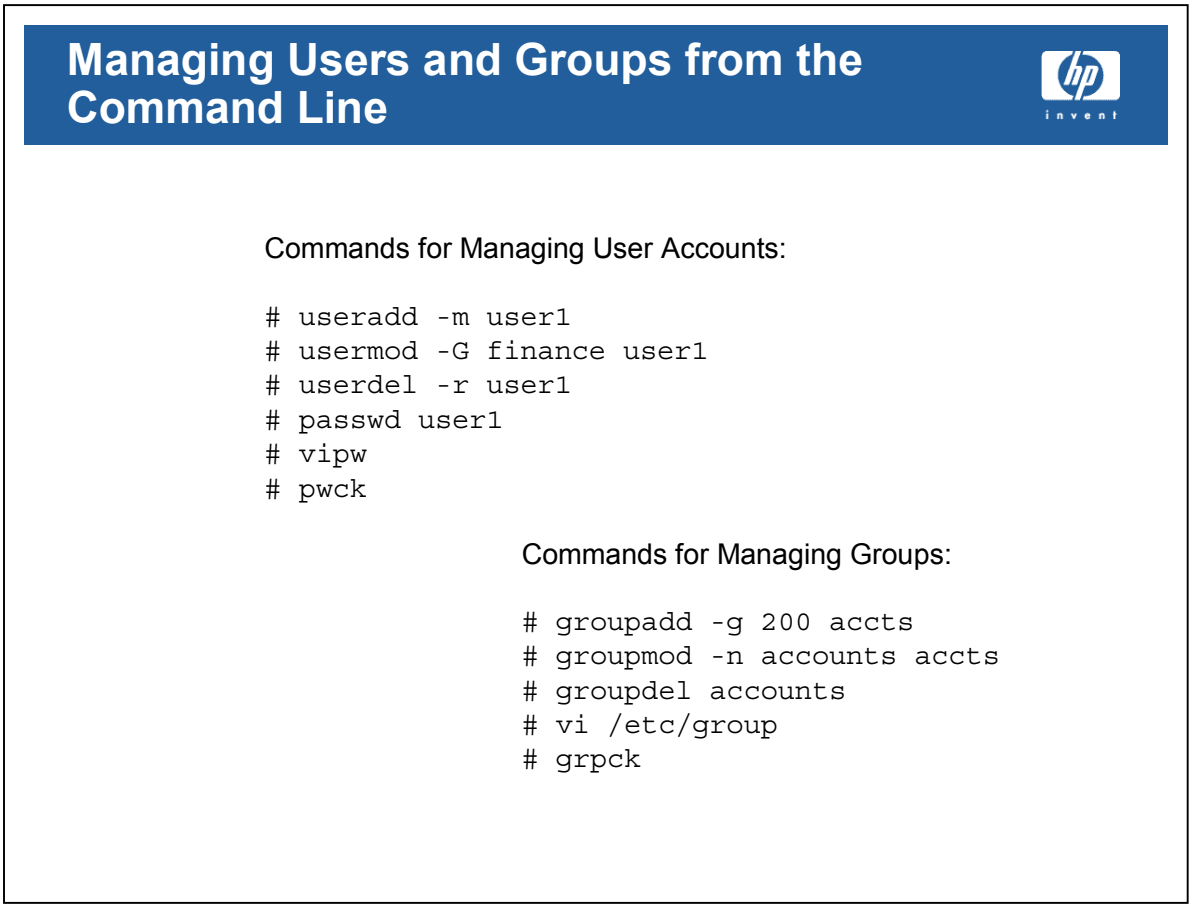
Add: To create a new group

Modify: To modify the group membership list

Remove: To remove the group entirely

© 2004 Hewlett-Packard Development Company, L.P.

2–10. SLIDE: Managing Users and Groups from the Command Line



Student Notes

Up to this point, we have used SAM to create and modify users and groups on the system. Some administrators, however, prefer to manage user accounts from the command line. Using the command line offers two major advantages over SAM:

- Issuing commands from the command line is much faster.
- User account management may be automated by calling commands from shell scripts.

The examples on the slide show some of the more common commands HP-UX provides for managing users and groups.

Managing Users from the Command Line

The first example creates a new user account for a new user, "user1". The **`-m`** option creates a home directory for "user1" as well. Note, however, that **`useradd`** does not set a password for the user initially!

```
# useradd -m user1
```


© 2004 Hewlett-Packard Development Company, L.P.

Already existing user accounts may be modified using the **usermod** command. The example below adds user "user1" to the group called "finance":

```
# usermod -G finance user1
```

Both **useradd** and **usermod** recognize many options for setting or changing a user's account definition in the **/etc/passwd** file (see the **useradd** man page for additional options):

`-u 101`	*Use userid 101.*
`-g finance`	*Define the user's primary group.*
`-G finance,accounts`	*Define the user's other group memberships.*
`-c "111-1111"`	*Define the user's comment field in /etc/passwd.*
`-d /home/guest`	*Define the user's home directory.*
`-s /usr/bin/csh`	*Define the user's default login shell.*
`–o`	*Allows the creation of users with non-unique userids.*

User accounts may be removed via the **userdel** command. The command below removes user1's account from the system:

`# userdel user1`	*Remove only user1's /etc/passwd definition.*
`# userdel -r user1`	*Also remove user1's home directory.*

If you prefer to manually edit the **/etc/passwd** file, use **vipw** and **pwck**:

```
# vipw
# pwck
```

vipw locks the passwd file to prevent users from making changes until you save your changes to **/etc/passwd** and exit out of **vi**. **vipw** does not require any arguments. After making changes to **/etc/passwd**, it is prudent to run the **pwck** utility to check for syntax errors

Managing Groups from the Command Line

HP-UX also supports several commands for managing groups on the system:

`# groupadd -g 200 accts`	*Create a group called accts with GID 200.*
`# groupmod -n accounts accts`	*Change the name of accts to accounts.*
`# groupdel accounts`	*Delete the accounts group.*

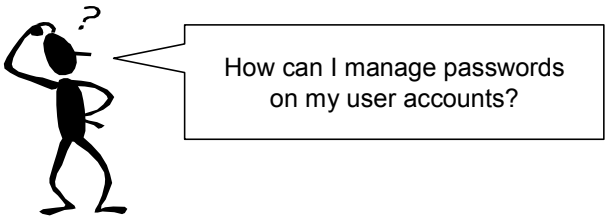
You may also manually modify the **/etc/group** file, but be sure to do a sanity check on the file afterwards with **grpck**:

```
# vi /etc/group
# grpck
```


© 2004 Hewlett-Packard Development Company, L.P.

2–11. SLIDE: Managing Passwords from the Command Line

Managing Passwords from the Command Line



`# passwd user1`	reset a user's password
`# passwd -d user1`	set a null password
`# passwd -f user1`	force a password change at next login
`# passwd -l user1`	lock or disable an account
`# passwd -n 7 -x 70 -w 7 user1`	configure password aging
`# passwd -n 7 -x 0 user1`	prevent a user from changing passwords
`# passwd -s user1`	check password aging settings

Student Notes

In order to ensure the integrity and security of an HP-UX system, you must ensure that your users are choosing secure passwords and are changing them frequently.

Default HP-UX Password Restrictions

By default, HP-UX places a number of restrictions on user-defined passwords:

- Passwords must be at least 6 characters in length.
- Each password must contain at least 1 non-alphabetic character.
- Each password must contain at least 2 alphabetic characters.
- Passwords must differ from the user's user name, and any circular or reverse shift of the user's username.

NOTE: None of these restrictions apply to passwords set by the system administrator.

© 2004 Hewlett-Packard Development Company, L.P.

Resetting a User's Password

If a user forgets his or her password, the administrator may reset the password by issuing the **passwd** command with the user's username as an argument. The system will prompt the administrator to enter the new password twice.

Setting a Null Password

Some administrators choose to set a null password on newly created user accounts. Use the **-d passwd** option to delete a user's password. Deleting a user's password in this manner will allow the user to log in by simply hitting return when prompted for their password.

Forcing a Password Change at Next Login

You may wish to force new users to change their assigned password at next login. To force a password change at next login, use **passwd -f username** as shown on the slide.

Disabling a User Account

If a user plans to be out of the office for an extended period of time, consider locking or disabling the user's account using **passwd -l username**. Disabling an account simply replaces the user's password with an "*" in the **/etc/passwd** file. When the user returns, re-enable their account with **passwd username**.

Configuring Password Aging

Password aging forces users to change their passwords on a regular basis. Three options on the **passwd** command enable password aging. **-x** defines the maximum number of days a user can wait before they are required to change their password. **-n** defines the minimum number of days a user must retain a password before they are allowed to change it. Administrators often use the **–n** option to prevent users from changing their passwords, then immediately reverting to their previous passwords. The –w option specifies a password warning time. Several days before a user's password expires, the user will see a warning message at login time indicating that their password expiration date is approaching. Usually these options are used in combination. Although **–x**, **-n**, and **–w** all expect their associated arguments to be specified in days, HP-UX rounds the values up to the nearest week.

On a non-shadowed system, password aging causes HP-UX to append a comma, followed by several additional characters to the password field in **/etc/passwd**. For a description of these special characters see the **passwd(4)** man page.

On a shadowed system, the password aging information is stored in **/etc/shadow**. See the **shadow(4)** man page for details.

Preventing a User from Changing Passwords

If several users share a common account, you may wish to prevent unauthorized password changes. If the minimum number of weeks between changes is greater than the maximum number of weeks between changes for a given account, then only the **superuser** will be allowed to change that account's password.

© 2004 Hewlett-Packard Development Company, L.P.

Checking Password Aging Settings

You can determine if password aging is enabled for a user using the **`passwd -s`** command. If password aging has been enabled, this command displays the minimum and maximum password age settings, as well as the week in which the user's password was last changed.

© 2004 Hewlett-Packard Development Company, L.P.

2–12. LAB: Hands-On Adding Users

Perform the following tasks. Record the commands you use, and anwer any questions.

Part 1: Creating and Modifying Users and Groups

1. Use SAM to create a user account for user **`mickey`** on your system. Let SAM choose a UID for you. Assign the user to the **`users`** group, accept the default home directory name, and use the Posix shell as the startup shell. You may use any password you wish.

2. What did SAM add to the **`/etc/passwd`** file?
   Is the new user listed in the **`/etc/group`** file? Explain!

3. Oops! We misspelled **`mickey`**'s username. Can you use SAM to change **`mickey`**'s username to **`mickie`**? Try it! Then do an **`ll`** of the **`/home`** directory. Did SAM automatically modify Mickey's home directory, too? If not, make the change manually.

4. Create another user account for user **`donald`**, but this time don't use SAM. Make sure you include the option to create a home directory for the new user; otherwise, you may accept the defaults.

© 2004 Hewlett-Packard Development Company, L.P.

5. Can **donald** log in at this point? What else needs to be done to make his account fully functional? Make it so!

6. Oops! **donald** wants to use the Korn shell as his default startup program. Make this change without using SAM.

7. **mickie** and **donald** need to collaborate on a project, and have asked you to create a group ID for them. Create a group called **project**, and ensure that **mickie** and **donald** both have access to the group. Their primary group IDs, however, should remain set to **users**. Again, use the command line rather than SAM.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Deactivating and Removing User Accounts

1. Deactivate/lock **mickie**'s account, but leave all of his files on the system.
   Remove **donald**'s account, but leave his files on the system, too.
   You may use either SAM or the command line.

2. What changed in the **/etc/passwd** file because of the commands in the previous question?

3. What happens now when **donald** and **mickie** attempt to log in? **telnet** to your local host, and try to login using both usernames. What happens?

   ```
   # telnet localhost
   ```

4. What happened to **donald** and **mickie**'s home directories? Do a long listing of **/home**. Can you explain what you see?

   ```
   # ll /home
   ```

5. Re-enable mickie's account. Choose a password for him, then do whatever is necessary to force him to choose a new password at next login.

© 2004 Hewlett-Packard Development Company, L.P.

6. Verify your configuration by **`telnet`**'ing to your **`localhost`** using username **`mickie`**.

```
# telnet localhost
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Implementing Shadow Passwords

1. Run **pwconv** to create the **/etc/shadow** file.

   a. What is in the password field in **/etc/passwd** now?

   b. What fields are populated in **/etc/shadow**?

   c. What are the permissions on **/etc/shadow**? Why is this significant?

2. Will **useradd** add new users be added to the **/etc/passwd** file, the **/etc/shadow** file, or both? Is the **passwd** command shadow aware? Try it! Use the **useradd** command to create an account for user25, then use the **passwd** command to set a null password for the user and force a password change at next login. Then check the **/etc/passwd** and **/etc/shadow** files to see what happened.

3. Enable shadow password aging on the user1 account.

   a. Ensure that the password is changed at least twice per year.

   b. Ensure that users wait at least one week between password changes.

   c. Provide a one-week warning before the user's password expires.

4. Before you continue on to the next part, revert to a non-shadowed password file.

© 2004 Hewlett-Packard Development Company, L.P.

Part 4: (Optional) Automating User Account Creation

1. Pretend for a moment that you are a system administrator at a large university. Fifty students have just enrolled to start classes, and you need to create user accounts for them. Can you write a simple shell script to automatically create the user accounts? Initially, you can assign the students null passwords, but force them to change their passwords after their first successful login.

   Hint: Try running the sample shell script below. What must be changed in the shell script to automatically create the desired accounts?

   ```
   #!/usr/bin/sh
   n=1
   while ((n<=50))
   do
      echo stud$n
      ((n=n+1))
   done
   ```


© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 3 — Customizing User Accounts

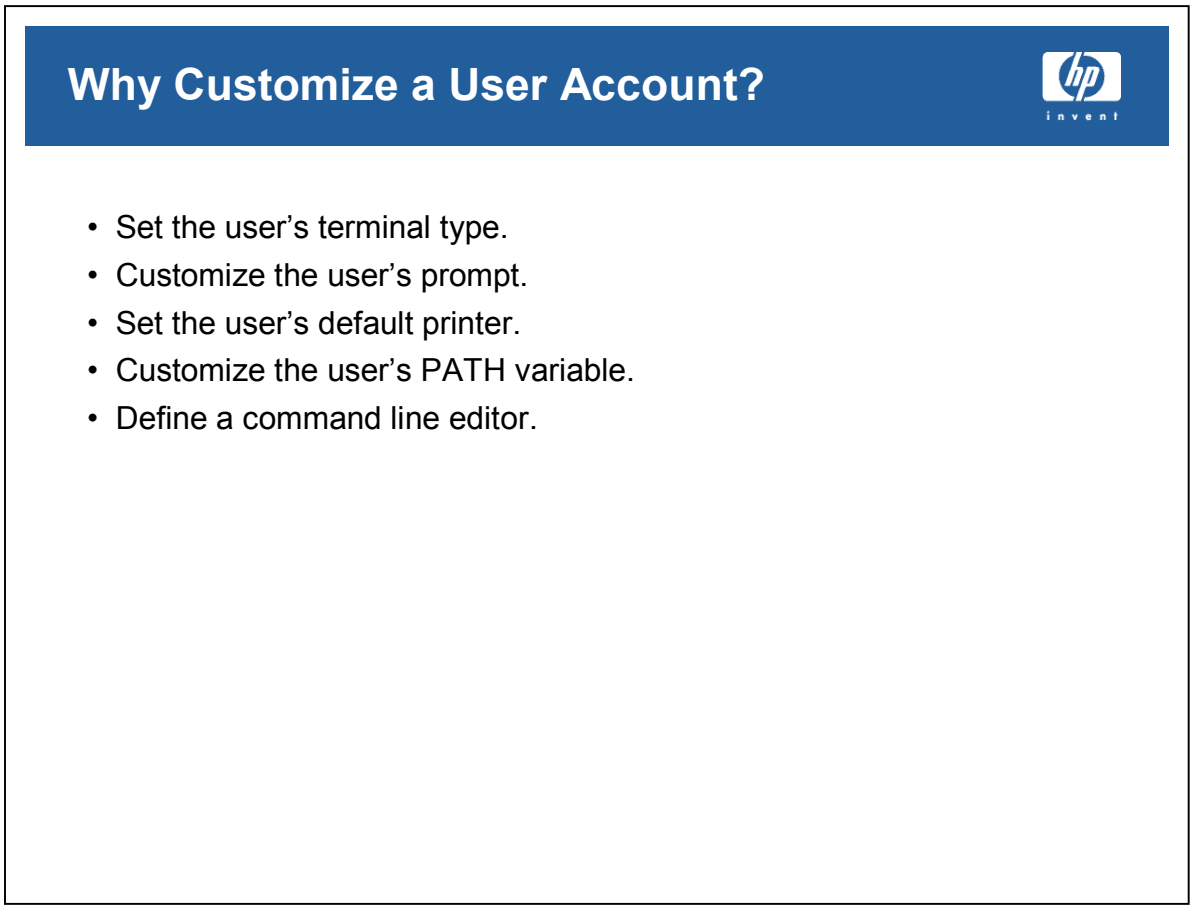
Objectives

Upon completion of this module, you will be able to do the following:

- List configuration files read during the login process.
- Change the user's default PATH.
- Change the user's default terminal type.
- Change the user's prompt string.
- Change the user's command line editor.
- Change the user's default printer.
- Manage default configuration files in `/etc/skel`.

© 2004 Hewlett-Packard Development Company, L.P.

3–1. SLIDE: Why Customize a User Account?



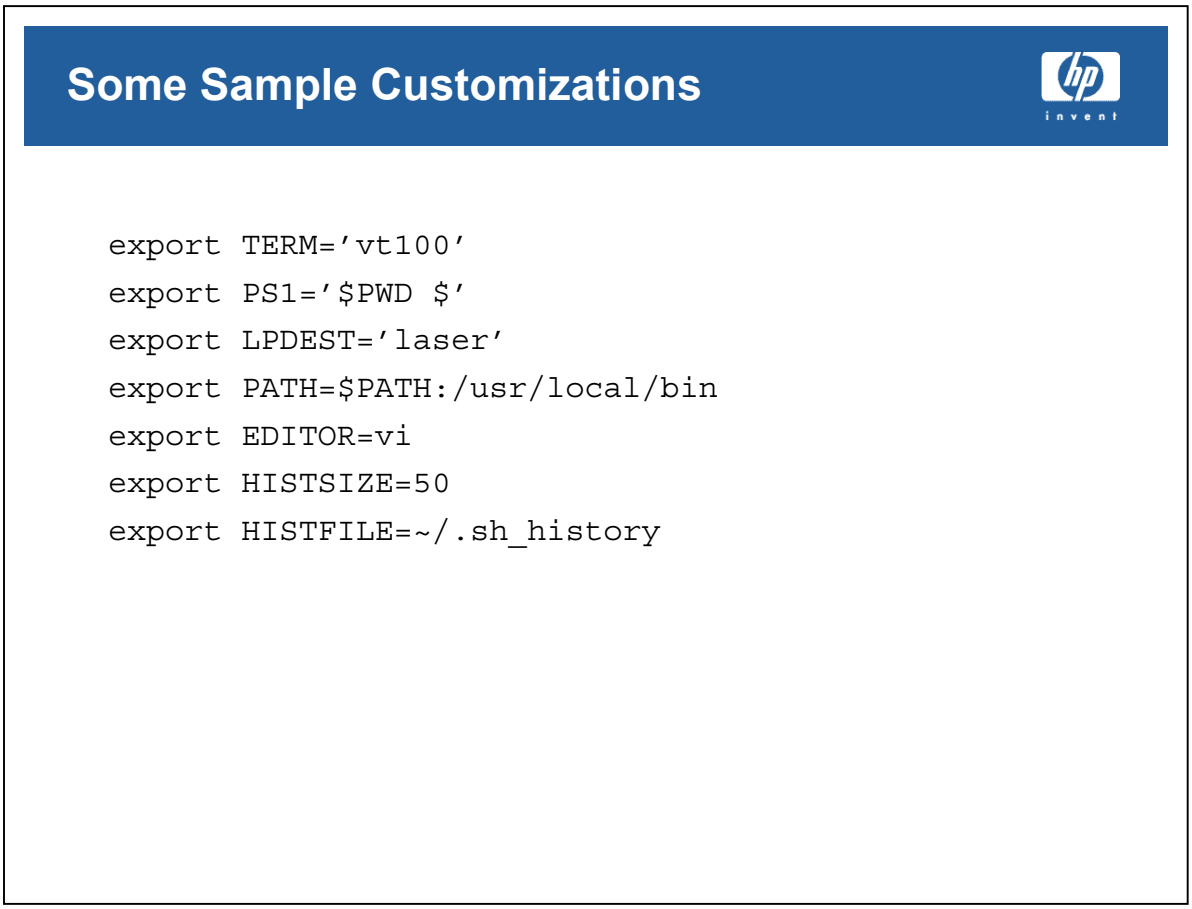
Student Notes

Simply creating an entry for a user in **`/etc/passwd`** and **`/etc/group`** may not give the user all the functionality needed.

- You may need to define the user's terminal type so applications can properly write to the user's screen.
- You may want to customize the user's prompt. Many users like the present working directory to appear in the prompt string.
- Although the system administrator defines a system default printer, individual users may choose to select a different default destination printer.
- If the user accesses third party applications, you may need to modify their PATH variable so their shell can find the application executables.
- Some special configuration is required if the user wishes to use command line editing and the command history mechanism. You may wish to configure this functionality for new user accounts.

© 2004 Hewlett-Packard Development Company, L.P.

3–2. SLIDE: Some Sample Customizations



Student Notes

All of the features mentioned on the previous slide are configured via "environment variables" that are set during the login process. Some of the most commonly modified environment variables are listed below:

TERM — The TERM variable defines the user's terminal type. If the TERM variable is set incorrectly, applications may not be able to write to the user's terminal properly. Valid terminal types are listed in the **`/usr/lib/terminfo/*`** directories. You can explicitly set an appropriate TERM value using a command similar to the following:

```
export TERM=vt100     # for a vt100 type terminal
export TERM=hp        # for an HP ASCII terminal
export TERM=dtterm    # for a dtterm terminal emulator window
```

More commonly, however, the TERM variable is set using the ttytype command, which can usually automatically determine your terminal type. The following portion of code can be included in one of the scripts that runs at login to set your terminal type for you:

© 2004 Hewlett-Packard Development Company, L.P.

```
if [ "$TERM" = "" -o \
             "$TERM" = "unknown" -o \
             "$TERM" = "dialup"  -o \
             "$TERM" = "network" ]
        then
                eval `ttytype -s -a`
        fi
        export TERM
```

PS1 The PS1 variable defines your shell prompt string. This, too, can be changed by the user. Some useful sample PS1 values are shown below:

```
export PS1='$ '                  # Use a simple "$ " prompt
export PS1='$PWD $'              # Include the user's pwd in the prompt
export PS1='$PWD ($LOGNAME) $'   # Include the user's username ,too
```

LPDEST LPDEST defines the user's default printer. The printer named in LPDEST takes precedence over the system-wide default printer configured by the system administrator. Examples:

```
export LPDEST=laser       # use "laser" as the default printer
export LPDEST=printera    # use "printera" as the default printer
```

PATH Every time the user enters a command, the shell must find the executable associated with the requested command. The PATH variable contains a ":" separated list of directories that the shell should search for executables. If users need access to new applications and utilities, you may need to modify their PATH variables. You can append a new directory to the user's PATH using syntax similar to the following syntax:

```
PATH=$PATH:/usr/local/bin      # adds /usr/local/bin
                               # to the existing PATH
```

The initial PATH variable value usually taken from the **/etc/PATH** file. Oftentimes installing an application automatically updates the **/etc/PATH** file for you, so it may not be necessary to update individual users' PATHs.

EDITOR Three variables must be defined if your users want to use command line editing:

```
export EDITOR=vi
export HISTFILE=~/.sh_history
export HISTSIZE=50
```

EDITOR defines the user's preferred command line editor. **emacs** and vi are the only allowed values. HISTFILE determines the file that should be used to log commands entered by the user. HISTSIZE determines the number of commands retained in the shell's command buffer.

These are just some of the more commonly defined environment variables that you can define for your users. Other environment variables are defined in the man page for the POSIX shell (**man 1 sh-posix**), and still others may be required by your applications.

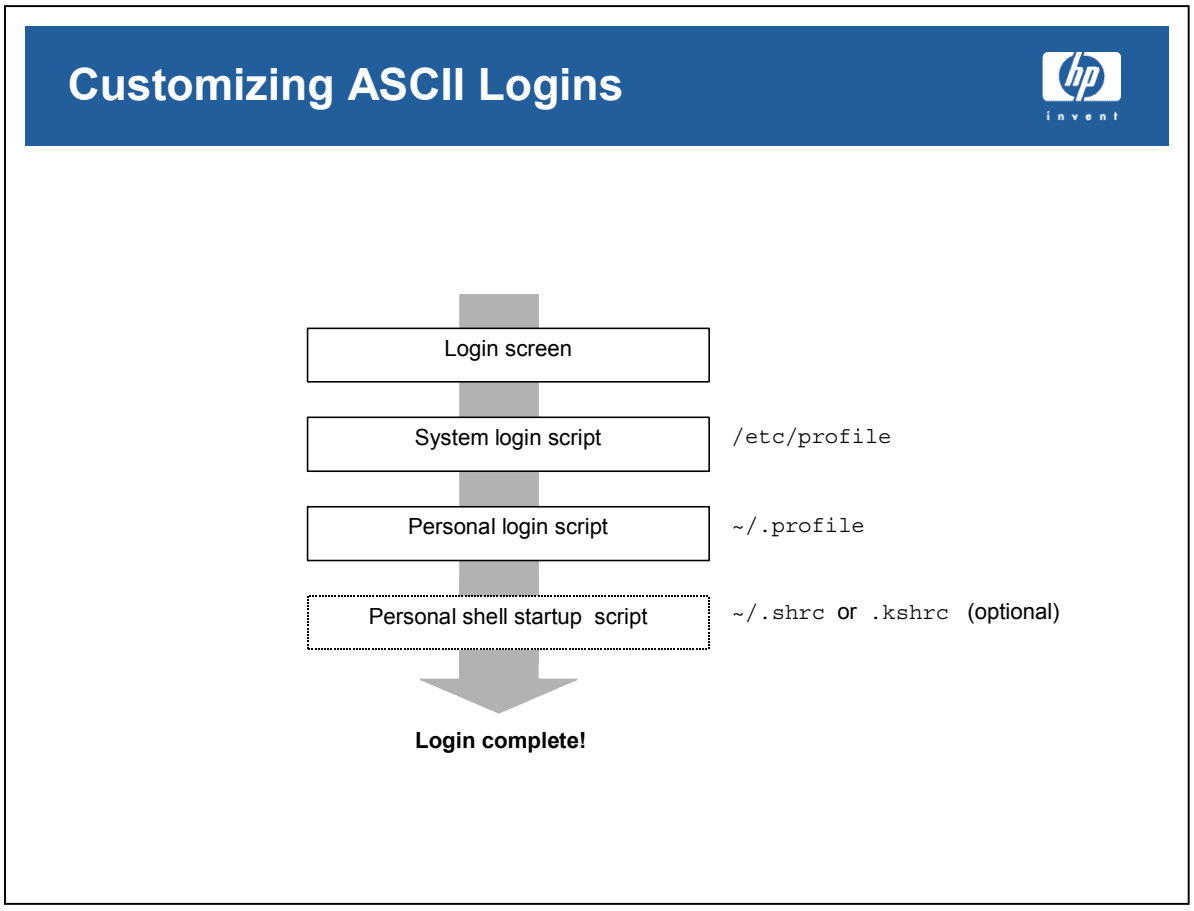
© 2004 Hewlett-Packard Development Company, L.P.

Environment variables can be set from the command line, but are more commonly defined in the login configuration files, which will be covered later in this chapter. You can view a list of currently defined environment variables by executing the **`env`** command:

```
# env
```


© 2004 Hewlett-Packard Development Company, L.P.

3–3. SLIDE: Customizing ASCII Logins



Student Notes

Although environment variables used to customize a user's login environment can be defined from the command line, they are more commonly defined by several scripts that execute during the login process. Typically, three scripts execute at the start of every ASCII terminal or **`telnet`** session to initialize the user's environment.

The first is a system-wide script maintained by the system administrator to define general environment variables required by all users.

The second is a local login script maintained by each user. Local login scripts can override or modify the system defaults on a user-by-user basis.

The optional additional local login script can define additional environment variables or set additional shell features. In the POSIX shell, this additional script is often used to define shell aliases.

© 2004 Hewlett-Packard Development Company, L.P.

POSIX and Korn Shell Initialization

The login script names vary from shell to shell. A detailed discussion of the login scripts used by the POSIX/ Korn shells follows below:

1. After the user enters a username and password, the **`/usr/bin/login`** script checks the **`/etc/passwd`** file to verify that the user has a valid account. If the user's username and password are correct, the login program launches a shell for the user.

2. Next, the newly launched shell executes a script called **`/etc/profile`**. **`/etc/profile`** is a POSIX/Korn shell script that is maintained by the system administrator to configure a default environment for all users. The script accesses the **`/etc/PATH`**, **`/etc/MANPATH`**, and **`/etc/TIMEZONE`** files to set initial values for the PATH, MANPATH, and TZ variables. The script attempts to define the TERM variable automatically, too. Since **`/etc/profile`** executes every time any user logs in, the administrator can modify this file to set global default environment variables for all users at login time.

3. Next, the user's personal **`~/.profile`** script executes. Each user has a **`.profile`** script that executes at login time to define additional environment variables, or to override the default environment variable values that the administrator defined in **`/etc/profile`**.

4. Finally, the shell looks for an environment variable called ENV. The ENV variable identifies a personal shell startup program that users may optionally choose to configure. POSIX shell users often create a **`~/.shrc`** shell startup script, while Korn shell users typically define a **`~/.kshrc`** shell startup script. Unlike the **`~/.profile`** script, which only executes at login, the shell startup script executes every time the user logs in, runs a shell script, opens a terminal emulator window, or launches a shell. The POSIX and Korn shell startup scripts are typically used to define shell aliases.

C-Shell Initialization

The C-Shell initialization process is similar, but somewhat different:

1. The **`/usr/bin/login`** program verifies the user's username and password.

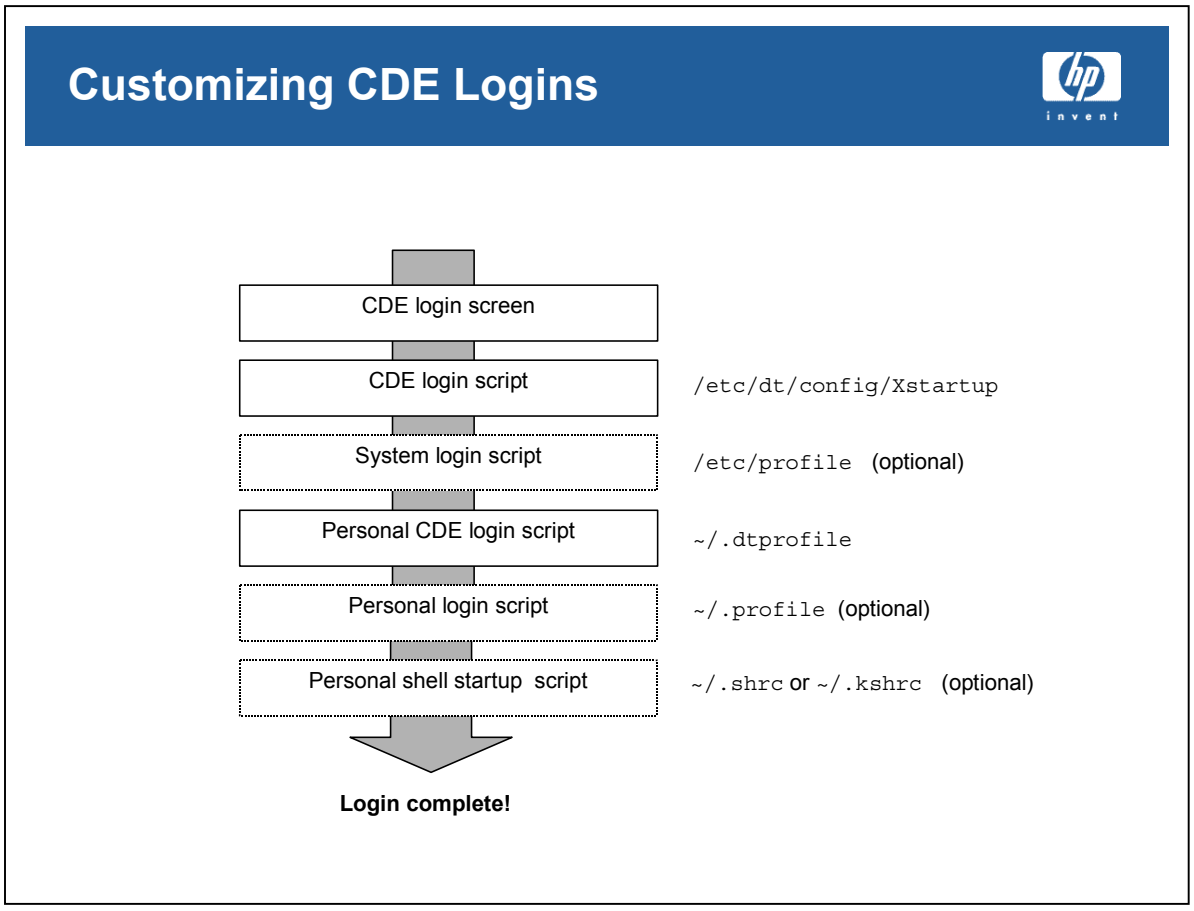
2. Next, the system-wide C-Shell startup script, **`/etc/csh.login`**, executes. This script serves the same purpose as the POSIX/Korn **`/etc/profile`** script.

3. At login, and with each subsequent launch of another C-shell, the user's personal **`~/.cshrc`** executes. The **`~/.cshrc`** script serves the same purpose as the POSIX **`~/.shrc`** or Korn **`~/.kshrc`**.

4. The user's personal **`~/.login`** script executes next. **`~/.login`** serves the same purpose as the POSIX/Korn **`~/.profile`** script. Note however that the **`.cshrc`** file will be executed *before* the **`.login`** script at login time, whereas **`~/.profile`** executes before **`~/.shrc`** in the POSIX login initialization process.

© 2004 Hewlett-Packard Development Company, L.P.

3–4. SLIDE: Customizing CDE Logins



Student Notes

The CDE login process is entirely different from the login process used by **`telnet`** and ASCII terminal sessions:

1. The **`/usr/dt/bin/dtlogin`** program is responsible for displaying the CDE login screen and validating user names and passwords. The color and appearance of the dtlogin screen may be customized by copying **`/usr/dt/config/Xconfig`** to **`/etc/dt/config/Xconfig`**, and manually editing the file.

2. Next, the **`/usr/dt/config/Xstartup`** program sets initial values for the DISPLAY, HOME, PATH, USER, SHELL, and TZ variables. This is a system-wide script that executes anytime any user logs in via CDE. If you wish, you may copy **`/usr/dt/config/Xstartup`** to **`/etc/dt/config/Xstartup`** and define additional environment variables as you would in **`/etc/profile`**.

3. The final script that executes by default at CDE login is the user's personal **`~/.dtprofile`**. This script contains little but comments by default, but the user may define additional environment variables in this script if desired.

© 2004 Hewlett-Packard Development Company, L.P.

Note that, by default, **/etc/profile**, **~/.profile**, and **~/.shrc** are not referenced during the CDE login process!

Executing /etc/profile During CDE Login

Although **/etc/profile** doesn't run by default during CDE login, you may want to force it to run so users have the same environment variables defined whether they log in via an ASCII terminal or CDE. To force CDE to run **/etc/profile**, create a link to **/etc/profile** in the **/etc/dt/config/Xsession.d** directory. CDE automatically executes all scripts in **Xsession.d** during the login process. Make sure the scripts in this directory have both "r" and "x" permission.

```
# ln /etc/profile /etc/dt/config/Xsession.d/profile
# chmod +rx /etc/dt/config/Xsession.d/profile
```

Executing ~/.profile During a CDE Login

By default, CDE executes the **~/.dtprofile** script at login rather than **~/.profile**. If you wish to execute users' **~/.profile** scripts in addition to **~/.dtprofile** during CDE login, do the following:

First, you must ensure that your users' **.profile** scripts are CDE-safe. Make sure that **stty**, **tset**, **ttytype**, and other ASCII-terminal related commands only execute during terminal or telnet logins. Do this by enclosing such commands within an if statement as shown below:

```
# vi ~/.profile
...
if [ ! "$DT" ]; then
    TERM=$(ttytype)
    stty erase "^H" kill "^U" intr "^C" eof "^D" susp "^Z"
    stty brkint hupcl ixon ixoff
fi
...
```

Once your users' **.profile** scripts are CDE-safe, you will need to ensure that the **DTSOURCEPROFILE** variable is set to "true" in **~/.dtprofile**. Setting this variable to true causes CDE to read a user's **.profile** in addition to their **.dtprofile**.

```
# vi ~/.dtprofile
...
export DTSOURCEPROFILE=true
...
```

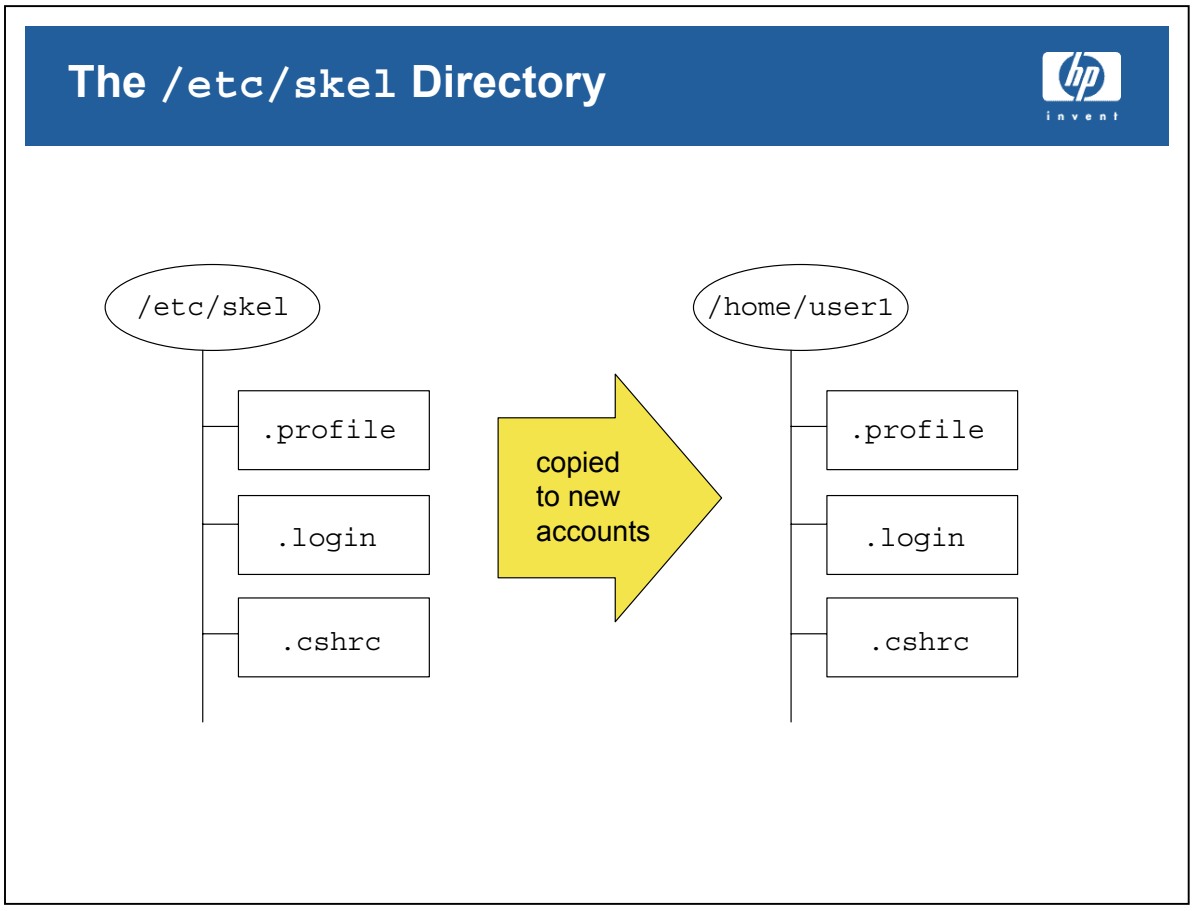
Executing ~/.shrc During a CDE Login

In order for **~/.shrc** to execute, the ENV variable must be defined. If you have already ensured that **.profile** executes at CDE login, then you may define ENV in **~/.profile**. If you have not configured **.profile** to execute at CDE login, then define ENV in **~/.dtprofile**. In either case, the variable should be defined as follows:

```
export ENV=~/.shrc
```


© 2004 Hewlett-Packard Development Company, L.P.

3–5. SLIDE: The `/etc/skel` Directory



Student Notes

When a new user account is created with SAM or **useradd**, default configuration files are copied from the **/etc/skel** directory to the new user's home. Several files are included in **/etc/skel** by default:

```
/etc/skel/.profile      # ksh/posix local login script
/etc/skel/.login        # csh local login script
/etc/skel/.cshrc        # csh additional login script
/etc/skel/.exrc         # vi startup configuration file
```

If you wish to change the default configuration files that are copied to new users' home directories, modify the files in **/etc/skel**. Note that changes made in **/etc/skel** won't affect existing users' home directories.

Additional files can be copied into **/etc/skel** as well, if your applications require configuration files in users' home directories.

© 2004 Hewlett-Packard Development Company, L.P.

NOTE: There is no CDE **.dtprofile** customization script in **/etc/skel**. The first time a user logs in via CDE, HP-UX attempts to copy either **/etc/dt/config/sys.dtprofile** (if it exists) or **/usr/dt/config/sys.dtprofile** to the user's **~/.dtprofile**. Use the following procedure to customize the default **.dtprofile**:

```
# cp -p /usr/dt/config/sys.dtprofile /etc/dt/config/sys.dtprofile
# vi /etc/dt/config/sys.dtprofile
```


© 2004 Hewlett-Packard Development Company, L.P.

3–6. LAB: Customizing User Accounts

Part 1: Customizing Accounts via `.profile`

1. Modify the appropriate configuration file so that root will have access to command line editing at next login.

2. Modify the appropriate configuration file so that the system administrator's shell prompt displays the present working directory and user name after the next login.

3. If your system is running CDE, modify root's CDE login script to ensure that the system consults your **`.profile`** at next login.

4. Linux users often use the **`bash`** shell rather than HP-UX. Try running the **`bash`** shell. What happens? Why?

   ```
   # bash
   ```

5. Where is the **`bash`** shell executable on the system? How can you find out?

© 2004 Hewlett-Packard Development Company, L.P.

6. What do you need to do to ensure that you can run **bash** by simply typing the command name?

7. Shell aliases are sometimes used to abbreviate commonly used commands. For instance, the **alias** below automatically pipes the output from **ps -ef** to the **grep** command. Try it!

```
# alias psgrep="ps -ef | grep"
# psgrep nfs
# psgrep init
```

   Add this useful alias to your personal POSIX shell startup script. If the script does not exist yet, create it. Make sure that **.profile** calls your personal POSIX shell startup script by adding the following line to **.profile**: **export ENV=~/.shrc**

8. Log out, then log back in again to see if your changes were successful.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Customizing New Accounts via `/etc/skel`

1. New users added to your system may appreciate having access to the same functionality you configured for **`root`** in Part 1 of this lab. How can you ensure that all new user accounts have the same custom functionality you configured for **`root`** in Part 1? Make it so.

2. **`.profile`** will be read only if your new users have a modified **`.dtprofile`**, too. How can you ensure that new home directories automatically get a copy of the modified **`.dtprofile`** that you created in Part I? Make it so.

3. Create a new user account using SAM or **`useradd`**; then log in as that new user and see if your customizations worked.

4. Do changes to **`/etc/skel`** affect already-existing accounts? Try logging in as **`user24`** to find out. Does **`user24`** have the custom prompt you configured in **`/etc/skel/.profile`**?

© 2004 Hewlett-Packard Development Company, L.P.

Module 4 — Guided Tour of the HP-UX File Hierarchy

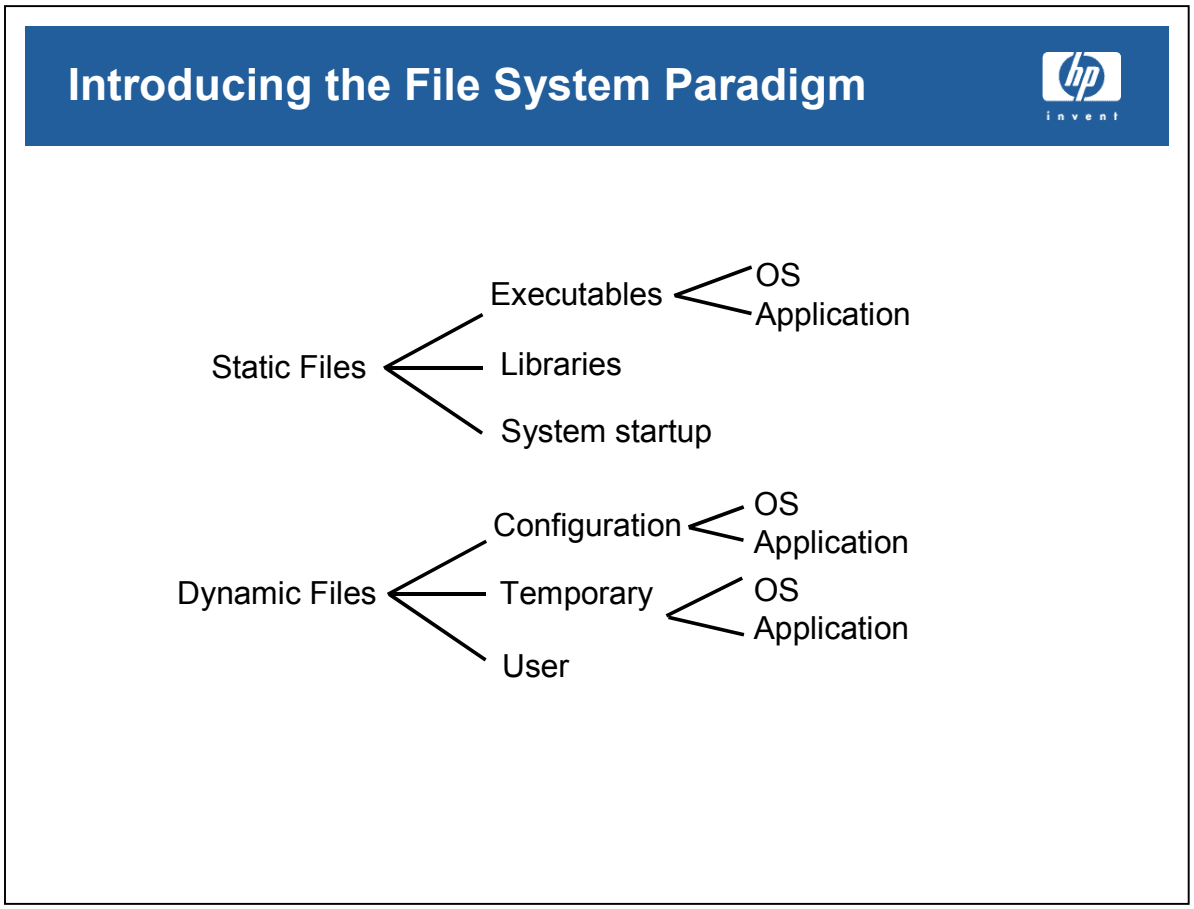
Objectives

Upon completion of this module, you will be able to do the following:

- Describe the reasons for separating dynamic and static file systems.
- Describe the key contents of **/sbin**, **/usr**, **/stand**, **/etc**, **/dev**, **/var** (OS-related directories).
- Describe the key contents of **/opt**, **/etc/opt**, and **/var/opt** (application-related directories).
- Use **find**, **whereis**, and **which** to find files in the HP-UX file system.

© 2004 Hewlett-Packard Development Company, L.P.

4–1. SLIDE: Introducing the File System Paradigm



Student Notes

Many HP-UX system administration tasks require the administrator to find and manipulate system and application configuration and log files. Understanding the philosophy behind the organization of the file system will ensure that you can successfully find the resources you need to perform administration tasks.

Files in the HP-UX file system are organized by various categories. *Static* files are separated from *dynamic* files. *Executable* files are separated from *configuration* files. This philosophy provides a logical structure for the file system and simplifies administration as well.

HP-UX Separates Static and Dynamic Portions of the File System

Files and directories in HP-UX may be categorized as *static* or *dynamic*. The contents of static files and directories rarely change, except when patching or installing the operating system or applications. Executable files, libraries, and system start-up utilities are all considered to be *static*.

Dynamic files and directories change frequently. They are stored in a separate portion of the file system. Configuration, temporary, and user files are all considered to be *dynamic*.

© 2004 Hewlett-Packard Development Company, L.P.

Separating dynamic and static data offers the following advantages:

- System backups are easier.
- Disk space management is simplified.

HP-UX Separates Executable Files from Configuration Files

Configuration data is kept separate from the executable code that uses that data. Separating executable files from configuration files offers the following advantages:

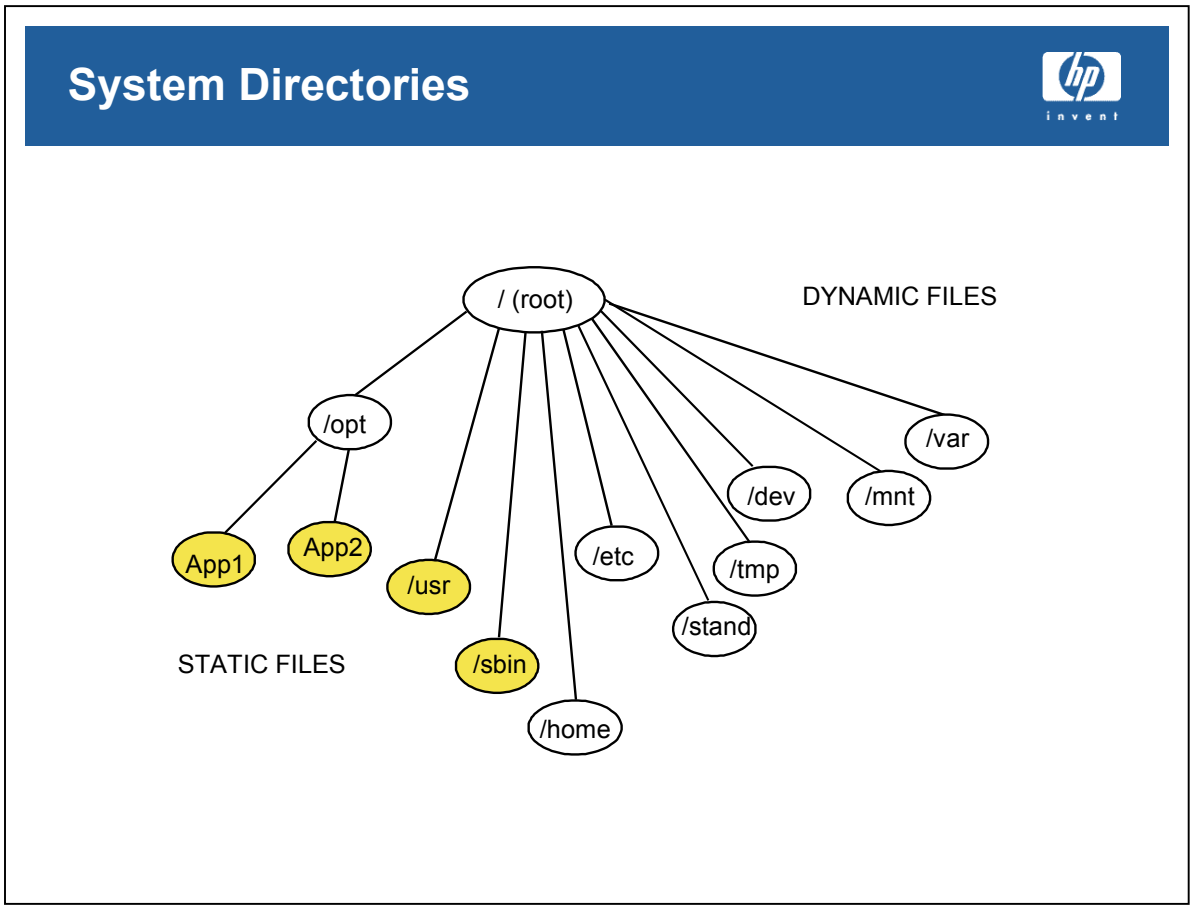
- Changes made to configuration data are not lost when updating the operating system.
- Executable files can be easily shared across the network, while host-specific configuration data is stored locally on each host.

HP-UX Follows the AT&T SVR4 Standard File System Layout

Though there are minor differences from vendor to vendor, the file system layout used in HP-UX is very similar to that used in other flavors of UNIX. This simplifies administration tasks for administrators with responsibilities on multiple vendors' machines.

© 2004 Hewlett-Packard Development Company, L.P.

4–2. SLIDE: System Directories



Student Notes

The shaded directories in the diagram on the slide contain *static* data, while unshaded directories in the diagram contain *dynamic* data. The sharable portion of the operating system is located beneath **`/usr`** and **`/sbin`**. Only the operating system can install files into these directories. Applications are located beneath **`/opt`**.

The directories **`/usr`**, **`/sbin`**, and the application subdirectories below **`/opt`** can be shared among networked hosts. Therefore, they must not contain host-specific information. The host-specific information is located in directories in the dynamic area of the file system.

General definitions for these directories are:

Directory	Definition
`/usr`	Sharable operating system commands, libraries, and documentation.
`/sbin`	Minimum commands needed to boot the system and mount other file systems.
`/opt`	Applications.

© 2004 Hewlett-Packard Development Company, L.P.

/etc	System configuration files. No longer contains executable files
/dev	Device files
/var	Dynamic information such as logs and spooler files (previously in **/usr**).
/mnt	Local mounts
/tmp	Operating system temporary files
/stand	Kernel and boot loader
/home	User directories

A Closer Look at /usr

The **/usr** directory contains the bulk of the operating system, including commands, libraries and documentation. The **/usr** file system contains operating system files, such as executable files and ASCII documentation.

The allowed subdirectories in **/usr** are defined below; no additional subdirectories should be created.

Examples of files that live here are

/usr/bin	Operating system user commands.
/usr/conf	Kernel configuration.
/usr/contrib	Unsupported contributed software.
/usr/lbin	Back-ends to other commands
/usr/local	User-contributed software.
/usr/newconfig	Default operating system configuration data files.
/usr/sbin	System administration commands.
/usr/share	Architecture independent sharable files.
/usr/share/man	Operating system man pages.
/usr/share/doc	Release notes.

A Closer Look at /var

The **/var** directory is for multipurpose log, temporary, transient, variable sized, and spool files. The **/var** directory is extremely variable in size, hence the name. In general, any files that an application or command creates at runtime, and that are not critical to the operation of the system, should be placed in a directory that resides under **/var**. For example,

© 2004 Hewlett-Packard Development Company, L.P.

/var/adm will contain log files and other runtime-created files related to system administration. **/var** will also contain variable size files like **crontabs**, and print and mail spooling areas.

In general, files beneath **/var** are somewhat temporary. System administrators that wish to free up disk space are likely to search the **/var** hierarchy for files that can be purged. Some sites may choose not to make automatic backups of the **/var** directories.

Examples of files that reside here are

/var/adm	Common administrative files and log files.
/var/adm/crash	Kernel crash dumps.
/var/mail	Incoming mail.
/var/opt/	Application-specific runtime files (e.g. logs, temporary files). Each application will have its own directory.
/var/spool	Spooled files used by subsystems such as **lp**, **cron**, software distributor.
/var/tmp	Temporary files generated by commands in the **/usr** hierarchy

A Closer Look at /var/adm

This directory hierarchy is used for common administrative files, logs, and databases. For example, files generated by syslog(3C), files used by cron(1M), and kernel crash dumps will be kept here and in subdirectories.

Examples of files that reside here are

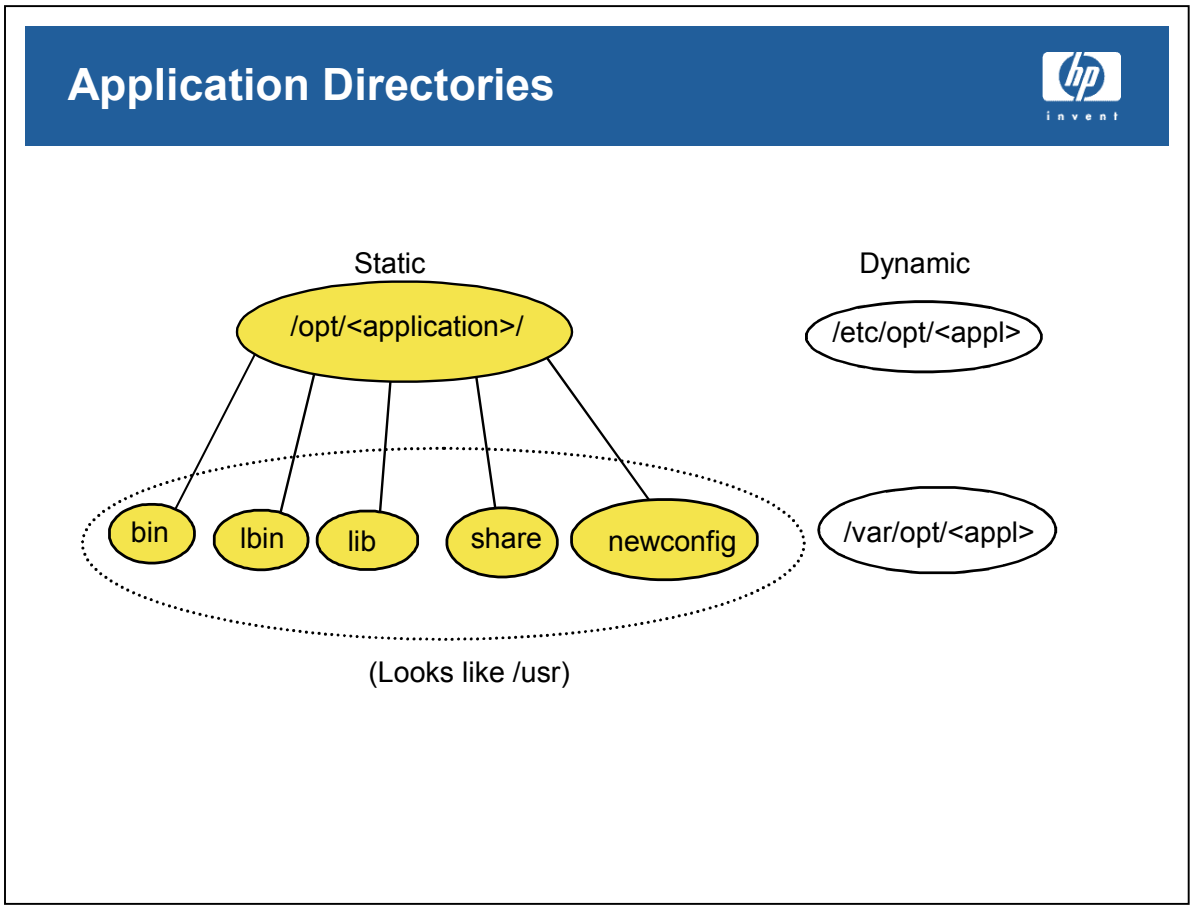
/var/adm/crash	Kernel crash dumps will be located in this directory.
/var/adm/cron	Used for log files maintained by **cron**. **cron** is a subsystem that allows you to schedule processes to run at a specific time or at regular intervals.
/var/adm/sw	Used for log files maintained by the Software Distributor.
/var/adm/syslog	System log files. Applications as well as the kernel can log messages here. The **syslogd** daemon is responsible for writing the log messages. The behavior of the **syslogd** daemon can be customized with the**/etc/syslog.conf** file. The name of the default log file is **/var/adm/syslog/syslog.log**. At boot time this file is copied to **OLDsyslog.log**, and a new **syslog.log** is started. The **syslog.log** file is an ASCII file.
/var/adm/sulog	This file contains a history of all invocations of the switch user command. **sulog** is an ASCII log file.

© 2004 Hewlett-Packard Development Company, L.P.

/var/adm/wtmp	This file contains a history of successful logins. This file is not ASCII. The **last** command is used to display this information. The **wtmp** file will continue to grow and should be trimmed by the administrator from time to time.
/var/adm/btmp	This file contains a history of unsuccessful logins. This file is not ASCII. The **lastb** command is used to display this information. The **btmp** file will continue to grow and should be trimmed by the administrator from time to time.
/etc/utmp	This file contains a record of all users logged onto the system. This file is used by commands such as **write** and **who**. This file is not an ASCII file and can not be directly viewed.

© 2004 Hewlett-Packard Development Company, L.P.

4–3. SLIDE: Application Directories



Student Notes

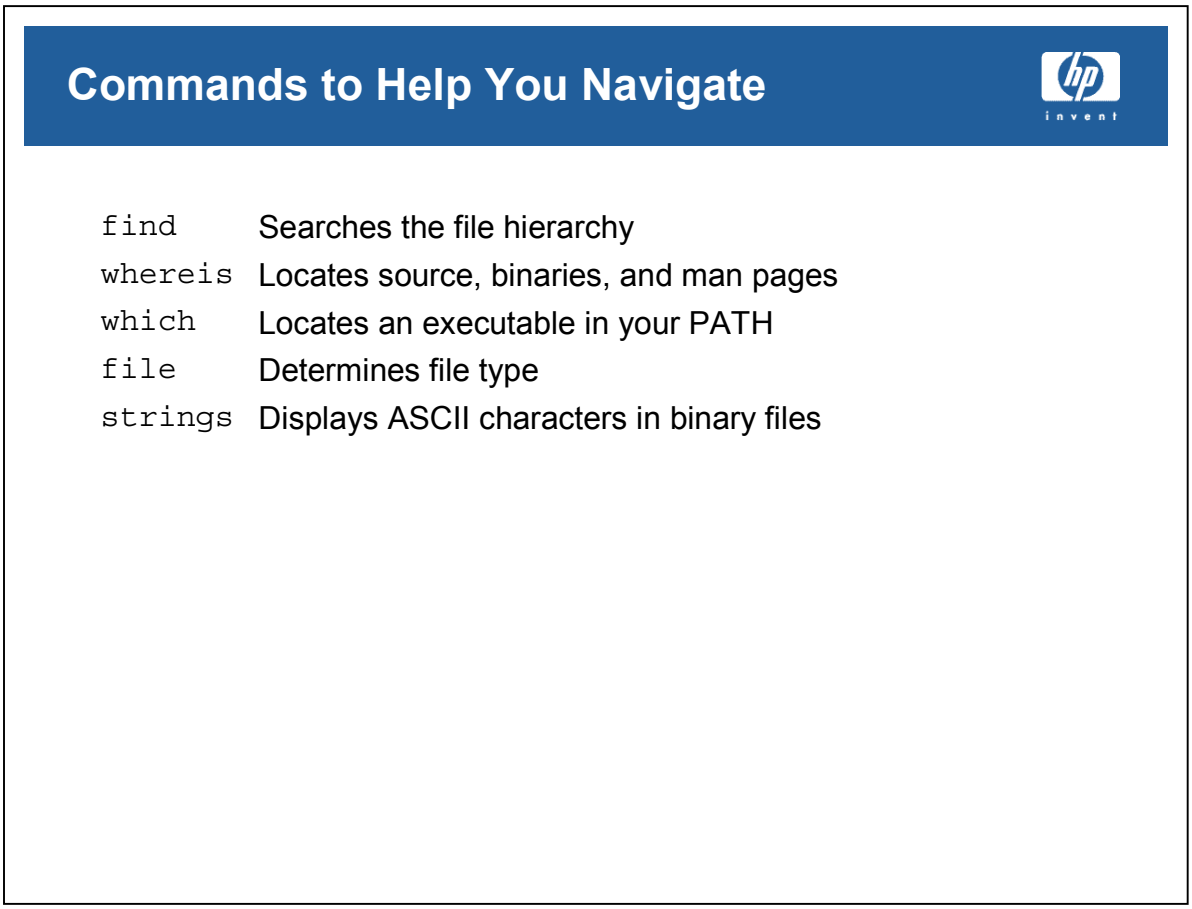
Each application will have its own subdirectory under **`/opt`**, **`/etc/opt`**, and **`/var/opt`**. The sharable, or static, part of the application is self-contained in its own **`/opt/application`** directory, which has the same hierarchy as the operating system layout:

`/opt/application/bin`	User commands.
`/opt/application/share/man`	**`man`** pages.
`/opt/application/lib`	Libraries.
`/opt/application/lbin`	Back end commands.
`/opt/application/newconfig`	Master copies of configuration files.

The application's host-specific log files are located under **`/var/opt/application`**, and host-specific configuration files are located under **`/etc/opt/application`**.

© 2004 Hewlett-Packard Development Company, L.P.

4–4. SLIDE: Commands to Help You Navigate



Student Notes

As a system administrator, you will need to reference files in directories all over the HP-UX file system. HP-UX offers several tools for finding the files and executable files you need to perform administration tasks.

The `find` Command

The **`find`** command is a powerful tool for system administrators. It searches the file hierarchy starting at a specified point and finds files that match the criteria you select. You can search for files by name, owner, size, modification time, and so on. **`find`** also allows you to execute a command with the files found used as an argument.

Examples

- Find all files belonging to the user **`greg`**:

```
# find / -user greg
```

- Find files in **`/tmp`** that have not been accessed in 7 days:

```
# find /tmp -type f -atime +7
```


© 2004 Hewlett-Packard Development Company, L.P.

- Remove core files:

```
# find / -name core -exec rm –i {} \;
```

The `whereis` Command

The **`whereis`** command is useful when you receive "not found" error messages. It searches a predefined list of directories. By default, **`whereis`** looks for source, binaries, and **`man`** pages. You can limit the search to binary files by using the **`-b`** option.

Example

```
# whereis -b sam
sam: /usr/sbin/sam
```

The `which` Command

The **`which`** command is useful for determining which version of a command will be used. Some commands have multiple homes. Which version you execute is determined by the order of the directories in your PATH variable.

The `file` Command

The **`file`** command performs a series of tests on a file and attempts to classify it. It can be useful for determining if a command is a shell script or a binary executable.

Examples

```
# file /sbin/shutdown
/sbin/shutdown: s800 shared executable

# file /etc/passwd
/etc/passwd: ascii text
```

The `strings` Command

The **`strings`** command is useful when trying to find information in a binary file. It will print any printable characters in the file.

© 2004 Hewlett-Packard Development Company, L.P.

4–5. LAB: HP-UX File System Hierarchy

Directions

Answer all the questions below.

1. Which of the following directories are dynamic?

   `/etc`

   `/usr`

   `/sbin`

   `/dev`

   `/tmp`

2. Viewing a report on your disk space usage, you note that **`/usr`**, **`/var`**, and **`/opt`** are all nearing 90% capacity. Which of these directories should you be most concerned about? Why?

3. Match the directory with its contents:

1. **`/usr/share/man`**	A. kernel, boot loader
2. **`/stand`**	B. system configuration files
3. **`/var/adm`**	C. shareable operating system commands
4. **`/etc`**	D. man pages
5. **`/usr`**	E. application directories
6. **`/opt`**	F. common admin files and logs

© 2004 Hewlett-Packard Development Company, L.P.

4. Where would you expect to find the **`cp`** and **`rm`** OS user executables? See if you are correct.

5. Where would you expect to find the **`sam`**, **`useradd`**, and **`userdel`** executables? See if you are correct.

6. The **`pre_init_rc`** utility executes in the early stages of the system start-up procedure to check for file system corruption. Where would you expect to find this executable? See if you are correct.

7. There is a system log file that maintains a record of system shutdowns. Where would you expect to find the shutdown log file? See if you are correct.

8. In which directory would you expect to find the "hosts" configuration file, which contains network host names and addresses? See if you are correct.

© 2004 Hewlett-Packard Development Company, L.P.

9. Though many utilities and daemons maintain independent log files, many daemons and services write their errors and other messages to a log file called **`syslog.log`**. See if you can find the path for this file, then check to see if any messages have been written to the file in the last day.

10. Find all of the directories (if any) under **`/home`** that are owned by root.

11. (Optional) Find all the files under **`/tmp`** that haven't been accessed within the last day.

12. (Optional) Find all the files on your system that are greater than 10000 bytes in size. If you needed to make some disk space available on your system, would it be safe to simply remove these large files?

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 5 — Configuring Hardware

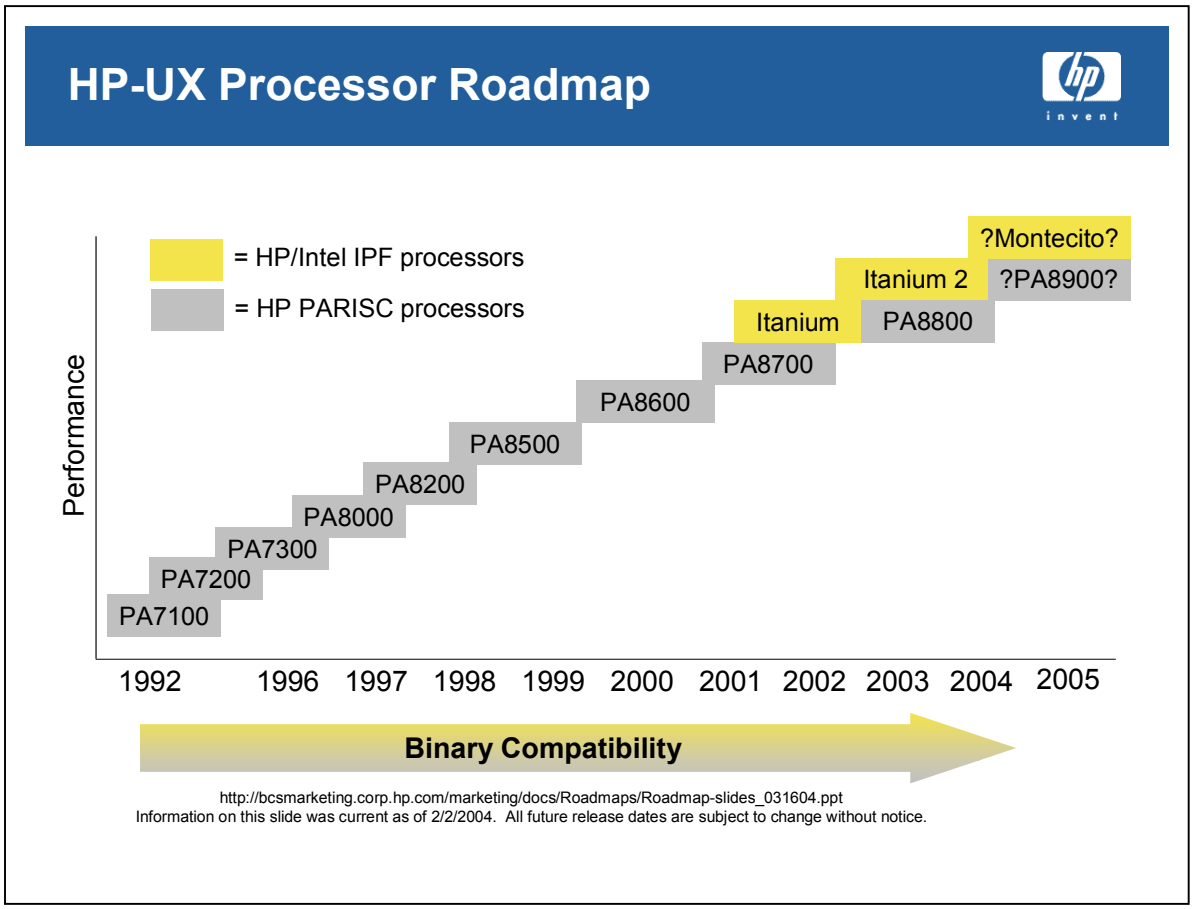
Objectives

Upon completion of this module, you will be able to do the following:

- List HP's current processor, hardware, and HP-UX operating system offerings
- Describe the major hardware components of an HP-UX system
- Describe the components of an HP-UX hardware path
- Display a system's hardware components with **`ioscan`**, **`rad`**, and **`olrad`**
- Determine SCSI devices' card instance, target, and LUN addresses from **`ioscan`**
- Describe the role and purpose of Node Partitions in HP-UX
- Describe the role and purpose of Virtual Partitions in HP-UX
- Add or replace a PCI interface cards with and without HP OLAR functionality
- Add or remove pluggable or non-hot-pluggable devices

© 2004 Hewlett-Packard Development Company, L.P.

5–1. SLIDE: HP-UX Processor Roadmap



Student Notes

HP's HP-UX systems are based on two processor families.

The PA-RISC Processor Family

Since 1986, HP has been building HP-UX systems based on our proprietary Precision Architecture RISC (PA-RISC) technology. PA-RISC uses Reduced Instruction Set Computing (RISC) principles to provide high performance, and high reliability. HP has offered several iterations of its PA-RISC technology over the years. The early PA7000 series of chips were built on a 32-bit architecture, while the newer PA8000 series chips are built on a 64-bit architecture.

HP's latest servers are based on the new PA8800 processor, which offers 20-40% better performance than the previous PA8700 processor. In addition, a single PA8800 processor module may contain one or two processor cores, thus allowing twice as many processors in a single system. The PA8800 Superdome supports up to 64 processor modules, a total of up to 64 processors, and up to 512GB of memory!

HP continues to develop the PA-RISC processor family, and still plans to introduce additional enhanced PA-RISC chips in the future.

© 2004 Hewlett-Packard Development Company, L.P.

The Itanium Processor Family (IPF)

HP now ships a full complement of Itanium 2 based servers and workstations, too. The Itanium processor chip was developed by an HP/Intel partnership to serve as a successor to both the IA32 architecture currently used in most Intel-based servers, and the PA-RISC technology that has been the cornerstone of HP's HP-UX server and workstation offerings.

The Itanium 2 architecture uses a variety of techniques to increase **parallelism**—the ability to execute multiple instructions during each machine cycle. Parallelism improves performance because it allows multiple instructions to be executed simultaneously. The Itanium architecture is designed to make certain the processor can execute as many instructions per cycle as possible.

A key to the high performance of Itanium chips is the design philosophy at the heart of the Itanium processor family, Explicitly Parallel Instruction Computing (EPIC). The EPIC philosophy is a major reason why Itanium processors are different from other 64-bit processors, providing much higher instruction-level parallelism without unacceptable increases in hardware complexity. How does EPIC achieve such performance? By placing the burden of finding parallelism squarely on the compiler. Although processor hardware can extract a limited sort of parallelism, the best approach is to let the compiler, which can see the whole code stream, find the parallelism and make global optimizations. The compiler communicates this parallelism explicitly to the processor hardware by creating a three-instruction bundle with directions on how the instructions should be executed. The hardware focuses almost entirely on executing the code as quickly as possible.

The EPIC architecture, together with several other architecture innovations, gives Itanium a significant advantage over both IA32 and 64-bit RISC systems. As co-developer of the Itanium architecture, HP has been able to take the lead in bringing production-ready Itanium-based servers and workstations to market.

Itanium supports three major operating systems currently: HP-UX, Linux, and 64-bit Windows. In the future, VMS and HP's Non-Stop server operating systems will also be supported.

PA-7000, PA-8000, and Itanium2 Compatibility

Since the introduction of the HP-UX operating system on PA-RISC in 1986, HP has maintained a strong commitment to delivering source and binary compatibility with each update of the operating system. Most HP-UX applications that were compiled for older 32-bit PA7000 systems still run on the current 64-bit PA8800 without modification.

In fact, many binaries compiled for PA-RISC will run without modification on Itanium-based systems, too! These applications execute under a software-enabled compatibility mode subsystem called "Aries." The Aries subsystem consists of a dynamic translator that translates PA-RISC instruction sequences into the native instruction set, and a system interface layer that emulates the PA-RISC system API. Aries is invoked automatically by the HP-UX kernel whenever a PA-RISC application is executed, and is transparent to the user of that application. Some applications may be dependent on specific features of the hardware or the operating system, and may therefore be unsupported in compatibility mode.

The Aries subsystem does not support applications that contain a mixture of PA-RISC code and native code within the same program ("mixed-mode" programs). Such applications must be partitioned manually into two separate programs, one that consists entirely of PA-RISC

© 2004 Hewlett-Packard Development Company, L.P.

code, and another that consists entirely of native code. For a complete discussion of this technique, see the white paper "Mixed-mode on Itanium®-based platforms—why and how." More information is also available at **http://www.hp.com/go/hpux**.

Because binary compatibility is achieved through dynamic instruction translation, the performance of an application running in compatibility mode will be less than that of the same application recompiled and running in native mode. Applications that are largely interactive or I/O intensive should experience little to no noticeable degradation in performance, while those that perform heavy computation may run noticeably slower than on a recent PA-RISC system. HP recommends recompilation for all applications and libraries where performance is a concern.

Applications compiled for Itanium-based systems cannot be executed on PA-RISC systems.

To learn more about HP-UX compatibility issues, search for the HP-UX compatibility white paper (document 5981-7108EN) on **http://www.hp.com**.

Determining your Processor Type

On PA-RISC systems, you can determine your processor type via the SAM system properties screen.

```
# sam -> Performance Monitors -> System Properties -> Processor
```


© 2004 Hewlett-Packard Development Company, L.P.

On IPF systems, you can determine your processor type and configuration via the **machinfo** command.

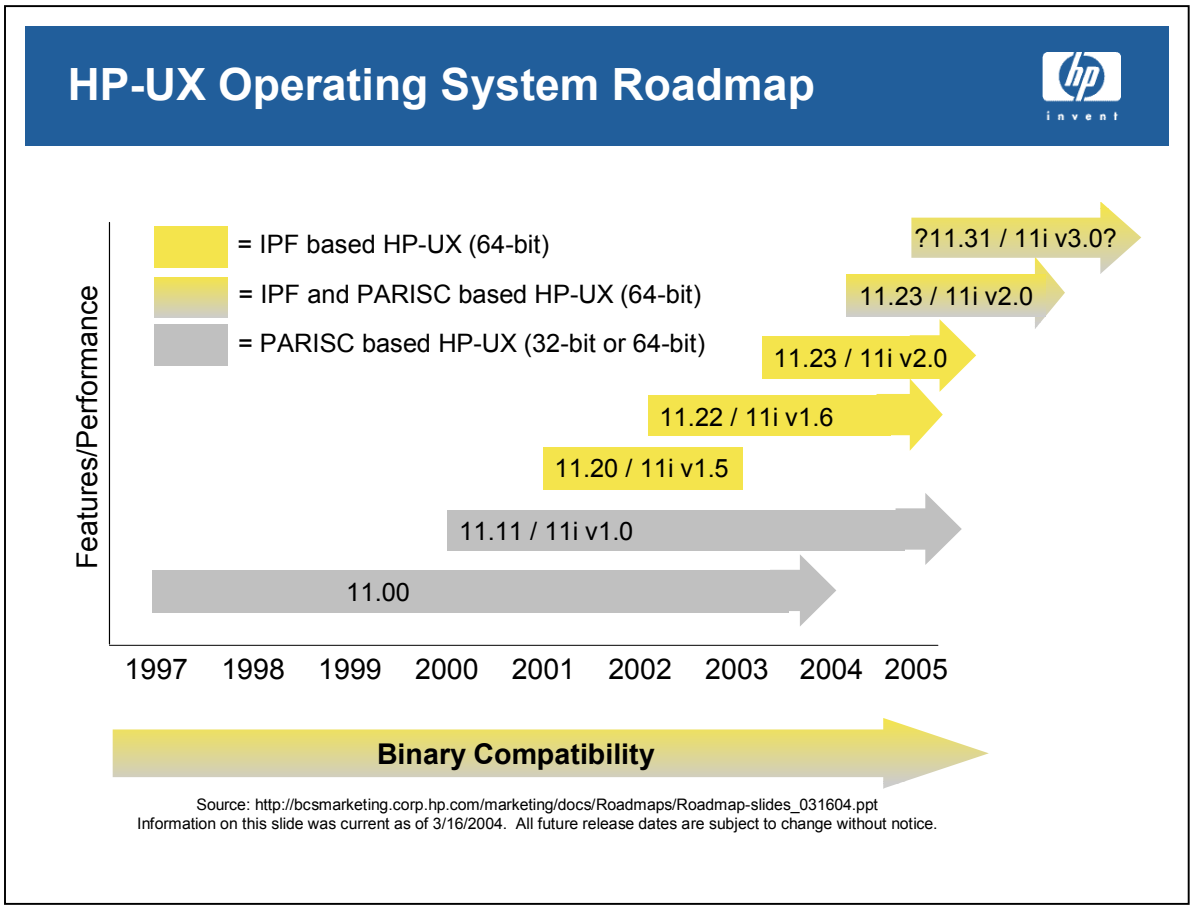
```
# machinfo
CPU info:
   Number of CPUs = 2
   Clock speed = 900 MHz
   CPUID registers
      vendor information =       "GenuineIntel"
      processor serial number =  0x0000000000000000
      processor version info =   0x000000001f000504
         architecture revision:       0
         processor family:           31   Intel(R) Itanium 2 Family
         processor model:             0   Intel(R) Itanium 2 processor
         processor revision:          5   stepping B1
         largest CPUID reg:           4
      processor capabilities =   0x0000000000000001
         implements long branch:      1
   Bus features
      implemented =  0xbdf0000060000000
      selected    =  0x0000000040000000
         Bus Lock Signal masked

Cache info:
   L1 Instruction: size =   16 KB, associativity = 4
   L1 Data:        size =   16 KB, associativity = 4
   L2 Unified:     size =  256 KB, associativity = 8
   L3 Unified:     size = 3072 KB, associativity = 12

Memory = 4088 MB (3.992188 GB)

Firmware info:
   Firmware revision = 01.82
   FP SWA driver revision: 1.C
   BMC version: v01.40

Platform info:
   model string =          "ia64 hp server rx2600"
   machine id number =
   machine serial number = US21729721

OS info:
   sysname  = HP-UX
   nodename = r848c39
   release  = B.11.22
   version  = U (unlimited-user license)
   machine  = ia64
   idnumber =
   /stand/vmunix _release_version:
@(#)   $Revision: vmunix:    B.11.22_LR FLAVOR= DFILE=dfile.em_64.isu
Thu Aug  $
```

For More Information

For more information on HP's Itanium strategy, visit our Itanium web page at **http://www.hp.com/go/itanium/**.

© 2004 Hewlett-Packard Development Company, L.P.

5–2. SLIDE: HP-UX Operating System Roadmap



Student Notes

All of HP's PA-RISC and Itanium based servers and workstations run HP's HP-UX Operating System. HP-UX has been through many revisions since it was first released, and each revision has added additional features and functionality. In a 2002 DH Brown survey, HP-UX was rated the #1 UNIX operating system for its functionality, serviceability, security, and manageability (See **`http://www.hp.com/products1/unix/operating/infolibrary/reports/2002Unix_execsummary.pdf`** for details).

Several versions of HP-UX are currently available:

- HP-UX 11.00 was the first 64-bit version of HP-UX, and is still supported on many of our PA-RISC servers and workstations. A 32-bit version of 11.00 is available for some older PA-RISC systems. Both the 64-bit and 32-bit versions of 11.00 are able to run 32-bit application binaries.

- HP-UX 11i v1.0 (also known as 11.11) was released in 2000, and is the most current, supported version of HP-UX on 32-bit and 64-bit PA-RISC systems. Both the 64-bit and 32-bit versions of 11.00 are able to run 32-bit application binaries.

© 2004 Hewlett-Packard Development Company, L.P.

- HP-UX 11i v1.5 (also known as 11.20) was released in 2001, and was the first HP-UX release that supported IPF based systems. 11i v1.5 was never supported on PARISC platforms, and hasn't been supported on the IPF platform, either, since February 2003.

- HP-UX 11i v1.6 (also known as 11.22) was released in 2002. This version was only released for IPF, and continues to be supported on IPF systems today.

- HP-UX 11i v2.0 (also known as 11.23) is the most current version of HP-UX supported on IPF systems, and offers many of the same OS features that are supported on 11i v1.0 for PA-RISC, and more. 1i v2.0 also offered several important new features such as an improved kernel configuration mechanism, compressed crash dumps, and a variety of performance enhancements.

- HP-UX 11i v2.0 will be extended to the PA-RISC platform later in 2004. This will be the first HP-UX release supported on both IPF and HP-UX.

- HP-UX 11i v3.0 (also known as 11.31) will support many new features such as HP's AdvFS file system and enhanced clustering technology. The final release date for 11i v3.0 hasn't yet been announced.

Compatibility

HP has provided binary compatibility for its PA-RISC systems since 1986. Most HP-UX applications that were compiled for 32-bit PA7000 systems run on the current 64-bit PA8000 systems without modification.

Because Itanium provides backward compatibility with the PA-RISC architecture, most existing HP-UX applications should run on HP-UX 11i v1.5, v1.6, and v2.0 without modification. To fully harness the power of Itanium, however, you may choose to recompile applications to native Itanium binaries.

Updating from 11.00 to 11i v1.0, or from 11i v1.6 to v2.0 is a relatively straightforward process using HP's **`update-ux`** command. For more information, consider attending HP Education's "hands-on" with 11i v2.0 course (u5523s), or read the Install/Update manuals that are available for each release on **`http://docs.hp.com`**.

Determining Your Operating System Version

You can determine your processor type via the uname command.

```
# uname –r or
# uname -a
```

For More Information

For more information a description of some of the key features of HP-UX visit **`http://www.hp.com/go/hpux`**.

© 2004 Hewlett-Packard Development Company, L.P.

5–3. SLIDE: HP-UX Hardware Product Overview

HP-UX Hardware Product Overview

	PA-RISC	IPF
Servers (800s)	rp Series Servers and Superdome Servers	rx Integrity Series Servers and Integrity Superdome Servers
Workstations (700s)	B/C/J Series Workstations	zx Series Workstations

Student Notes

HP offers a wide variety of HP-UX systems to meet the needs of any IT environment, from single processor Itanium/IPF and PA-RISC workstations, to the 128-processor Superdome server with 1TB of RAM and 192 I/O expansion slots.

In 2001, HP implemented a new naming scheme for its servers and workstations to more clearly communicate the architecture of each product line.

- The rp series systems are rack-mounted, entry-level to mid-range servers with 1 to 16 PA-RISC processors.
- The HP 9000 Superdome is our high-end, standalone server that supports up to 128 PA-RISC processors, 1TB of RAM, and 192 PCI expansion slots.
- The rx series systems are rack-mounted, entry-level to mid-range IPF servers that support up to 16 Itanium 2 CPUs.
- The new Integrity Superdome server is our high end IPF server offering, which supports up to 64 Itanium-2 processors, 512GB of RAM, and 192 PCI expansion slots.

© 2004 Hewlett-Packard Development Company, L.P.

- The zx series workstations support up to 2 Itanium 2 CPUs.
- The B, C, and J series workstations support up to 2 PA-RISC CPUs.

For the most current information on HP's hardware products, visit the HP Web Site at **`http://www.hp.com`**, or contact your local HP sales representative.

Upgrade Paths

Customers often find that as their business grows, their transaction volumes demand greater capacity and performance. Hewlett-Packard provides a comprehensive upgrade program that protects customers' investments in hardware, software and training. The upgrade program includes simple board upgrades, system swaps with aggressive trade-in credits, and 100 percent return credit on most software upgrades.

iCOD

Hewlett-Packard's Instant Capacity On Demand (iCOD) program provides access to additional CPU resources beyond the number originally purchased with your server. These iCOD CPUs reside in the purchased system, but aren't initially activated. Customers that wish to implement iCOD must purchase the iCOD software product, and pay a nominal "Right-To-Access Fee" for each iCOD processor.

When additional capacity is required, the iCOD CPUs can be "activated" via the **`icod_modify`** command. No physical hardware changes are required to enable the new CPU, nor is any HP Engineer intervention required. Once an iCOD processor has been activated, its processing power becomes fully accessible to the system, an email message is automatically sent to HP, and you will be billed for the additional CPU.

PPU

Customers with cyclical load requirements may be better served by HP's Pay Per Use (PPU) utility pricing program.

PPU allows customers to better align their information technology (IT) costs with revenues as technology use fluctuates over time. With HP Pay per Use pricing, customers can meet unexpected capacity needs and pay for that capacity only when it is used. Customers are billed based on the actual computing power used during any given month, allowing service-oriented companies to bill their end-users in a highly accurate, utility-like manner.

HP's utility infrastructure includes remote management and metering technology, so HP Pay per Use customers receive monthly statements that reflect the average of peaks and down-cycles in usage over a given month. The invoice reflects a base payment and a variable, usage-based payment, similar to the payment structure of a typical cellular phone service.

Determining your System Model Type

You can determine your system's model type and number via the **`model`** command.

```
# model
ia64 hp server rx2600
```

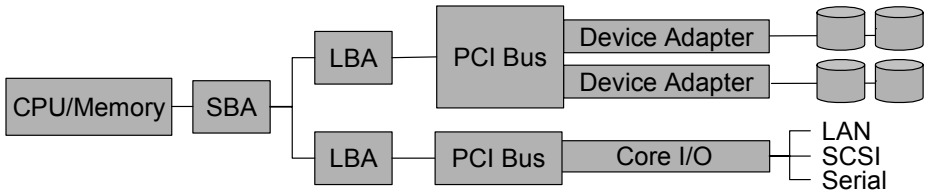

© 2004 Hewlett-Packard Development Company, L.P.

5–4. SLIDE: Hardware Component Overview

Hardware Component Overview

Every recent HP-UX system has several hardware components:

- One or more PA-RISC or Itanium CPU chips for processing data
- One or more memory modules for storing data
- One or more System / Local Bus Adapters that provide connectivity to I/O expansion buses
- One or more PCI I/O expansion buses with slots for add-on Device Adapter cards
- One or more Device Adapter cards with ports for cabling peripheral devices
- One or more Core I/O cards with built-in support for LAN, console, and boot disk connectivity



Student Notes

Every recent HP-UX system has several hardware components:

- PA-RISC or Itanium CPU chips for processing data
- Memory modules for storing data
- System and Local Bus Adapters that provide connectivity to I/O expansion buses
- PCI I/O expansion buses with slots for add-on Device Adapter cards
- Device Adapter cards with ports for cabling peripheral devices
- Core I/O cards with built-in support for LAN, console, and boot disk connectivity

Each of these components will be described in detail later in the chapter.

© 2004 Hewlett-Packard Development Company, L.P.

5–5. SLIDE: CPU, Memory, and Cell Boards

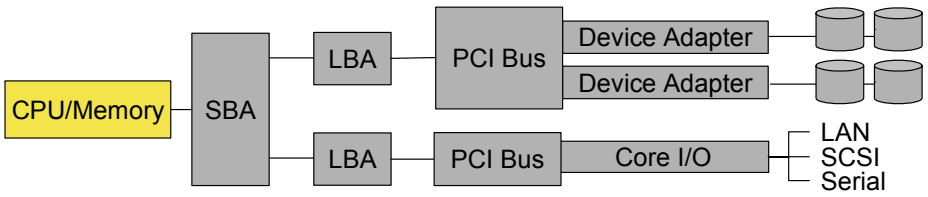
CPU, Memory, and Cell Boards

Processor-based systems

- HP-UX workstations and low-end servers use a processor-based architecture
- CPU and memory are physically hosted on a single system board

Cell-based systems

- High-end HP-UX servers use a cell based architecture
- Each system is comprised of one or more cell boards
- Each cell board contains one or more CPUs, and a portion of the system's memory
- All cell boards are interconnected via low latency system crossbars



Student Notes

CPU and memory chips are at the heart of every computer system.

Processor-Based Systems and the zx1 Chipset

HP's workstations and low-end servers with 1 to 4 processor modules are all "processor-based" systems. In processor-based servers, the major components of the system are discrete components, residing directly on the system bus. These systems include CPU and memory boards, as well as bus interface boards and I/O adapters, plus a management interface, referred to as a service or management processor.

HP's PA8800 and Itanium2 processor-based systems all utilize HP's zx1 scalable processor chipset. The zx1 chipset includes three modular components:

- The HP zx1 Memory and I/O Controller connects to the processor bus and contains the memory controller and the I/O cache controller. It interfaces to the processor bus and provides a low latency connection to memory either directly or through zx1 Scalable Memory Expander. It can connect up to 12 Memory Expanders for quadruple the base memory capacity at the same time that eight I/O adapters are handling 4GB/sec of I/O bandwidth.

© 2004 Hewlett-Packard Development Company, L.P.

- The HP zx1 I/O Adapter chip is a scalable solution designed to support PCI, PCI-X and AGP bus architectures. It provides scalable I/O implementation for a wide variety of systems.

- The HP zx1 Scalable Memory Expander is an optional component used to increase memory capacity up to four times and increase bandwidth to the main memory to 12.8 GB/s. Acting like as a memory hub, it decreases the number of signal loads on the memory bus, thereby allowing the system to dial up its memory transfer rate.
  The zx1 chipset supports both PA8800 and Itanium2 processors, and provides a simple upgrade path for customers who prefer to continue using PA-RISC today, but plan to upgrade to Itanium in the future.

Cell-Based Systems and the sx1000 Super-Scalable Chipset

HP's mid- to high-end servers use a cell-based architecture. These servers have two or more **cell boards**. Each cell board hosts one or more CPUs, up to 16GB of RAM, and a cell controller that facilitates communication between the cell board components, crossbar controller, and PCI I/O chassis. The system cell boards are interconnected by a low latency crossbar that provides efficient data transfers between cell boards.

HP's cell-based PA8800 and Itanium2 systems utilize the sx1000 chipset rather than the zx1 chipset to provide greater scalability and throughput. The HP sx1000 Chipset consists of five distinct chips:

- a cell controller, which routes data between the processors on a cell board
- a memory controller, which manages the memory subsystem
- a PCI-X host bridge adapter, which provides connectivity to one or more PCI-X I/O buses
- a system bus adapter, which provides connectivity to the PCI-X host bridge
- and a crossbar, which manages communication with other cell boards on the system

HP's high-end Integrity Superdome system is based on the sx1000 chip and currently supports up to 64 processors, 512GB of RAM, and 192 PCI expansion slots.

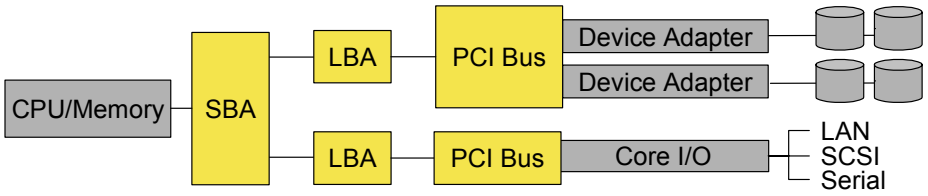
The sx1000 chipset supports both PA8800 and Itanium2 processors, and provides a simple upgrade path for customers who prefer to continue using PA-RISC today, but plan to upgrade to Itanium in the future.

© 2004 Hewlett-Packard Development Company, L.P.

5–6. SLIDE: SBAs, LBAs, and PCI Expansion Buses

SBAs, LBAs, and PCI Expansion Buses

- System and Local Bus Adapters provide connectivity to I/O expansion buses
- PCI buses provide one or more slots for device adapter cards
- HP supports a variety of PCI bus types and bandwidths up to ~1GB/sec
- All current HP PCI slots support “adaptive signaling technology” for maximum flexibility
- Most HP PCI slots have dedicated buses to minimize downtime and maximize performance
- PCI buses on high-end servers support HP’s “Online Addition & Replacement” functionality



Student Notes

Every cell or system board has a System Bus Adapter (SBA) that provides connectivity between the system processor and the system’s I/O buses.

The SBA connects to one or more Local Bus Adapters (LBAs) on the system’s I/O backplane via a high-speed communications channel known as a “rope”. Some LBAs have a single rope connection to the SBA. Other LBAs have two ropes to the SBA to increase bandwidth.

Each LBA provides connectivity to one or two Peripheral Component Interconnect (PCI) buses. PCI is a bus architecture that provides high-speed connectivity to and between device adapter cards. PCI was developed by Intel, but has become an industry standard that is used on many platforms.

PCI Slot Types

Since it was first introduced, the PCI standard has been modified several times to accommodate the greater bandwidth and shorter response times demanded from the input/output (I/O) subsystems of enterprise computers. The table below shows the evolution of PCI technology.

© 2004 Hewlett-Packard Development Company, L.P.

Specification	HP Terminology	Bus Width	Bus Frequency	Bandwidth
PCI 2.2	PCI	32 bits	33.3 MHz	133 MB/s
PCI 2.2	PCI 2x / Turbo	64 bits	33.3 MHz	266 MB/s
PCI 2.2	PCI 4x / Twin Turbo	64 bits	66.6 MHz	533 MB/s
PCI-X	PCI-X	64 bits	133.3 MHz	1066 MB/s

HP's latest servers, which are based on the zx1 and sx1000 chipsets, utilize PCI-X technology to provide the best possible performance. HP's PCI-X slots also offer backwards compatibility with earlier PCI technology. Thus, any supported HP PCI device adapter card may be installed in any HP PCI-X slot. Using HP's adaptive signaling technology, each slot adapts to the data width (64-bit or 32-bit) and frequency (33MHz, 66MHz, or 133MHz) of the installed card. For best performance, it is best to use 64-bit, 133.3MHz cards in PCI-X slots.

PCI Online Addition and Replacement (OLA/R)

On older systems, adding, removing, or replacing an interface card traditionally required a system shutdown. Newer servers that support HP's Online Addition and Replacement (OLA/R) functionality now make it possible to add and remove interface cards without shutting down the system. If a card needs to be replaced, that card's slot can be powered down while the OS and other slots remain functional. Cards that are used to access system-critical resources such as swap, or the **`/`**, **`/usr`**, or **`/var`** file systems may not be replaced online via OLA/R.

Older I/O Expansion Bus Types

HP's current systems are based on the PCI and PCI-X I/O expansion bus architecture. Older systems, though, have used a variety of industry standard and HP proprietary bus technologies. Some of these older bus types are described briefly below:

EISA Extended Industry Standard Architecture. The EISA bus was the preferred I/O expansion bus on most early workstations, and some servers. Some current low-end workstations and servers continue to offer EISA expansion slots. Cards plugged into EISA slots must be configured using the **`eisa_config`** utility, which executes automatically in HPUX 10.x and beyond.

HP-PB Hewlett-Packard Precision Bus. This HP proprietary bus architecture was the primary I/O expansion bus type offered on the old E, F, G, H, I, K, and T series servers.

GSC General System Connect. This HP proprietary bus architecture was only offered on workstations, beginning with the early J-series systems. HP's current workstations use PCI rather than GSC buses.

HSC High Speed System Connect. The HSC bus is the server version of the GSC bus. All current servers use PCI rather than HSC buses.

Learning More about Your Server's Expansion Slots

To learn more about the expansion slots and cards available for your server, visit **`http://www.hp.com/go/servers`**.

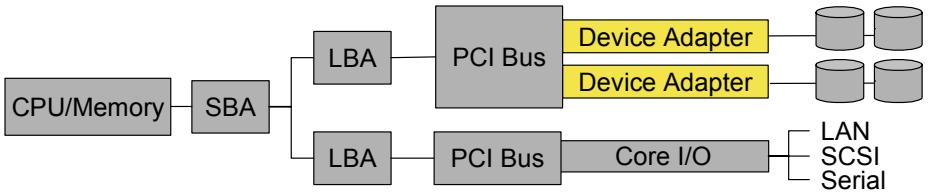
© 2004 Hewlett-Packard Development Company, L.P.

5–7. SLIDE: Device Adapter Cards

Device Adapter Cards

- Device adapters / interface cards provide connectivity to additional devices
- HP offers a variety of interface card types to support a variety of devices:

Interface Type	Typical Usage
SCSI	Disks, Tapes, CDROMs
Fibre Channel	Tape Libraries, Disk Arrays
10/100BaseT	LAN Connectivity
10/100/1000BaseT	High Speed LAN Connectivity
1000BaseSX, FDDI	High Speed Fiber Optic LAN Connectivity
ATM, X.25	WAN Connectivity



Student Notes

Peripheral devices may be attached to your system via a variety of device adapter cards. Additional device adapter cards may connect to your system via the expansion slots on your system's I/O expansion bus(es). On some models, the expansion slots are accessible directly from the system's backplane. On other models, it may be necessary to open the system's cabinet to add additional device adapter cards. The chart on the slide briefly describes some of the types of interface cards and device adapters you may find on your HP-UX system.

WARNING:	When purchasing additional interface cards, be sure to order interface cards to match your system's bus type! For example, HP-PB adapter cards may not be used on PCI buses!

For the latest information on HP's supported interface card types, consult the **`http://www.hp.com`**. website.

© 2004 Hewlett-Packard Development Company, L.P.

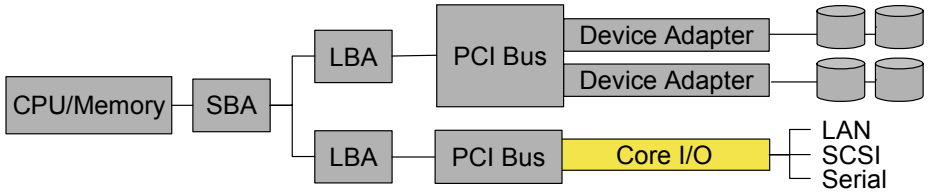
5–8. SLIDE: Core I/O Cards

Core I/O Cards

- All HP servers and workstations include at least one Core I/O card
- Core I/O cards provide multiple core Device Adapter “functions” on a single card
- Sample Device Adapters found on many current core I/O cards include:

Interface Type	Typical Usage
SCSI	Boot disk, tape, and DVD connectivity
10/100BaseT	LAN connectivity
Serial	Serial terminal/modem connectivity
Parallel*	Printers & Scanners
USB/PS1*	Keyboards & Mice
Audio*	Speakers & Headphones

* Workstations only



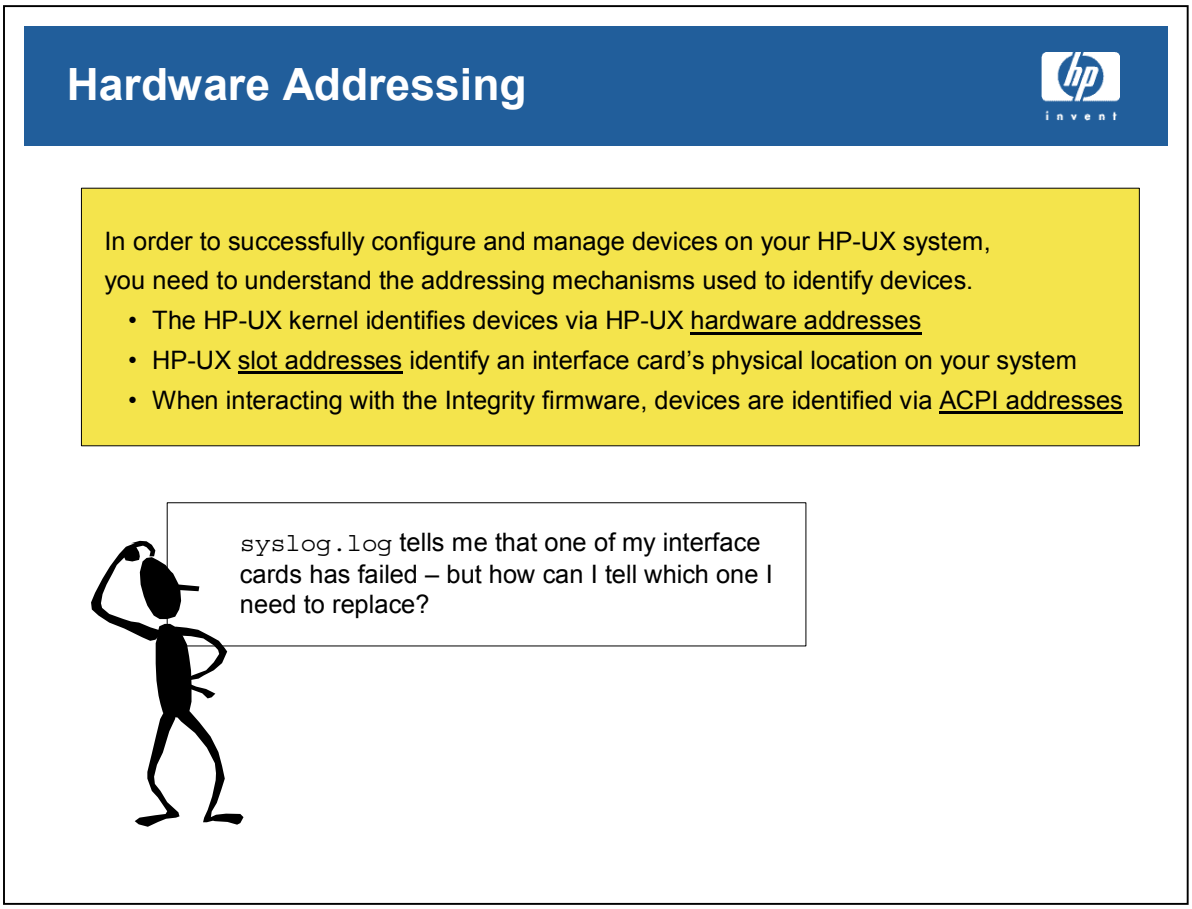
Student Notes

All HP 9000 and Integrity systems include at least one integrated **Core I/O Card** or **Multi-Function I/O Card (MFIOC)** that provides several device adapters on a single circuit board. The Core I/O card typically includes one or two SCSI interfaces that may be used to connect the boot disk(s) and a tape or DVD drive, a LAN interface, and one or two serial/console ports. Workstation Core I/O cards may include a parallel port for a local printer/scanner, and PS2 or USB ports for connecting a keyboard and mouse. See the hardware documentation on **`http://docs.hp.com`** for more specific information about your system’s Core I/O capabilities.

Cell-based systems often have multiple Core I/O cards.

© 2004 Hewlett-Packard Development Company, L.P.

5–9. SLIDE: Hardware Addressing



Student Notes

In order to successfully configure and manage devices on your HP-UX system, you need to understand the addressing mechanisms used to identify devices. HP-UX administrators need to be familiar with three different types of hardware paths and addresses.

- HP-UX slot addresses identify an interface card's physical location on your system. In order to add a new interface card, you must power-down the slot in which you intend to add the card. The command required to power-down a slot requires a slot address. Powering-down the wrong slot could be disastrous!

- The HP-UX kernel identifies devices via HP-UX hardware addresses. Many HP-UX systems have dozens or even hundreds of disks. When you configure a new disk, you must know the disk's hardware address. Choosing the wrong address might cause loss of data!

- When interacting with the Integrity firmware, devices are identified via ACPI addresses. During the boot process, you can specify which disk, DVD, or tape you wish to boot from. If you have multiple potential boot devices on your system, you must be able to determine which address corresponds to the desired boot device.

© 2004 Hewlett-Packard Development Company, L.P.

The next few slides discuss these addressing schemes in detail.

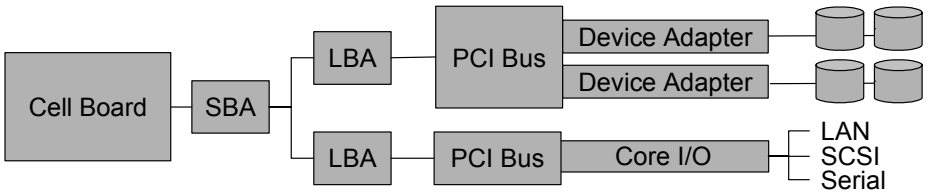
© 2004 Hewlett-Packard Development Company, L.P.

5–10. SLIDE: HP-UX Hardware Addresses

HP-UX Hardware Addresses

The HP-UX kernel identifies cards and devices via HP-UX hardware paths, which describe a device's logical location in the kernel's I/O tree structure in memory.

Processor-based hw path format:	SBA/LBA/device/function.target.LUN
Processor-based hw path example:	0/0/2/0.6.0
Processor-based hw path explanation:	SBA=0, LBA=0, dev=2, function=0, target=6, LUN=0
Cell-based hw path format:	Cell/SBA/LBA/device/function.target.LUN
Cell-based hw path example :	1/0/0/2/0.6.0
Cell-based hw path explanation:	Cell=1,SBA=0, LBA=0, dev=2, function=0, target=6, LUN=0



Student Notes

During the HP-UX boot process, the kernel automatically scans the system hardware and assigns a unique HP-UX hardware address to every bus adapter, interface card, and device. In order to configure new devices on your system, you need to be able to read and understand HP-UX hardware addresses.

Since hardware architectures vary from model to model, the structure of the HP-UX hardware paths also vary from model to model. Most current systems based on PCI expansion buses, though, use a fairly consistent hardware addressing scheme.

Processor-Based Servers

HP-UX hardware paths on processor-based servers are expressed in the following format:

`SBA/LBA/device/function.target*.LUN*` *(*SCSI devices only)*

SBA — The first portion of the HP-UX hardware address identifies the System Bus Adapter (SBA) to which the device is attached. On processor-based systems, there is only one SBA, so the first component of the hardware path will always be 0. Processors and

© 2004 Hewlett-Packard Development Company, L.P.

memory aren't associated with an SBA, but typically display a non-zero number in this component of the hardware path; the meaning of these numbers is model-specific.

LBA

The SBA connects to one or more Local Bus Adapters (LBAs) via high-speed communication channels known as "ropes". Some LBAs have just one rope to the SBA. Other LBAs have two ropes to the LBA to provide enhanced throughput.

The LBA component in the HP-UX hardware path actually identifies the rope number that connects the LBA to the SBA. When an LBA has two ropes to the SBA, the lower rope number is used in the hardware path.

Because some LBAs utilize two ropes and others utilize just one, a device's rope/LBA number typically isn't the same as its slot number.

The hardware manual for each server model contains a table that maps LBA numbers to physical slot addresses. Look for your server's hardware manual on **`http://docs.hp.com`**.

Device/Function

Each LBA provides connectivity to one or two PCI-X or PCI slots. Most slots provide one controller per card. However, Core I/O or Multifunction cards support multiple controllers per card. The Device/Function numbers together uniquely identify a single controller on a PCI or PCI-X card. Consult your server's hardware manual on **`http://docs.hp.com`** to determine the device/function numbers on your system.
If a card isn't a multi-function card, a device/function combination 0/0 indicates that it is a PCI card, and 1/0 indicates that it is PCI-X.

Target/LUN

The remainder of the hardware path varies from interface to interface. For SCSI devices, the two dot separated numbers on the end of the hardware path identify the SCSI target address and the SCSI Logical Unit Number (LUN). These will be discussed in detail on the SCSI addressing slide later in the chapter.

Cell-Based Servers

HP-UX hardware paths on cell-based servers are similar to hardware paths on processor-based servers, but include a global cell number on the beginning of the hardware path:

`Cell/SBA/LBA/device/function.target*.LUN*` *(*SCSI devices only)*

Cell

On cell-based servers, the first component of the HP-UX hardware path identifies the global cell number of the cell board to which the device is attached. On Superdome systems, cells in cabinet 0 are numbered 0-7, and cells in cabinet 1 are numbered 8-15. rx/rp7xxx servers have two cell boards numbered 0-1. rx/rp8xxx servers have four cell boards numbered 0-3.

© 2004 Hewlett-Packard Development Company, L.P.

SBA

The next portion of the HP-UX hardware address identifies the System Bus Adapter (SBA) to which the device is attached. On today's cell-based systems, there is only one SBA per cell board, so the SBA component of the hardware path will usually be 0. Processors and memory aren't associated with an SBA, but typically display a non-zero number in this component of the hardware path; the meaning of these numbers is model-specific.

LBA

The SBA connects to one or more Local Bus Adapters (LBAs) via high-speed communication channels known as "ropes". Some LBAs have just one rope to the SBA. Other LBAs have two ropes to the LBA to provide enhanced throughput.

The LBA component in the HP-UX hardware path actually identifies the rope number that connects the LBA to the SBA. When an LBA has two ropes to the SBA, the lower rope number is used in the hardware path.

Because some LBAs utilize two ropes and others utilize just one, an device's rope/LBA number typically isn't the same as its slot number.

The hardware manual for each server model contains a table that maps LBA numbers to physical slot addresses. Look for your server's hardware manual on **`http://docs.hp.com`**.

Device/Function

Each LBA provides connectivity to one or two PCI-X or PCI slots. Most slots provide one controller per card. However, Core I/O or Multifunction cards support multiple controllers per card. The Device/Function numbers together uniquely identify a single controller on a PCI or PCI-X card. Consult your server's hardware manual on **`http://docs.hp.com`** to determine the device/function numbers on your system.

If a card isn't a multi-function card, a device/function combination 0/0 indicates that it is a PCI card, and 1/0 indicates that it is PCI-X.

Target/LUN

The remainder of the hardware path varies from interface to interface. For SCSI devices, the two dot separated numbers on the end of the hardware path identify the SCSI target address and the SCSI Logical Unit Number (LUN). These will be discussed in detail on the SCSI addressing slide later in the chapter. Hardware addresses for fibre channel devices are significantly more complicated.

© 2004 Hewlett-Packard Development Company, L.P.

5–11. SLIDE: Viewing HP-UX Hardware Addresses (ioscan)

Viewing HP-UX Hardware Addresses (ioscan)

You can view devices' HP-UX hardware addresses, properties, and states via the `ioscan` command.

`# ioscan`	short listing of all devices
`# ioscan -f`	full listing of all devices
`# ioscan -fk`	full listing, using cached information
`# ioscan -fH 0/0/0/1/0`	full listing of device at 0/0/0/1/0
`# ioscan -fC disk`	full listing of "disk" class devices

```
# ioscan -f
Class    I  H/W Path       Driver S/W State H/W Type    Description
=============================================================================
root     0                 root   CLAIMED   BUS_NEXUS
cell     0  0              cell   CLAIMED   BUS_NEXUS
ioa      0  0/0            sba    CLAIMED   BUS_NEXUS   SBA
ba       0  0/0/0          lba    CLAIMED   BUS_NEXUS   LBA
ext_bus  0  0/0/0/2/0      c720   CLAIMED   INTERFACE   SCSI Fast Wide
disk     0  0/0/0/2/0.6.0  sdisk  CLAIMED   DEVICE      HP Disk
ctl      0  0/0/0/2/0.7.0  sctl   CLAIMED   DEVICE      Initiator
```

Student Notes

You can view a list of the device on your system and their hardware addresses via the **`ioscan`** command. **`ioscan`** supports a number of useful options.

`# ioscan` — Scans hardware and lists all devices and other hardware devices found. Shows the hardware path, class, and a brief description of each component.

`# ioscan -f` — Scans and lists the system hardware as before, but displays a "full" listing including several additional columns of information. See the **`ioscan`** field output descriptions below for more information.

`# ioscan -fk` — Lists the system hardware as before, but uses cached information. On a large system with dozens of disks and interface cards, **`ioscan -kf`** is much faster than **`ioscan -f`**.

© 2004 Hewlett-Packard Development Company, L.P.

`# ioscan -fH 0/0/0/2/0`	Shows a full listing for the device at the specified hardware address. This is useful on a large system if you just need to view information about a single device.
`# ioscan -fC disk`	Lists devices of the specified class only. Two other common classes are "tape" and "lan".
`# ioscan –fn`	Lists device file names associated with each device. Device files are discussed at length in the next chapter.

Fields in the `ioscan` Output

Class	The device “class” associated with the device or card. For example, all LAN cards, from Ethernet cards to Token Ring cards, belong to the “lan” class. All tape devices belong to the “tape” class. A device’s class is determined by the device’s kernel driver.
Instance	The instance number associated with the device or card. It is a unique number assigned by the kernel to each card or device within a class. If no driver is available for the hardware component, or if an error occurs binding the driver, the kernel will not assign an instance number and a `-1` will be listed. Instance numbers are used when creating device files, and will be discussed in more detail in the next chapter.

© 2004 Hewlett-Packard Development Company, L.P.

H/W Path	A string of numbers, separated by "/"'s and "."'s that uniquely identifies each hardware component on the system. Each number in the string represents the location of a hardware component on the path to the device. Some administrative tasks require the administrator to identify devices by hardware path.
Driver	The name of the kernel driver that controls the hardware component. If no driver is available to control the hardware component, a question mark (?) is displayed in the output.
S/W state	The result of software binding. CLAIMED means the driver for this device was successfully bound to the device. UNCLAIMED means no driver was found in the kernel for this device. See the **ioscan** man page for further explanation.
Hardware Type	Entity identifier for the hardware component. Common examples include **INTERFACE** (for interface cards) or **DEVICE** (for peripheral devices).
Description	Describes the device, and in some cases, identifies the device's manufacturer and model number.

Troubleshooting with `ioscan -f`

After adding an interface card or SCSI device to your system, you should execute **ioscan** to see if your system recognizes the device.

First, simply check to see that your new device appears in the **ioscan** output. If not, shutdown your machine and check to ensure that all the cables are connected properly. In the case of an interface card, ensure that the card is firmly inserted in the interface card slot in the backplane of your machine.

Next, ensure that the hardware path is correct. Did you set the correct SCSI address? Add the device and its hardware path and description to the hardware diagram in your system log book.

Assuming the hardware path is correct, check the "S/W state" column in the **ioscan -f** output. In order to communicate with your new device or interface card, your kernel must have the proper device drivers configured. If the proper driver already exists in your kernel, the "S/W State" column should say "CLAIMED". If this isn't the case, you will have to add the driver to the kernel. A later chapter discusses kernel configuration.

If your new device appears to be "CLAIMED" by the kernel, proceed to the next chapter and learn how to create and use device files to access your new device.

© 2004 Hewlett-Packard Development Company, L.P.

5–12. SLIDE: Slot Addresses

Slot Addresses

On cell-based systems, administrators can identify a card slot's physical location by its slot address, which is based on:

- The slot's cabinet (each Superdome complex may have 1-4 cabinets)
- The slot's I/O bay (each Superdome cabinet may have 1-2 I/O bays)
- The slot's I/O chassis (each Superdome I/O bay may have 1-2 I/O chassis)
- The slot's slot number (each Superdome I/O chassis has twelve card slots)

Slot Address Format:	Cabinet-Bay-Chassis-Slot
Slot Address Example:	0-1-3-4
Slot Address Explanation:	Cabinet 0, Bay 1, Chassis 3, Slot 4

Student Notes

Though the kernel identifies devices via HP-UX hardware paths, it is often easier to determine an interface card's physical location via its interface card's slot address. The slot address consists of four components, which identify an interface card's exact location on a system.

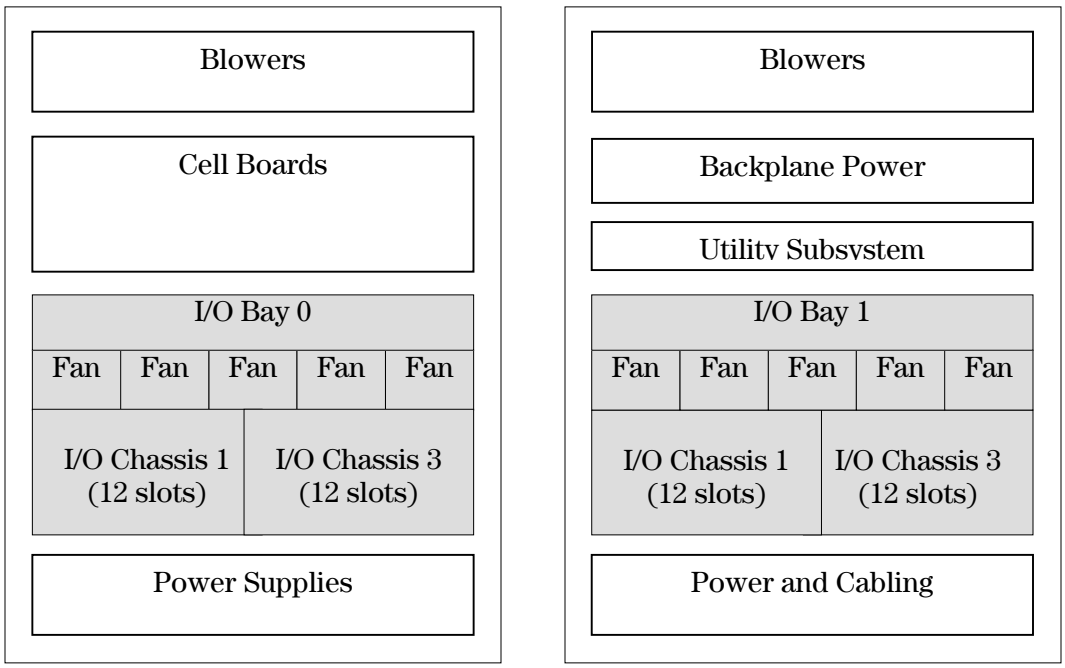
- The first portion of the slot address identifies a slot's cabinet number. A Superdome complex may have one or two system cabinets, and two additional I/O expansion (IOX) cabinets. Superdome interface card slots that are connected to cell boards in the first cabinet will have a 0 in cabinet portion of the slot address. Superdome interface card slots that are connected to cell boards in the second cabinet will have a 1 in the cabinet portion of the slot address. Interface cards in the expansion cabinets will have an 8 or 9 in the first portion of the slot address. On non-Superdome systems, the cabinet number will always be 0. The photo below shows a 64-way Superdome with two I/O expansion cabinets.

© 2004 Hewlett-Packard Development Company, L.P.

Cabinet 0 Cabinet 1 Cabinet 8 Cabinet 9

© 2004 Hewlett-Packard Development Company, L.P.

- The second component of the slot address identifies the slot's I/O bay. Each Superdome cabinet has two I/O bays. I/O bay 0 is located on the front of the cabinet, and I/O bay 1 is located in the rear of the cabinet. Each Superdome IOX cabinet can have three vertically stacked I/O bays, numbered 1 to 3 from bottom to top. Additional space in the IOX can be used to install peripheral devices. On non-Superdome systems, the I/O bay number will always be 0. The diagram below shows the location of the I/O bays on the front and back of a Superdome cabinet.



Cabinet #0 Front Cabinet #0 Rear

- The third component of the slot address identifies the slot's I/O chassis number. Each I/O bay contains up to two I/O chassis. On Superdome systems, the I/O chassis are physically distinct components; the chassis on the left is I/O chassis 1 and the I/O chassis on the right is I/O chassis 3. On the rp7xxx, rx7xxx, rp8xxx, and rx8xxx servers, there are two logical I/O chassis are numbered 0 and 1, but they are combined in a single physical card cage.

- The fourth component of the slot address identifies the PCI card slot number. Each Superdome I/O chassis has twelve slots, numbered 0- 11. On rp7xxx, rx7xxx, rp8xxx, and rx8xxx servers, each I/O chassis has eight slots numbered 0-7.

© 2004 Hewlett-Packard Development Company, L.P.

5–13. SLIDE: Viewing Slot Addresses (`olrad` and `rad`)

Viewing Slot Addresses (`olrad` and `rad`)

You can view slot addresses, properties, capabilities, and usage via the `rad` command (11i v1) or `olrad` command (11i v2).
These commands are only available on cell-based systems.

```
# olrad -q
Slot      Path      Bus Max Spd Pwr Occu Susp OLAR OLD Max   Mode
                    Num Spd                           Mode
0-1-3-1   2/0/1/1   21  133 133 Off No   N/A  N/A  N/A PCI-X N/A
0-1-3-2   2/0/2/1   42  133 N/A On  Yes  No   Yes  N/A PCI-X N/A
0-1-3-3   2/0/3/1   63  133 N/A On  Yes  No   Yes  N/A PCI-X N/A

# rad -q
Slot      Path       Bus   Speed   Power   Occupied   Suspended   Capable
0-0-0-1   0/0/8/0    64    66      On      Yes        No          Yes
0-0-0-2   0/0/10/0   80    66      On      Yes        No          Yes
0-0-0-3   0/0/12/0   96    66      On      Yes        No          Yes
```

Student Notes

You can view slot addresses and correlate those addresses to HP-UX hardware addresses, via the **`rad`** command (11i v1) or **`olrad`** command (11i v2). These commands are only available on mid- to high-end servers.

For each slot, the commands report:

Slot: The “Slot” column reports the slot’s slot address.

Path: The “Path” column reports the slot’s corresponding HP-UX hardware path.

Bus Num: The “Bus Num” column reports the slot’s bus number.

Max Spd: The ”Max Spd” column reports the maximum speed (MHz) supported by the slot. PCI-X slots support speeds up to 133MHz. PCI slots support speeds up to 33MHz or 66MHz.

Spd: The “Spd” column reports the maximum speed (MHz) supported by the card currently in the slot. PCI-X cards operate up to 133MHz. PCI slots support speeds up to 33MHz or 66MHz.

© 2004 Hewlett-Packard Development Company, L.P.

Pwr: The "Pwr" column reports the power status of the slot. Slots can be powered up/down to facilitate I/O card replacement. This functionality will be described in detail later in the chapter.

Occu: Is the slot currently occupied by an interface card?

Susp: The "Susp" column identifies slots in the suspended state. A card must be suspended before it can be replaced.

OLAR/Capable: The "OLAR" column (**olrad**) or "Capable" column (**rad**) indicates if the slot supports HP's online addition and replacement functionality.

Max Mode: The "Max Mode" column distinguishes PCI-X slots from PCI slots.

Sample Output from `olrad` and `rad`

Here is some sample output from the **olrad** and **rad** commands (only a portion of the slots on each system have been listed here):

```
# olrad -q
Slot     Path     Bus Max Spd Pwr Occu Susp OLAR OLD Max   Mode
                  Num Spd                          Mode
0-1-3-1  2/0/1/1  21  133 133 Off No   N/A  N/A  N/A PCI-X N/A
0-1-3-2  2/0/2/1  42  133 N/A On  Yes  No   Yes  N/A PCI-X N/A
0-1-3-3  2/0/3/1  63  133 N/A On  Yes  No   Yes  N/A PCI-X N/A

# rad -q
Slot     Path      Bus  Speed  Power  Occupied  Suspended  Capable
0-0-0-1  0/0/8/0   64   66     On     Yes       No         Yes
0-0-0-2  0/0/10/0  80   66     On     Yes       No         Yes
0-0-0-3  0/0/12/0  96   66     On     Yes       No         Yes
```

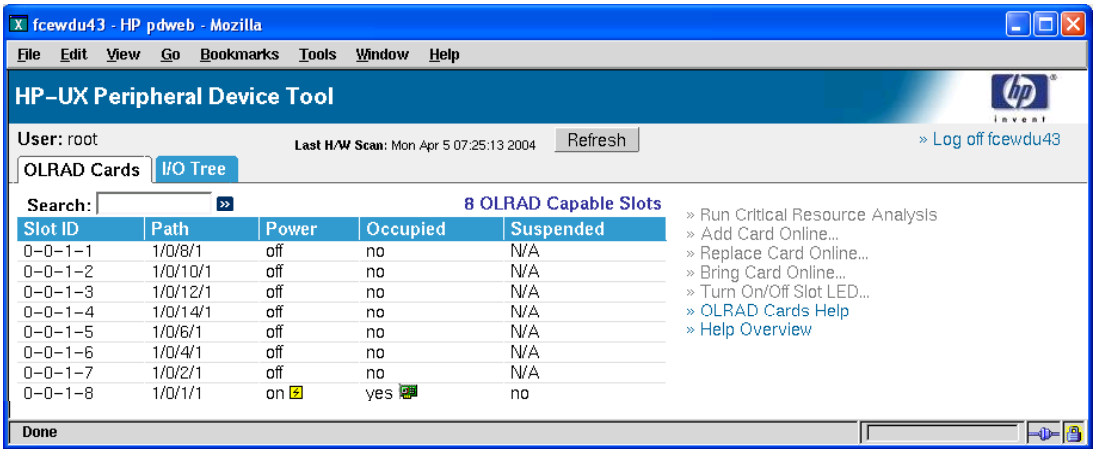

© 2004 Hewlett-Packard Development Company, L.P.

5–14. SLIDE: Viewing Slot Addresses (`pdweb` and `sam`)

Viewing Slot Addresses (`pdweb` and `sam`)

- On 11i v1, you can view slot addresses via the SAM Interface Cards GUI
- On 11i v2, you can view slot addresses via the new `pdweb` web interface

```
# kcweb [-F] or
# mozilla http://server:1110/pd/pd.cgi
```



Student Notes

Both HP-UX 11i v1 and HP-UX 11i v2 provide graphical alternatives to the **`rad`** and **`olrad`** commands discussed on the previous page. On 11i v1, you can browse slot and HP-UX hardware addresses via the SAM Interface Cards screen. On 11i v2, the **`pdweb`** web interface provides similar functionality.

© 2004 Hewlett-Packard Development Company, L.P.

Viewing Slot Addresses via `sam` in 11i v1

In 11i v1, the **`sam -> Interface Cards`** screen lists interface card slot addresses, hardware paths, and state information.

```
# sam -> Peripheral Devices -> Cards
```

```
+------------------------- Cards ----------------------------------+
|File View Options Actions                                    Help |
|I/O Cards                                        2 of 23 selected|
|-----------------------------------------------------------------|
|        Hardware                        Slot                     |
|  Slot  Path     Driver State          Power Description        |
|+---------------------------------------------------------------+ |
|| -     0/0/2/1  c720   not OLAR-able -     SCSI C87x Fast Wid | |
|| -     0/0/4/0  func0  not OLAR-able -     PCI BaseSystem (10 | |
|| -     0/0/4/1  asio0  not OLAR-able -     Service Processor  | |
|| 5     0/2/0    -      -              on    empty slot         | |
|| 6     0/5/0    -      -              on    empty slot         | |
|| 7     0/1/0    -      -              on    empty slot         | |
|| 8     0/3/0    -      -              on    empty slot         | |
|| 9     0/9/0/0  c8xx   active         on    SCSI C1010 Ultra W | |
|| 9     0/9/0/1  c8xx   active         on    SCSI C1010 Ultra W | |
|| 10    0/8/0/0  c8xx   active         on    SCSI C1010 Ultra W | |
|| 10    0/8/0/1  c8xx   active         on    SCSI C1010 Ultra W | |
|+---------------------------------------------------------------+ |
+-----------------------------------------------------------------+
```

Viewing Slot Addresses via `pdweb` and a Local Web Browser in 11i v2

In 11i v2, the **`pdweb`** web-based interface replaces the SAM Interface Cards screen. Administrators can access the **`pdweb`** interface from a browser running locally on the target server, or from a browser running remotely on another UNIX system or PC.

Several steps are required if you wish to access the **`pdweb`** interface on the localhost:

1. Verify that your **`DISPLAY`** variable is set. Running **`pdweb`** via a remote X-windows session or X-windows emulator such as ReflectionsX is risky since X-windows traffic isn't secure. In fact, **`pdweb`** automatically verifies that the **`pdweb`** server's hostname matches the hostname in the **`DISPLAY`** variable. If the hostnames don't match, **`pdweb`** won't start unless you add the **`-F`** (force) option. If you need to configure a kernel on a remote target server, consider using the more secure technique described in the next section of the notes.

   ```
   # echo $DISPLAY
   ```

2. Verify that your **`BROWSER`** variable is set. If the **`BROWSER`** variable isn't defined, **`pdweb`** launches **`/opt/mozilla/mozilla`**, the default browser in 11i v2.

   ```
   # export BROWSER=/opt/mozilla/mozilla
   ```


© 2004 Hewlett-Packard Development Company, L.P.

3. Run **pdweb** and enter the root username and password when prompted. If **pdweb** is executed without any options, the server daemon will stop automatically after a period of inactivity. If you are accessing the target system via a remote X-windows session, add the **-F** (force) option, too.

   ```
   # pdweb [-F] &
   ```

   - **pdweb** will:
     - Create SSL certificates for the server if they haven't already been created.
     - Launch the Apache/hpwebadm server daemon if it isn't already running.
     - Launch the web browser defined in your **BROWSER** environment variable.

4. The slide shows a screenshot from the **pdweb** interface. Click on any slot address to view detailed information about a specific slot, the cards in the slot, or devices attached to the card in the slot. Click the I/O tree tab to view an **ioscan**-like list of hardware paths. On some mid- and high-end servers, you can use **pdweb** to add and replace interface cards. Online addition and replacement will be discussed later in the chapter.

Viewing Slot Addresses via pdweb and a Remote Web Browser in 11i v2

It is also possible to access **pdweb** from a web browser on a remote PC or UNIX system. In order for this to work, though, you must first start the Apache/hpwebadm daemon on the system you wish to manage. This is most easily accomplished by configuring the **inetd** daemon to "autostart" the necessary daemon for you automatically on an as-needed basis.

The autostart functionality causes the **inetd** daemon to listen for incoming **pdweb** requests on port#1110. When a browser contacts port#1110, **inetd** launches the **/opt/hpwebadmin/lbin/autowa** (auto web admin) server process. **autowa**, in turn, launches the Apache/hpwebadm on port#1188 using a Secure Socket Layer (SSL) connection to encrypt network traffic between your browser and the daemon. Shortly after the session terminates, the Apache/hpwebadm daemon dies. When **inetd** receives another request for service, it launches **autowa** again.

Two simple steps are required to configure the autostart functionality.

1. Enable/disable **inetd** autostart functionality for the Apache/hpwebadm service in **/etc/inetd.conf** and **/etc/services**. You can modify the appropriate lines automatically by running the **waconf** command, or you can use the **vi** editor to manually remove the **#** sign preceding the **autowa** lines in **/etc/inetd.conf** and **/etc/services**. Verify that **inetd** is **LISTEN**ing for incoming requests on port 1110.

   ```
   server# waconf –a on|off
   server# netstat –an | grep 1110
           tcp 0 0 *.1110 *.* LISTEN
   ```

2. Launch a web browser and enter the root username/password when prompted. If this doesn't work, make sure your firewall permits traffic from your browser client to port#1110 and port#1188 on the target server.

© 2004 Hewlett-Packard Development Company, L.P.

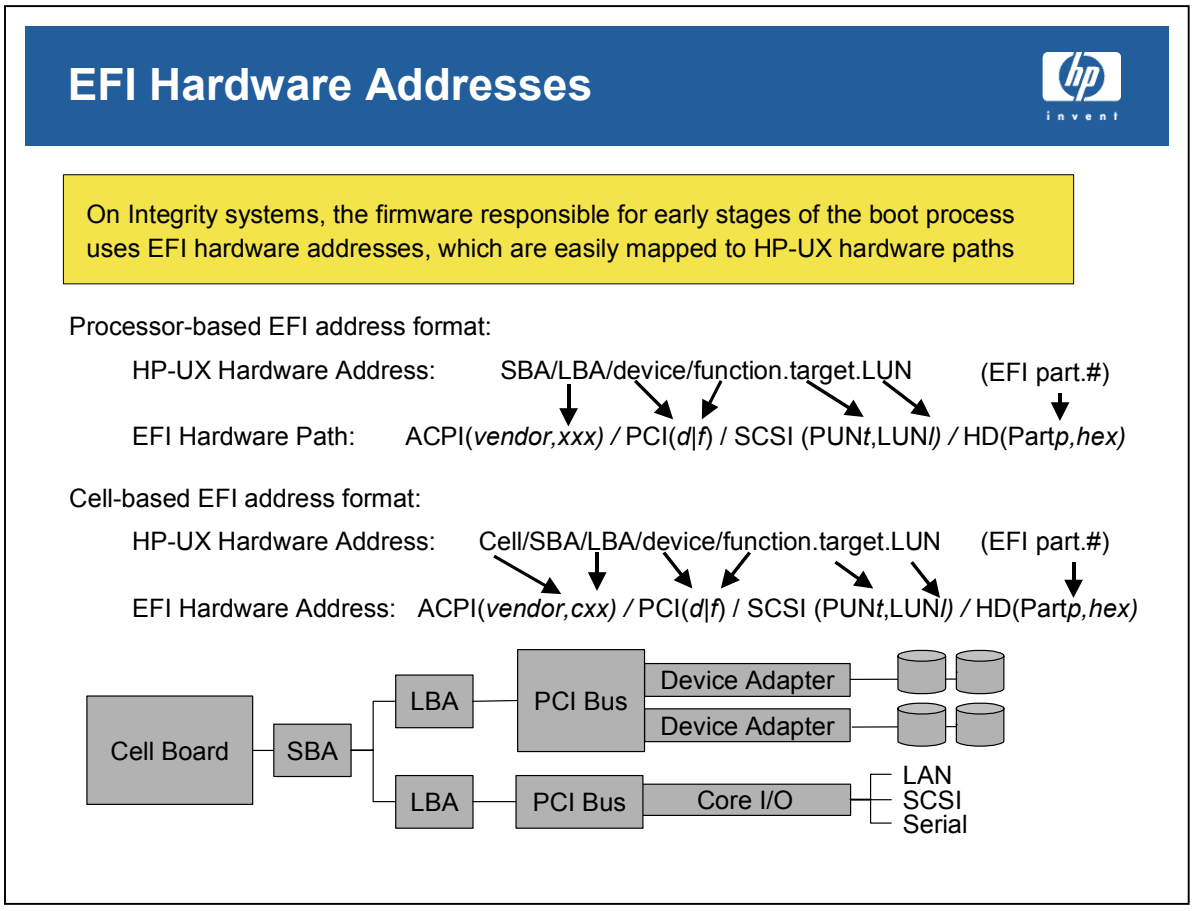
```
client# mozilla http://server:1110/pd/pd.cgi
```

3. The slide shows a screenshot from the **pdweb** interface. Click on any slot address to view detailed information about a specific slot, the cards in the slot, or devices attached to the card in the slot. Click the I/O tree tab to view an **ioscan**-like list of hardware paths. On cell-based servers, you can use **pdweb** to add and replace interface cards. Online addition and replacement will be discussed later in the chapter.

NOTE:	**pdweb** is only supported on the Internet Explorer and Mozilla browsers. Cookies must be enabled, and you must have connectivity to the target server via port numbers 1110 and 1188.

© 2004 Hewlett-Packard Development Company, L.P.

5–15. SLIDE: EFI Hardware Addresses



Student Notes

The HP-UX kernel identifies devices via HP-UX hardware addresses. Slot addresses identify interface card physical locations. In order to successfully manage the boot process on an Integrity server, though, the administrator must also be familiar with EFI/ACPI hardware addresses.

EFI/ACPI Addresses

The Extensible Firmware Interface (EFI) is a standard firmware interface that was developed by Intel and its partners to provide a consistent, OS- and platform-neutral mechanism for selecting a boot disk and managing the boot process on Itanium-based systems. The EFI boot process and commands will be discussed in detail later in the course.

The Advanced Configuration and Power Interface (ACPI) is the portion of the EFI specification that defines the mechanism used to address and access devices from the EFI. If you wish to change your default boot disk, boot from a recovery tape, or install the OS from the install DVD on an Integrity server, you need to understand EFI/ACPI addressing.

© 2004 Hewlett-Packard Development Company, L.P.

EFI Addressing

EFI hardware addresses are structured differently than the HP-UX hardware addresses that we discussed earlier.

The EFI address consists of several four major address components. On processor-based servers, the addresses look like this:

ACPI(*vendor*,*cxx*)/PCI(*d*|*f*)/SCSI(PUN*t*,LUN*l*)/HD(Part*p*,*hex*)

ACPI(*vendor*,*cxx*) The vendor field, which precedes the comma in the first set of parentheses, contains a hexadecimal value that identifies the device manufacturer.

The meaning of the number immediately following the comma in the first set of parentheses differs from model to model. Consult your system's owner's manual for details specific to your server, or use the **ioscan -e** command as described on the next page to correlate these addresses with HP-UX hardware paths.

On the Integrity Superdome, the first hexadecimal digit after the comma (c in the example) identifies the device's cell number. For cell 0 the cell number is omitted. The two hex digits that follow identify the device's LBA/rope number. The following table shows the relationship of the EFI rope numbers to physical slot numbers:

EFI#	slot#
0	0
0x15	1
0x2A	2
0x3F	3
0x54	4
0x69	5
0xEA	6
0xDF	7
0xBE	8
0xA8	9
0x93	10
0x7F	11

For processor-based systems the number in the parentheses that follows the vendor ID simply identifies the LBA/rope. The LBA/rope/physical slot correlations may vary from model to model. The following table shows the mapping for an rx5670 server:

© 2004 Hewlett-Packard Development Company, L.P.

EFI #	rope	slot
0	0	1
0	0	2
100	1	3
400	4	4
400	4	5
700	7	6
700	7	7
600	6	8
600	6	9
500	5	10
200	2	11
300	3	12

PCI(*d*|*f*) This component of the EFI path represents the device and function numbers. The "|" is literal. As in HP-UX hardware paths, the device and function numbers are assigned differently from model to model. Again, see your server's hardware guide for details or run **ioscan -e**.

SCSI(PUN*t*,LUN*l*) This component of the EFI path identifies the SCSI target (***t***) and Lun (***l***), both given in hex. The ***t*** is the HP-UX target address translated into decimal. The LUN is the HP-UX LUN translated into decimal.

HD(Part*p*,*hex*) The **Part*p*** portion of the EFI path identifies the EFI partition on the disk (if any) that contains the utilities necessary to load and boot the kernel. Usually, the kernel loader is found in EFI partition 1. EFI partitions and the structure of an Integrity/EFI boot disk will be discussed in the EFI boot chapter. EFI partition numbers aren't referenced in HP-UX hardware paths.

Additional Fields

Device-dependent information may be provided in additional fields, following those defined above.

© 2004 Hewlett-Packard Development Company, L.P.

5–16. SLIDE: Viewing EFI Hardware Addresses (`ioscan`)

Viewing EFI Hardware Addresses (`ioscan`)

From HP-UX, use the `ioscan -e` command to view HP-UX *and* ACPI hardware paths

```
# ioscan -eC disk
H/W Path       Class    Description
============================================================
0/1/1/0.0.0    disk     HP 73.4GST373405LC
Acpi(000222F0,100)/Pci(1|0)/Scsi(Pun0,Lun0)/HD(Part1,SigF9)

0/1/1/0.1.0    disk     HP 73.4GST373405LC
Acpi(000222F0,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part1,SigB8)
```

From the EFI Shell, use the `map` command to view ACPI hardware paths

```
Shell> map -r -fs -b
Device mapping table
fs0: Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun0,Lun0)/HD(Part1,SigB8D)
fs1: Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part1,SigB8D)
```

Student Notes

In 11i v2, you can use the **`ioscan -e`** command to correlate disk devices' EFI addresses to HP-UX hardware addresses. The EFI hardware paths appear immediately below the corresponding HP-UX hardware paths. (The sample output below was truncated slightly to fit on the page.)

```
# ioscan -eC disk
H/W Path       Class    Description
============================================================
0/1/1/0.0.0    disk     HP 73.4GST373405LC
Acpi(000222F0,100)/Pci(1|0)/Scsi(Pun0,Lun0)/HD(Part1,SigF9)

0/1/1/0.1.0    disk     HP 73.4GST373405LC
Acpi(000222F0,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part1,SigB8)
```

From the EFI shell prompt, you can view a list of EFI devices and addresses via the **`map -r -fs`** command. This will be described in detail in the Integrity Boot chapter elsewhere in the course. The command displays an arbitrarily assigned drive name (eg: "**`f1:`**") on the far left, followed by each disk device's EFI address. The **`-r`** option forces the map command to rescan the devices to avoid stale cached information. The **`-fs`** option limits output to

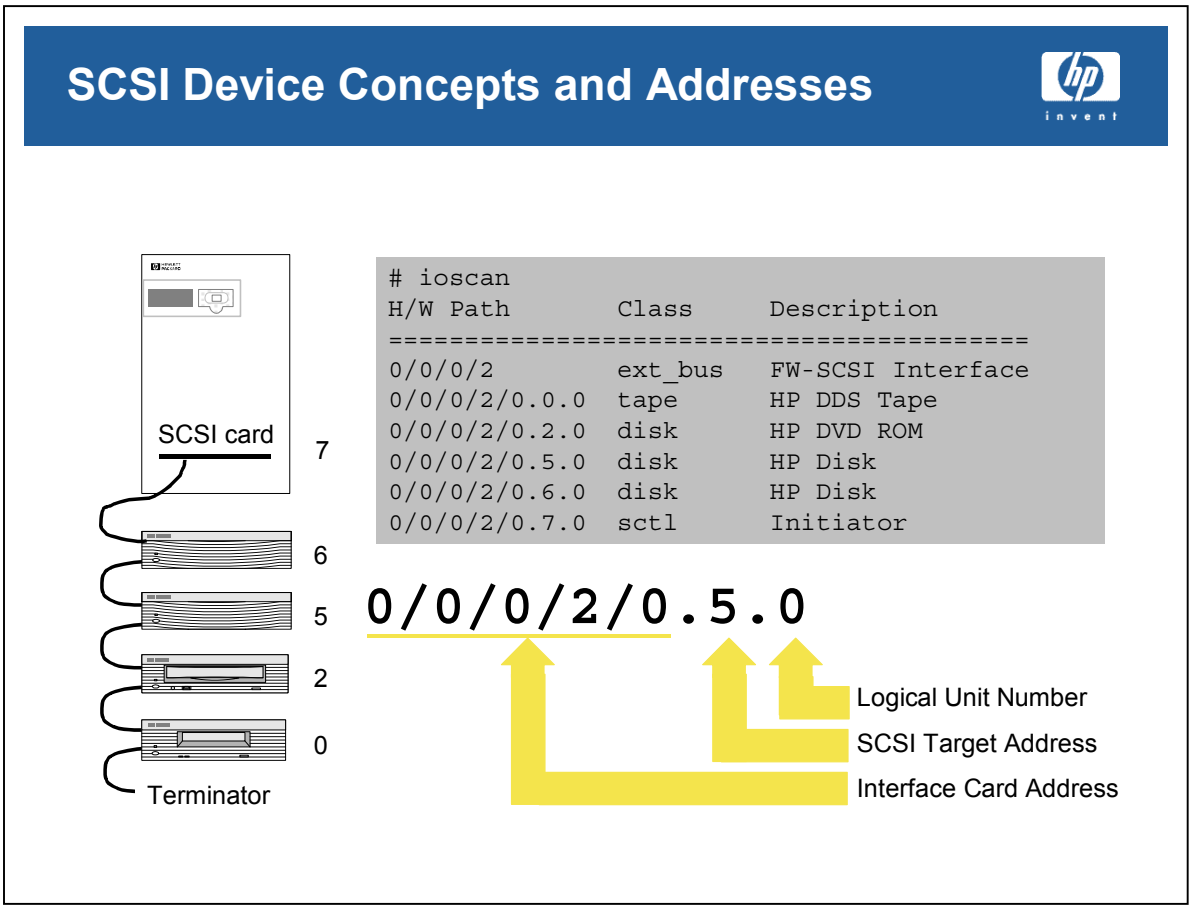
© 2004 Hewlett-Packard Development Company, L.P.

devices that contain EFI/FAT32 file systems that may contain the utilities necessary to boot HP-UX. The **-b** option displays one screen of output at a time.

```
Shell> map –r –fs -b
Device mapping table
fs0: Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun0,Lun0)/HD(Part1,SigB8D)
fs1: Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part1,SigB8D)
```


© 2004 Hewlett-Packard Development Company, L.P.

5–17. SLIDE: SCSI Device Concepts and Addresses



Student Notes

Disk drives, CDROMs, and tape drives are among the most commonly used peripherals. These mass storage devices are all typically attached to HP-UX systems via some variation on a SCSI ("Small Computer Systems Interface") interface.

SCSI Types

The SCSI standard was first endorsed by the American Standards Institute as a 5MB/second interface for mass storage devices in 1986. Over the years, there have been many variations and improvements on this original SCSI standard. The next few paragraphs below explain some of the terminology you will likely encounter when dealing with SCSI devices and interfaces:

"Narrow" vs. "Wide":

Every SCSI bus has a predefined "bus width" that determines how much data can traverse the bus at any given time. A "Narrow" SCSI bus carries 8-bits of data, while a "Wide" SCSI bus carries 16-bits of data. All Ultra2 SCSI cards and devices are "wide". Otherwise, if a device or interface card isn't labeled "wide" or "narrow", assume that it is narrow.

© 2004 Hewlett-Packard Development Company, L.P.

"Single-Ended" vs. "LVD" vs. "HVD"

Over the years, SCSI devices have used a variety of different signal protocols to carry data over SCSI buses. These protocols define what voltages are supported, and how bits are represented as electrical signals. Early SCSI devices used the "Single-Ended" SE-SCSI protocol. The "High Voltage Differential" (HVD) or "Differential SCSI" protocol, which was introduced several years later, allowed much greater distances between devices on a SCSI bus. Most newer SCSI cards and devices, however, use the "Low Voltage Differential" (LVD) protocol, which allows higher speeds than HVD.

If you have an interface card labeled LVD/SE, this indicates that the card supports both the SE-SCSI and LVD protocols, which makes it possible to attach both LVD and SE devices. Note, however, that the presence of a single SE-SCSI device on the bus causes the entire bus to operate using the SE-SCSI protocol, which typically yields poorer performance and shorter supported bus lengths.

HVD interface cards are incompatible with both SE-SCSI and LVD devices. All devices on an HVD SCSI bus must be differential/HVD.

The table below summarizes the common SCSI device/interface types that may encounter as a system administrator. Note the maximum supported transfer rate (labeled "Max. MB/s") and maximum number of devices per bus (labeled "Max. Devices") for each interface card type. Cells labeled "n/a" identify unsupported protocols (eg: the Ultra3 SCSI standard doesn't support the "HVD" signal protocol)

© 2004 Hewlett-Packard Development Company, L.P.

Interface Type	Bus Width	Bus Length			Max. MB/s	Max. Devices
		SE	LVD	HVD		
Narrow SE-SCSI	8 bits	6m	n/a	25m	5	8
Narrow Fast SCSI	8 bits	3m	n/a	25m	10	8
Fast Wide SCSI	16 bits	3m	n/a	25m	20	16
Narrow Ultra SCSI	8 bits	1.5m	n/a	25m	20	8
Narrow Ultra SCSI	8 bits	3m	n/a	n/a	20	4
Wide Ultra SCSI	16 bits	n/a	n/a	25m	40	16
Wide Ultra SCSI	16 bits	1.5m	n/a	n/a	40	8
Wide Ultra SCSI	16 bits	3m	n/a	n/a	40	4
Wide Ultra2/LVD SCSI	16 bits	n/a	12m	n/a	80	16
Wide Ultra160/LVD SCSI	16 bits	n/a	12m	n/a	160	16
Wide Ultra320/LVD SCSI	16 bits	n/a	12m	n/a	320	16

Notes:
- If a device doesn't specify whether it is "wide" or "narrow", assume "narrow"...
- Exception: After Ultra2, all new SCSI technologies are wide only.
- The terms "HVD" and "Differential" are often used interchangeably.
- Interfaces that don't contain the term "LVD", "HVD", "Ultra2", "Differential", or "Ultra3" are usually single-ended.

To determine what types of SCSI interface cards are installed on your server, execute **`ioscan -f`** and look at the **`Description`** column. Your server and SCSI device owner manuals may provide some guidance, too.

SCSI Cable Types

The SCSI standards support a variety of cable types, with anywhere from 50 to 80 pins, in a variety of shapes and sizes. Read your owner's manuals to determine what type of cable is required to support your SCSI devices.

SCSI Chains

All SCSI interface types allow multiple devices to be daisy chained together off of a single SCSI interface card, as shown on the slide. The resulting string of devices is known as a **SCSI chain**. The number of devices allowed in a single SCSI chain varies, depending on the SCSI interface card type. In all cases, however, you may freely mix tape drives, CDROMs, and disks on a single SCSI chain.

Cabling a SCSI chain is simply a matter of running an appropriate SCSI cable from the SCSI interface card to the first device in the chain. Each SCSI device should have two ports. Attach the cable from your system's SCSI interface card to either one of the ports on the first SCSI device. Then run a cable from the second port to the next device, from the second device to the third, and so forth.

SCSI Terminators

The final device on every SCSI chain must have a SCSI **terminator** attached to its second port. This terminator provides matching impedance on the bus circuits; without a terminator,

© 2004 Hewlett-Packard Development Company, L.P.

the devices on the bus will not work properly. There are several types of terminators; read your documentation to determine which type is appropriate for your interface card.

If an interface card doesn't have any attached devices, the interface card may require a terminator. Some of HP's newer interface cards are "auto-terminating", and have termination circuitry built into the card itself. You may need to manually enable this feature; see your owner's manual for more information.

SCSI Target Addresses

In order to differentiate among devices on a SCSI chain, you must assign each device on the chain a unique SCSI **bus/target address**. Older SCSI interface cards only provided eight target addresses from 0 to 7 (though only seven devices were allowed, since the controller itself required one address, typically target address 7). The newer SCSI technologies provide up to 16 addresses from 0 to 15 (though again, only fifteen devices are allowed since the controller requires one address). Priority is given to devices on the SCSI bus according to the following scheme:

7 (highest priority) ----------> 0 ----> 15 ----------> 8 (lowest priority)

SCSI target addresses may be set several different ways. Some devices use a series of binary DIP switches to set the address. Other devices use a series of jumper pins. Consult your device documentation to determine how to set your device's SCSI target address.

The SCSI interface card itself also requires a target address. Most interface cards ship from the factory with SCSI target address 7. To determine which target address has been assigned to your interface card, execute the **`ioscan`** command, and find the line containing the word **`Initiator`** in the description column:

```
# ioscan -fu | grep Initiator
```

The interface card's target address is listed as the second to last component of the Initiator's H/W Path.

When you attach a new device, make sure that you choose a unique SCSI address! Connecting two devices with the same address to a single SCSI interface will very likely cause loss of data!

For more information on SCSI technology, take a look at the SCSI Technical Association's website at **`http://www.scsita.org`** or the FAQ sheets on **`http://www.cablemakers.com`**.

SCSI Hardware Addresses

The previous slide noted that every device has a unique hardware path; SCSI devices are no different. SCSI device hardware paths have three components. Working from right to left:

Logical Unit Number	The right-most element in a SCSI device's hardware path is the device Logical Unit Number (LUN). The LUN will only be non-zero for arrays and tape autochangers. The next slide discusses LUNs in more detail. LUNs will be discussed in detail later in the chapter.

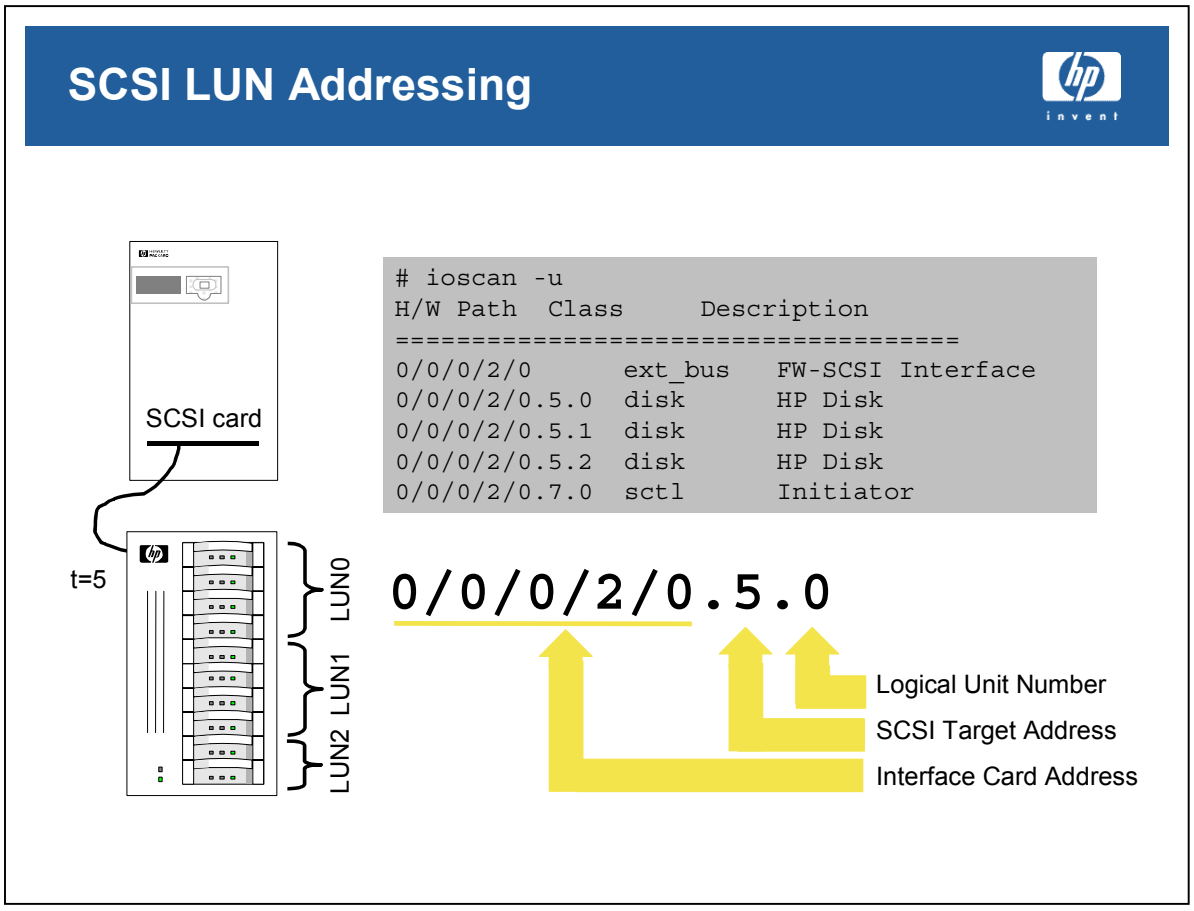
© 2004 Hewlett-Packard Development Company, L.P.

Target Address	The second to last element in every SCSI device hardware path identifies the device's SCSI target address.
SCSI Interface Card Address	The leftmost portion of the SCSI hardware path identifies the hardware path of the SCSI interface card to which the device is attached.

The **`ioscan`** command will display the hardware paths of all of the devices on your system for you automatically.

© 2004 Hewlett-Packard Development Company, L.P.

5–18. SLIDE: SCSI LUN Addressing



Student Notes

Some devices such as disk arrays may have a single SCSI target address, with multiple addressable units within the device. HPUX uses the Logical Unit Number (LUN) portion of the hardware path to identify these components.

LUNs for Disk Arrays

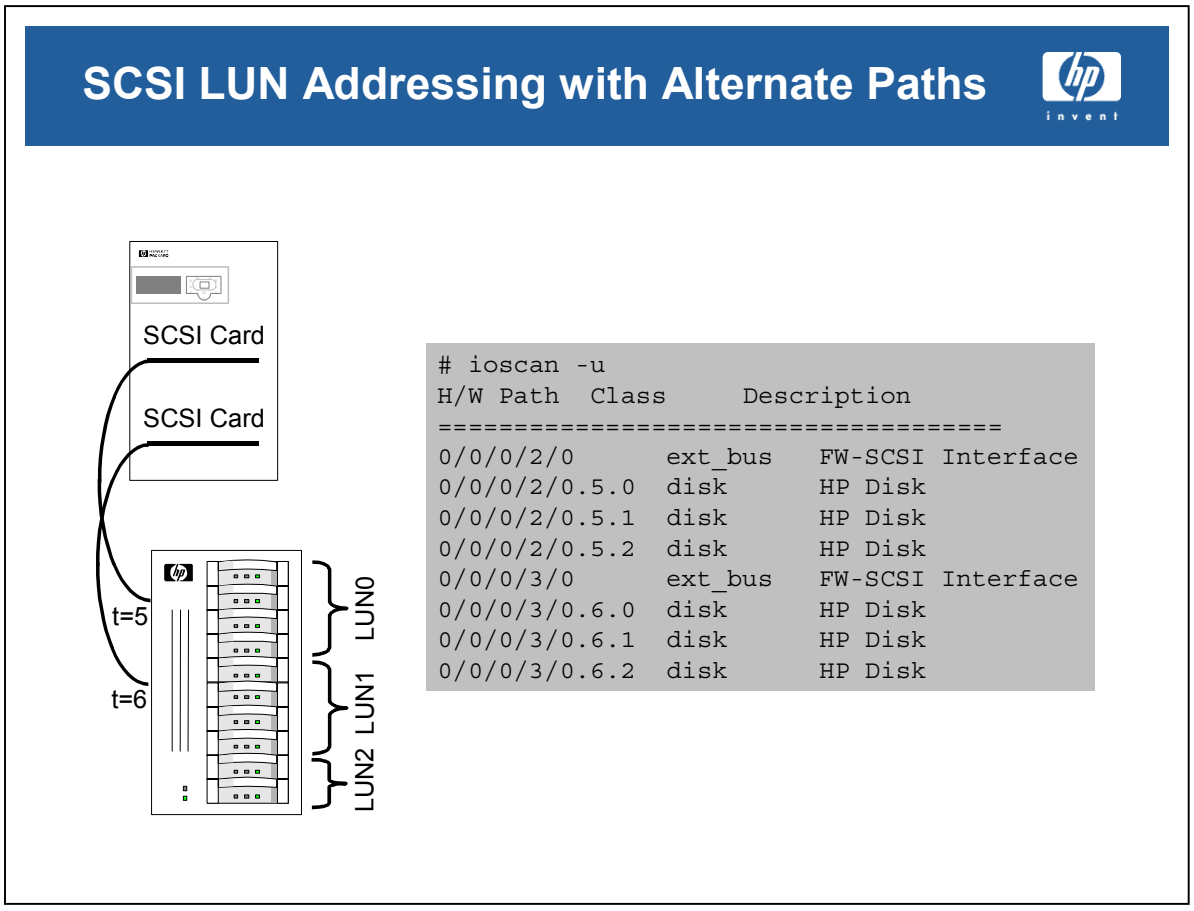
Consider the disk array shown in the diagram on the slide. The disk array has ten physical disks. However, software included with the array makes it possible to treat several physical disks as if they were a single, larger, "logical unit" also known as a LUN. The disk array shown on the slide has three LUNs configured. Each LUN is assigned a unique LUN number, and appears as a distinct entry in the **`ioscan`** output. The LUN number for each **`ioscan`** entry is reflected in the last portion of the hardware path.

© 2004 Hewlett-Packard Development Company, L.P.

```
# ioscan -u
H/W Path       Class     Description
=========================================
0/0/0/2/0      ext_bus   FW-SCSI Interface
0/0/0/2/0.5.0  disk      HP Disk
0/0/0/2/0.5.1  disk      HP Disk
0/0/0/2/0.5.2  disk      HP Disk
0/0/0/2/0.7.0  sctl      Initiator
```


© 2004 Hewlett-Packard Development Company, L.P.

5–19. SLIDE: SCSI LUN Addressing with Alternate Paths



Student Notes

In order to increase system uptime, many disk arrays have not one, but two array controllers. Each controller has a distinct SCSI target address, and connectivity to a separate SCSI interface card on the host server. On most arrays, any LUN may be accessed via either controller.

The **`ioscan`** output on systems like this can be somewhat confusing. Since each LUN is accessible via two different paths, each LUN has two different hardware addresses and appears twice in the **`ioscan`** output.

The example on the slide shows a server connected to a disk array with two array controllers. Both controllers are connected to the server: the controller with target address 5 is connected via the SCSI card at 0/0/0/2/0, and the controller with target address 6 is connected via the SCSI card at 0/0/0/3/0. All three LUNs are accessible through both cards. As a result, each LUN presents two hardware paths to the server:

```
0/0/0/2/0.6.0 = 0/0/0/3/0.5.0
0/0/0/2/0.6.1 = 0/0/0/3/0.5.1
0/0/0/2/0.6.2 = 0/0/0/3/0.5.2
```


© 2004 Hewlett-Packard Development Company, L.P.

The **ioscan** output on such a system would look something like this:

```
# ioscan -u
H/W Path       Class     Description
====================================
0/0/0/2/0         ext_bus   FW-SCSI Interface
0/0/0/2/0.5.0  disk      HP Disk
0/0/0/2/0.5.1  disk      HP Disk
0/0/0/2/0.5.2  disk      HP Disk
0/0/0/3/0      ext_bus   FW-SCSI Interface
0/0/0/3/0.6.0  disk      HP Disk
0/0/0/3/0.6.1  disk      HP Disk
0/0/0/3/0.6.2  disk      HP Disk
```

As long as at least one path remains between the server and array, the LUNs remain physically accessible.

LVM Physical Volume Links

In order to fully utilize redundant physical paths to disk array, your disk space management software may require additional configuration. If you use HP's Logical Volume Manager (LVM) to manage disk space, "Physical Volume Links" (PV Links) must be configured in order for LVM to recognize the alternate links.

LVM designates one path to the array as the primary path, and the second path as an alternate. Initially, all I/O requests are channeled through the array's primary path. If the primary path fails, the alternate path becomes the primary, and the failed path becomes the alternate. On LVM systems, both PV Links must be assigned to the same LVM volume group for LVM to properly use and control access to the alternate paths.

Not all disk arrays support LVM PV Links. Check with your array vendor before you attempt to implement this functionality.

VxVM Dynamic Multipathing

Veritas Volume Manager (VxVM) is more sophisticated than LVM; its Dynamic Multipathing (DMP) daemon recognizes both paths to the disk array automatically. Much like LVM, the base VxVM product that is included with HP-UX channels all I/O requests through one of the paths initially. If that path fails, VxVM automatically redirects requests to the other still functional path.

If you purchase the add-on "Full" version of VxVM (B9116AA), VxVM will interleave requests across both physical links to increase I/O throughput.

Not all disk arrays support VxVM DMP. Check with your array vendor before you attempt to implement this functionality.

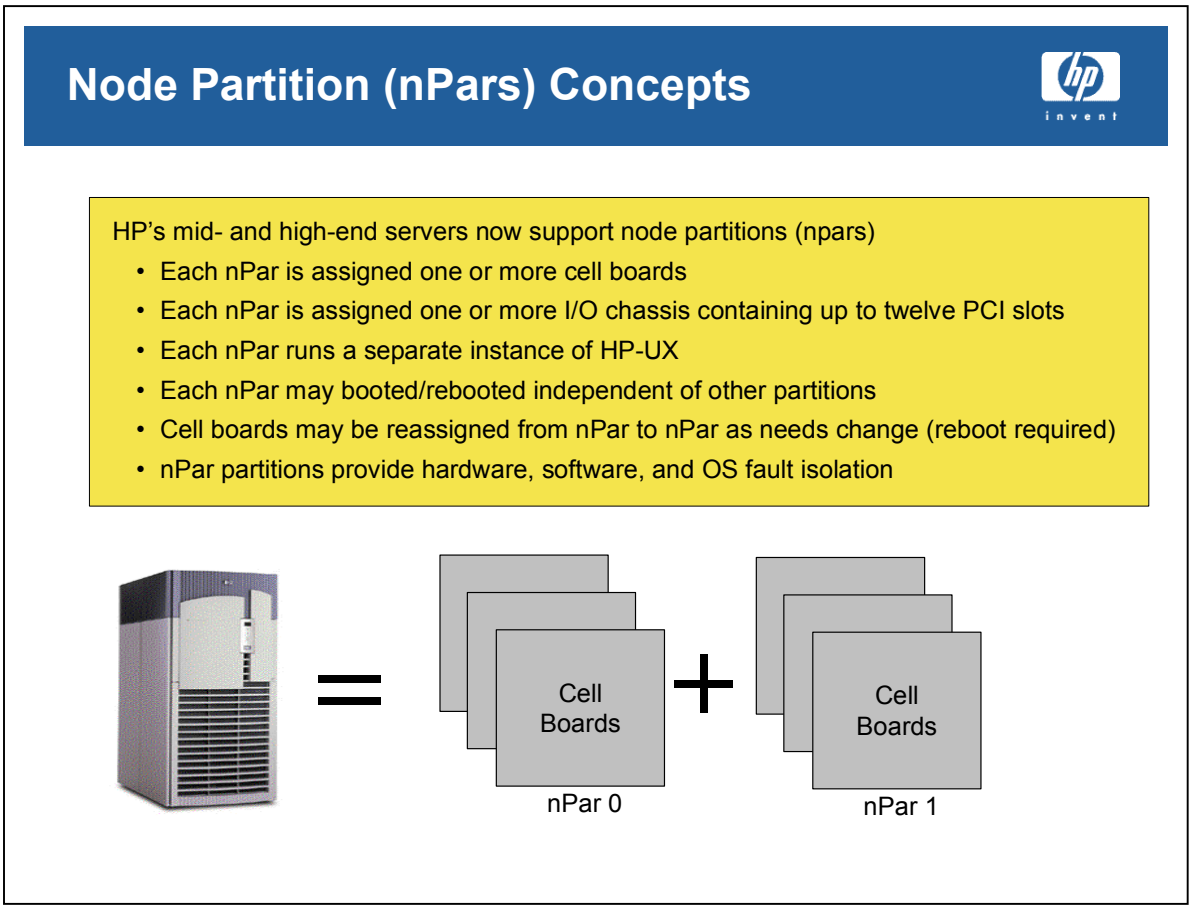
© 2004 Hewlett-Packard Development Company, L.P.

Array Specific Multipathing

Many disk array vendors offer additional software that can be configured in the kernel to provide multi-pathing capabilities independent of LVM or VxVM. HP's Storageworks Secure Path product provides this functionality for HP's XP, EVA, and VA disk arrays. The Power Path product from EMC provides similar functionality for EMC disk arrays. To learn more about HP Storageworks Secure Path, visit **`http://www.hp.com`**.

© 2004 Hewlett-Packard Development Company, L.P.

5–20. SLIDE: Node Partition (nPar) Concepts



Student Notes

HP's nPar functionality makes it possible to subdivide a server's resources into one or more **Node Partitions** (**nPartitions** or **nPars**). Each nPar includes one or more cell boards (containing processors and memory), and all of the I/O chassis connected to those cells. Each nPar runs a separate instance of HP-UX 11i that may be tuned, managed, and rebooted independently from the other partitions.

You can reconfigure a server's nPar definitions without moving any cables or cards. Thus, if a partition's resources are underutilized, cell boards (and their associated I/O chassis and peripheral devices) may be reallocated from the underutilized partition to other overburdened partitions. If a cell board in a partition fails, the cell board may be replaced without shutting down or otherwise affecting the other partitions. Changing the configuration of a node partition does, however, require the nPar to be rebooted.using HP's special **`-R`** and **`-H`** "Reset-to-Reconfig" options on the **`shutdown`** and **`reboot`** commands. The partition must then be manually booted to allow the cells to download the new partition configuration.

HP provides a GUI-based partition configuration tool called **`parmgr`**, a collection of command-line nPar utilities, and an enhanced "Guardian Service Processor" or "Management

© 2004 Hewlett-Packard Development Company, L.P.

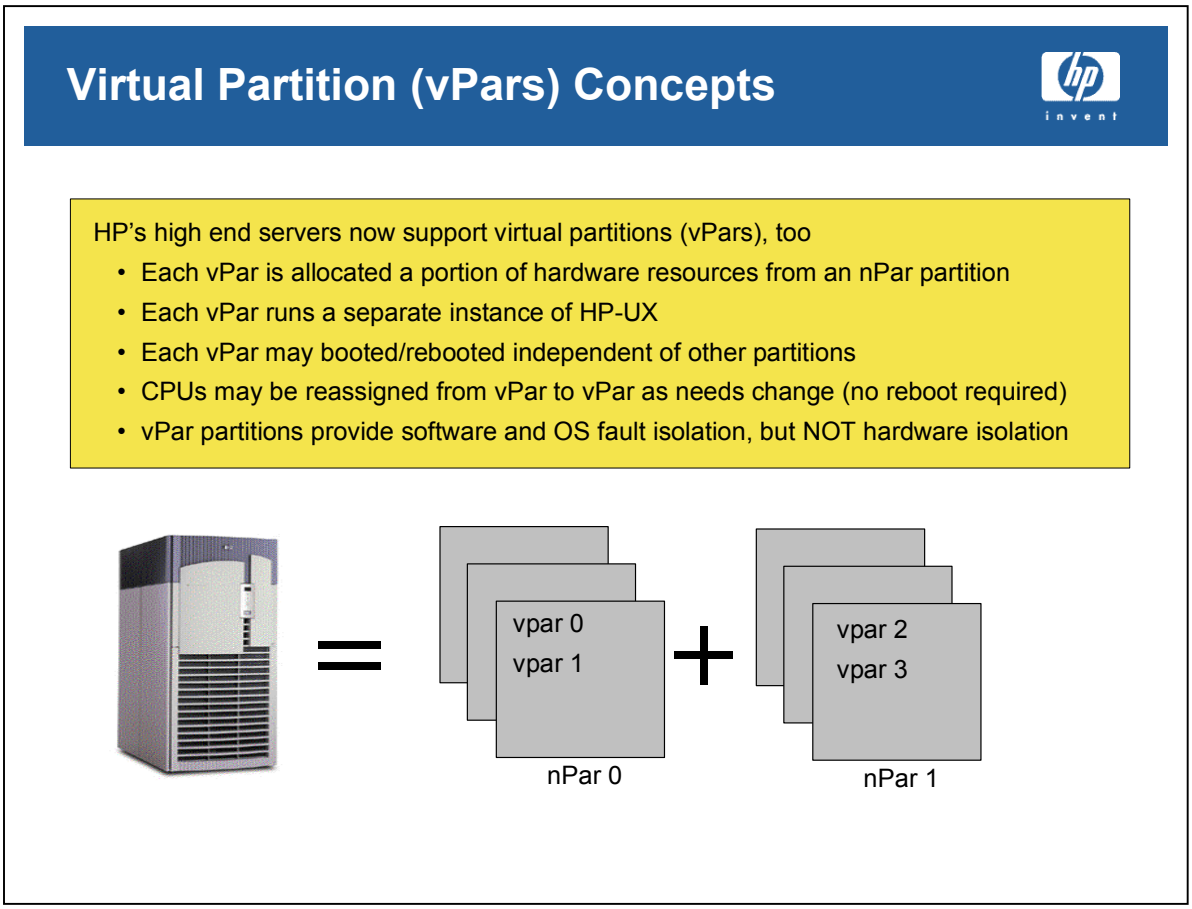
Processor" partition-aware console interface that may be used to boot, reboot, troubleshoot, and manage partitions.

nPar partitions provide hardware, software, and O/S fault isolation that is particularly attractive to administrators who are managing server consolidation projects.

This feature is currently offered on all of HP's PA-RISC and Integrity cell-based servers. To determine if your system supports node partitioning, or to learn more about partition management, read the HP System Partitions Guide on **`http://docs.hp.com`**. HP Education's Superdome curriculum courses also provide extensive coverage of nPar configuration and management.

© 2004 Hewlett-Packard Development Company, L.P.

5–21. SLIDE: Virtual Partition (vPar) Concepts



Student Notes

Many PA-RISC servers now support **Virtual Partitions (vPars)** functionality, too. Using HP's vPars product, you can subdivide an nPar's resources into multiple, smaller virtual machines. Currently, vPars are only supported on select PA-RISC systems. Support for vPars on Integrity will be added in a future release.

Each Virtual Partition is assigned a subset of its nPar's hardware. At a minimum, each vPar must be assigned one CPU, and one I/O bus (LBA) that has a boot device with HP-UX 11i and the vPars software product installed. An I/O bus with a network interface card also should be assigned to each vPar to support networking. Only the active hardware assigned to the local nPar can be used by vPars within the nPar. Hardware that is assigned to other "remote" nPars cannot be used by vPars in the local nPar.

You can reconfigure vPars and dynamically reallocate certain processors among vPars in the local nPar. Currently, only processors can be dynamically reallocated.

Much like an nPar, each vPar runs its own instance of HP-UX that can be independently tuned, managed, and rebooted. However, vPars offer less robust hardware isolation than nPar partitions. If an nPar's cell board fails, all vPars within the nPar may be affected. Other

© 2004 Hewlett-Packard Development Company, L.P.

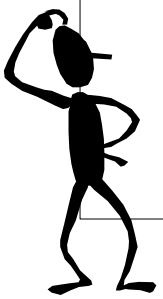
nPar partitions on the system remain functional even while the failed partition's cell board is being replaced.

© 2004 Hewlett-Packard Development Company, L.P.

5–22. SLIDE: nPar and vPar Hardware Addresses

nPar and vPar Hardware Addresses

- Hardware resources allocated to one partition aren't visible to other partitions
- `ioscan`, `rad`, `olrad`, and map only display devices in the current partition
- The `parstatus` command reports which components are assigned to each nPar



Q: Why do I only see half of my interface cards and cell boards in my `ioscan` output?

A: `ioscan` only reports the devices available in my current partition!

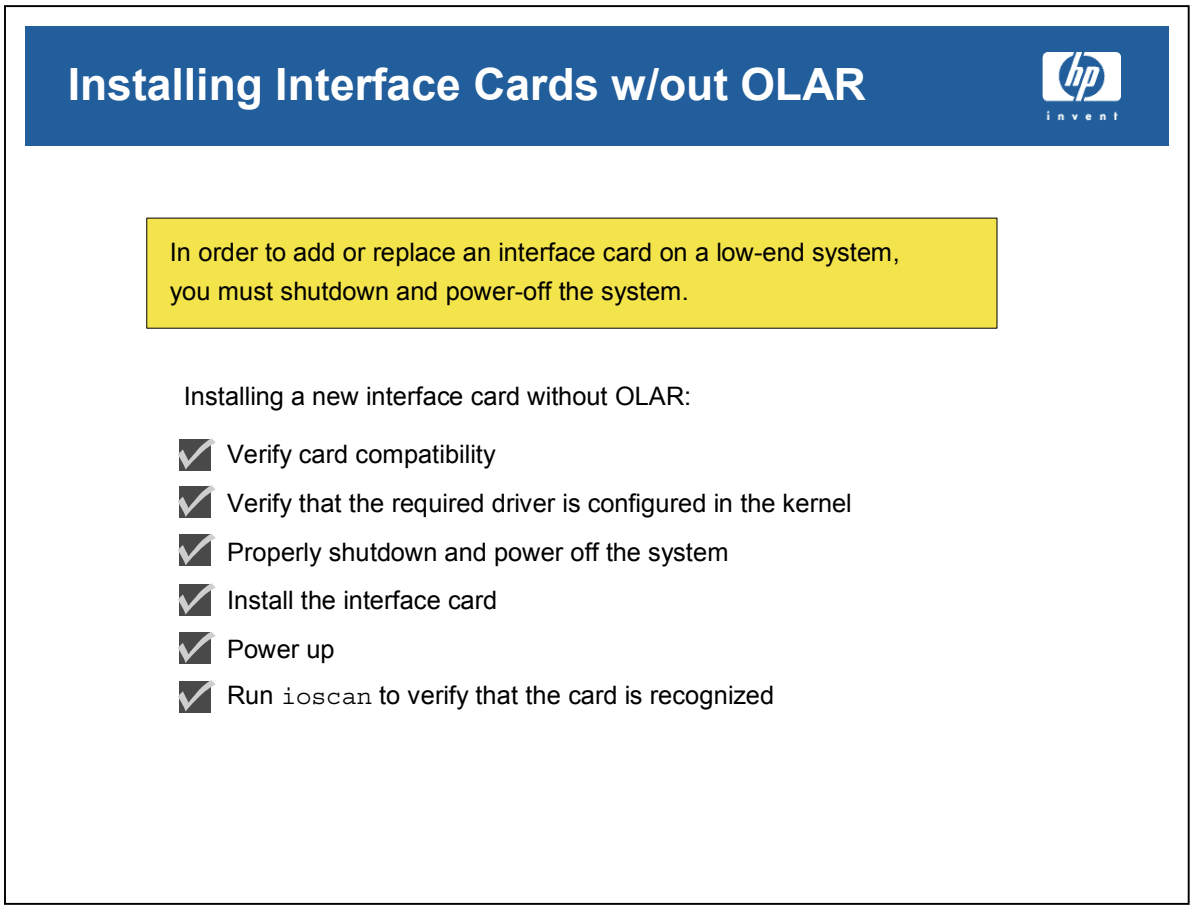
Student Notes

Hardware resources allocated to one partition aren't visible to other partitions. **`ioscan`**, **`rad`**, **`olrad`**, **`pdweb`**, **`sam`**, and **`map`** only display devices in the current node or virtual partition.

If you want to see which resources have been assigned to other node partitions on your system, run the **`parstatus`** command. Similarly, the **`vparstatus`** command reports which resources have been allocated to other virtual partitions.

© 2004 Hewlett-Packard Development Company, L.P.

5–23. SLIDE: Installing Interface Cards without OLA/R



Student Notes

In past years, customers often relied on HP Customer Engineers to install and remove hardware devices for them. These days, however, many administrators choose to add and remove hardware components themselves. The procedures on this and the following two slides describe how to properly install additional interface cards on your server or workstation.

On workstations and entry-level servers, you must shutdown and power-off your system as described below in order to add or replace an interface card. On servers that support HP's OnLine Addition and Replacement (OLA/R) functionality, you can shutdown a single card slot without shutting down your entire server. The OLA/R process is described over the next couple slides.

Installing a new interface card without OLA/R is a multi-step process:

1. Verify card compatibility. HP servers and workstations support a variety of I/O expansion buses. Make sure your new interface card matches the card slot that you intend to use. PCI cards, for instance, may not be installed on an EISA bus! See your owner's manual to determine which card types can be used in each card slot.

© 2004 Hewlett-Packard Development Company, L.P.

2. Verify that the required driver is configured in the kernel. Check your interface card documentation to determine what driver is required, then use **sam** or **kcweb** to determine if the required driver is configured in your kernel. A later chapter in this course discusses kernel configuration in detail.

3. Use the **shutdown** command to properly shut down the system. When you see a message indicating that it is safe to power-off, either press the power button, or use the Management Processor power control (**pc**) command to power-off your system.

   ```
   # shutdown -hy 0
   ```

4. Install the interface card. Static discharge can easily damage interface cards. Be sure to follow the anti-static guidelines that come with your interface card.

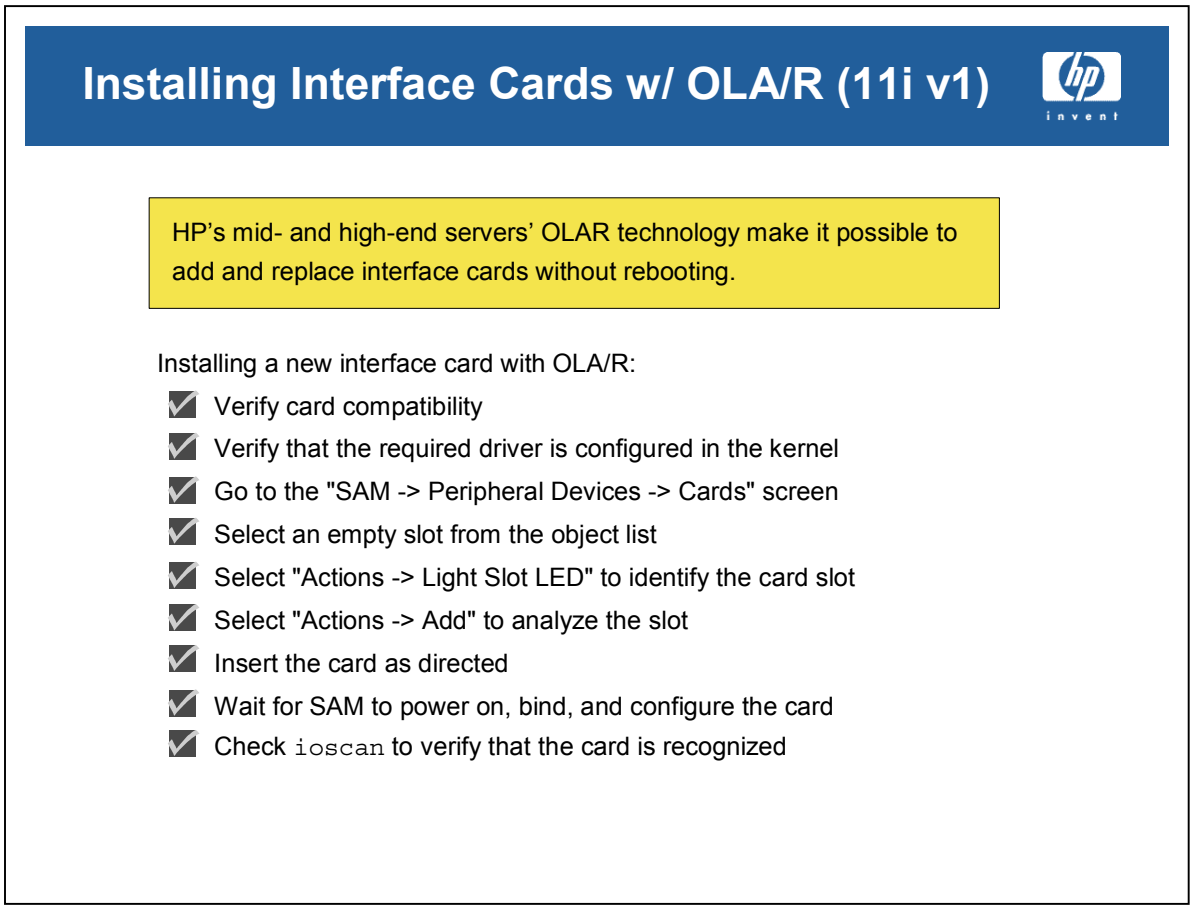
5. Power-up the system.

6. During the system boot process, the kernel should scan the system for new interface cards and devices. Use **ioscan** to verify that the system recognized your new interface card. Does your new device appear in the device list? Is the new device CLAIMED? If not, it may be necessary to add a new driver to the kernel. See our kernel configuration chapter later in the course!

WARNING:	Be sure to check your support agreement before you open the cabinet of any HP system. Attempting to service hardware components without the assistance of an HP engineer may invalidate your warranty or support agreement.

© 2004 Hewlett-Packard Development Company, L.P.

5–24. SLIDE: Installing Interface Cards with OLA/R (11i v1)



Student Notes

Prior to HP-UX 11i, adding or removing an interface card always required a system reboot, as described in the process on the preceding slide. HP-UX 11i v1 introduced a new technology called "On Line Addition and Replacement" (OLA/R), which provides the ability to add, remove, and replace PCI interface cards without a system reboot. The OLA/R functionality is currently only supported for selected PCI interface cards, on selected servers running HP-UX 11i. See your system hardware manual to determine if your server supports this functionality. The notes below describe the OLA/R process on 11i v1. The next slide describes the 11i v2 OLA/R process.

Installing a New Interface Card with OLA/R

Installing an interface card using the new OLA/R functionality is a multi-step process that may be performed from the command line via the **`/usr/bin/rad`** command, or by using the SAM GUI/TUI. HP strongly recommends using SAM for OLA/R administration. The procedure for adding an OLA/R interface card via SAM is described below. If you prefer to work from the command line, see chapter 2 in HP's *Configuring Peripherals for HP-UX* manual on **`http://docs.hp.com`**.

1. Verify card compatibility. Check the documentation accompanying your interface card to verify that the card is OLA/R compatible. Check your system owner's manual for details.

© 2004 Hewlett-Packard Development Company, L.P.

2. Verify that the required driver is configured in the kernel. Without the proper driver configured, you may be able to physically install the card, but the card will be unusable. Check your interface card documentation to determine what driver is required, then use the **`sam -> Kernel Configuration -> Drivers`** screen to determine if the required driver is configured in your kernel. A later chapter will describe the process required to add a new driver.

3. Go to the **`sam -> Peripheral Devices -> Cards`** screen. This screen lists all of the interface cards installed on your system, and includes several items in the **`Actions`** menu for managing OLA/R interface cards.

```
+------------------------- Cards ---------------------------------+
|File View Options Actions                                   Help |
|I/O Cards                                        2 of 23 selected|
|-----------------------------------------------------------------|
|        Hardware                      Slot                       |
|  Slot  Path     Driver State         Power Description          |
|+---------------------------------------------------------------+ |
|| -     0/0/2/1  c720   not OLAR-able -     SCSI C87x Fast Wid | |
|| -     0/0/4/0  func0  not OLAR-able -     PCI BaseSystem (10 | |
|| -     0/0/4/1  asio0  not OLAR-able -     Service Processor  | |
|| 5     0/2/0    -      -             on    empty slot         | |
|| 6     0/5/0    -      -             on    empty slot         | |
|| 7     0/1/0    -      -             on    empty slot         | |
|| 8     0/3/0    -      -             on    empty slot         | |
|| 9     0/9/0/0  c8xx   active        on    SCSI C1010 Ultra W | |
|| 9     0/9/0/1  c8xx   active        on    SCSI C1010 Ultra W | |
|| 10    0/8/0/0  c8xx   active        on    SCSI C1010 Ultra W | |
|| 10    0/8/0/1  c8xx   active        on    SCSI C1010 Ultra W | |
|+---------------------------------------------------------------+ |
+-----------------------------------------------------------------+
```

4. Select an empty slot from the object list. Slots that are available for use by new interface cards should be marked either **`empty slot`** or **`unclaimed card`** in the **`Description`** field. Select one of these slots.

```
+------------------------- Cards ---------------------------------+
|File View Options Actions                                   Help |
|I/O Cards                                        2 of 23 selected|
|-----------------------------------------------------------------|
|        Hardware                      Slot                       |
|  Slot  Path     Driver State         Power Description          |
|+---------------------------------------------------------------+ |
|| -     0/0/2/1  c720   not OLAR-able -     SCSI C87x Fast Wid | |
|| -     0/0/4/0  func0  not OLAR-able -     PCI BaseSystem (10 | |
|| -     0/0/4/1  asio0  not OLAR-able -     Service Processor  | |
|| 5     0/2/0    -      -             on    empty slot         | |
|| 6     0/5/0    -      -             on    empty slot         | |
|| 7     0/1/0    -      -             on    empty slot         | |
|| 8     0/3/0    -      -             on    empty slot         | |
|| 9     0/9/0/0  c8xx   active        on    SCSI C1010 Ultra W | |
|| 9     0/9/0/1  c8xx   active        on    SCSI C1010 Ultra W | |
|| 10    0/8/0/0  c8xx   active        on    SCSI C1010 Ultra W | |
|| 10    0/8/0/1  c8xx   active        on    SCSI C1010 Ultra W | |
|+---------------------------------------------------------------+ |
+-----------------------------------------------------------------+
```


© 2004 Hewlett-Packard Development Company, L.P.

5. Select **`Actions -> Light Slot LED`** to identify the card slot on the system backplane. This should light an LED on the selected PCI slot to indicate which slot should be used for the new interface card.

6. Select **`Actions -> Add`** to analyze the slot. In order to insert the new interface card, the selected PCI slot must be powered down. On some servers, multiple slots may share a common "power domain". Slots within a power domain are powered on or off as a unit. Powering off the power domain containing the interface card for the system boot disk or other critical system resources could be disastrous! SAM automatically analyzes the selected slot's power domain to ensure that it is safe to temporarily disable the power domain while the new card is being added.

7. Insert the card as directed by SAM.

8. SAM will power-on the card, identify and "bind" an appropriate kernel driver, and run a post addition script to finish configuring the card, if necessary.

9. Check **`ioscan`** to verify that the card is recognized. Does the card appear in the hardware list? Is it CLAIMED?

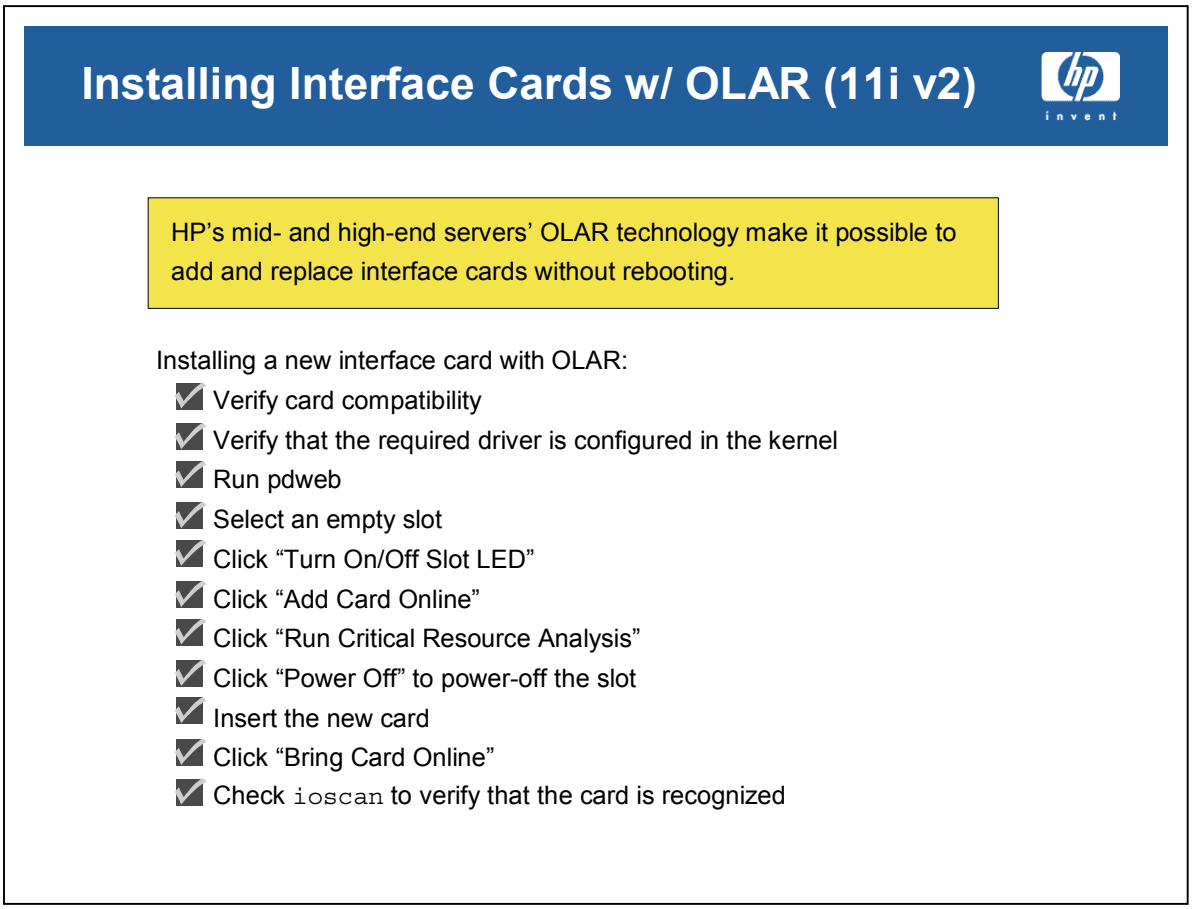
Other OLA/R Possibilities

OLA/R also makes it possible to replace interface cards without rebooting. Simply select the **`Actions -> Replace`** menu item rather than **`Actions -> Add`**. There are some restrictions, however. Generally speaking, the replacement card must be identical to the original card. See Chapter 2 in HP's "*Configuring Peripherals for HP-UX*" manual on **`http://docs.hp.com`** for more information on these procedures.

WARNING:	Be sure to check your support agreement before you open the cabinet of any HP system. Attempting to service hardware components without the assistance of an HP engineer may invalidate your warranty or support agreement.

© 2004 Hewlett-Packard Development Company, L.P.

5–25. SLIDE: Installing Interface Cards with OLA/R (11i v2)



Student Notes

The **pdweb** web-based interface in 11i v2 provides an even more intuitive interface for adding and replacing OLA/R capable interface cards.

Installing a New Interface Card with OLA/R

Installing an OLA/R capable interface card in 11i v2 is a multi-step process that may be performed from the command line via the **/usr/bin/olrad** command, or by using the **pdweb** web interface. HP strongly recommends using SAM for OLA/R administration. The procedure for adding an OLA/R interface card via **pdweb** is described below. If you prefer to work from the command line, see chapter 2 in HP's *Configuring Peripherals for HP-UX* manual on **http://docs.hp.com**.

1. Verify card compatibility. Check the documentation accompanying your interface card to verify that the card is OLA/R compatible. Check your system's hardware manual for details.

2. Verify that the required driver is configured in the kernel. Without the proper driver configured, you may be able to physically install the card, but the card will be unusable. Check your interface card documentation to determine what driver is required, then use

© 2004 Hewlett-Packard Development Company, L.P.

the **kcweb** command to determine if the required driver is configured in your kernel. A later chapter will describe the process required to view and add drivers.

3. Launch **pdweb**. If you are logged in via CDE directly on the server you wish to manage, simply type **kcweb**. If the hostname in your **DISPLAY** variable is different from the server's own hostname, you will have to add the **-F** (force) option. Note, however, that setting an X-Windows remote **DISPLAY** variable can introduce serious security problems on your system. Consider using the remote browser access method below instead.

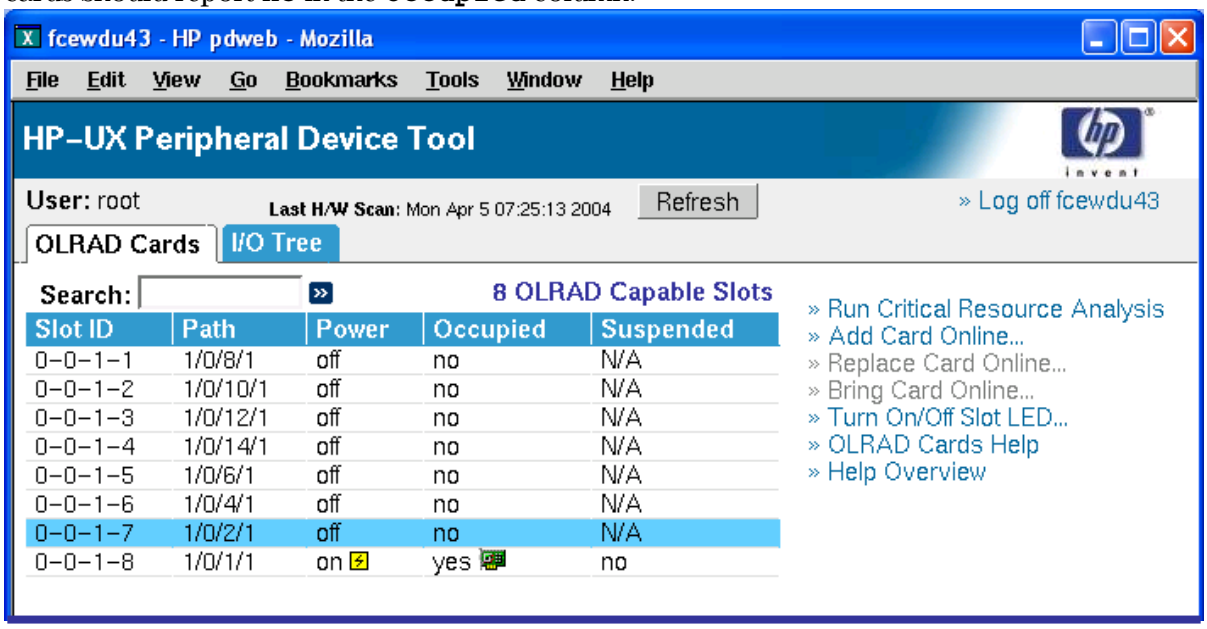
   ```
   # kcweb [-F]
   ```

   If you are accessing the server from a remote client, enable the daemon autostart functionality on the server if you haven't done so previously, then point your web browser to the pdweb URL.

   ```
   server# waconf -a on
   client# mozilla http://server:1110/pd/pd.cgi
   ```

   Log into **pdweb** using the root username and password.
   An earlier slide in this chapter discussed this process in much greater detail.

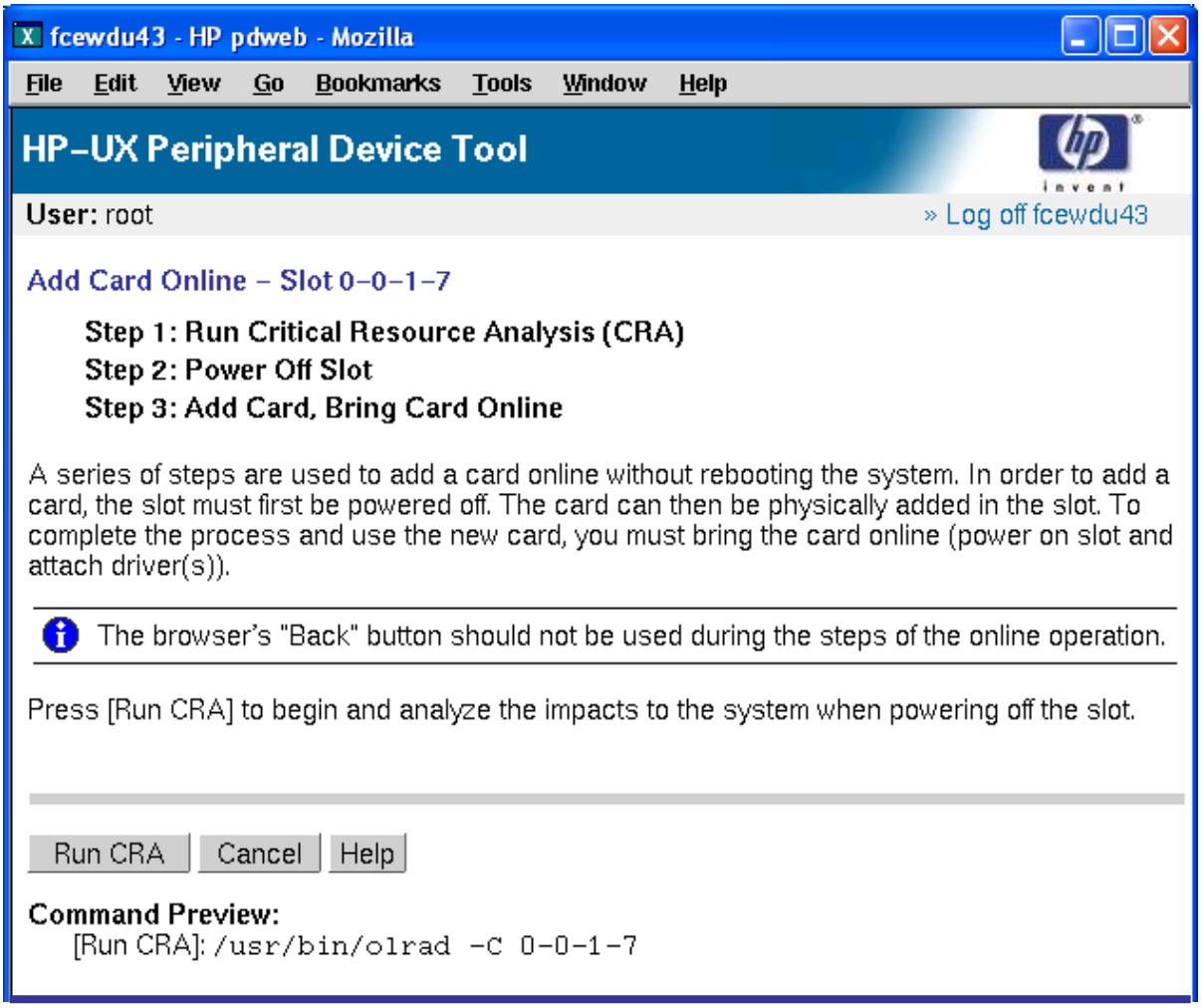
4. Select an empty slot from the object list. Slots that are available for use by new interface cards should report **no** in the **Occupied** column.



5. Click Turn On/Off Slot LED. Check the slot LEDs on the backplane of your system to verify that you selected the right slot.

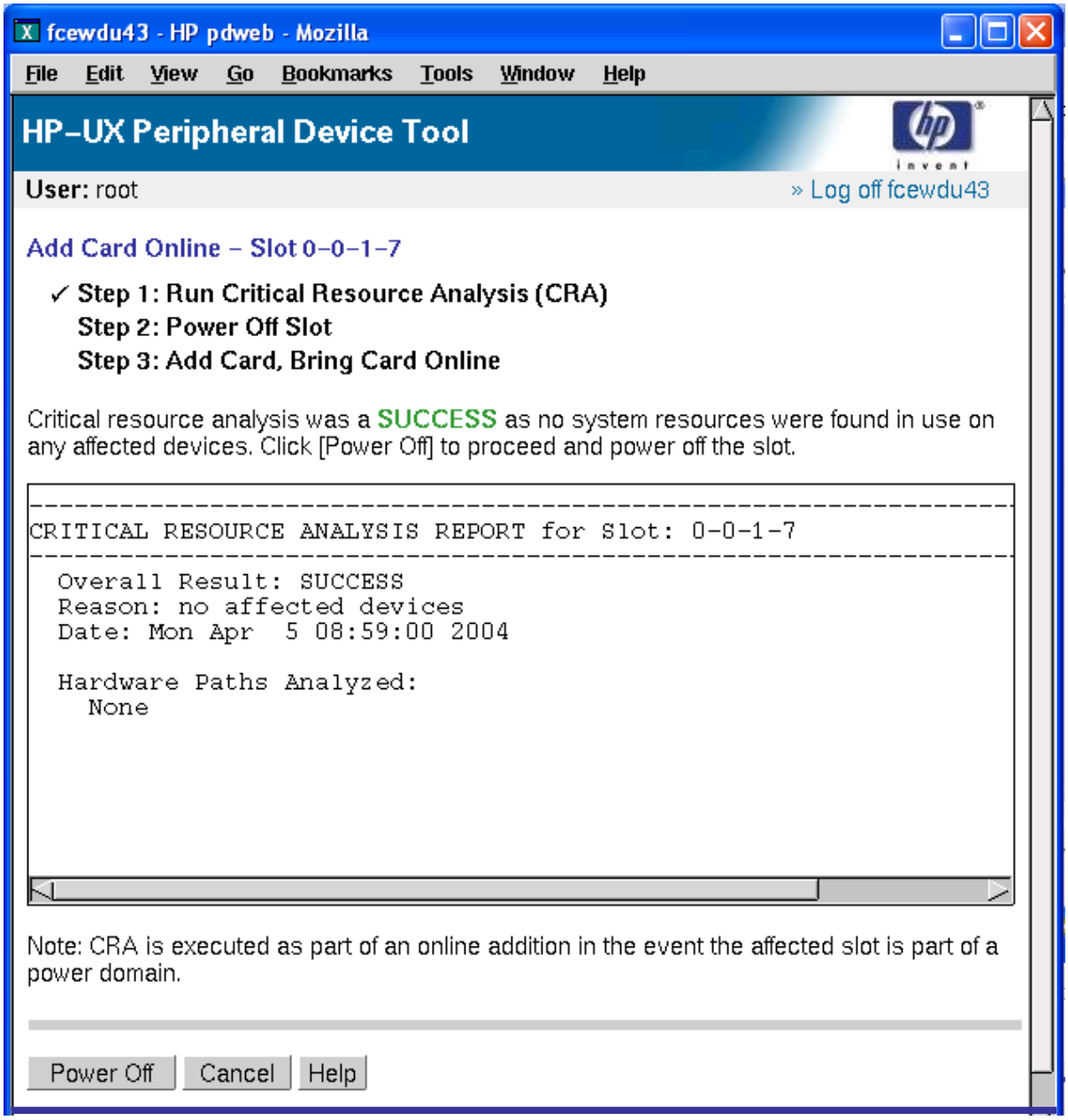
© 2004 Hewlett-Packard Development Company, L.P.

6. Click **`Add Card Online`**. This should display a dialog box similar to the following:



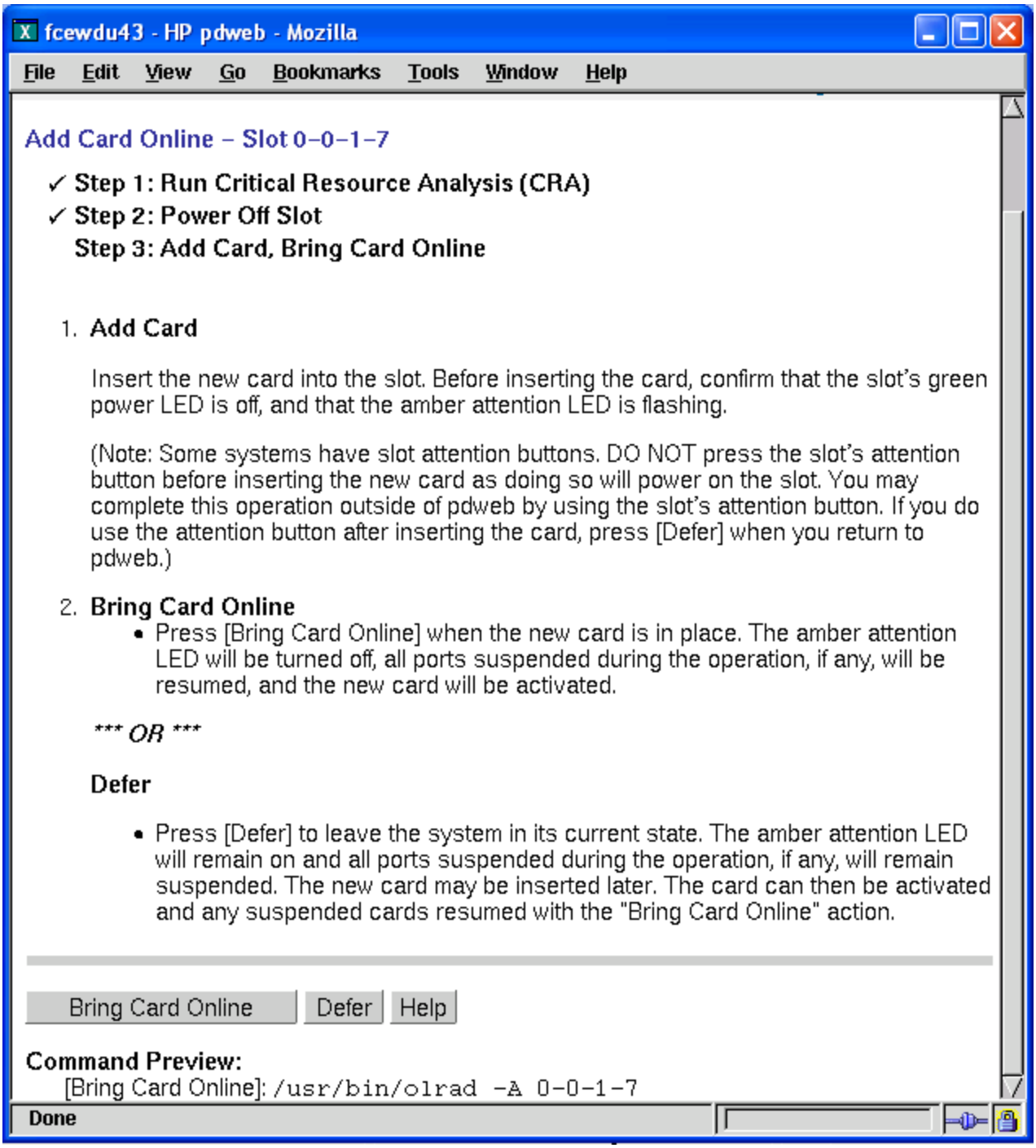
© 2004 Hewlett-Packard Development Company, L.P.

7. Click **`Run CRA`** (Critical Resource Analysis) to analyze the slot. In order to insert the new interface card, the selected PCI slot must be powered down. On some servers, multiple slots may share a common OLA/R "power domain". Slots within a power domain are powered on or off as a unit. Powering off the power domain containing the interface card for the system boot disk or other critical system resources could be disastrous! **`pdweb`** automatically analyzes the selected slot's power domain to ensure that it is safe to temporarily disable the power domain while the new card is being added. If the CRA succeeds, you should see a screen like this:



© 2004 Hewlett-Packard Development Company, L.P.

8. If the CRA succeeds, click the **`Power Off`** button to power down the slot. Momentarily you should see a screen similar to the following to indicate that it is safe to insert the new card:



9. Insert the new card.

10. Click **`Bring Card Online`**

© 2004 Hewlett-Packard Development Company, L.P.

11. Finally, check **`ioscan`** to verify that the card is recognized. Does the card appear in the hardware list? Is it CLAIMED?

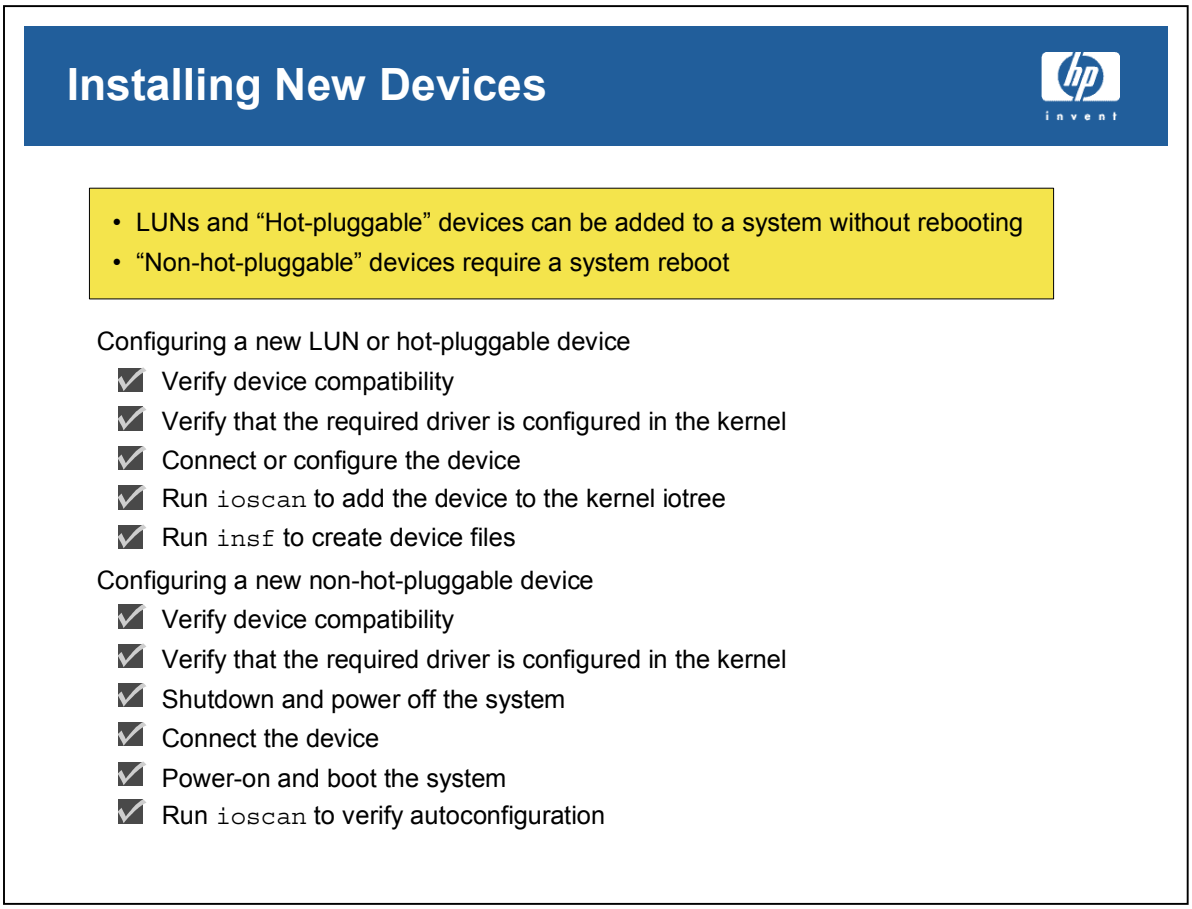
Other OLA/R Possibilities

`pdweb` also makes it possible to replace interface cards without rebooting. Simply select a card and select the **`Replace Card Online`** link on the main menu. There are some restrictions, however. Generally speaking, the replacement card must be identical to the original card. See Chapter 2 in HP's "*Configuring Peripherals for HP-UX*" manual on **`http://docs.hp.com`** for more information on these procedures.

WARNING:	Be sure to check your support agreement before you open the cabinet of any HP system. Attempting to service hardware components without the assistance of an HP engineer may invalidate your warranty or support agreement.

© 2004 Hewlett-Packard Development Company, L.P.

5–26. SLIDE: Installing New Devices



Student Notes

After installing a new interface card, you may choose to attach new devices, too. The procedures below explain the process to install both “hot-pluggable” and “non-hot-pluggable” devices.

Configuring a New LUN or Hot-Pluggable Device

Many devices today, such as the media bays on most of the current servers, are “hot-pluggable”. Hot pluggable devices can be installed or removed without shutting down the system. If you create or remove LUNs on a Storage Area Network, this same procedure may be used to force HP-UX to recognize the new configuration.

1. Verify device compatibility. Check HP’s website or call the response center to verify that the device is supported on your system.

2. Verify that the required driver is configured in the kernel via **`sam`** (11i v1) or **`kcweb`** (11i v2).

3. Connect or configure the device.

© 2004 Hewlett-Packard Development Company, L.P.

4. Run **ioscan** to add the device to the kernel iotree. Don't include the **-u** or **-k** options; in order to recognize the new device, **ioscan** must scan the buses rather than simply report the devices already recorded in the iotree.

5. Run **insf** to create device files. Device files allow users and applications to access peripheral devices. Device files are discussed in detail in the next chapter.

NOTE:	Even if a card is hot-pluggable, you must shutdown any daemons using the device before you remove the device.

Configuring a New Non-Hot-Pluggable Device

In order to install some devices, a reboot may be required.

1. Verify device compatibility. Check HP's website or call the response center to verify that the device is supported on your system.

2. Verify that the required driver is configured in the kernel via **sam** (11i v1) or **kcweb** (11i v2).

3. Shutdown and power off the system.

4. Connect the device.

5. Power-on and boot the system.

6. Run **ioscan** to verify auto-configuration. Verify that the device appears in the **ioscan** output and is **CLAIMED**.

Additional Configuration

Some additional configuration may be required after physically connecting a new device. Terminals and modems may require new device files. Disks may need to be partitioned before they may be used. The next couple chapters will discuss these additional configuration tasks in detail.

© 2004 Hewlett-Packard Development Company, L.P.

5–27. LAB: Viewing the Hardware Configuration with `ioscan`

Directions

The **`ioscan`** command is a powerful tool for exploring your system's hardware configuration.

The sample **`ioscan`** output shown below was taken from an rp8400 server. Use the sample output to answer the questions that follow.

```
# ioscan
H/W Path      Class             Description
============================================================================
0             cell
0/0              ioa               System Bus Adapter (804)
0/0/0               ba             Local PCI Bus Adapter (782)
0/0/0/0/1              tty         PCI Serial
0/0/0/1/0              lan         HP A3639-60019 1000Base-T Built-in I/O
0/0/0/2/0              ext_bus     SCSI C87x Fast Wide Single-Ended
0/0/0/2/0.6               target
0/0/0/2/0.6.0                disk  HP 36.4GST336704LC
0/0/0/2/0.7               target
0/0/0/2/0.7.0                ctl   Initiator
0/0/0/2/1              ext_bus     SCSI C87x Fast Wide Single-Ended
0/0/0/2/1.2               target
0/0/0/2/1.2.0                disk  HP DVD-ROM 305
0/0/0/2/1.7               target
0/0/0/2/1.7.0                ctl   Initiator
0/0/0/3/0              ext_bus     SCSI C896 Fast Wide Single-Ended
0/0/0/3/0.6               target
0/0/0/3/0.6.0                disk  HP 36.4GST336704LC
0/0/0/3/0.7               target
0/0/0/3/0.7.0                ctl   Initiator
0/0/0/3/1              ext_bus     SCSI C896 Fast Wide LVD
0/0/0/3/1.7               target
0/0/0/3/1.7.0                ctl   Initiator
0/0/1               ba             Local PCI Bus Adapter (782)
0/0/1/0/0              lan         HP A5230A PCI 10/100Base-TX Addon
0/0/2               ba             Local PCI Bus Adapter (782)
0/0/2/0/0              ext_bus     SCSI C87x Fast Wide Differential
0/0/2/0/0.7               target
0/0/2/0/0.7.0                ctl   Initiator
0/0/4               ba             Local PCI Bus Adapter (782)
0/0/4/0/0              ext_bus     SCSI C87x Fast Wide Differential
0/0/4/0/0.7               target
0/0/4/0/0.7.0                ctl   Initiator
0/0/4/0/1              ext_bus     SCSI C87x Fast Wide Differential
0/0/4/0/1.3               target
0/0/4/0/1.3.0                disk  HP 36.4GST336704LC
0/0/4/0/1.6               target
0/0/4/0/1.6.0                disk  HP 36.4GST336704LC
0/0/4/0/1.7               target
0/0/4/0/1.7.0                ctl   Initiator
0/0/6               ba             Local PCI Bus Adapter (782)
0/0/6/0/0              fc          HP Tachyon Fibre Channel Adapter
```


© 2004 Hewlett-Packard Development Company, L.P.

```
0/0/8                ba             Local PCI Bus Adapter (782)
0/0/8/0/0               fc          HP Tachyon Fibre Channel Adapter
0/0/10               ba             Local PCI Bus Adapter (782)
0/0/10/0/0              fc          HP Tachyon Fibre Channel Adapter
0/0/12               ba             Local PCI Bus Adapter (782)
0/0/12/0/0              fc          HP Tachyon Fibre Channel Adapter
0/0/14               ba             Local PCI Bus Adapter (782)
0/0/14/0/0              fc          HP Tachyon Fibre Channel Adapter
0/5               memory            Memory
0/10              processor         Processor
0/11              processor         Processor
0/12              processor         Processor
0/13              processor         Processor
2              cell
2/5               memory            Memory
2/10              processor         Processor
2/11              processor         Processor
2/12              processor         Processor
2/13              processor         Processor
```

1. How many cell boards are available on this system?

2. How many CPUs are configured on this system?

3. How many LAN interfaces are available? What is the hardware path of the high-speed LAN interface?

4. When you install HP-UX from CD, you will need to know the hardware path of the system's DVD/CD drive. What is the DVD/CD hardware path on this system?

© 2004 Hewlett-Packard Development Company, L.P.

5. How many SCSI interfaces are available on this system? Which card would provide best performance for an Ultra-2 SCSI LVD device?

6. If an administrator chose to put a tape drive on the 0/0/0/2/1 interface, which target addresses could be assigned to the drive?

7. If an administrator chose to put a tape drive with target address 0 on the 0/0/0/2/1 interface card, what would the tape drive's hardware path be?

8. Assuming there are no redundant physical links on this system, if the SCSI interface card at address 0/0/4/0/1fails, will any disks be affected? If so, which ones?

9. How many fibre channel interfaces are available on this system? Are any of these interfaces connected to disk arrays currently?

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 6 — Configuring Device Files

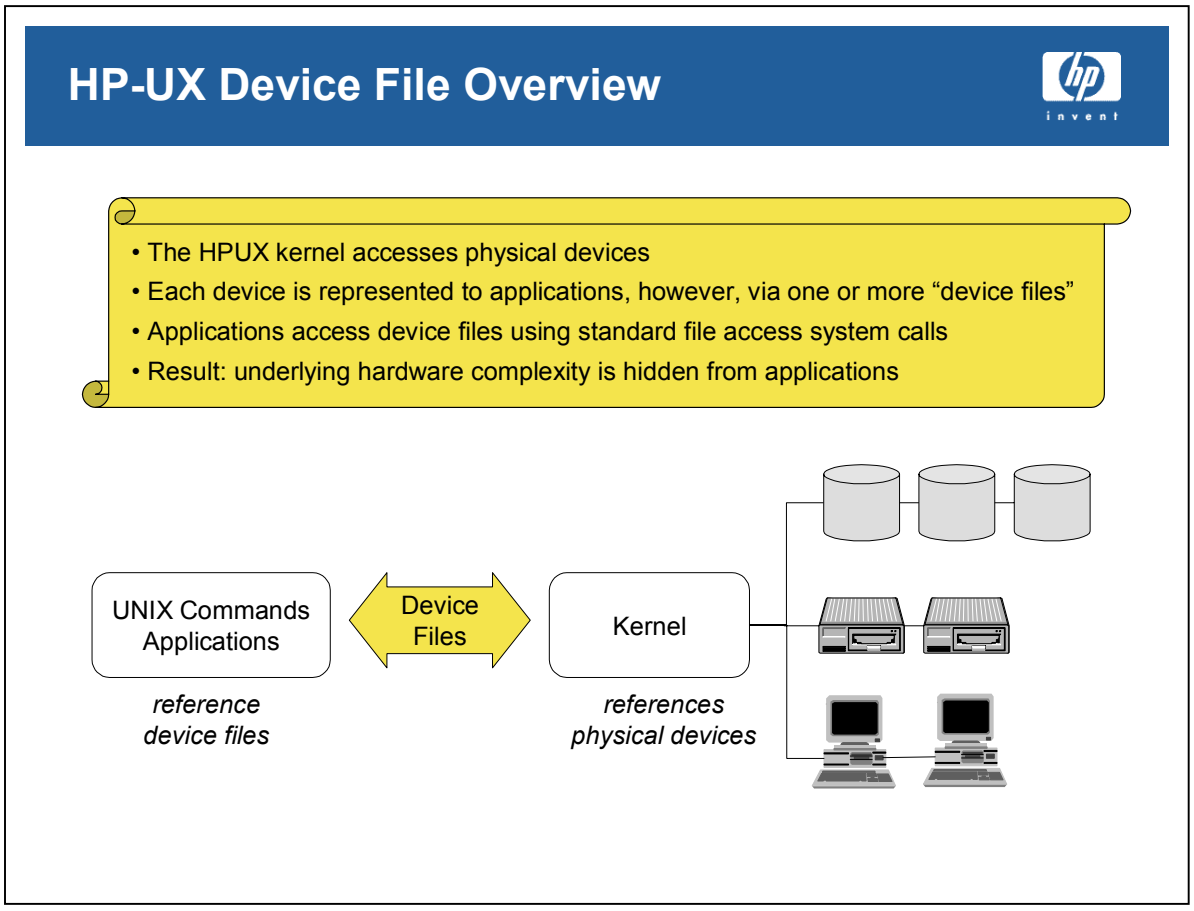
Objectives

Upon completion of this module, you will be able to do the following:

- Explain the purpose of a device file.
- Explain the significance of major and minor numbers.
- Differentiate between block and character I/O.
- Use **`lsdev`** to list kernel driver major numbers.
- Use **`ll`** to determine a device file's major and minor numbers.
- Use **`ioscan`** to list device files associated with a specific device.
- Use **`lssf`** to interpret the characteristics of a device file.
- Given a disk, tape, or CD device file name, determine the controller card instance number and target address of the associated device.
- Given a modem or terminal device file name, determine the controller card instance number and port number of the associated device.
- Create device files using autoconfiguration, SAM, **`insf`**, **`mksf`**, and **`mknod`**.
- Remove device files using **`rmsf`**.

© 2004 Hewlett-Packard Development Company, L.P.

6–1. SLIDE: HP-UX Device File Overview



Student Notes

HP-UX communicates with peripheral devices (such as tape drives, disk drives, printers, terminals, and modems) through files called **device files**. HP-UX treats I/O to a peripheral device in the same manner as I/O to a file. Before HP-UX can communicate with a peripheral device, the device must have a device file. For example, each terminal has its own device file through which HP-UX writes data (which appears on the terminal screen) and reads data (typed by the user at the keyboard).

A device file does not contain data, as a regular file does.

Instead, a device file simply specifies how HP-UX is to communicate with a device. Device files are stored in the **`/dev`** directory.

NOTE:	Device file is synonymous with **special file**. You will see these terms used interchangeably.

The following examples demonstrate the use of device files by HP-UX commands:

```
# tar -cvf /dev/rmt/0m /usr
```


© 2004 Hewlett-Packard Development Company, L.P.

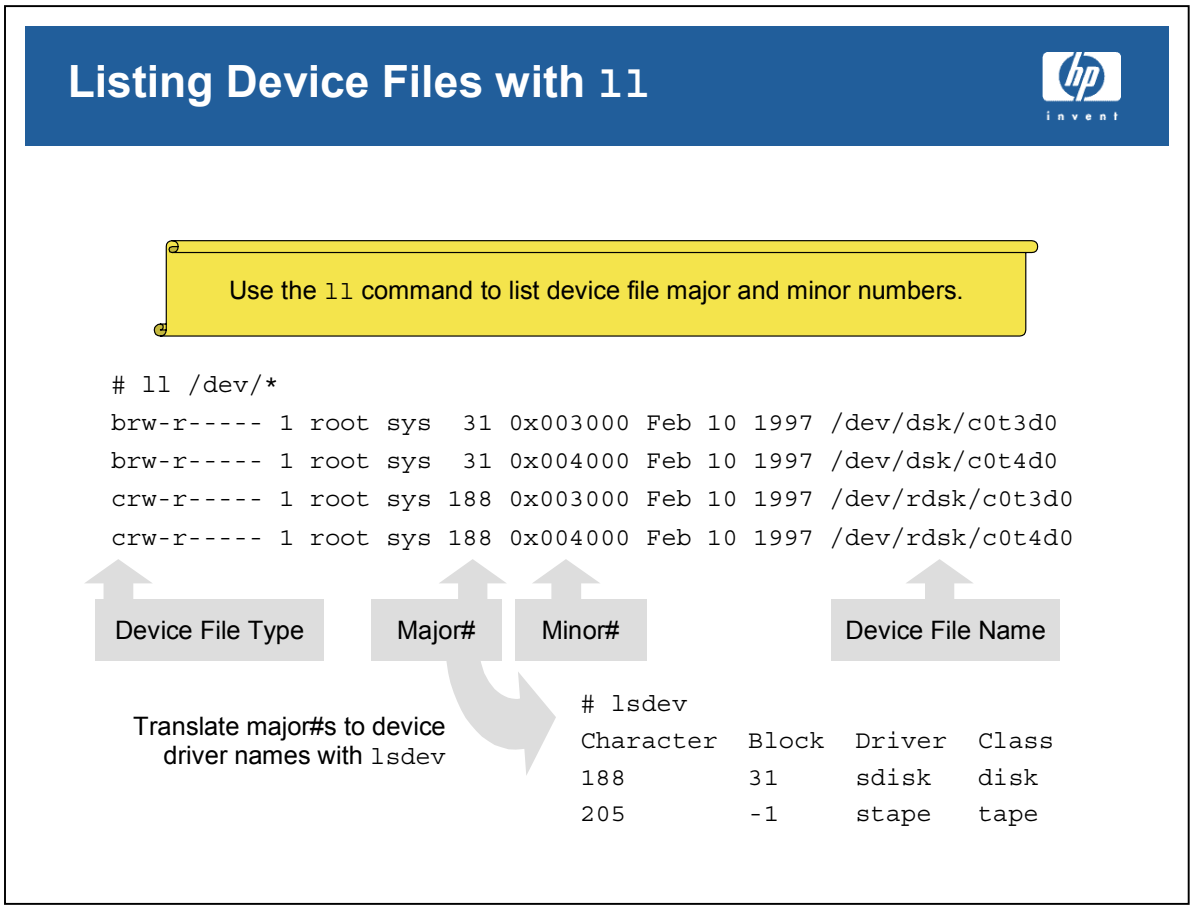
The **tar** application opens the file specified by the **-f** option for output. Since device files allow applications to access devices using the same system calls that are used to access files, the **-f** option may be used to write a tar archive to either a tape drive (eg: **/dev/rmt/0m**) or a file (eg: **/tmp/archive.tar**).

In this second example, the standard output of the **echo** command is redirected to a terminal via the terminal's device file.

```
# echo hello > /dev/tty0p1
```


© 2004 Hewlett-Packard Development Company, L.P.

6–2. SLIDE: Listing Device Files with `ll`



Student Notes

Device files typically reside in the **`/dev`** directory. You can list the device files in this directory with the **`ll`** command, just as you would list files in any other directory. Note, however, that the **`ll`** output for device files is a bit different.

Device File Types

The very first character in the **`ll`** output for a device file indicates the device file type.

Character Device Files — A "c" in the first character position identifies a **character** device file. Character device files transfer data to the device one character at a time. Devices such as terminals, printers, plotters, modems, and tape drives are typically accessed via character device files. Character device files are sometimes called "raw" device files.

© 2004 Hewlett-Packard Development Company, L.P.

Block Device Files	A "b" in the first character position identifies a **block** device file. When accessing a device via a block device file, the system reads and writes data through a **buffer** in memory, rather than transferring the data directly to the physical disk. This can significantly improve I/O for disks and CD-ROMs. Block device files are sometimes called "cooked" device files.

Terminals, modems, printers, plotters, and tape drives typically only have character device files. Disks and CD-ROMs may be accessed in either character or block mode, and thus typically have both types of device files. Some applications and utilities prefer to access disks directly via character device files. Other utilities require a block device file. Read the application or utility documentation to determine which device file is required.

Device File Major Numbers

Every device file has a "major number" that appears in the fifth field of the `ll` output. The major number identifies the "kernel driver" that should be used when the device is accessed. A kernel driver is a portion of code in the HP-UX kernel that controls I/O for a particular type of device. Most HP-UX machines have multiple drivers; the major number on every device file determines which driver should be used. The `lsdev` command lists the drivers configured in your kernel, and their associated major numbers.

Device File Minor Numbers

Every device file has an associated minor number. The minor number is a 24-bit hexadecimal number that identifies:

- The physical location of the device on the system.
- Device-specific access options. Tape drives, for instance, have special access options that enable/disable hardware compression and define the density format used when writing to the tape.

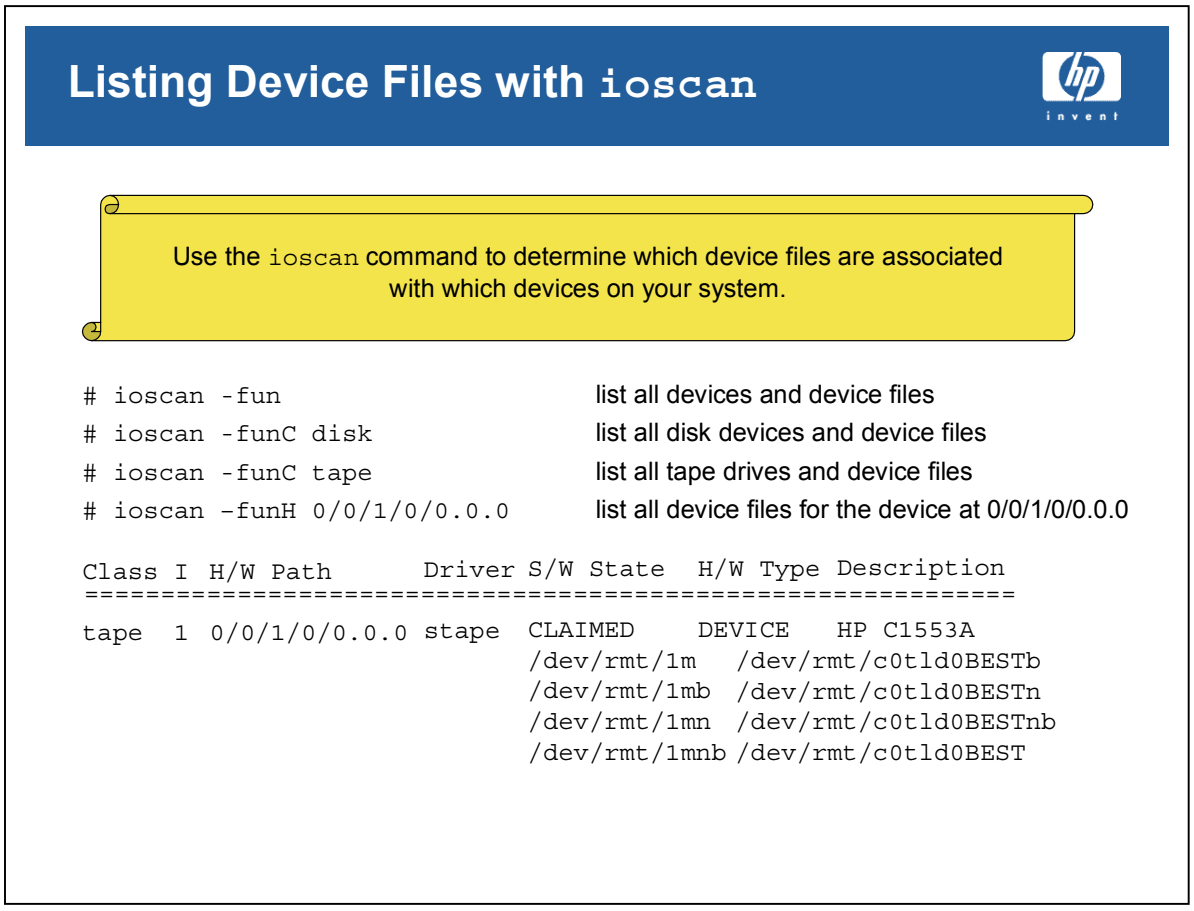
Minor numbers are formulated differently for various types of devices. See the Configuring HP-UX for Peripherals manual for more information about formulating and interpreting major and minor numbers.

Device File Names

Device file names follow a standard naming convention that makes it fairly easy to determine which device files are associated with which devices. Later slides in this chapter discuss the naming convention in detail.

© 2004 Hewlett-Packard Development Company, L.P.

6–3. SLIDE: Listing Device Files with `ioscan`



Student Notes

Although the **`ll`** command lists the device files on your system, it does not indicate which device each device file accesses.

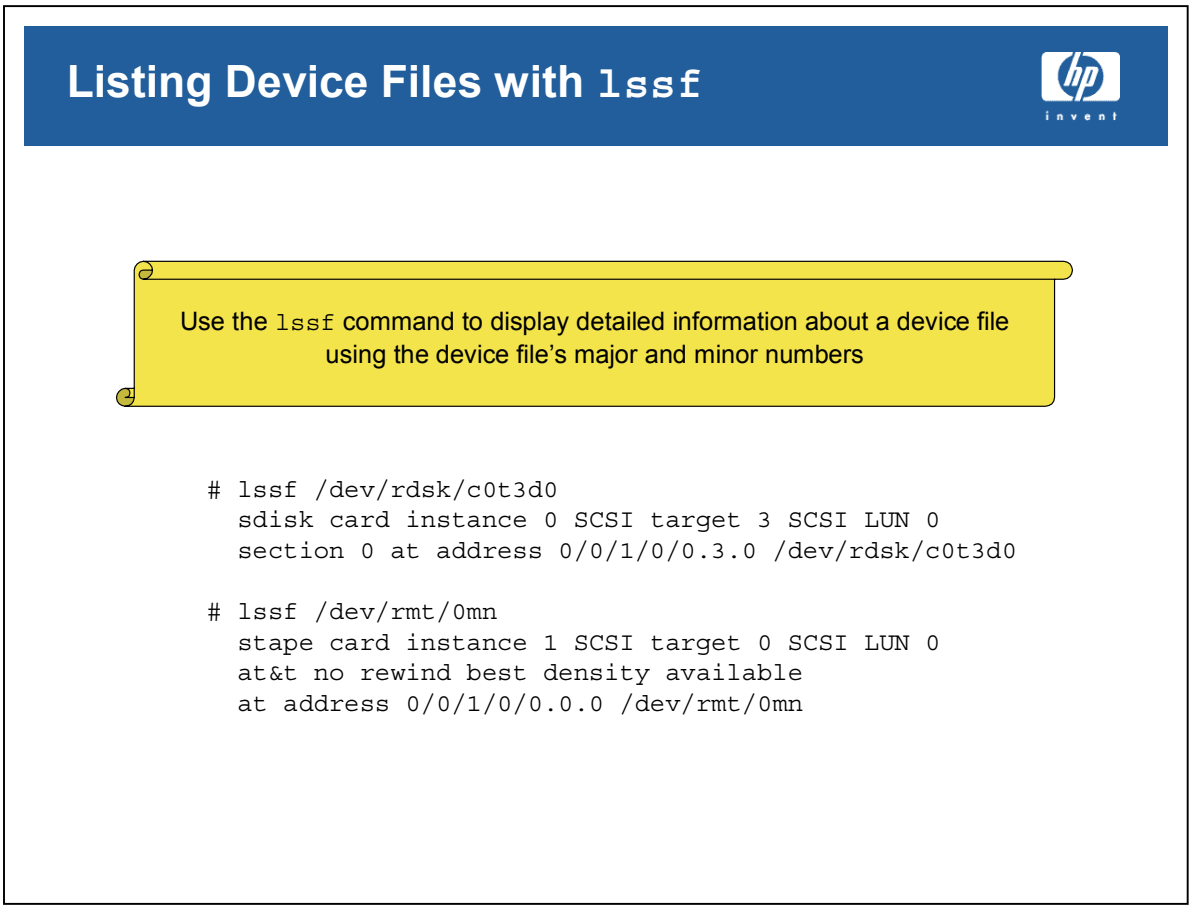
The **`ioscan -fun`** command provides a convenient mechanism for determining which device files are associated with each hardware path on your system. Under each hardware path, **`ioscan -fun`** lists all the device files associated with each hardware path. Since some devices have multiple access options, **`ioscan`** can list multiple device files for a single device.

Examples

```
# ioscan –fun                    # list all devices, and their device files
# ioscan -funC disk              # only list disk class device files
# ioscan -funC tape              # only list tape drives and their device files
# ioscan -funH 0/0/1/0/0.0.0     # only list device files for 0/0/1/0/0.0.0
```


© 2004 Hewlett-Packard Development Company, L.P.

6–4. SLIDE: Listing Device Files with `lssf`



Student Notes

Many devices have multiple associated device files. For instance, the tape drive shown on the previous slide had eight device files. This is because many devices may be accessed with a variety of combinations of access options. Each combination of access options requires a separate device file.

`ioscan` lists the device files for each device, but does not indicate which device-specific options each device file enables. The **`lssf`** command is the utility of choice for determining exactly which features each device file provides. **`lssf`** tells you:

- Which driver the device file uses.
- The device's hardware address information.
- Any device-specific access options used by the device file.

© 2004 Hewlett-Packard Development Company, L.P.

Questions

We have seen three commands for viewing device files: **`ll`**, **`lssf`**, and **`ioscan`**. Each command has a somewhat different purpose. Determine which command would be appropriate for each of the following situations:

1. Which command would you use to list the device files for the tape drive at hardware address 0/0/2/0.0.0?

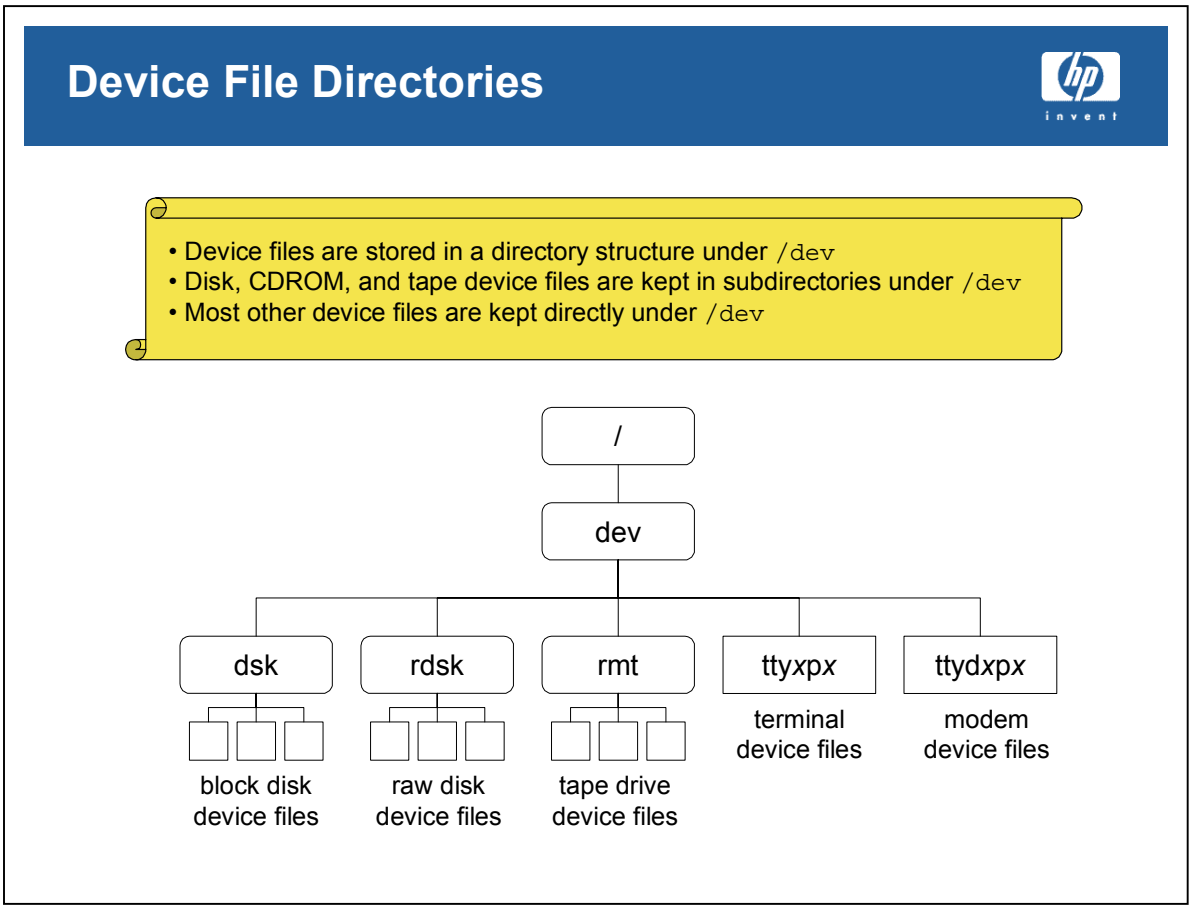
2. Which command(s) could you use to list all of your disk device files?

3. Which command would you use to determine the hardware path of the device accessed by **`/dev/rmt/c0t0d0BEST`**?

4. Which command will tell you the device-specific options used by the **`/dev/rmt/c0t0d0BESTnb`** device file?

© 2004 Hewlett-Packard Development Company, L.P.

6–5. SLIDE: Device File Directories



Student Notes

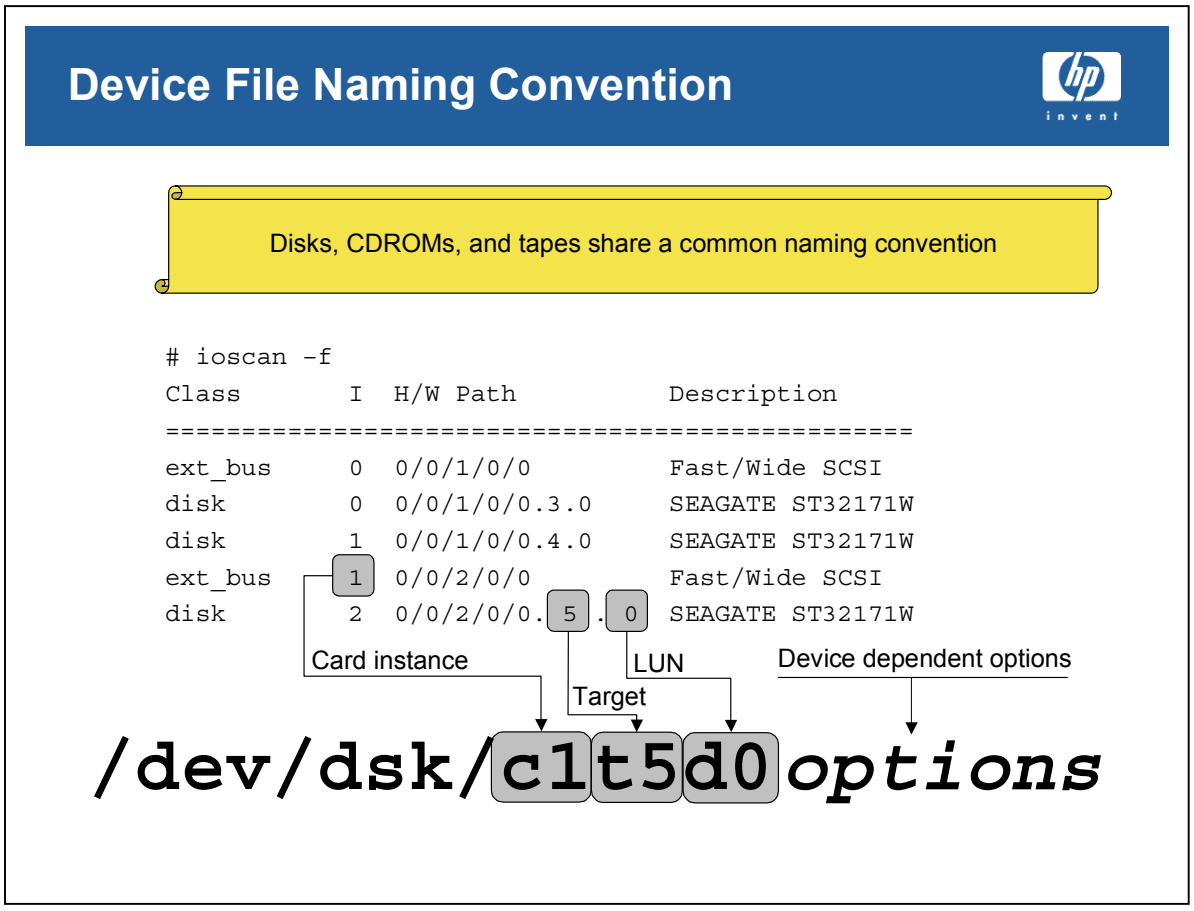
The next few slides introduce the naming convention used by the system when assigning names to device files. An understanding of the device file naming convention will allow you to more easily choose and use device files on your system.

On most HP-UX systems, device files live under the **/dev** directory. Some device files sit directly in **/dev**, while others are grouped in subdirectories under **/dev**. The list below highlights some of the more important device file directories.

/dev	Contains all terminal, modem, LAN, and printer device files.
/dev/dsk	Contains all block disk device files.
/dev/rdsk	Contains all raw disk device files.
/dev/rmt	Contains all tape device.

© 2004 Hewlett-Packard Development Company, L.P.

6–6. SLIDE: Device File Naming Convention



Student Notes

By default, HP-UX names your system device files following a standard naming convention. Although you can arbitrarily assign any device file names you wish, using this standard convention makes it easier to locate your device files. Most tape, disk, and CDROM device file names adhere to the convention shown on the slide. The notes below explain each component of these device file names.

The Interface Card Instance Number

The kernel automatically assigns an **instance number** to every device and interface card on an HP-UX system. Instance numbers are shown in the "I" column of the **`ioscan -f`** output.

The "c" in a disk, tape, or CD-ROM device file name identifies the instance number of the interface card to which the device is attached. The last disk shown in the **`ioscan`** on the slide would have device files beginning with "c1", since the instance number of the SCSI interface card is "1". Note that each device has an instance number as well. Instance numbers for individual devices are used internally by the OS, but are not used to formulate device file names.

© 2004 Hewlett-Packard Development Company, L.P.

The Target Address

The "t#" portion of the device file name identifies the SCSI target address of the device file's associated device. The SCSI target address is set via jumper pins or DIP switches on the device itself. Look at the second to last digit in a SCSI device's hardware path to determine the SCSI target address. For instance, in the **ioscan** shown above, the disk at 0/0/2/0/0.5.0 has SCSI target address "5".

The Logical Unit Number

The **Logical Unit Number** (LUN) can be used to identify the robotic mechanism in a tape changer, or a logical unit in a disk array. For simple SCSI devices, the LUN number will simply be "0". The LUN number for each SCSI device appears in the last digit of the device's hardware path. Note that all of the disks in the sample **ioscan** above have a "0" in the LUN digit.

Device-Dependent Access Options

The last part of the device file name lists device-specific access options enabled by the device file. Tape drive device file names may have a variety of options listed in this portion of the device file name. Access options vary from device to device.

Consider the following **ioscan** output:

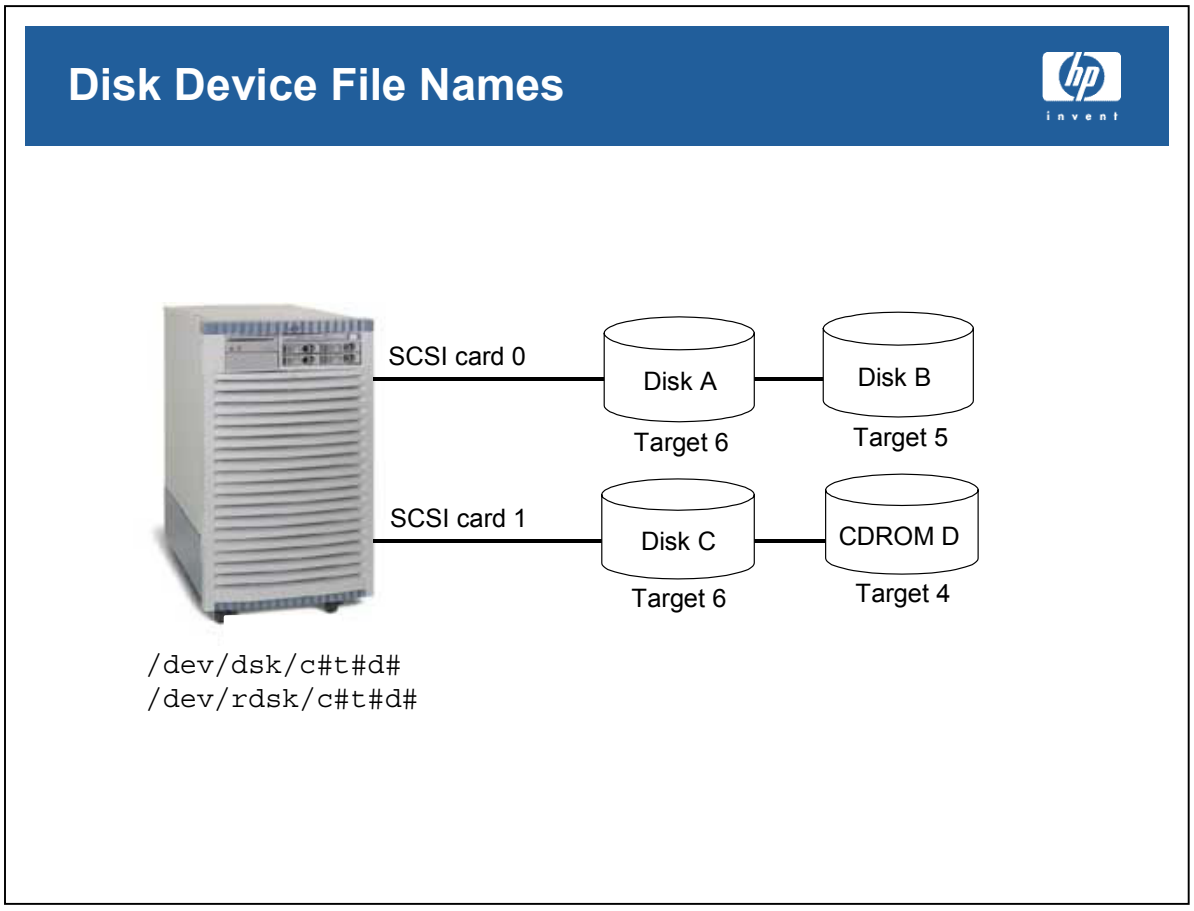
```
 # ioscan -fu
Class       I  H/W Path    Driver      S/W State H/W Type  Description
ext_bus     0  8/12        c720        CLAIMED   INTERFACE GSC Fast/Wide
disk        0  8/12.5.0    sdisk       CLAIMED   DEVICE    SEAGATE
disk        1  8/12.6.0    sdisk       CLAIMED   DEVICE    SEAGATE
ext_bus     1  8/16/5      c720        CLAIMED   INTERFACE Built-in SCSI
disk        2  8/16/5.3.0  sdisk       CLAIMED   DEVICE    SEAGATE
```

1. What is the interface card instance number of the disk at 8/12.5.0?
2. What is the target address of the disk at 8/12.5.0?
3. What is the interface card instance number of the disk at 8/16/5.3.0?
4. What is the target address of the disk at 8/16/5.3.0?

Answers are provided in the solutions appendix.

© 2004 Hewlett-Packard Development Company, L.P.

6–7. SLIDE: Disk Device File Names



Student Notes

Every disk and CD-ROM has two device files:

- A block device file in **`/dev/dsk`**
- A character device file in **`/dev/rdsk`**

The device files within these directories follow the standard naming convention described on the previous slide.

Examples

The device files for disks "A" and "B" on the slide above would be:

`/dev/dsk/c0t6d0`	Block device file for disk "A"
`/dev/dsk/c0t5d0`	Block device file for disk "B"
`/dev/rdsk/c0t6d0`	Raw device file for disk "A"
`/dev/rdsk/c0t5d0`	Raw device file for disk "B"

© 2004 Hewlett-Packard Development Company, L.P.

Questions

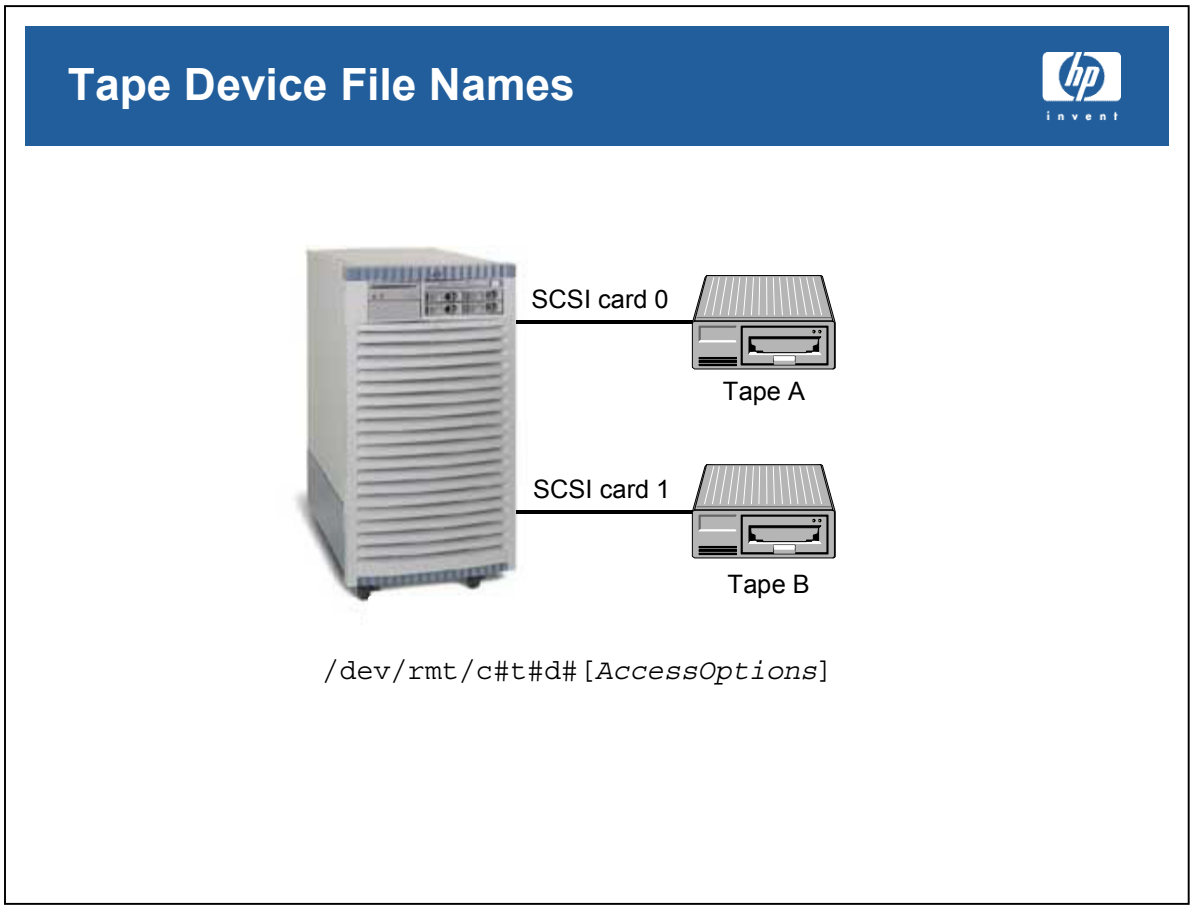
1. What would be the device file name for the disk labeled "C" on the slide?

2. What would be the device file name for the CD-ROM labeled "D" on the slide?

Answers are provided in the solutions appendix.

© 2004 Hewlett-Packard Development Company, L.P.

6–8. SLIDE: Tape Device File Names



Student Notes

Tape device files are stored in the **`/dev/rmt`** directory, and follow the **`c#t#d#`** [options] naming convention. Unlike disks and CD-ROMSs, however, tape drives often support numerous access options in the [options] portion of the device file name.

These options include:

density Density or format used in writing data to tape. This field is designated by the following values:

BEST	Highest-capacity density or format will be used, including data compression, if the device supports compression
NOMOD	Maintains the density used for data previously written to the tape. Behavior using this option is dependent on the type of device. This option is only supported on DDS and 8MM drives.
DDS *n*	Selects one of the known DDS formats; can be used to specify DDS1 or DDS2, as required.
D [#]	Specifies density as a numeric value to be placed in the SCSI mode select block descriptor.

© 2004 Hewlett-Packard Development Company, L.P.

C[#] Write data in compressed mode, on tape drives that support data compression. If a number is included, use it to specify a compression algorithm specific to the device. Note, compression is also provided when the density field is set to BEST.

n No rewind on close. Unless this mode is requested, the tape is automatically rewound upon close.

b Specifies Berkeley-style tape behavior. When the b is absent, the tape drive follows AT&T-style behavior.

w Writes wait for physical completion of the operation before returning status. The default behavior (buffered mode or immediate reporting mode) requires the tape device to buffer the data and return immediately with successful status. For each tape device present, eight device files are automatically created when the system is initialized. Four of these device files utilize the standard naming conventions. These four files contain the density specification "BEST". There are four such files because each of the four different permutations of the "n" and "b" options.

Examples

The following are some device files for tape A:

`/dev/rmt/c0t0d0BEST`	Use the best density and compression options available
`/dev/rmt/c0t0d0BESTn`	Same as above, but use the "no-rewind" feature
`/dev/rmt/c0t0d0DDS1`	Use the DDS1 density for compatibility with older drives

9.x Compatibility

Prior to version 10.x of HP-UX, device files followed an entirely different naming convention:

`/dev/rmt/0m`	First tape drive on the system
`/dev/rmt/1m`	Second tape drive on the system
`/dev/rmt/2m`	Third tape drive on the system
`/dev/rmt/2mn`	Third tape drive on the system, "no-rewind" feature enabled

Each device file includes an arbitrary number to distinguish the tape drive from all other tape drives, followed by the letter "m", followed by a series of access options as described earlier in the notes for this slide. For the sake of backwards compatibility, these old style device files still exist. They are simply links to the **`/dev/rmt/c0t0d0BEST[options]`** device files.

Questions

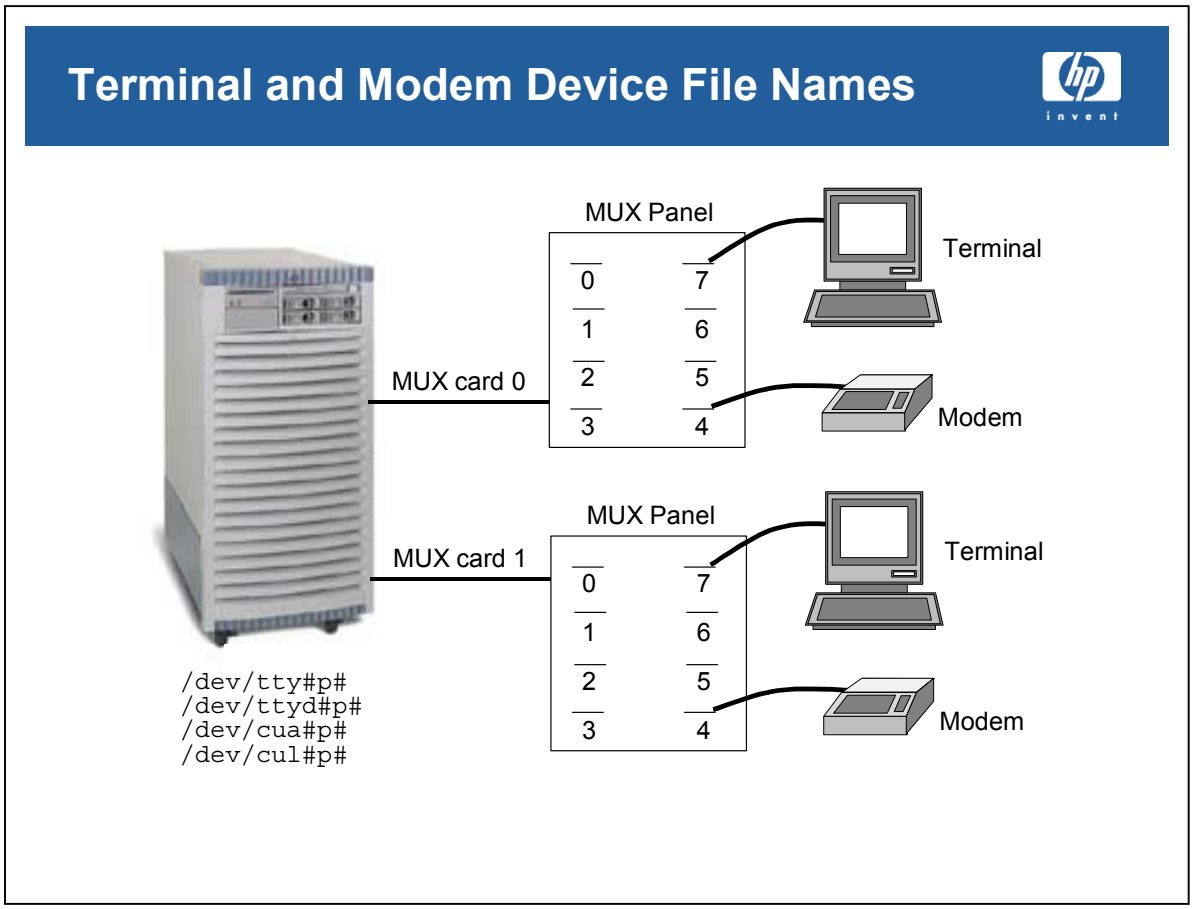
1. What device file would you use to access tape "B" in the diagram on the slide using the best available density and compression features?

2. What device file would you use to access tape "B" in the diagram on the slide using the best available density and compression features, and with the auto-rewind feature disabled?

Answers are provided in the solutions appendix.

© 2004 Hewlett-Packard Development Company, L.P.

6–9. SLIDE: Terminal and Modem Device File Names



Student Notes

The naming convention for terminals and modems is slightly different from the naming convention for disks, CD-ROMs, and tape drives.

- Terminals and modems don't have a dedicated subdirectory; their device files are kept directly in the **`/dev`** directory.
- Terminals and modems don't follow the **`c#t#d#`** naming convention

Terminal Device Files

Terminal device file names identify both the interface card and MUX port number to which the terminal is attached. If the terminal is attached to a serial port rather than a MUX panel, use port#0 in the device file name.

Examples

`/dev/tty0p7`	Device file for the terminal on the first MUX, port #7
`/dev/tty1p7`	Device file for the terminal on the second MUX, port #7

© 2004 Hewlett-Packard Development Company, L.P.

Modem Device Files

A fully functional modem requires three device files.

Examples

Device files for the modem on the first MUX, port #4

```
/dev/cua0p4
/dev/cul0p4
/dev/ttyd0p4
```

Device files for the modem on the second MUX, port #4

```
/dev/cua1p4
/dev/cul1p4
/dev/ttyd1p4
```

Pseudo Terminals

Pseudo terminals are used by applications that provide terminal emulation capabilities, such as **hpterm, xterm, telnet,** etc. The pseudo terminal driver provides support for a device-pair termed a pseudo terminal. A pseudo terminal is a pair of character devices, a master device and a slave device.

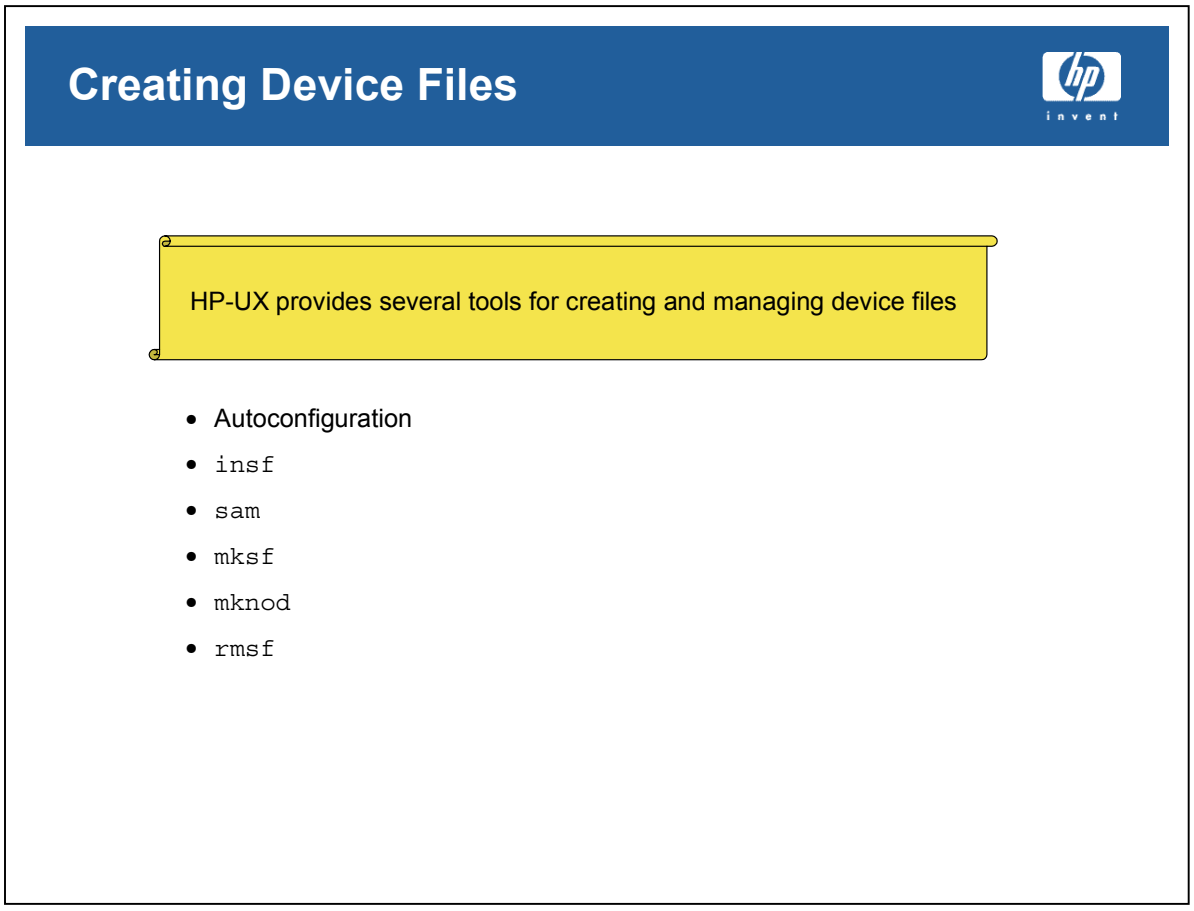
The device files for pseudo terminals are found in the following places:

slave	**/dev/tty *xx***. These are links to files in the **/dev/pty** directory **/dev/pty/tty *xx***
master	**/dev/pty *xx***. These are links to files in the **/dev/ptym** directory **/dev/ptym/pty *xx***
streams- based pseudo slave	**/dev/pts/*n***. This is used by the **dtterm** terminal emulator
streams- based master	**/dev/ptymx**. This is used by the **dtterm** terminal emulator

By default, HP-UX creates 60 pseudo-terminals of each type. If your server is likely to service more than 60 concurrent telnet sessions, or more than 60 concurrent terminal emulator windows, then you may need to increase the number of pseudo terminals. To increase the number of pseudo terminals, increase the value of the **npty** and **nstrpty** kernel parameters, and reboot. During the boot process, HP-UX executes the **insf** command, which then creates the additional pseudo terminal device files. Kernel configuration will be covered in a later chapter.

© 2004 Hewlett-Packard Development Company, L.P.

6–10. SLIDE: Creating Device Files



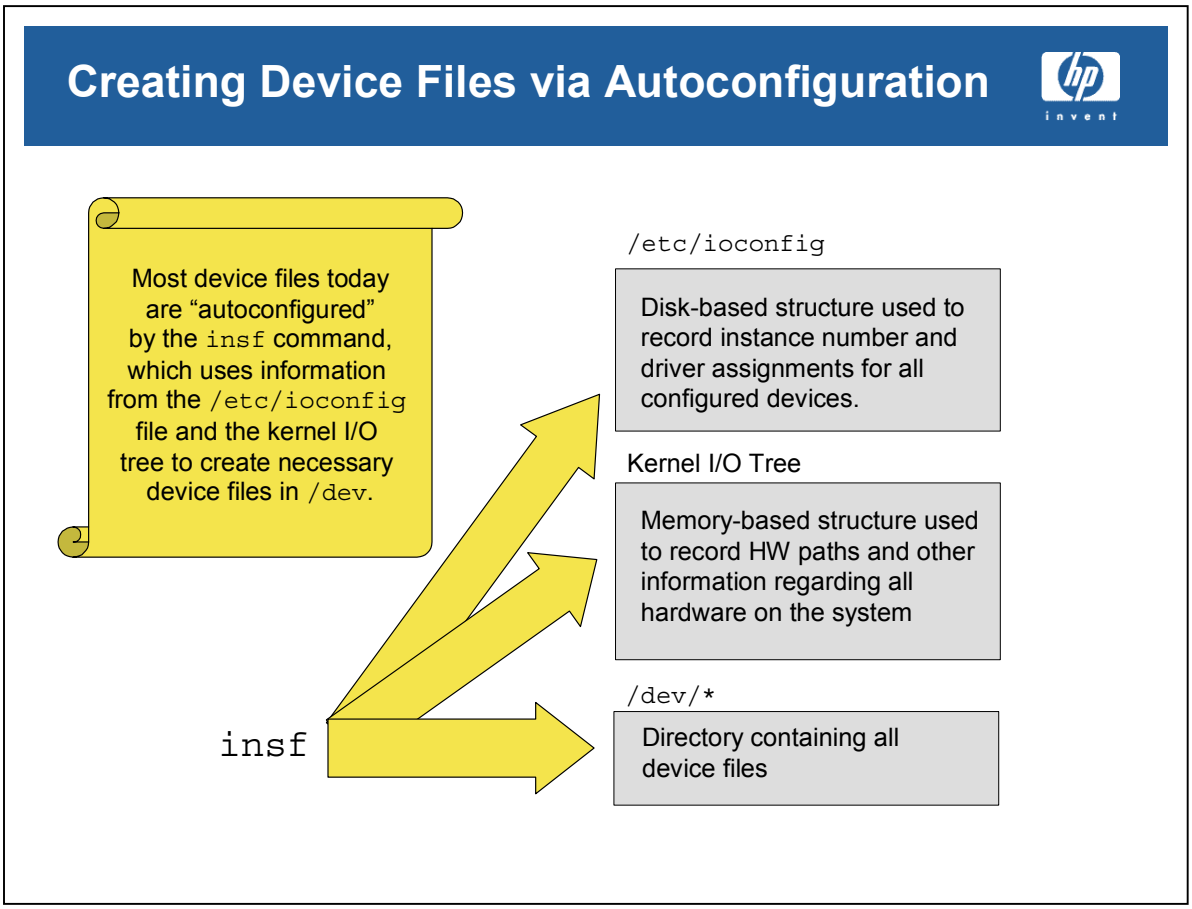
Student Notes

HP-UX provides several tools for creating and managing device files. Each of these will be described in detail in the pages that follow.

- Autoconfiguration
- `insf`
- `sam`
- `mksf`
- `mknod`
- `rmsf`

© 2004 Hewlett-Packard Development Company, L.P.

6–11. SLIDE: Creating Device Files with Autoconfiguration



Student Notes

These days, most HP-UX device files are created automatically via the HP-UX auto-configuration process.

What Is Autoconfiguration?

During the system boot process, the kernel automatically probes all hardware installed on the system and identifies all bus adapters, device adapters, and external devices that can be auto-configured. The kernel "binds" (matches) an appropriate driver to each auto-configurable device detected at each hardware address.

Later in the boot process, the kernel invokes the **`init`** command. The **`init`** process reads the **`/etc/inittab`** file and invokes several system startup commands listed in the file, including **`/sbin/ioinitrc`**. The **`/sbin/ioinitrc`** command usually starts **`ioinit`**, which does several things.

First, **`ioinit`** reads the contents of the **`/etc/ioconfig`** file and transfers the device mapping information found there to the **`io_tree`** kernel data structures. Next, **`ioinit`** executes **`insf`**.

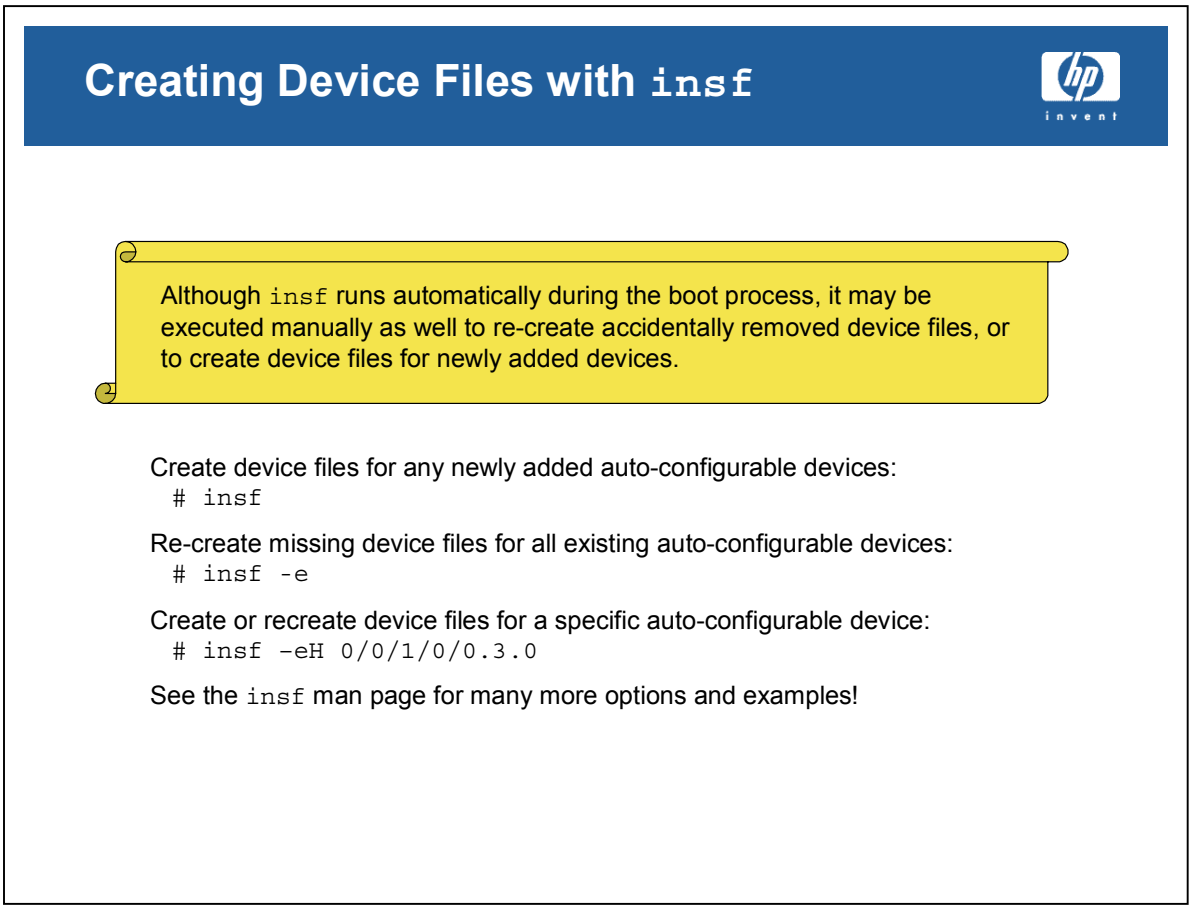
© 2004 Hewlett-Packard Development Company, L.P.

`insf` (install special files) creates device files for new devices and updates **`/etc/ioconfig`** and the kernel **`io_tree`**.

All Hewlett-Packard peripheral devices supported by HP-UX Release 10.20 and beyond are automatically configurable. Device files are automatically created during the reboot process for new devices and I/O cards.

© 2004 Hewlett-Packard Development Company, L.P.

6–12. SLIDE: Creating Device Files with `insf`



Student Notes

`insf` usually runs automatically during the boot process, but you may choose to run the command manually when:

- device files are accidentally removed; or
- you add devices to your system without rebooting

If you don't specify any options, **`insf`** will install device files for all auto-configurable devices that haven't yet been configured.

If you want to create device files for an "existing" device that is already recorded in **`/etc/ioconfig`**, include the **`-e`** option. Use this option if device files have been accidentally deleted.

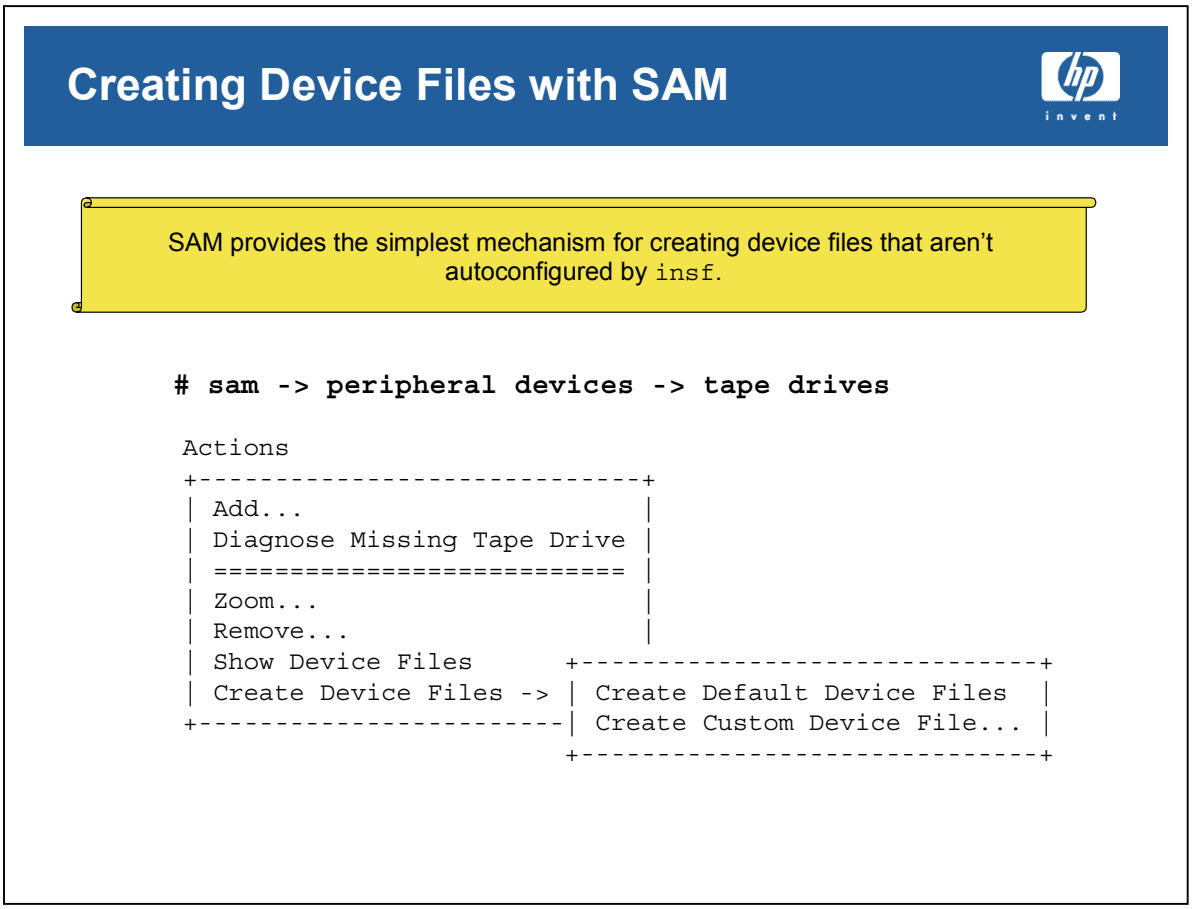
© 2004 Hewlett-Packard Development Company, L.P.

If you don't want to create device files for all devices, the **-d**, **-H**, and **-C** options are used to select devices with a specified driver, device class, or hardware path address. Use the **lsdev** command to determine drivers and classes in the kernel (**/stand/vmunix**). Use the **ioscan** command to list the hardware paths in the kernel I/O tree.

-d *driver*	selects particular devices by driver name.
-C *class*	matches devices that belong to a given class, like disks.
-H *path*	matches a device at a given Hardware Path.
-I *instance#*	selects devices by card instance number

© 2004 Hewlett-Packard Development Company, L.P.

6–13. SLIDE: Creating Device Files with SAM



Student Notes

insf only configures standard device files for auto-configurable devices. For example, **insf** automatically creates the following device files for auto-configurable DDS tape drives:

```
crw-rw-rw-   2 bin  bin  205 0x030000 Aug 30 16:45 0m
crw-rw-rw-   2 bin  bin  205 0x030080 Aug 30 16:45 0mb
crw-rw-rw-   2 bin  bin  205 0x030040 Aug 30 16:45 0mn
crw-rw-rw-   2 bin  bin  205 0x0300c0 Aug 30 16:45 0mnb
crw-rw-rw-   2 bin  bin  205 0x030000 Aug 30 16:45 cxtxdxBEST
crw-rw-rw-   2 bin  bin  205 0x030080 Aug 30 16:45 cxtxdxBESTb
crw-rw-rw-   2 bin  bin  205 0x030040 Aug 30 16:45 cxtxdxBESTn
crw-rw-rw-   2 bin  bin  205 0x0300c0 Aug 30 16:45 cxtxdxBESTnb
crw-rw-rw-   1 bin  bin  205 0x030001 Aug 30 16:45 cxtxdxDDS
crw-rw-rw-   1 bin  bin  205 0x030081 Aug 30 16:45 cxtxdxDDSb
crw-rw-rw-   1 bin  bin  205 0x030041 Aug 30 16:45 cxtxdxDDSn
crw-rw-rw-   1 bin  bin  205 0x0300c1 Aug 30 16:45 cxtxdxDDSnb
```

If you wanted a device file for an unusual combination of special options, that device file would have to be manually configured via **sam** or **mksf**.

© 2004 Hewlett-Packard Development Company, L.P.

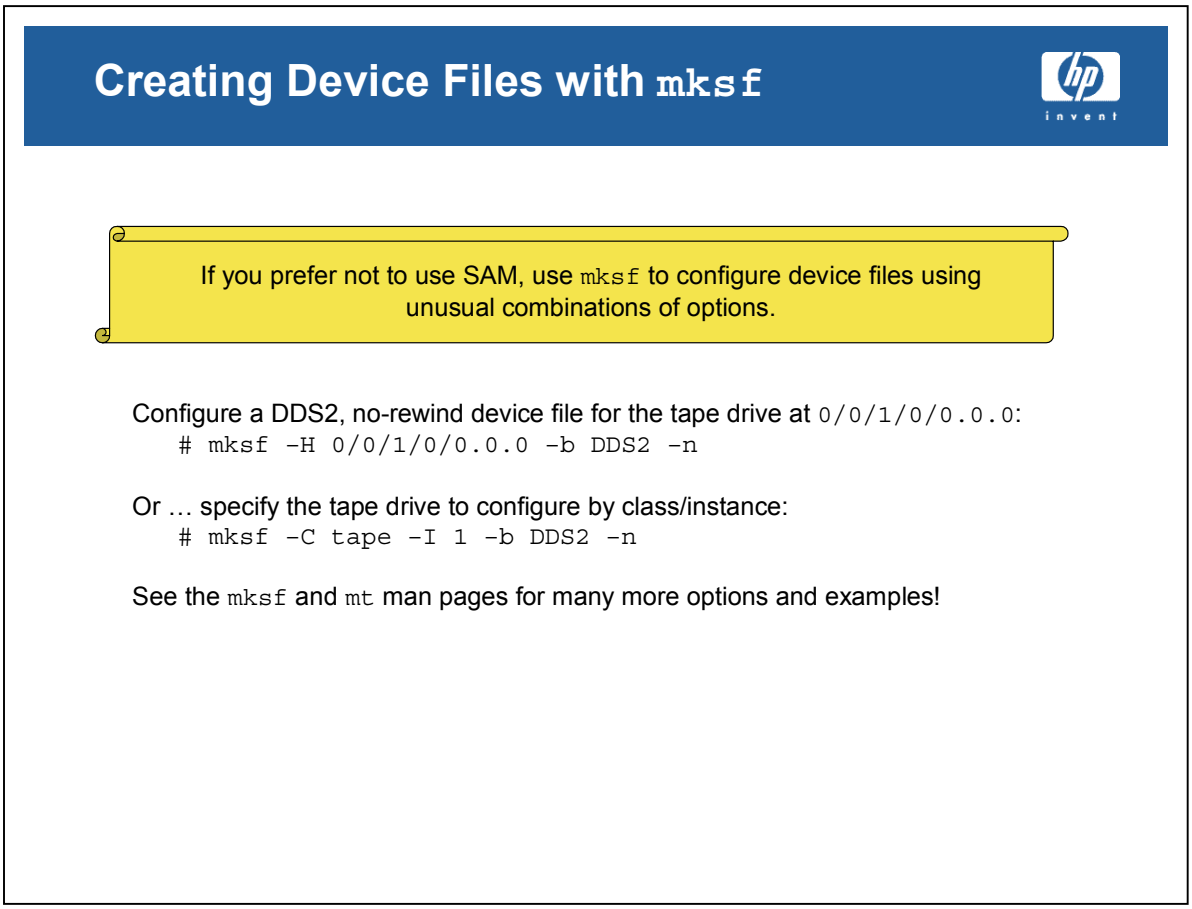
For instance, while autoconfiguration automatically creates a “BEST” device file for tape drives, it doesn’t automatically create a DDS2 device file, which may be useful if you need to ensure that a tape will be readable on an older system with an older tape drive.

The SAM peripheral devices functional area makes it possible to create custom device files for tape drives, terminals, and modems.

- Navigate to the SAM **`Peripheral Devices`** functional area
- Select the functional area for the peripheral device type you wish to configure
- Select the desired device’s hardware path from the resulting device list
- Select **`Actions -> Configure`**, or **`Actions -> Create Device File`**
- Specify the type of device file you wish to create in the resulting dialog box

© 2004 Hewlett-Packard Development Company, L.P.

6–14. SLIDE: Creating Device Files with `mksf`



Student Notes

If you prefer to work from the command line, use **`mksf`** rather than **`sam`** to create non-standard device files for autoconfigurable devices. The example below configures a DDS2 device file with the auto-rewind feature disabled for a device at hardware address 0/0/1/0/0.0.0:

```
# mksf -H 0/0/1/0/0.0.0 -b DDS2 -n
```

The device you wish to configure may be specified several different ways:

`-d` ***`driver`***	selects a particular device by its driver name.
`-I` ***`instance#`***	selects *device instance*.
`-C` ***`class`***	matches devices that belong to a given class, like *disks*.
`-H` ***`path`***	matches a device at a given hardware path.
`-D` ***`directory`***	overrides the default device installation directory and install special file in *directory*. Note that *directory* must exist.

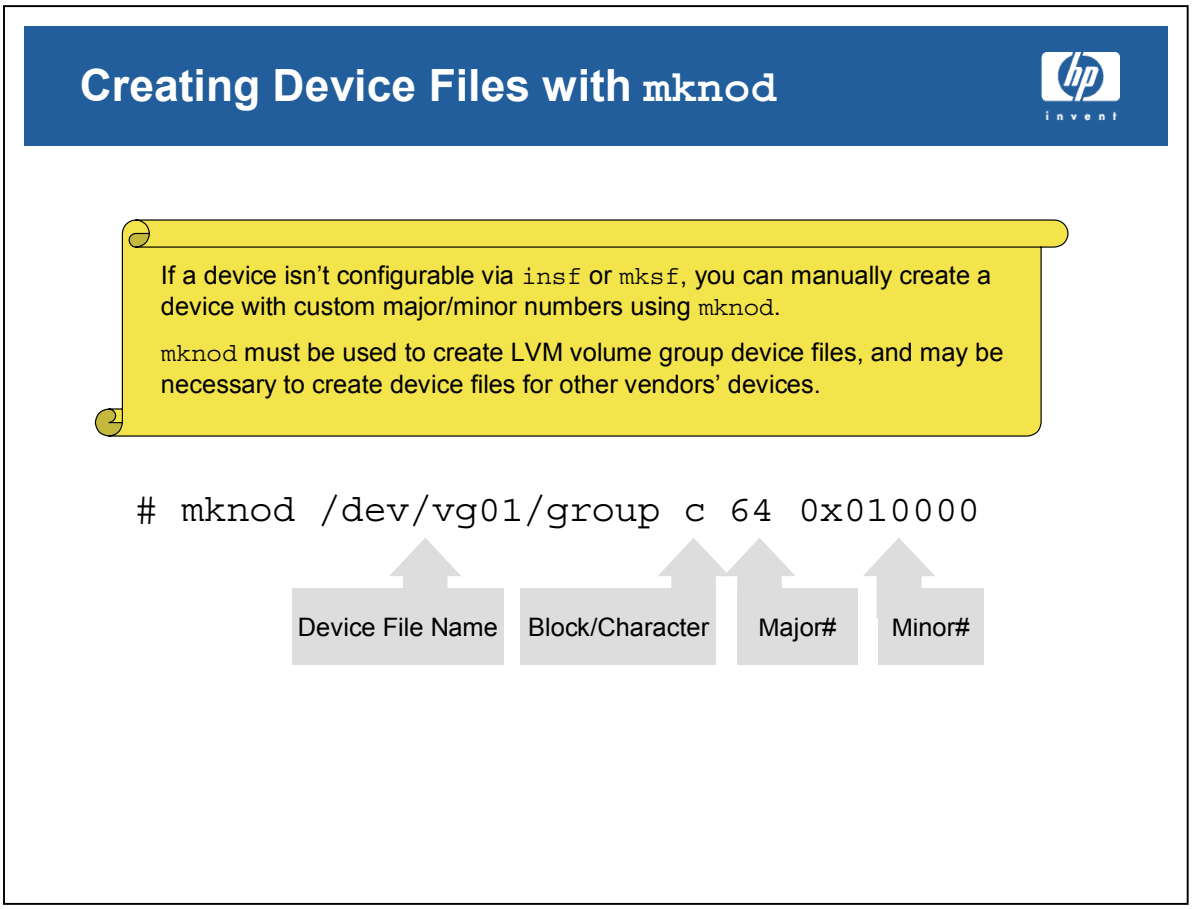
© 2004 Hewlett-Packard Development Company, L.P.

mksf options for each driver vary widely. Options that are meaningful to one device driver are meaningless to another. The **mksf(1m)** and **mt(7)** man pages describe dozens of options that may be used to configure device files with various combinations of options.

Fortunately, the **mksf** command is rarely needed since most commonly used device files are auto-configured by **insf**.

© 2004 Hewlett-Packard Development Company, L.P.

6–15. SLIDE: Creating Device Files with `mknod`



Student Notes

The vast majority of devices on HP-UX systems are auto-configurable. Devices that aren't auto-configurable must be configured via **`mknod`**.

The **`mknod`** command requires several options and arguments:

- The full pathname for the new device file. Device files are typically stored in **`/dev`**.
- **`b`** or **`c`**, to indicate whether you wish to create a block or character device file.
- The major number of the kernel driver used to access the device. Use the **`lsdev`** command to view a list of drivers and their associated major numbers.
- The minor number. Consult your device's documentation to determine an appropriate minor number.

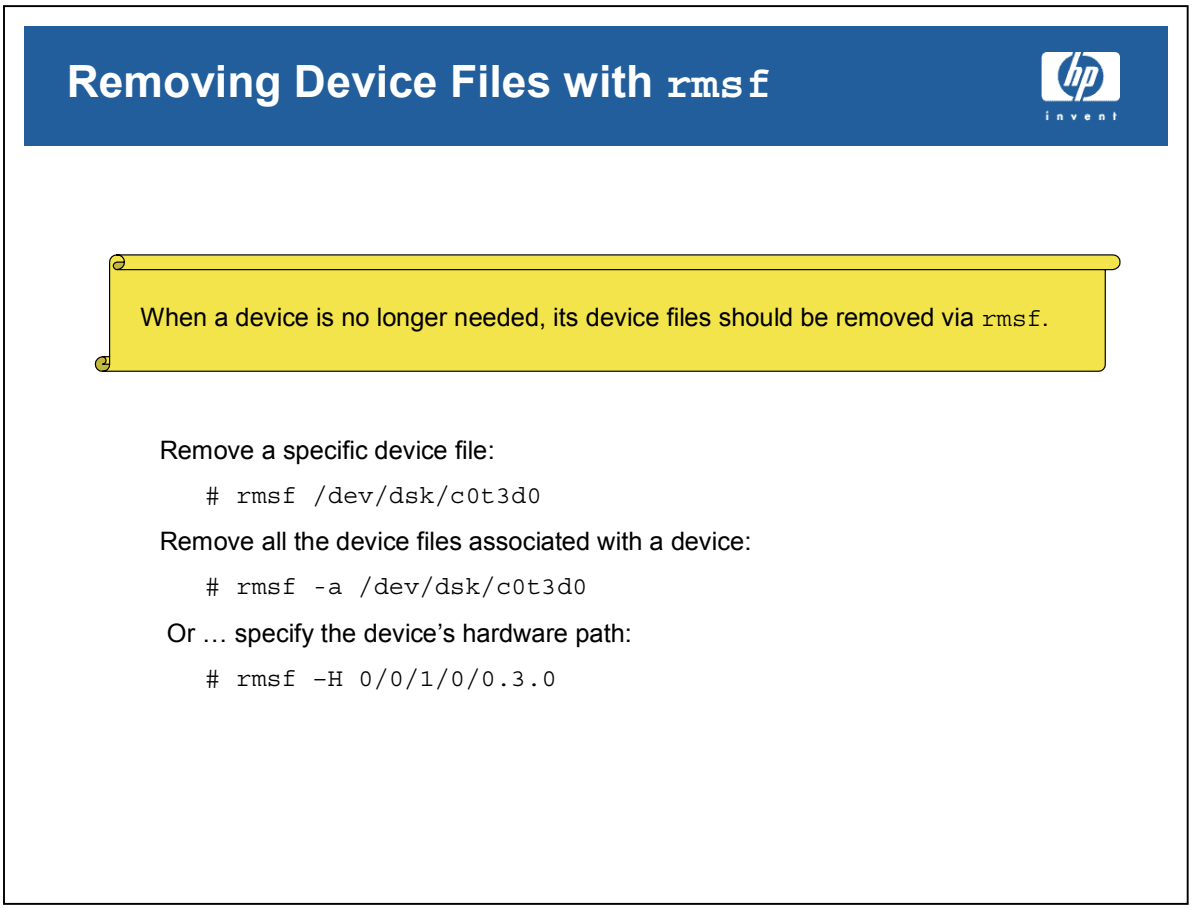
© 2004 Hewlett-Packard Development Company, L.P.

HP's Logical Volume Manager (LVM) volume group device files are perhaps the most common device files that must be manually configured using **mknod**. LVM concepts and commands will be covered more formally in a later chapter; for now, we will simply concentrate on the **mknod** options required to create an LVM volume group device file. The example on the slide shows the full syntax for the **mknod** command required to create a volume group device file.

After creating a device file with **mknod**, you may need to execute the **chmod** command to set appropriate permissions. A volume group device file should be owned by root, with 640 permissions. Other device files' permissions will vary.

© 2004 Hewlett-Packard Development Company, L.P.

6–16. SLIDE: Removing Device Files with `rmsf`



Student Notes

If you remove a device from your system, the device files for the device must be manually removed using the **`rmsf`** command.

`rmsf` may be used several different ways. You can remove a single device file:

```
# rmsf /dev/dsk/c0t3d0
```

If you want to remove all of the device files associated with a specific device, use the **`-a`** option:

```
# rmsf -a /dev/dsk/c0t3d0
```

Alternatively, you can specify the hardware path of the device you wish to remove:

```
# rmsf -H 0/0/1/0/0.3.0
```

`rmsf` removes the device's device files, as well as its definition in **`/etc/ioconfig`**.

© 2004 Hewlett-Packard Development Company, L.P.

6–17. LAB: Device Files

Directions

Part 1: Viewing and Interpreting Device Files

1. Use **`ioscan`** to find the names of the device files for all of your disk class devices.

2. Use **`ioscan`** to find the names of the device files for your system's LAN card(s).

3. Choose one of the disk device files in your **`/dev/dsk`** directory. Execute the command that lists the characteristics of this device file. What kernel driver is associated with this device file? What is the hardware path of the disk associated with this device file?

4. How does **`lssf`** know which kernel driver is associated with each device file?

5. (Optional) If you have a tape drive on your system, run **`lssf /dev/rmt/*`**. Which device file(s) access your tape drive with the no-rewind feature?

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Creating Device Files with SAM

You should have at least two available serial ports on your machine. In this portion of the lab, you will create device files to support a modem on one of your serial ports, and a hard-wired ASCII terminal on the other. SAM provides a simple interface for creating device files for both of these devices.

1. Go to **`SAM --> Peripheral Devices --> Terminals and Modems`**. Are there any modems or terminals currently?

2. First, configure a modem on your first available serial port.

   a) Choose **`Actions --> Add Modem`**.
   b) Choose a serial port/MUX interface hardware path.
   c) Click Port Number to choose an available MUX port (Choose 0 on workstations).
   d) Choose any baud rate. (It doesn't really matter since we don't have a real modem.)
   e) Choose to support both incoming and outgoing calls.
   f) Click OK.

   Which device files does SAM create for you?

3. Now configure a hard-wired terminal on another available serial port.

   a) Choose **`Actions --> Add Terminal`**.
   b) Choose a serial port/MUX hardware path.
   c) Click Port Number to choose an available MUX port.
   d) Choose "H" as the baud rate.
   e) Click OK.

   Which device files does SAM create for you?

© 2004 Hewlett-Packard Development Company, L.P.

4. (Optional) If you have a tape drive on your machine, use SAM to create a non-standard device file for it. Specify DDS1 density, no compression, and Berkeley-style semantics. Let the other options default.

   This DDS1 device file may be useful if you create tapes that may need to be read on older tape drives that don't support the high-density format used by "BEST" device files.

   What device file does SAM create?

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Creating Device Files with `insf`

Though SAM must be used when you want to create serial-device device files or non-standard device files, most device files can be created automatically with **`insf`**. In this exercise you will have a chance to try using **`insf`**.

1. Remove all of the device files for one of the non-vg00 disks on your system.

   ```
   # rmsf -a /dev/dsk/cxtxdx
   ```

2. Run **`ioscan -funC disk`**. Do you see the disk's device files? Do you see the disk's hardware path?

   ```
   # ioscan –funC disk
   ```

3. Force the system to recognize the disk by running **`ioscan`** without the –u option. Does this recreate the device files?

   ```
   # ioscan –fnC disk
   ```

4. Recreate the disk device file with **`insf -C disk`**.

   ```
   # insf -C disk
   ```


© 2004 Hewlett-Packard Development Company, L.P.

5. Verify that the device file was recreated.

```
# ioscan -funC disk
```


© 2004 Hewlett-Packard Development Company, L.P.

Module 7 — Configuring Disk Devices

Objectives

Upon completion of this module, you will be able to do the following:

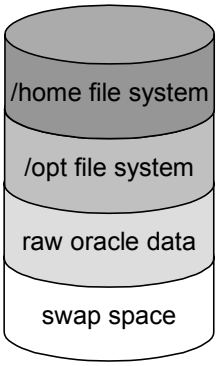
- Describe the reasons for disk partitioning.
- Partition a disk using the whole disk layout approach.
- Define the terms **volume group**, **logical volume**, and **physical volume**, and explain how they relate to each other.
- Create physical volumes, volume groups, and logical volumes from the command line.
- Compare and contrast the advantages and disadvantages of the whole disk layout approach, LVM, and the Veritas Volume Manager.

© 2004 Hewlett-Packard Development Company, L.P.

7–1. SLIDE: Disk Partitioning

Disk Partitioning

- Each HP-UX disk can have one or more partitions.
- Each partition can be used for:
  - A file system
  - Swap space
  - Raw application data
  - A Boot area
- Partitions can be configured using:
  - The Whole Disk Approach
  - Logical Volume Manager
  - Veritas Volume Manager



Student Notes

Disk space is organized into *partitions*. A partition is nothing more than a portion of disk space allocated for a particular purpose. A partition can span one disk, multiple disks, or a portion of a disk. Each partition can contain one of the following:

- A file system (space allocated for files and directories)
- A swap area (space used by the kernel to supplement physical memory)
- Raw data (data accessed directly by an application, such as a database)
- A boot area (space containing utilities used during the boot process)

HP-UX offers three different approaches for creating and managing disk partitions:

- The whole disk approach
- The Logical Volume Manager (LVM)
- The Veritas Volume Manager (VxVM)

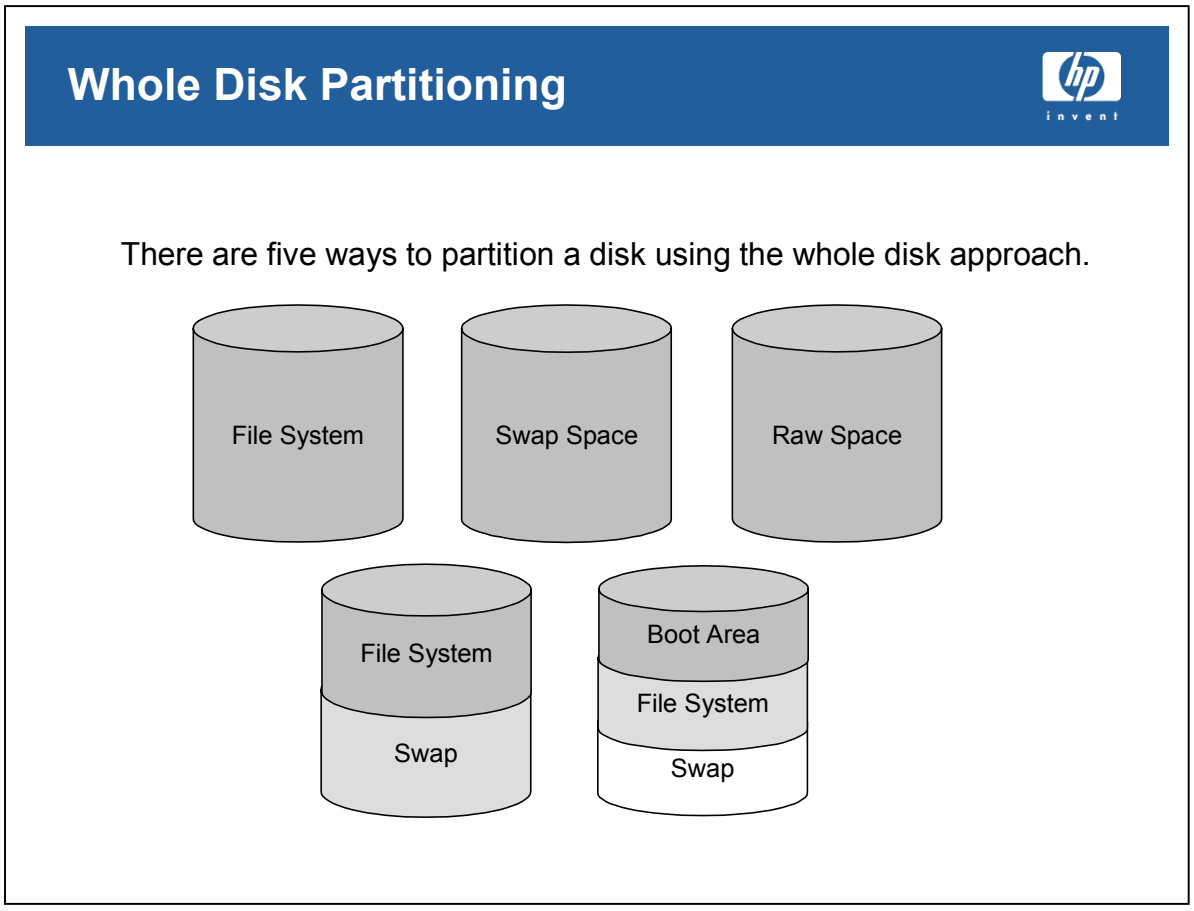
Some of the disks on your system can be configured using the whole disk layout approach, while others can be configured using LVM. All three techniques can be used concurrently on the same system, but not on the same disk.

© 2004 Hewlett-Packard Development Company, L.P.

All three approaches have advantages and disadvantages. This chapter will discuss all three disk-partitioning techniques, but will emphasize LVM, which is currently the most common of the three.

© 2004 Hewlett-Packard Development Company, L.P.

7–2. SLIDE: Whole Disk Partitioning



Student Notes

Using the whole disk approach, a disk may be configured five different ways:

- The disk can be dedicated entirely for use by a single file system.
- The disk can be dedicated entirely for use as swap.
- The disk can be dedicated entirely for use as a raw partition.
- A portion of the disk can contain a file system, with any remaining space used as swap.
- A disk can be configured as a boot disk, containing the root file system, a swap area, and a 2-MB special boot area containing utilities used during the boot process.

SAM is the easiest way to configure a disk using the whole disk approach:

```
SAM --> Disks and File Systems
    --> Disk Devices
    --> Actions --> Add a Disk --> Not Using LVM
```


© 2004 Hewlett-Packard Development Company, L.P.

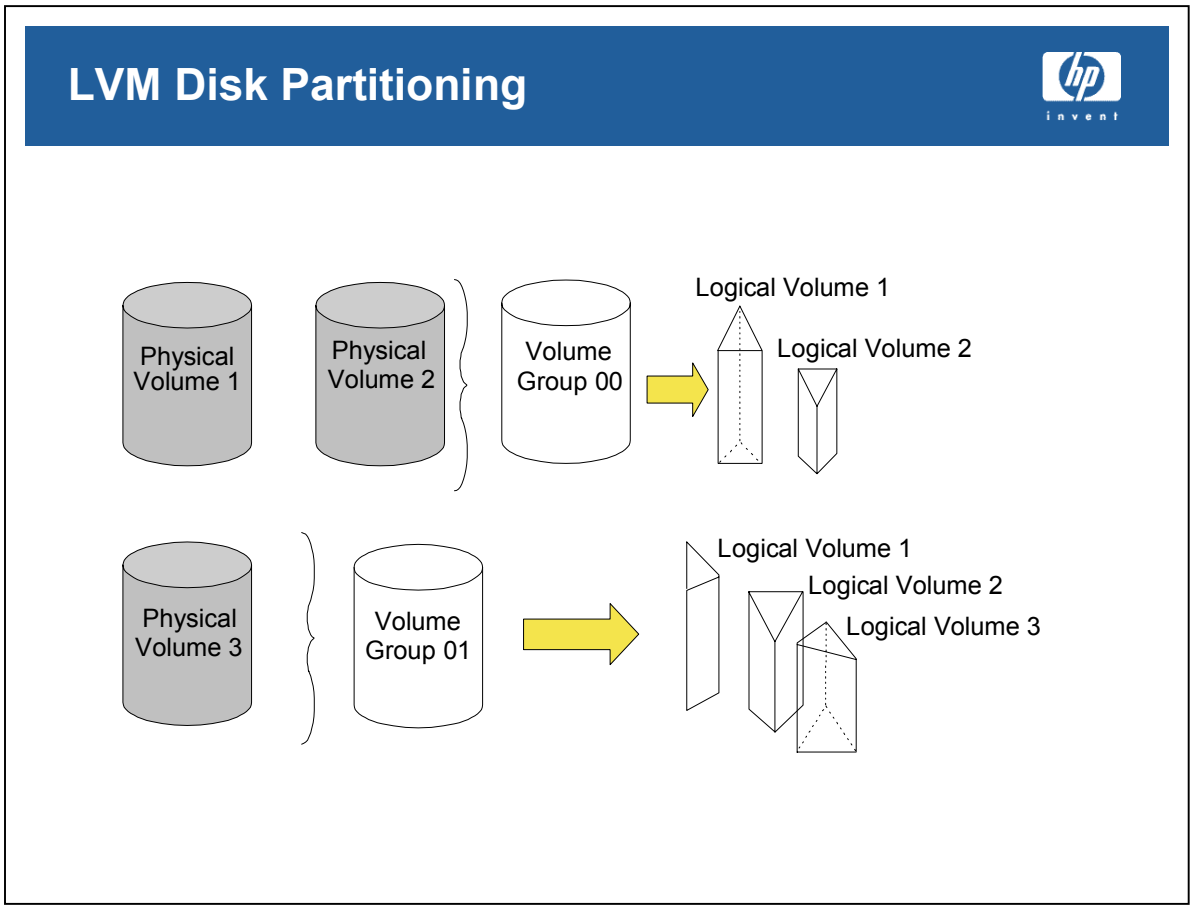
Though the whole disk approach is easy to use, it has several limitations:

- A file system cannot span multiple disks.
- There can only be one file system partition per disk.
- It is difficult to extend a file system if more space is needed.

For these reasons, many administrators choose to use the Logical Volume Manager to manage disk space instead of the whole disk approach.

© 2004 Hewlett-Packard Development Company, L.P.

7–3. SLIDE: LVM Disk Partitioning



Student Notes

Logical Volume Manager (LVM) enables you to pool space from several disks (known as "Physical Volumes" in LVM) to form a "Volume Group". You can then subdivide the space in the volume group into "Logical Volumes" (the LVM equivalent of a partition). The Logical Volume Manager (LVM) overcomes the limitations of the whole disk layout scheme by making it possible to do the following:

- Create logical volumes that span multiple disks.
- Create multiple logical volumes on a single disk.
- Extend and reduce logical volumes as necessary.

Physical Volumes

A disk managed by LVM is known as a physical volume. Several special data structures must be created on a disk before it can be used by LVM. Once these data structures have been created, the disk is considered to be a physical volume, and may be added to a volume group.

© 2004 Hewlett-Packard Development Company, L.P.

Volume Groups

A volume group is a group of one or more physical volumes. The physical volumes in a volume group form a pool of disk space which may be allocated to one or more logical volumes. Volume groups usually follow the naming convention:

- **`/dev/vg00`**
- **`/dev/vg01`**
- **`/dev/vg02`**
- and so on

However, you may use any naming convention you wish. **`vg00`** is a special volume group known as the "root volume group" which typically contains the default boot disk and the majority of the HP-UX operating system. You may have other volume groups on your system for applications, and other user and application data.

Logical Volumes

Disk space from a volume group may be allocated to one or more logical volumes. A logical volume is analogous to a partition, and may contain a file system, swap area, or raw partition.

Logical volumes can

- encompass all, or any portion of, the space on a physical volume.
- span multiple LVM physical volumes.
- be resized, or even moved to a different disk if the need arises.

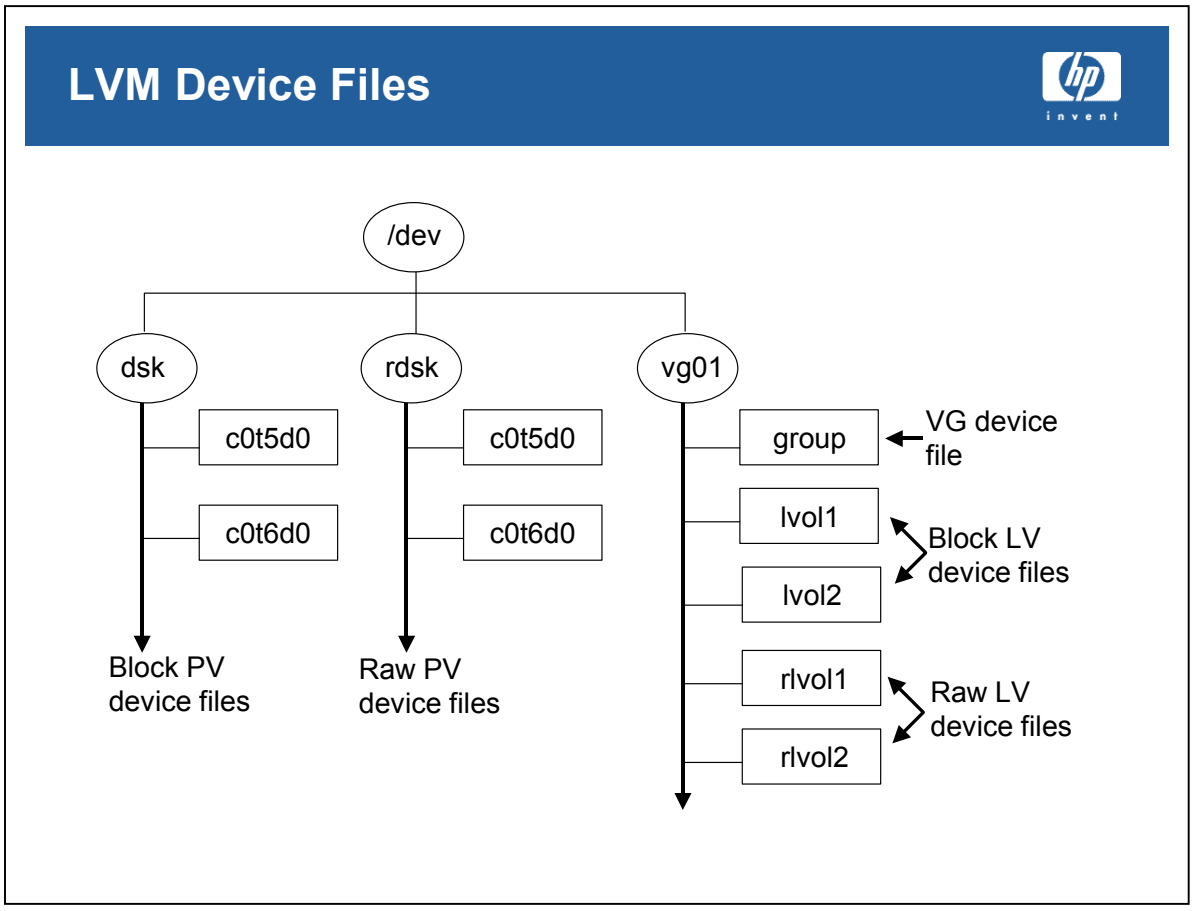
By default, logical volumes within a volume group are assigned names as follows:

- **`/dev/vg01/lvol1`**
- **`/dev/vg01/lvol2`**
- **`/dev/vg01/lvol3`**
- and so on

However, you may use any naming convention you wish.

© 2004 Hewlett-Packard Development Company, L.P.

7–4. SLIDE: LVM Device Files



Student Notes

Physical volumes, volume groups, and logical volumes are all referenced via device files, much as disk devices are referenced via device files.

Physical Volume Device Files

You can reference a physical volume using the physical volume's associated disk device files. Since disks may be referenced in either block or character mode, each physical volume has both a block and a character device file.

Examples

```
/dev/dsk/c0t5d0      # block device file for the disk at SCSI address 5.
/dev/rdsk/c0t5d0     # raw device file for the disk at SCSI address 5.
```

Volume Group Device Files

Volume groups are referenced via device files, too. Each volume group has a subdirectory under **`/dev`** containing a *group* device file for the volume group itself, as well as the device files for all of the logical volumes within the volume group. The name of the volume group's subdirectory determines the volume group's name.

© 2004 Hewlett-Packard Development Company, L.P.

Examples

```
/dev/vg01          # directory containing device files associated with vg01.
/dev/vg01/group    # device file for vg01 volume group.
```

Logical Volume Device Files

Logical volume device files are stored in the directory of the volume group to which they belong. Each logical volume has two device files: one is used when accessing the logical volume in character mode; the other is used when accessing the logical volume in block mode.

Examples

```
/dev/vg01/lvol1    # block device file for logical volume "lvol1" in vg01.
/dev/vg01/rlvol1   # raw device file for logical volume "lvol1" in vg01.
```

LVM Major and Minor Numbers

Like all other device files, every logical volume and volume group device file must have both a major and a minor number.

All LVM device files have major number **64**, the major number associated with the LVM kernel driver.

The first two digits of the minor number identify which volume group the device file is associated with. The last two digits identify the logical volume associated with the device file.

Name: **/dev/vg01/lvol2**
Major #: **64**
Minor #: **0x010002**

Example

- The major number for this device file, like all LVM device files, is **64**.
- The first two digits in the minor number (01) indicate that this is in **vg01**.
- The last two digits in the minor number indicate that this is **lvol2**.

Questions

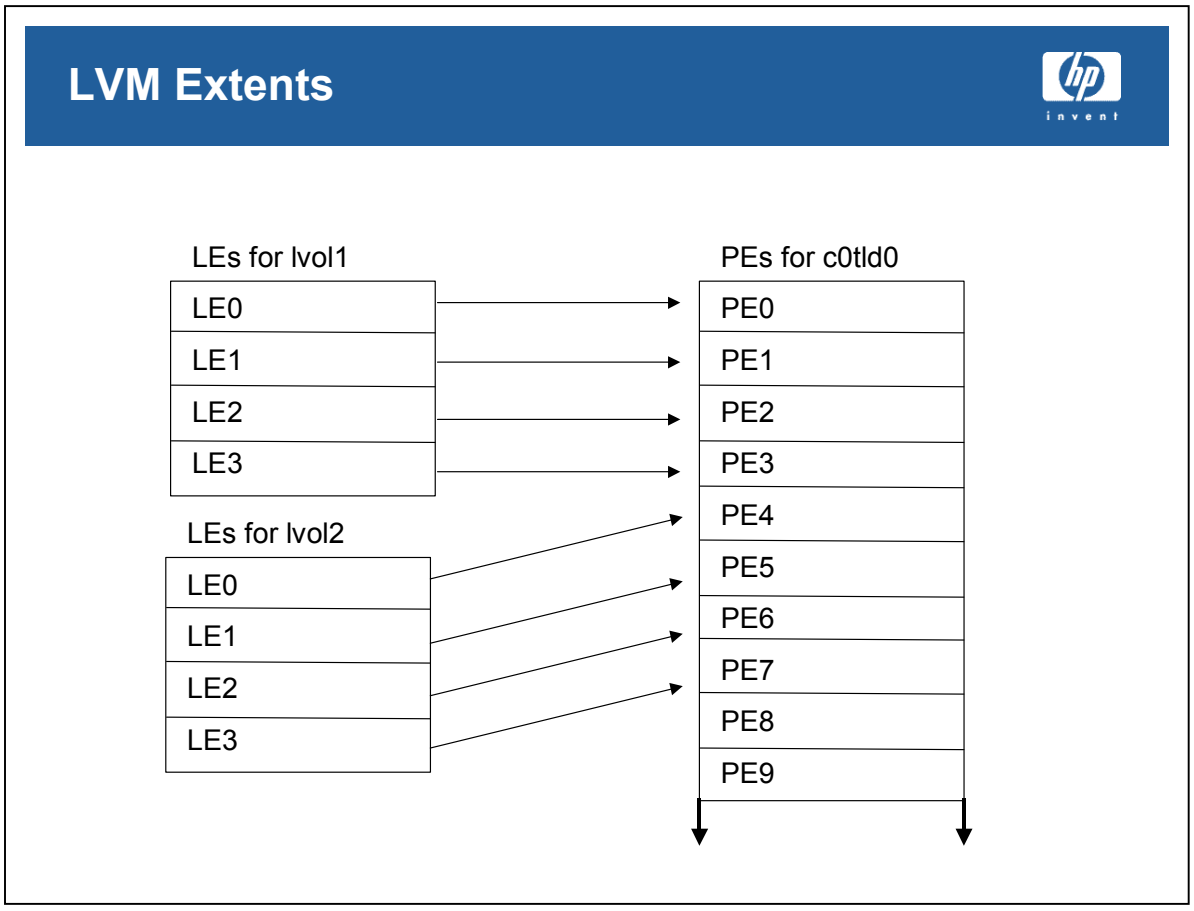
If **vg02** has three logical volumes created using the default naming convention:

1. What directory would contain the logical volumes' device files?
2. What would be the name of the volume group's device file?
3. What would be the name of the first logical volume's raw device file?
4. Overall, how many device files should you find in **/dev/vg02**?
5. What would be the minor number of the third logical volume's device file?

Answers are provided in the solutions appendix.

© 2004 Hewlett-Packard Development Company, L.P.

7–5. SLIDE: LVM Extents



Student Notes

We need to consider one more concept before looking at the practical side of configuring LVM.

The smallest allocatable unit of space in LVM is known as an *extent*. A physical volume is broken into *physical extents* (PEs), which are made available for allocation when the physical volume is added to a volume group.

A logical volume consists of a series of sequentially numbered *logical extents* (LEs). Each logical extent is nothing more than a pointer to a physical extent on disk. Larger logical volumes have more logical extents, and smaller logical volumes have fewer logical extents. In order to make a logical volume larger, LVM needs only allocate some additional extents.

The PE and LE sizes are consistent throughout a volume group, and may be set when the volume group is initially created. The default extent size is 4 MB.

The volume group shown on the slide, **`vg01`**, has two 16 MB logical volumes. Since the default extent size is 4 MB, each logical volume has four logical extents. Each logical extent is a pointer to a 4 MB physical extent on the disk.

© 2004 Hewlett-Packard Development Company, L.P.

Questions

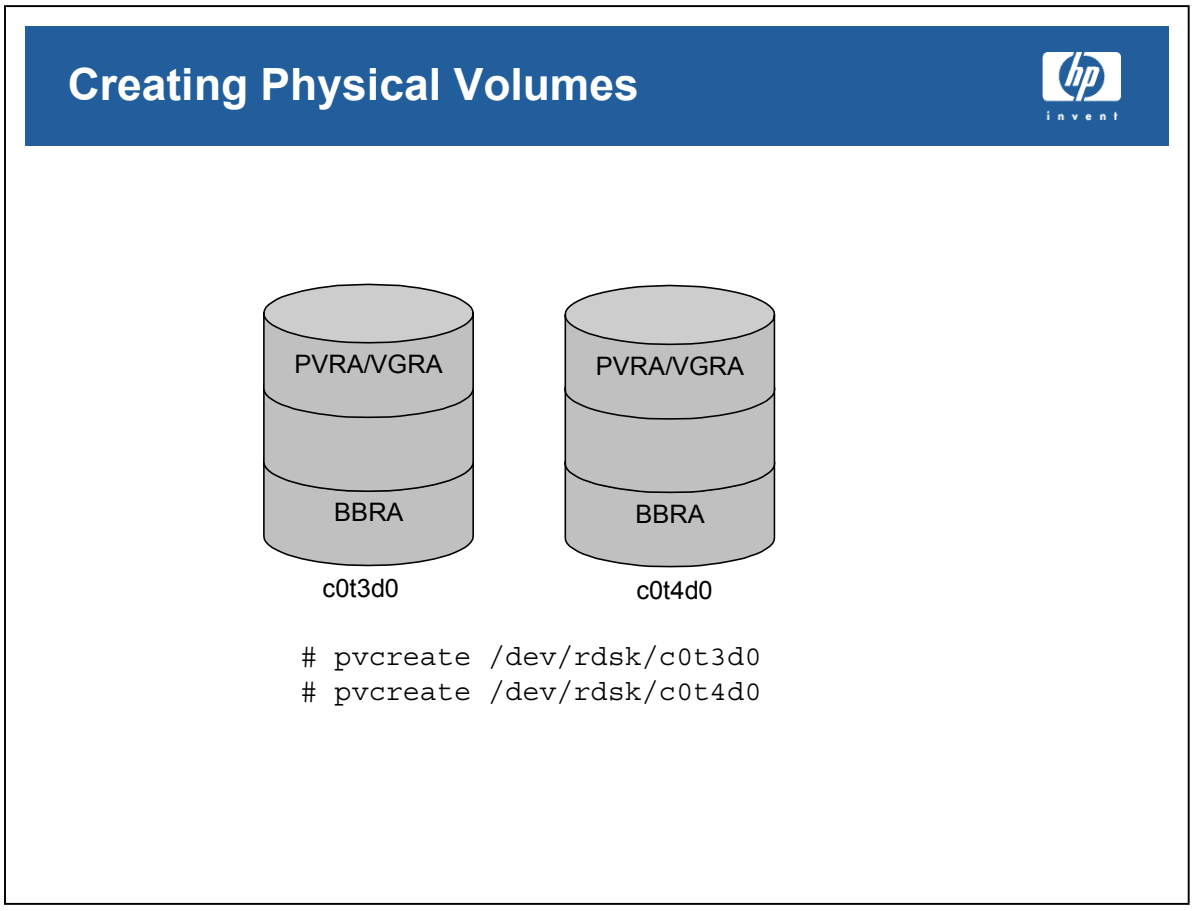
1. Assuming a volume group uses the default extent size, how many logical extents would you expect to find in an 88 MB logical volume?

2. Assuming a volume group uses the default extent size, how many physical extents would you expect to find on a 400 MB physical volume?

Answers are provided in the solutions appendix.

© 2004 Hewlett-Packard Development Company, L.P.

7–6. SLIDE: Creating Physical Volumes



Student Notes

Before you can use the space on a disk for logical volumes, you must designate the disk as an LVM physical volume. Once the disk has been configured as a physical volume, you can add the disk to a volume group and begin allocating space from the disk to logical volumes.

Preparing a Physical Volume

The disk you want to use as a physical volume must be connected to your system, be powered on, and belong to an existing volume group. Use the **`vgdisplay`** command determine which physical volumes are already part of a volume group, and the **`ioscan`** command to list the device file name for all the disks. By comparing the two lists, you can determine which disks are available for use as LVM physical volumes.

```
# vgdisplay -v | grep "PV Name"
# ioscan -funC disk
```


© 2004 Hewlett-Packard Development Company, L.P.

You can determine the size of a prospective physical volume via the **diskinfo** command.

```
# diskinfo /dev/rdsk/c0t3d0
SCSI describe of /dev/rdsk/c0t3d0:
             vendor: HP 18.2G
         product id: MAN3184MC
               type: direct access
               size: 17783240 Kbytes
   bytes per sector: 512
```

If the disk you choose to configure has old file system or Veritas Volume Manager header structures, those headers must be removed before you can configure the disk as an LVM physical volume. Use the **dd** command to write 1024 1K blocks to the top of the disk. If you execute the **dd** command without the **count=1024** argument, it will overwrite the entire disk.

```
# dd if=/dev/zero of=/dev/rdsk/c0t3d0 bs=1024 count=1024
# dd if=/dev/zero of=/dev/rdsk/c0t4d0 bs=1024 count=1024
```

Next, execute the **pvcreate** command to create the LVM data structures needed by LVM to begin using the disk as a physical volume. If the disk was previously part of another volume group, you may need to use the **-f** option on **pvcreate**. To overwrite any existing LVM or file system structures on the disk:

```
# pvcreate -f /dev/rdsk/c0t3d0
# pvcreate -f /dev/rdsk/c0t4d0
```

Your disk is now ready to be added to a new or existing volume group.

LVM Data Structures

LVM stores information in data structures at the beginning of the Physical Volume.

- **Physical Volume Reserved Area (PVRA)** contains LVM information specific to that Physical Volume. It is created by the **pvcreate(1M).**

- **Volume Group Reserved Area (VGRA)** contains LVM information specific to the entire Volume Group. A carbon copy of the VGRA is found on each Physical Volume in the Volume Group. Within the VGRA is the Volume Group Status Area (VGSA) which contains quorum information for the Volume Group, and the Volume Group Descriptor Area (VGDA) which contains information the device driver needs to configure the volume group for LVM. The VGRA is created by **vgcreate(1M)**.

- User Data Area contains file systems, virtual memory (swap), or user applications. When a volume group is created, the user data area is divided into fixed-size physical extents, which map to logical extents. The map of Logical Extents is contained in the VGRA.

- **Bad Block Relocation Area (BBRA)**, contains information specific to the bad block recovery mechanism.

- LVM boot disks contain additional data structures required by the boot process.

© 2004 Hewlett-Packard Development Company, L.P.

LVM Overhead

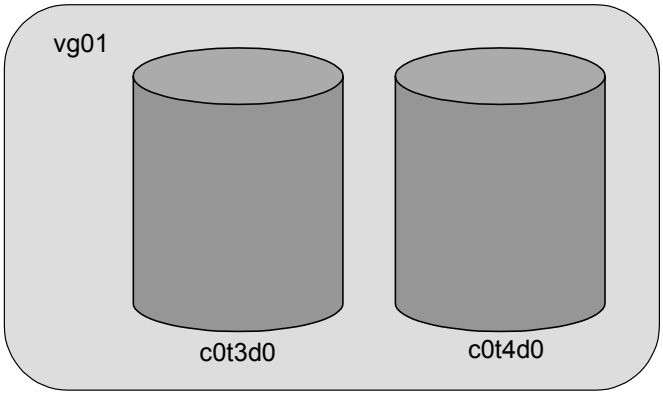
The data structures that are used by LVM consume some overhead from the disk space. This overhead is set at a fixed boundary for bootable LVM disks (2912 KB), and may vary in size for non-bootable LVM disks (typically 400 KB).

Overhead required on non-bootable disks depends on the parameters used on the **`vgcreate(1M)`** command or used in SAM. If you set a small extent size or create many physical volumes, your LVM data structures will be larger.

© 2004 Hewlett-Packard Development Company, L.P.

7–7. SLIDE: Creating Volume Groups

Creating Volume Groups



Create:
```
# mkdir /dev/vg01
# mknod /dev/vg01/group c 64 0x010000
# vgcreate vg01 /dev/dsk/c0t3d0 /dev/dsk/c0t4d0
```
Check:
```
# vgdisplay -v vg01
# pvdisplay -v /dev/dsk/c0t3d0
# pvdisplay -v /dev/dsk/c0t4d0
```

Student Notes

After the disks have been designated as LVM physical volumes you can create volume groups. Once the volume groups are created, you can then create the logical volumes in them.

Step 1

Create a directory for the volume group. The naming convention is **`/dev/vgnn`**, where *nn* is the volume group number. Use the next number in sequence on your system. If your system is pre-installed, you already have a volume group **`/dev/vg00`**.

For example, to create volume group **`01`**, which would be the second volume group on your system, you would type:

```
# mkdir /dev/vg01
```

Step 2

The group special file, or control file, provides the means by which LVM kernel and LVM commands communicate within the volume group you create.

© 2004 Hewlett-Packard Development Company, L.P.

Create the control file named **group** in the directory **/dev/vgnn**. Use the **mknod(1M)** command. The **group** file is a *character* device file. The major number is always **64**. The minor number is hexadecimal, always ends in **0000**, and has the following form:

```
0xhh0000
```

where *hh* is the hexadecimal representation of the volume group number.

For example, to create a **group** file for a volume group **01**, you would type:

```
# mknod /dev/vg01/group c 64 0x010000
```

Be sure to verify the permissions on the new **group** file after executing the **mknod** command. The **group** file should be owned by root, with **rw-r-----** permissions; otherwise, unauthorized users may be able to access or change data stored in your logical volumes.

Step 3

Now you can create the volume group and specify the physical volumes it will contain. You use the **vgcreate(1M)** command. You can assign several volumes to a group at one time.

```
# vgcreate /dev/vg01 /dev/dsk/c0t3d0  /dev/dsk/c0t4d0
```

Note that you are using the *block* device file to create the volume group.

The options on **vgcreate** include

-e *max_physical_extents*	Sets the maximum number of physical extents per physical volume in a volume group (default is 1016).
-l *max_logical_vols*	Sets the maximum number of logical volumes allowed in a volume group (default 255).
-p *max_physical_vols*	Sets the maximum number of LVM disks (physical volumes) allowed in a volume group (default 16).
-s *physical_extent_size*	Sets the size, in megabytes, for each physical extent in a volume group (default 4).

Step 4

You can verify that you have created the volume group using the **vgdisplay** and **pvdisplay** commands:

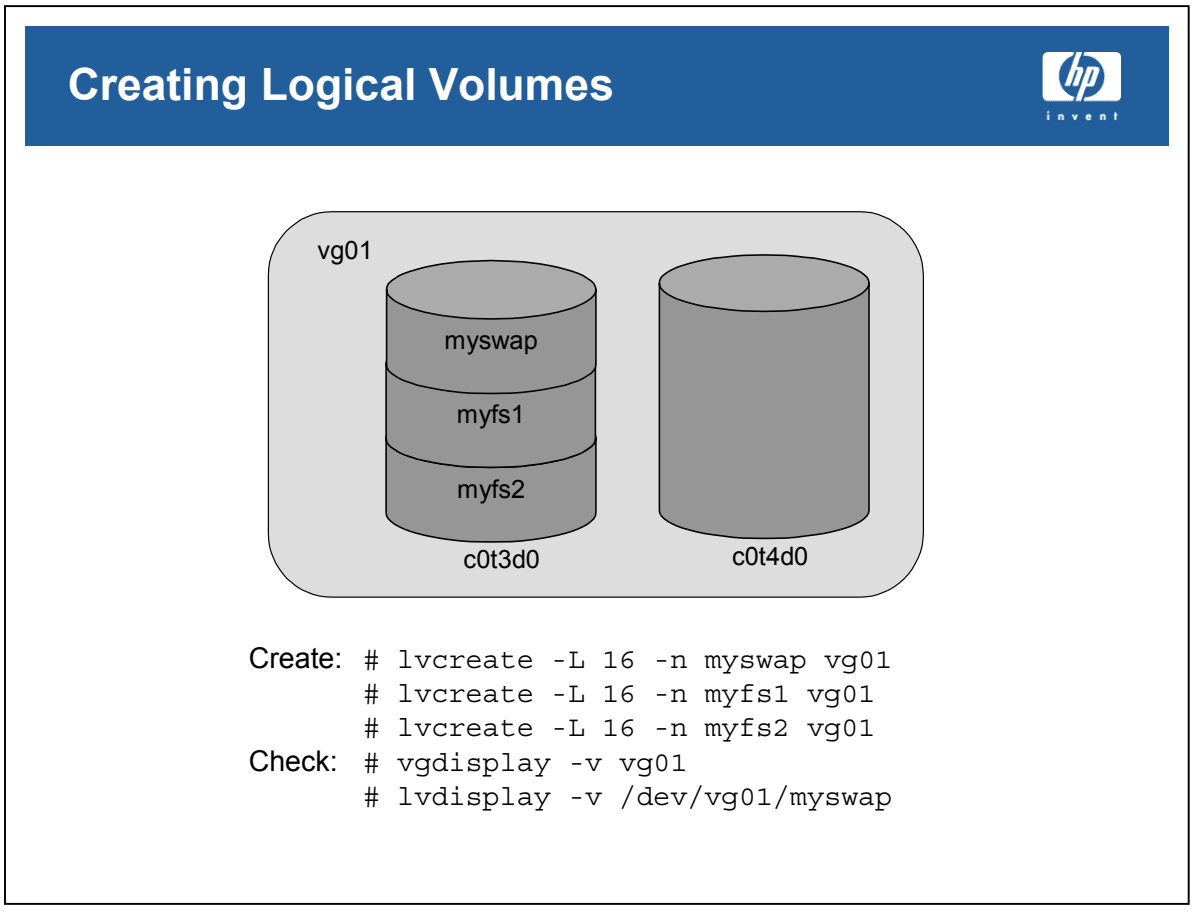
```
# vgdisplay -v vg01
--- Volume groups ---
VG Name                     /dev/vg01
VG Write Access             read/write
VG Status                   available
Max LV                      255
Open LV                     0
Max PV                      16
Cur PV                      2
```


© 2004 Hewlett-Packard Development Company, L.P.

```
Act PV                      2
Max PE per PV               1016
VGDA                        4
PE Size (Mbytes)            4
Total PE                    500
Alloc PE                    0
Free PE                     500
Total PVG                   0
Total Spare PVs             0
Total Spare PVs in use      0

   --- Physical volumes ---
   PV Name                     /dev/dsk/c0t3d0
   PV Status                   available
   Total PE                    250
   Free PE                     250

   PV Name                     /dev/dsk/c0t4d0
   PV Status                   available
   Total PE                    250
   Free PE                     250

# pvdisplay /dev/dsk/c0t3d0
 --- Physical volumes ---
PV Name                     /dev/dsk/c0t3d0
VG Name                     /dev/vg01
PV Status                   available
Allocatable                 yes
VGDA                        2
Cur LV                      0
PE Size (Mbytes)            4

Total PE                    250
Free PE                     250
Allocated PE                0
Stale PE                    0
IO Timeout                  default

# pvdisplay /dev/dsk/c0t4d0
 --- Physical volumes ---
PV Name                     /dev/dsk/c0t4d0
VG Name                     /dev/vg01
PV Status                   available
Allocatable                 yes
VGDA                        2
Cur LV                      0
PE Size (Mbytes)            4
Total PE                    250
Free PE                     250
Allocated PE                0
Stale PE                    0
IO Timeout                  default
```


© 2004 Hewlett-Packard Development Company, L.P.

7–8. SLIDE: Creating Logical Volumes



Student Notes

Create logical volumes in volume groups using the **`lvcreate(1M)`** command.

You can allocate disk space for file systems, swap, or raw data in either megabytes or LVM extents.

When **`lvcreate`** creates the logical volume, it creates the block and character device files and places them in the directory **`/dev/vg`*`nn`*`.`** It is created without size unless you specify a size when you use the **`lvcreate`** command. You can create a logical volume using the default characteristics, and change them later.

Options

`-L` ***`logical_volume_size`***	The size of the logical volume in megabytes. The size specified will be rounded up to the nearest whole logical extent. The default is zero.
`-l` ***`logical_extents_number`***	The number of logical extents in the logical volume. The default is zero.

© 2004 Hewlett-Packard Development Company, L.P.

-n *name*	A custom name you want to assign to the logical volume. The default name follows the naming convention.

Examples

To create a logical volume with the default characteristics, in the volume group **/dev/vg01**:

```
# lvcreate /dev/vg01
```

Only the name will be reserved; neither physical nor logical extents will be reserved. You will need later to extend the logical volume with the command **lvextend(1M)**.

To create a logical volume of 10 logical extents in size:

```
# lvcreate -l 10 /dev/vg01
```

To create a logical volume with a size of 100 MB:

```
# lvcreate -L 100 /dev/vg01
```

To create a logical volume with a nonstandard name:

```
# lvcreate -L 16 -n myswap vg01
```

Viewing Your Logical Volumes

You can use two commands to view information about your logical volumes:

```
# vgdisplay -v vg01                # to determine which LVs are in vg01
# lvdisplay -v /dev/vg01/myswap    # view details about the myswap LV
```

Full output from the **lvdisplay** command is shown below:

```
 # lvdisplay -v /dev/vg01/myswap
 --- Logical volumes ---
LV Name                     /dev/vg01/myswap
VG Name                     /dev/vg01
LV Permission               read/write
LV Status                   available/syncd
Mirror copies               0
Consistency Recovery        MWC
Schedule                    parallel
LV Size (Mbytes)                   16
Current LE                  4
Allocated PE                4
Stripes                     0
Stripe Size (Kbytes)        0
Bad block                   on
Allocation                  strict
IO Timeout (Seconds)        default
```


© 2004 Hewlett-Packard Development Company, L.P.

```
      --- Distribution of logical volume ---
      PV Name            LE on PV  PE on PV
      /dev/dsk/c0t3d0    4         4

      --- Logical extents ---
      LE   PV1                PE1  Status 1
      0000 /dev/dsk/c0t3d0    0000 current
      0001 /dev/dsk/c0t3d0    0001 current
      0002 /dev/dsk/c0t3d0    0002 current
      0003 /dev/dsk/c0t3d0    0003 current

# lvdisplay -v /dev/vg01/myfs1
  --- Logical volumes ---
LV Name                     /dev/vg01/myfs1
VG Name                     /dev/vg01
LV Permission               read/write
LV Status                   available/syncd
Mirror copies               0
Consistency Recovery        MWC
Schedule                    parallel
LV Size (Mbytes)            16
Current LE                  4
Allocated PE                4
Stripes                     0
Stripe Size (Kbytes)        0
Bad block                   on
Allocation                  strict
IO Timeout (Seconds)        default

   --- Distribution of logical volume ---
  PV Name            LE on PV  PE on PV
  /dev/dsk/c0t3d0    4         4

  --- Logical extents ---
LE   PV1                PE1  Status 1
0000 /dev/dsk/c0t3d0    0004 current
0001 /dev/dsk/c0t3d0    0005 current
0002 /dev/dsk/c0t3d0    0006 current
0003 /dev/dsk/c0t3d0    0007 current

# lvdisplay -v /dev/vg01/myfs2
  --- Logical volumes ---
LV Name                     /dev/vg01/myfs2
VG Name                     /dev/vg01
LV Permission               read/write
LV Status                   available/syncd
Mirror copies               0
Consistency Recovery        MWC
Schedule                    parallel
LV Size (Mbytes)            16
Current LE                  4
Allocated PE                4
```


© 2004 Hewlett-Packard Development Company, L.P.

```
Stripes                     0
Stripe Size (Kbytes)        0
Bad block                   on
Allocation                  strict
IO Timeout (Seconds)        default

   --- Distribution of logical volume ---
  PV Name            LE on PV  PE on PV
  /dev/dsk/c0t3d0    4         4

--- Logical extents ---
LE   PV1                PE1  Status 1
0000 /dev/dsk/c0t3d0    0008 current
0001 /dev/dsk/c0t3d0    0009 current
0002 /dev/dsk/c0t3d0    0010 current
0003 /dev/dsk/c0t3d0    0011 current
```


© 2004 Hewlett-Packard Development Company, L.P.

7–9. SLIDE: Disk Space Management Tool Comparison

Disk Space Management Tool Comparison

	Whole Disk	LVM	VxVM
HP-UX versions supported	10.x,11.x,11i	10.x,11.x,11i	11i
Boot disk support?	Yes	Yes	Yes
GUI configuration tool available?	SAM-based	SAM-based	Java-based
Available on other platforms?	Similar	Similar	Yes
Can partitions span disks?	No	Yes	Yes
Online resizing supported?	No	Yes	Yes
Online backups supported?	No	Yes*	Yes*
Striping (RAID 0) supported?	No	Yes	Yes
Mirroring (RAID 1) supported?	No	Yes*	Yes*
Mirrored stripes (RAID 0/1) supported?	No	Yes*	Yes*
RAID 5 supported?	No	No	Yes*
Dynamic relayout support?	No	No	Yes*
Dynamic multipathing support?	No	Partial	Full*

* Features indicated by an asterisk require an extra license

Student Notes

The first slide in this chapter noted that HP-UX now supports three different disk space management solutions. Depending on your system's needs, you can choose to use one, two, or all three solutions on your system. This slide compares the relative advantages and disadvantages of each approach.

HP-UX Versions Supported

The whole disk approach and LVM are available on all current versions of HP-UX. VxVM is only supported on HP-UX 11i.

Boot Disk Support?

Early versions of VxVM could only be used to configure HP-UX data disks. However, the current version of VxVM, version 3.5, may be used to manage both data and boot disks.

GUI Configuration Tool Available?

Whole disk and LVM partitions can be configured through SAM. VxVM, however, can be configured via a much more flexible, intuitive, Java-based interface called Veritas Enterprise Administrator (VEA).

© 2004 Hewlett-Packard Development Company, L.P.

Available on Other Platforms?

Some variation of the whole disk layout approach is available on almost all UNIX platforms. Logical Volume Manager is included with most Linux distributions, AIX (though the commands are somewhat different), and several other UNIX flavors. Veritas has ported VxVM to most major commercial UNIX flavors, too.

Can Partitions Span Disks?

Today's databases are oftentimes several hundred gigabytes in size, and can span multiple disks. LVM and VxVM make this possible by allowing logical partitions to span multiple physical disks. Partitions created using the whole disk approach cannot span disks.

Online Resizing Supported?

Partitions configured via the whole disk layout approach can't be resized. LVM and VxVM both allow the administrator to extend and reduce partitions at will, while disks are being accessed.

Online Backups Supported?

More and more IT shops today operate 24 hours per day, 7 days a week. These "always-on" environments make it difficult to perform system backups, since data files are likely to be constantly changing. Both LVM and VxVM make it possible to perform backups on a "live" partition.

Striping (RAID 0) Supported?

Striping maps a partition's data so that the data is interleaved among two or more physical disks to ensure that each disk shares an equal portion of the overall system I/O load. Load balancing the system disks in this manner can greatly improve system performance in some situations.

Mirroring (RAID 1) Supported?

Both LVM and VxVM support data "mirroring." Data written to a mirrored partition is physically written to two or more physical disks. Thus, even if a single disk fails, your applications can still access their data via the second mirror. LVM supports three-way mirroring, while VxVM allows up to 32-way mirrors of each partition!

Mirrored Stripes (RAID 0/1) Supported?

Disk mirroring provides improved reliability, while disk striping provides improved performance. LVM and VxVM both make it possible to mirror and stripe a partition simultaneously! LVM implements this functionality through a feature known as "distributed" logical volumes; VxVM provides a similar, but somewhat more flexible solution.

RAID 5 Supported?

The mirrored/striped partitions described above deliver both stability and performance. However, mirrored/striped partitions are also expensive to implement, since every megabyte of data requires two or more megabytes of disk space: one for each mirror. RAID 5 is a sophisticated technology that is included in VxVM to provide the same stability that one would expect from a mirrored partition, while simultaneously providing some of the same

© 2004 Hewlett-Packard Development Company, L.P.

performance benefits that one would expect from a striped partition. RAID 5 is not supported by LVM.

Dynamic Relayout Support?

The last few sections on this slide noted that VxVM supports striped, mirrored, mirrored/striped, and RAID 5 partitions. VxVM also makes it possible to dynamically convert a partition from one format to another. For instance, a RAID 5 partition may be converted to a mirrored partition. Striping parameters may be changed while users are accessing mounted file systems! This functionality is not supported in LVM.

Dynamic Multipathing Support?

Most disk arrays can be physically connected to two or more host interface cards. If one interface card fails, LVM and VxVM are capable of dynamically switching to the alternate path through the second host interface to avoid downtime and data loss. With the proper hardware, VxVM is even capable of utilizing both paths simultaneously to improve performance!

Add-on Functionality

The features in the table above that are preceded by an asterisk (*) require a special license.

The LVM Mirrordisk/UX license is now included with HP's Enterprise and Mission Critical Operating Environments, which are described in the OS install chapter later in the course. Customers that don't have the Mission Critical / Enterprise Operating Environments must purchase a license and install the Mirrordisk/UX bundle from the HP-UX Application CD (bundle B2491BA for servers, or B5403BA for workstations). To determine if you have the mirroring license on your machine, type the following:

```
# swlist LVM.LVM-MIRROR-RUN
```

All HP-UX Operating Environments include the **`Base-VxVM`** product, which makes it possible to configure simple VxVM volumes and disk groups. In order to use VxVM online features, though, you must install the B9116AA bundle from the HP-UX Application CD. To determine if you have access to the VxVM online features, type:

```
# swlist B9116AA
```

For More Information

For more information on the advanced features of LVM and VxVM that are described above, consider enrolling in HP Education's Hands-on with LVM and VxVM courses. LVM documentation can also be found on the **`http://docs.hp.com`** website in HP's *Managing Systems and Workgroups* manual. For more information on VxVM, look for the *Veritas Volume Manager 3.5 Administrators Guide*.

© 2004 Hewlett-Packard Development Company, L.P.

7–10. LAB: Logical Volume Manager

Directions

In this exercise you will have an opportunity to create several physical volumes, volume groups, and logical volumes. Except where noted, do all of the exercises from the command line. You will have a chance to create some logical volumes via SAM in a later chapter.

Part 1: Choosing a Disk for Use in a Volume Group

1. How many disks do you have on your system? Determine each disk's hardware path and device file names.

2. Use **`vgdisplay`** to determine which of the disks on your system are already members of active volume groups. You should have at least one disk that isn't currently in a volume group. Note the free disk's device file name; you will create a new volume group using your free disk in the next part of the lab exercise.

3. Before adding a disk to a volume group, you may want to check the size of the disk. This is accomplished via the **`diskinfo`** command. How large is your spare disk?

   ```
   # diskinfo /dev/rdsk/cxtxdx
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Creating a Physical Volume, Volume Group, and Logical Volumes

1. Configure your free disk as an LVM physical volume.

2. Can you **`pvdisplay`** your disk at this point? Try it.

3. Create a new **`vg01`** volume group using your newly created physical volume.

4. Use **`vgdisplay`** and **`pvdisplay`** to check the status of your new physical volume and volume group. How many physical volumes are in the volume group at this point? How many logical volumes are in the volume group at this point? What is the extent size?

5. Create two 24-MB logical volumes in your new volume group. Name the first logical volume **`cad`** and the second **`cam`**.

© 2004 Hewlett-Packard Development Company, L.P.

6. Use **vgdisplay** and **lvdisplay** to ensure that your new logical volumes were actually created.

7. Do an **ll** of the **/dev/vg01** directory. What is the name of the volume group device file for your new volume group? Each of your logical volumes should have two device files. Why?

8. Use SAM to remove your **vg01** volume group in preparation for the next portion of this lab. Volume groups can be managed in SAM from the Volume Groups object list:

   ```
   SAM --> Disks and File Systems --> Volume Groups
   ```

9. Try a **vgdisplay** back at the command line to ensure the volume group is gone. Did SAM remove the volume group device files, too?

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: (Optional) LVM "What-Ifs"

1. What happens if you **`pvcreate`** a disk that is already in use by another active volume group? Try it. See what happens if you **`pvcreate`** your boot disk in **`vg00`**. Did this work? Try again using the **`-f`** "force" option on **`pvcreate`**. What is the result?

2. Re-**`pvcreate`** the spare disk you used in the previous part of the lab. (This **`pvcreate`** should work.)

3. For the sake of variety, create a new volume group called **`vg02`** using the spare disk you just **`pvcreate`**d.

4. Now that you have a volume group, try creating a few logical volumes. Create a logical volume called **`test1`** in **`vg02`**. This time, though, don't specify a size for your logical volume. Based on the result of this experiment, what is the default logical volume size?

© 2004 Hewlett-Packard Development Company, L.P.

5. What happens if you don't specify a logical volume name when you **`lvcreate`**? Try it. Create two new logical volumes of size 12 MB and 16 MB, leaving off the **`-n`** option in both cases. What names did LVM assign to your new logical volumes? Why?

6. See what happens if you attempt to create an 11 MB logical volume in **`vg02`** called test2. Watch the output from **`lvcreate`** carefully. What size is your new logical volume? Explain.

7. At some point in your UNIX career, you will almost certainly use **`-l`** instead of **`-L`** on **`lvcreate`** accidentally. What appears to be the difference between these two options? Try it and find out.

```
# lvcreate -l 12 -n test3 vg02
# lvcreate -L 12 -n test4 vg02
# vgdisplay -v vg02 | more
```

8. Once again, use SAM to remove the volume group you just created.

```
SAM --> Disks and File Systems --> Volume Groups
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: (Optional) Exploring VxVM

As of HP-UX 11i, there is an alternative to the Whole Disk and LVM disk management solutions: VxVM. This optional portion of the lab gives you an opportunity to explore some of the basic VxVM functionality.

1. VxVM refuses to manage disks that contain LVM header information. Use the **dd** command to clobber any remnants of the LVM headers on your spare disk. Make sure you choose the right disk! If you accidentally overwrite the LVM headers on your boot disk, your system may be left in an unbootable state!

   ```
   # dd if=/dev/zero of=/dev/rdsk/cxtxdx bs=1024 count=1024
   ```
   Use your spare disk's device filename!

2. Verify that VxVM is installed on your system.

   ```
   # swlist Base-VXVM
   ```

3. Run the **vxinstall** program to initialize VxVM. You only need to run this utility once, when you first install VxVM.

   ```
   # vxinstall
   ```

4. When asked if you wish to enter a license key, answer “n”. We will only use the “base” VxVM functionality, which doesn’t require a license.

   ```
   Are you prepared to enter a license key [y,n,q,?] n
   ```

5. When asked if you wish to use enclosure based names for your disks, answer “n”. Enclosure-based names are most useful in disk array and SAN environments.

   ```
   Do you want to use enclosure based names for all disks ? n
   ```

6. The program will interactively walk you through a series of menus. Whenever **vxinstall** asks if you wish to continue, press Press **[Return]**. If you make a mistake, you can press **^C** to break out and start over again at any point.

   a) The program will search for and report your system's disk enclosures and/or controller cards. Press **[Return]**.

   b) Next, **vxinstall** asks how you wish to configure the disks on your system. You can choose either a "Quick Installation" (in which VxVM initializes all currently unused disks with little interaction from the administrator) or "Custom Installation" (in which you get to choose which disks should and shouldn't be configured). Choose "Custom Installation".

   c) If you chose "Custom Installation", **vxinstall** should list all of the currently unused disks on the system, and ask if you wish to initialize them. You can initialize selected disks, or all of the currently unused disks. For the sake of this exercise, choose "Install all disks as new disks".

© 2004 Hewlett-Packard Development Company, L.P.

d) Finally, VxVM gives you an opportunity to assign a meaningful name to each disk. These names will be used in place of device file names throughout the menus that follow. Accept the default disk names for the selected disks.

```
Use default disk names for these disks? [y,n,q,?] y
```

e) When asked to confirm that you wish to destroy the selected disks, answer "y".

```
Are you sure you want to destroy the contents of this disk? y
```

f) Proceed through the remaining confirmation screens.

7. VxVM can be managed from the command line, or via the "Veritas Enterprise Administrator" (**vea**) GUI-based management tool. In this lab, we will use **vea**. VEA is a client/server application. After you initially configure VxVM with **vxinstall**, it is necessary to stop and restart the VEA server daemon, **vxsvc**.

```
# /opt/VRTS/bin/vxsvc -k
# /opt/VRTS/bin/vxsvc
```

It may take several minutes for the daemon to fully launch. You can monitor its progress via the **vxsvc -m** command. When **vxsvc -m** reports that the daemon is Running, proceed on to the next step.

```
# /opt/VRTS/bin/vxsvc -m
Current state of server: RUNNING
```

8. Now launch the VEA client GUI, **vea**.

```
# /opt/VRTSob/bin/vea
```

9. On the VEA Connection screen, enter hostname **localhost**, and the root username and password. Click [OK]. Be patient! It may take several minutes for VEA to download all the necessary plugins.

10. The left pane displays a hierarchical tree of VxVM objects. Objects can be expanded and collapsed by clicking the "**+**" and "**-**" icons.

a) On the left pane, click the "**+**" sign to the left of **localhost** to expand the list of VxVM objects on your localhost.

b) You should now see a list of VxVM objects available on localhost such as **Controllers**, **Disk Groups**, and **Volumes**. Some of these objects can be further expanded. Click on the "**+**" sign to the left of **Disk Groups**.

© 2004 Hewlett-Packard Development Company, L.P.

11. Clicking on an object in the left pane (not the "**+**" sign preceding the object) displays that object's properties in the right pane.

    a) Click on **Controllers** to view a list of disk controllers.

    b) Click on **Disks** to view a list of disks on your system.

    c) Click on **File Systems** to view a list of file systems.

    d) Click on **Disk Groups** to view a list of VxVM disk groups. Disk groups are conceptually very similar to LVM volume groups. The **vxinstall** program should have automatically created a disk group for you called **rootdg**.

12. The **vea** interface may be used to view your VxVM configuration, but can also be used to modify your configuration. Let's create a VxVM volume in the **rootdg** disk group. VxVM volumes are conceptually very similar to LVM logical volumes." At this point, you shouldn't have any VxVM volumes configured.

    a) To configure a VxVM volume, left-click on Disk Groups in the left pane to display a list of available disk groups in right pane.

    b) In the right pane, right click on **rootdg**. A menu of available actions should appear. Choose **New Volume ...**.

    c) In the **New Volume** wizard dialog, click **Next>** to proceed to the **Select attributes** screen.

    d) On the **Select attributes** screen, enter the following information:

Disk Group Name:	**rootdg**
Volume Name:	**data**
Comment:	**Contains application data**
Size:	**12MB**

    Accept the defaults for all other fields.

    e) Click **Next>** to proceed to the **Select disks** screen.

    f) On the **Select disks** screen, click the **Let Volume Manager decide what disks to use** radio button.

    g) Click **Next>** to proceed to the **Create file system** screen.

    h) On the **Create file system** screen, click the "Create a file system" radio button, then enter the following information:

    File system type: VxFS
    Mount point: **/data**

© 2004 Hewlett-Packard Development Company, L.P.

i) Accept the defaults for the rest of the screen, then click **Next>** to proceed to the final confirmation screen.

j) Click **Finish** to create the volume and file system.

k) Click the **Volumes** object in the left pane to display a list of volumes in the right pane. Verify that your volume was created properly.

13. Verify that your file system was created properly. In the left pane, left-click **File Systems** to display a list of file systems in the right pane.

14. VxVM makes it very easy to make your volumes and file systems larger and smaller, too.

    a) In the left pane, left-click **File Systems** to display a list of file systems in the right pane.

    b) In the right pane, right click on **/data**, and select **Resize File System** from the resulting menu.

    c) Set the **New Size** to 24 MB.

    d) Click **OK**.

    e) Use the same procedure to resize the Volume back to 12M.

15. Can you guess what procedure is required to remove a Volume?

    a) In the left pane, left-click **Volumes** to display a list of volumes in the right pane.

    b) In the right pane, right click on the **data** volume, and select **Delete Volume** from the resulting menu.

    c) Confirm that you desire to delete the volume and file system.

16. If time permits, explore some of the other **vea** menus on your own.

17. If you create any additional volumes, remove them before proceeding.

18. Choose the **Exit** option from the **File** menu to exit out of **vea**.

19. Some disks on your system may be managed by VxVM, while others may be managed by LVM. However, no disk on the system can be managed by both VxVM and LVM concurrently. Your student workstation probably only has one disk, which will be needed for the last part of this lab exercise. Before proceeding, clobber the VxVM headers on your spare disk so it can again be used by LVM.

    Normally, **vea** makes it very easy to remove disks from VxVM disk groups. However, after running **vxinstall**, a minimum of one disk must remain under the control of VxVM. The only way to reclaim your spare disk for use in the LVM labs that remain in the next few modules is to resort to the command line. Replace ***cxtxdx*** with the device name of the disk in your **rootdg**.

© 2004 Hewlett-Packard Development Company, L.P.

```
# vxconfigd -k -r reset -m disable
# vxdg init rootdg
# vxdisk rm cxtxdx
# vxdctl stop
# dd if=/dev/zero of=/dev/rdsk/cxtxdx bs=1024 count=1024
```

NOTE: This lab only demonstrated VxVM's basic functionality. For more information on VxVM, attend one of HP's VxVM training courses, or take a look at the Veritas Volume Manager documentation on **http://docs.hp.com**.

© 2004 Hewlett-Packard Development Company, L.P.

Part 5: Creating One Last Volume Group

1. In preparation for the discussion of file systems and swap in the following chapters, create a volume group **`vg01`** with your spare disk. Create three 12 MB logical volumes in **`vg01`** called **`data`**, **`app`**, and **`tables`**.

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 8 — File System Concepts

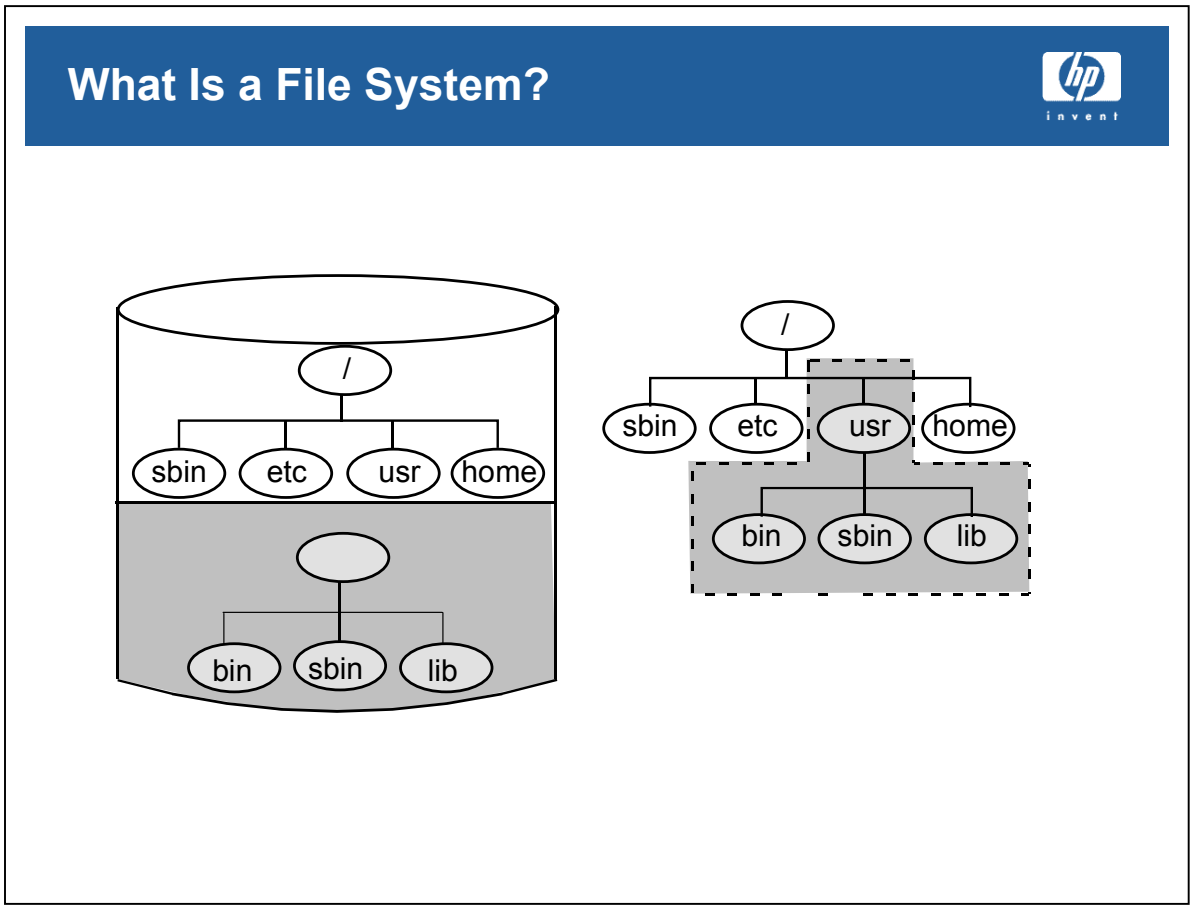
Objectives

Upon completion of this module, you will be able to do the following:

- List the file system types available in HP-UX.
- Describe the difference between **user data** and **metadata**.
- Describe the structure of an HFS file system.
- Describe the structure of a JFS file system.
- Compare the process used to update HFS metadata versus JFS metadata.
- Define the terms: **superblock** and **inode**.
- Define the terms: **block, fragment**, and **extent**.
- Compare **hard** and **soft** links.
- Create hard and soft links.

© 2004 Hewlett-Packard Development Company, L.P.

8–1. SLIDE: What Is a File System?



Student Notes

What Is a File System?

A UNIX file system is a collection of files and directories stored and managed together as a unit. Each file system is stored in a separate logical volume or whole disk partition. A typical HP-UX machine usually has multiple file systems. The following are just a few of the file systems you may have on your machine:

- The files under **`/usr`** are usually stored in one file system.
- The files under **`/var`** are usually stored in another file system.
- The files under **`/tmp`** are usually stored in yet another file system.

The **`/`** (root) file system is a special file system that incorporates the files under several directories including **`/etc`**, **`/dev`**, **`/sbin`**, and possibly others.

The Benefits of Configuring Multiple File Systems

Although all of your files and directories could be stored as part of the root file system, separating subtrees of the file hierarchy into several distinct file systems offers several advantages:

© 2004 Hewlett-Packard Development Company, L.P.

- The administrator can allocate a fixed amount of disk space to each file system to ensure that no single file system is allowed to monopolize an entire disk. The administrator might, for instance, allocate 100 MB to the **/tmp** file system. This ensures that temporary files under **/tmp** can use at most 100 MB of disk space on your system; remaining disk space could be preserved for other file systems.

- Each file system may be tuned independently. There are a number of parameters associated with each file system that can significantly affect system performance. It may be beneficial to optimize some file systems for storage of large files, while others are optimized for storage of smaller files.

- File system maintenance tasks may be performed on one file system, while other file systems remain accessible to your users.

Mounting File Systems

Each file system contains its own **root directory**. The root directory has no name of its own. At boot time, the root file system (typically the file system in **/dev/vg00/lvol3**) is mounted at location **/**. Mounting is the association of a *name* in the file hierarchy with a *device*. For example, if your root disk is on the device represented by **/dev/dsk/c0t6d0**, the name **/** is associated with that device.

Once the root file system has been mounted, other file systems can be mounted. The other file systems are mounted on directories from previously mounted file systems. These directories are known as **mount points**. As long as the file system is mounted the mount point directory's name is associated with the file system's device. If the file system is unmounted, the connection is broken. The system maintains a table, known as a *mount table* that maps names to device files. It is only necessary to keep the mount points in this map. The location of other files can be derived from there.

In the example shown on the slide, there is a disk that has been divided into two logical volumes. Each logical volume has its own device file. A file system has been created in each logical volume. The file system in the first logical volume is the root file system. At boot time it is mounted at **/**. In other words, the name **/** is associated with the first logical volume. In the root file system there is an empty directory named **usr**. The second file system is mounted on **/usr**. In other words, the name **/usr** is associated with the second logical volume, as long as it is mounted. If the second file system were unmounted the name **/usr** would again refer to the directory in the root file system.

The entire file hierarchy is built in this manner. The system administrator determines which portions of the hierarchy should be broken into separate file systems.

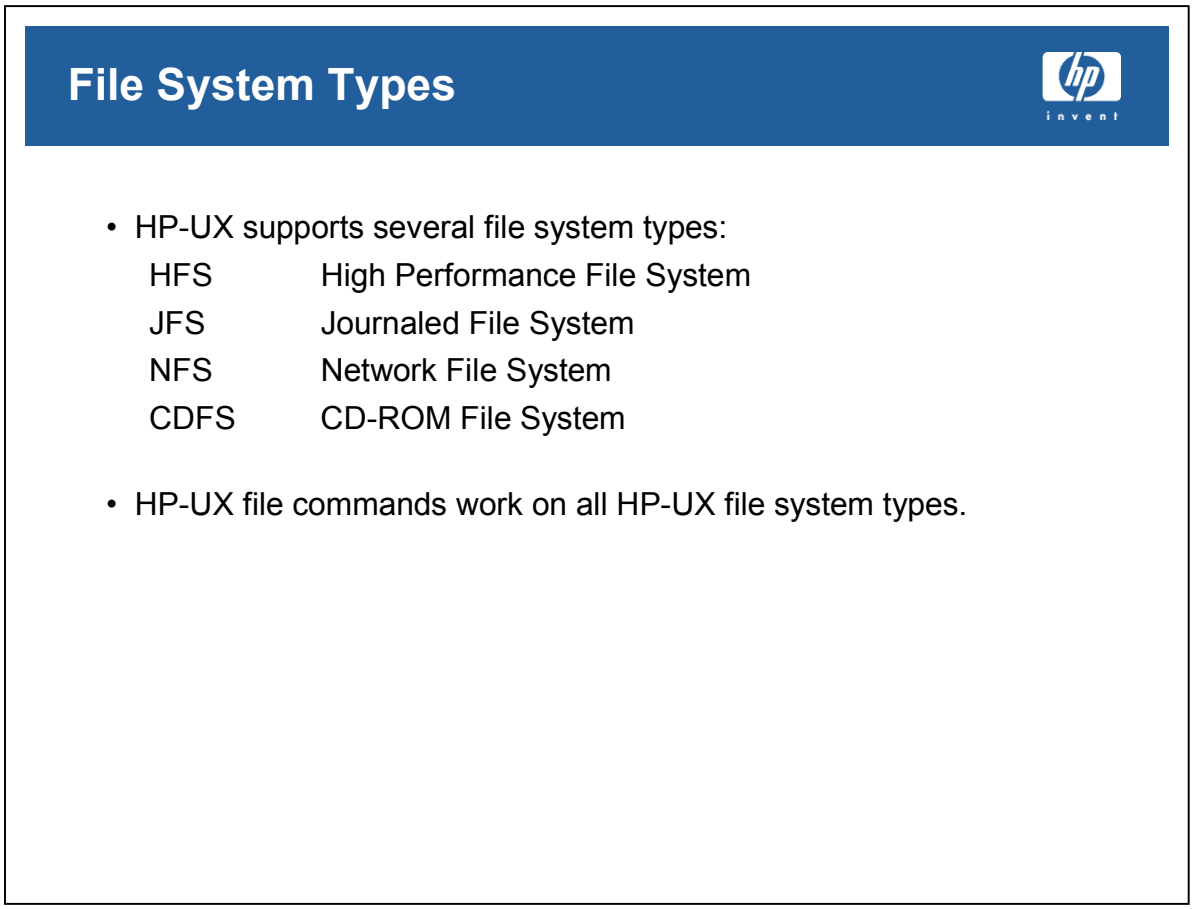
Viewing the Mounted File Systems

Two commands allow you to view a list of your currently mounted file systems:

```
# mount -v    # reports which file systems are mounted where
# bdf         # also reports file system sizes, and other info
```


© 2004 Hewlett-Packard Development Company, L.P.

8–2. SLIDE: File System Types



Student Notes

HP-UX supports several different file system types. The notes below briefly describe some of the features of the most common types of file systems.

High-Performance File System (HFS)

HFS represents HP-UX's standard implementation of the UNIX File System (UFS). HFS file systems reside on mass storage devices, usually hard disk drives. Prior to HP-UX Release 10.01 this was HP's only disk-based file system. HP's long term strategy is for JFS to become the default and for HFS to continue to exist for compatibility.

Journaled File Systems (JFS)

The HP-UX Journaled File System, also known as Veritas File System (VxFS), is an extent-based journaling file system which offers fast file system recovery and on-line features such as on-line backup, on-line resizing and on-line reorganization. An intent log contains recent file system data structure updates. After a failure the system checks the intent log and performs the required rollback or roll forward.

There are two JFS products, base and online. The base JFS file system has the fast recovery feature and is included in all 10.01 and later systems. Online JFS, also referred to as

© 2004 Hewlett-Packard Development Company, L.P.

Advanced VxFS is an optional product that adds extensions to JFS. It offers these additional capabilities:

- online defragmentation and reorganization
- online expansion and contraction of file system size
- online backup

Network File Systems (NFS)

NFS allows many systems to share the same files by using a client/server approach. Since access techniques are transparent, remote file access appears similar to local file access. Both JFS and HFS file systems can be shared with other systems using NFS. Network File System (NFS) provides transparent access to files from anywhere on the network. An NFS server makes a directory available to other hosts on the network by exporting the file system. An NFS client provides access to the NFS server's directory by mounting the file system. To users on the NFS client, the file system looks like one of the local file systems.

CD-ROM File Systems (CDFS)

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. CD-ROMs come in a variety of formats.

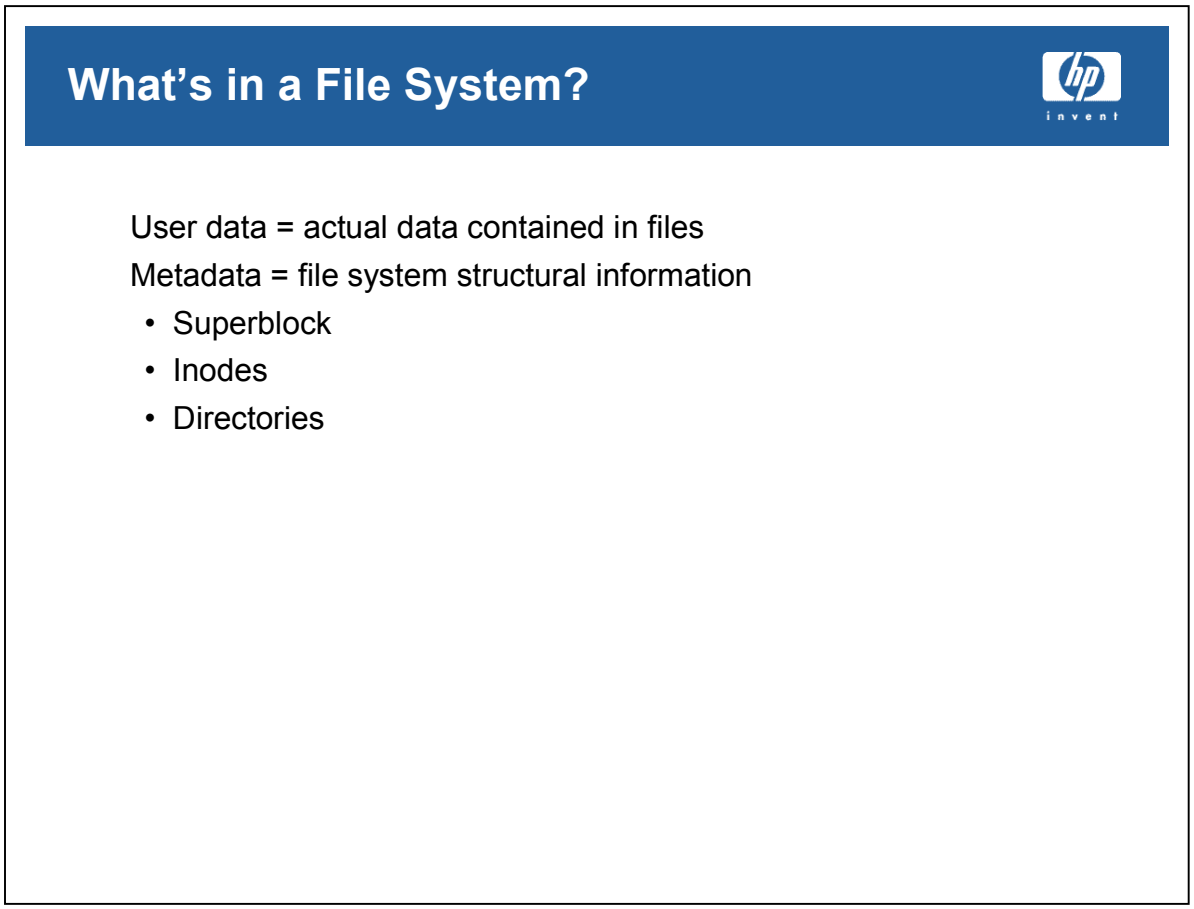
Mixing and Matching File Systems

All file systems on your machine may be the same type, but more typically, a machine's file system hierarchy is comprised of several different file system types. The kernel, for example, must reside in an "HFS" file system, so **`/stand`** is always HFS. However, since JFS file systems offer more flexibility and greater reliability, other file systems on your machine may be JFS. Fortunately for your users, the same file system navigation commands (such as **`cd`**, **`cp`**, and **`mv`**) work across all file system types. You can determine what types of file systems you have on your system with two commands:

```
# mount -v                  # what types of file systems are currently mounted?
# fstyp /dev/vg00/rlvol1    # what type of file system is in /dev/vg00/lvol1?
```


© 2004 Hewlett-Packard Development Company, L.P.

8–3. SLIDE: What's in a File System?



Student Notes

Disk space allocated to a file system, regardless of the file system type, is subdivided into multiple file system blocks. The blocks in a file system may be used for two different purposes.

Some of the blocks in a file system store the actual data contained in users' files. These data blocks account for the majority of the blocks in most file systems.

However, some blocks in every file system store the file system's metadata. A file system's metadata describes the structure of the file system. Some of the metadata structures that are common to most file system types are described below.

Superblock — Every file system has a superblock that contains general information about the file system. The superblock identifies the file system type, size, and status, and contains pointers to all of the other file system metadata structures. Since the superblock contains such critical information, HP-UX maintains multiple redundant copies of the superblock in every file system.

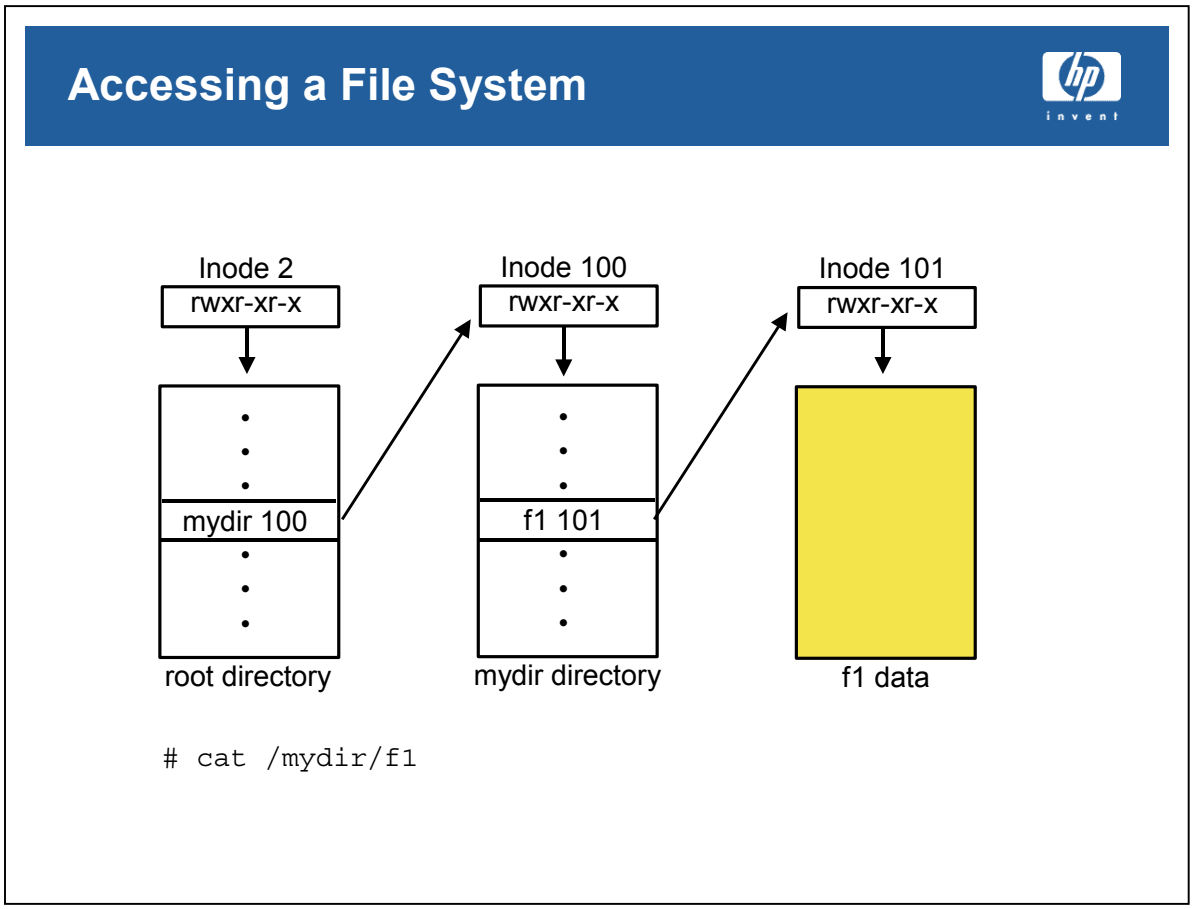
© 2004 Hewlett-Packard Development Company, L.P.

Inodes	Every file has an associated inode containing the attributes of the file. The inode identifies the file's type, permissions, owner, group, and size. A file's inode also contains pointers to the data blocks associated with the file. Each inode is identified by a unique inode number within the file system.
Directories	Users and applications generally reference files by name, not by inode number. The purpose of a directory is to make a connection between file names and their associated inode numbers.

The output and error messages from the HP-UX file system management utilities frequently refer to these metadata structures. A conceptual understanding of these structures and terms will contribute greatly to your success as a system administrator. The remainder of this module examines HFS and JFS metadata structures in somewhat greater detail.

© 2004 Hewlett-Packard Development Company, L.P.

8–4. SLIDE: Accessing a File System



Student Notes

The previous slide introduced several file system components: the superblock, directories, inodes, and data blocks. How are these structures used when users access a file or directory? Consider the example on the slide, which shows how a user might access a file called **`/mydir/f1`**.

First, HP-UX must find the **`/`** directory. The root directory of any file system is always found at inode number 2. HP-UX checks the permissions recorded in inode number 2 to ensure that the user has access to the / directory. The inode also contains pointers to the data blocks for the **`/`** directory.

A directory's data blocks contain a directory entry for each file or subdirectory within the directory. Each directory entry contains the name of a file or subdirectory and the inode associated with that file or subdirectory. HP-UX searches for the **`mydir`** directory entry in **`/`**.

The **`mydir`** directory entry identifies the inode associated with **`mydir`**. In the example on the slide, **`mydir's`** inode number is 100. HP-UX checks inode number 100 to determine if the user should be granted access to **`mydir`**.

© 2004 Hewlett-Packard Development Company, L.P.

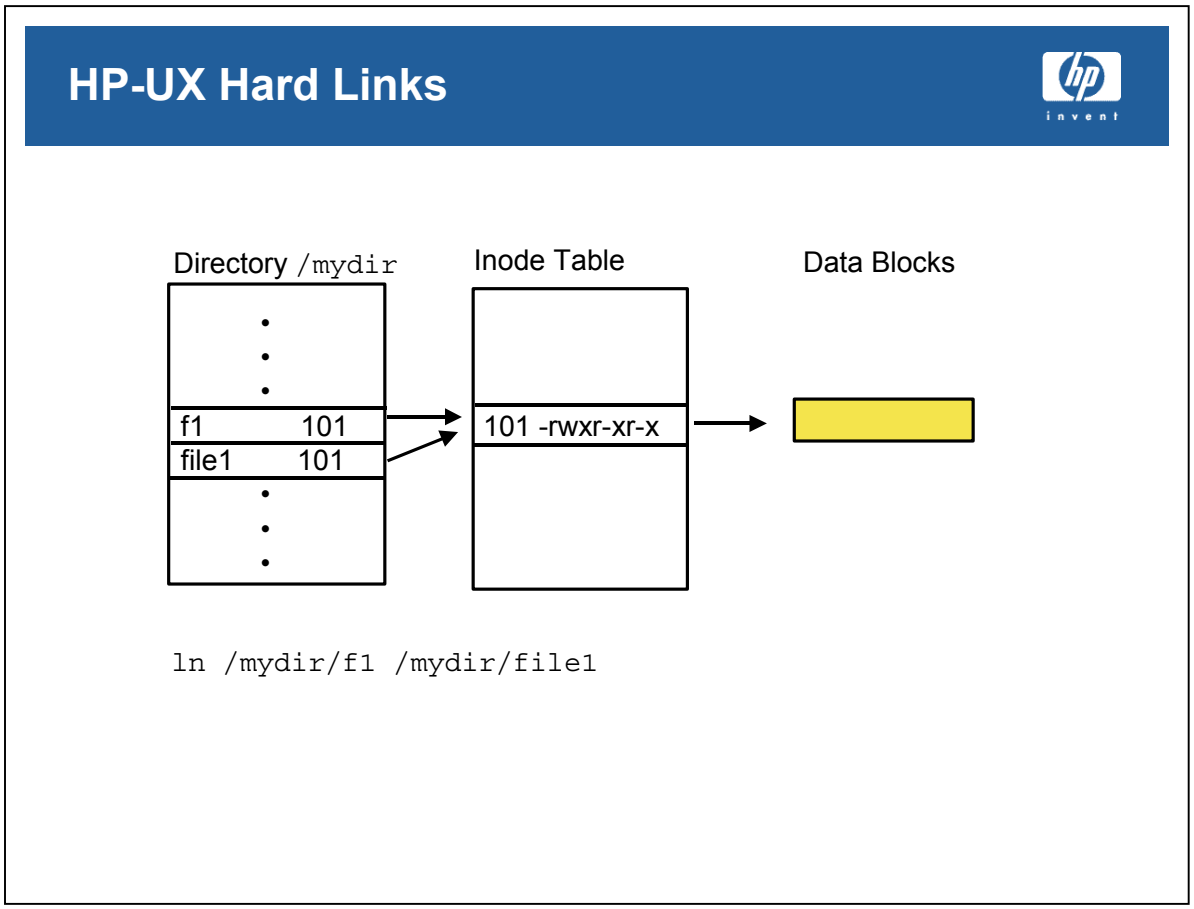
Next, HP-UX follows the pointers in inode 100 to the data blocks for **`mydir`**. These data blocks should contain a directory entry for file **`f1`**.

The **`f1`** directory entry identifies the inode associated with **`f1`**. In the example on the slide, **`f1`**'s attributes are recorded in inode number 101. HP-UX checks inode number 101 to ensure that the user has access to the file, then follows the pointers in the inode to **`f1`**'s data blocks.

Having found the data blocks for **`/mydir/f1`**, HP-UX displays the contents of those data blocks on **`stdout`**.

© 2004 Hewlett-Packard Development Company, L.P.

8–5. SLIDE: HP-UX Hard Links



Student Notes

Although most inodes are associated with exactly one directory entry, hard links make it possible to associate multiple directory entries with a single inode. This, in effect, allows your users to reference a single file via several different file names.

The example on the slide shows a file **`/mydir/f1`** that has been hard linked to **`/mydir/file1`**. Both names now reference the same inode, and thus share the same permissions, owner, and time stamp.

Since both file names reference the same inode, they also both ultimately reference the same data blocks. Changes made to **`f1`** will be reflected in **`file1`**, and vice versa. **`f1`** and **`file1`** are essentially the same file! Oftentimes it is useful to associate multiple file names with a single file in this manner.

A hard link may be created with the **`ln`** command. The first argument identifies the file name of the existing file, and the second identifies the name of the new link:

```
# ln /mydir/f1 /mydir/file1     Creates a link to f1
# ll /mydir                     Shows the number of links to each file
```


© 2004 Hewlett-Packard Development Company, L.P.

Creating a hard link creates a new directory entry for the new link, and increments the link count field in the inode. The second field in the output from the **ll** command shows the number of links to each file.

Hard links are oftentimes used to associate multiple file names with a single device file. For instance, some administrators do the following:

```
# ln /dev/rmt/c0t0d0BEST /dev/tape
```

After creating this link, users can access the c0t0d0 tape drive using the intuitive name **/dev/tape**, rather than the cryptic default name **/dev/rmt/c0t0d0BEST**. Be aware of two hard link limitations:

- Hard links cannot cross file system boundaries.
- Hard links cannot link directories.

Questions

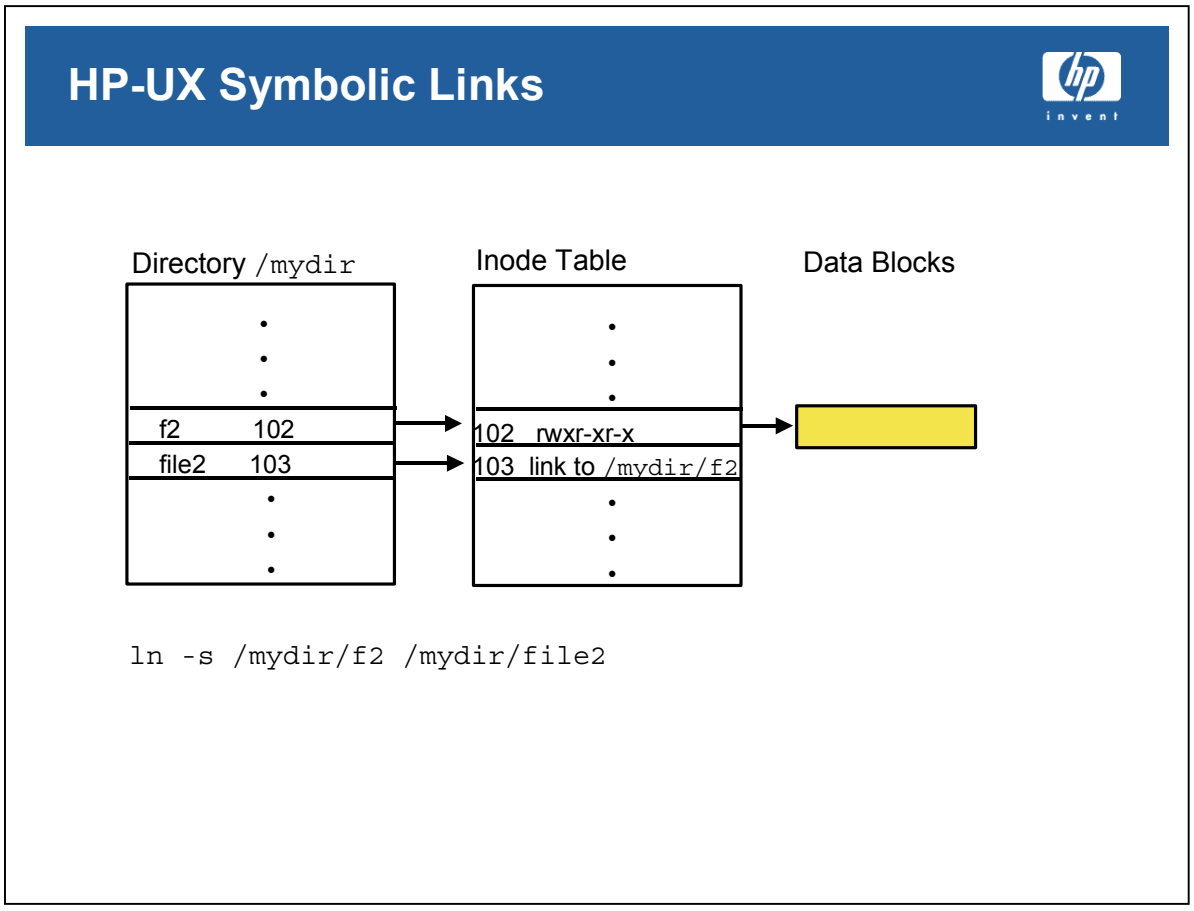
1. Why isn't it possible to link across file system boundaries?

2. How would **/mydir/file1** be affected if you were to remove **/mydir/f1**?

Answers are provided in the solutions appendix.

© 2004 Hewlett-Packard Development Company, L.P.

8–6. SLIDE: HP-UX Symbolic Links



Student Notes

Symbolic links, like hard links, make it possible to associate multiple file names with a single file. Unlike hard links, however, symbolic links:

- can cross file system boundaries
- can link directories

In the example on the slide, **/mydir/file2** is a symbolic link to /**mydir/f2**. **f2** and **file2** have distinct directory entries and inodes. However, as shown on the slide, **/mydir/file2** is nothing more than a pointer to **/mydir/f2**! Accessing **/mydir/file2** yields the same data one would see when accessing **/mydir/f2**.

Symbolic links are particularly useful when you must move files from one file system to another, but still wish to be able to use the file's original path name. At version 9 of HP-UX, system executables were stored in the **/bin** directory. At HP-UX 10, many operating system executables were moved to **/usr/bin**. However, a symbolic link exists from **/bin** to **/usr/bin** so users and applications can still use the version 9 path names. This is just one situation where symbolic links are commonly used.

© 2004 Hewlett-Packard Development Company, L.P.

Use the **ln** command with **-s** to create a symbolic link. The first argument identifies the existing file that you wish to link to. Additional arguments specify the path names of the symbolic links you wish to create to the existing file.

```
# ln -s /mydir/f2 /mydir/file2
```

The **ll** command identifies symbolic links with an **l** in the first character position. Also, the file name field in the **ll** output identifies the file to which a symbolic link leads.

NOTE:	In the example above, removing **/mydir/f2** will not automatically remove the **/mydir/file2** link! After removing **/mydir/f2**, accessing the link will result in an error message.

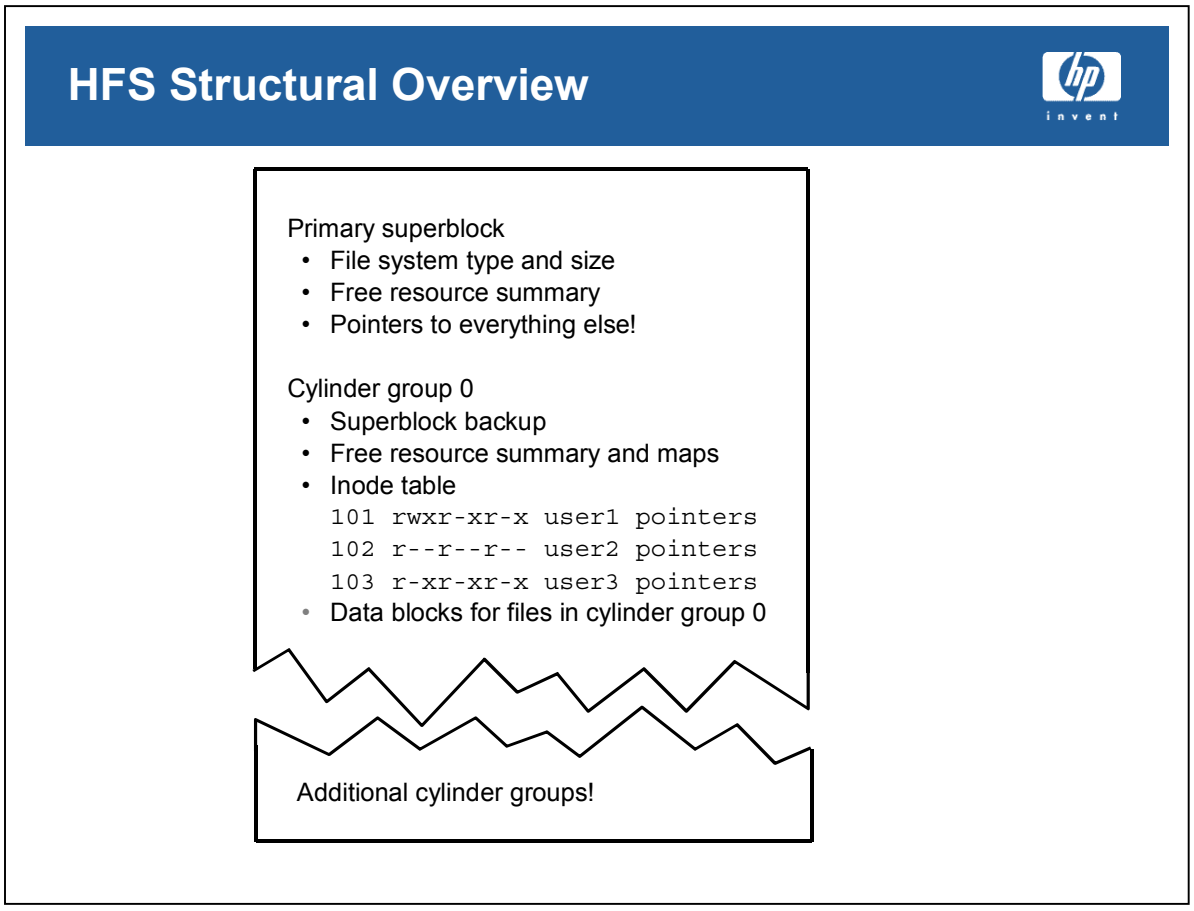
Question

1. Why is it possible to create symbolic links, but not hard links, across file system boundaries?

Answers are provided in the solutions appendix.

© 2004 Hewlett-Packard Development Company, L.P.

8–7. SLIDE: HFS Structural Overview



Student Notes

The last few slides introduced some of the structures found in an HP-UX file system and how those structures are used to find and access a file. Although this procedure is fairly consistent across file system types, there are some important differences between HFS and JFS file systems. The remainder of this module highlights some of the significant differences between HFS and JFS file systems.

The HFS Superblock

The second 8 KB of every HFS file system contains the file system's primary superblock. Recall that the superblock contains general information about the file system's size, usage, status, and configuration. Perhaps most importantly, the superblock contains pointers to all the other HFS metadata structures in the file system.

Because the superblock is so critical, HFS maintains multiple redundant superblock copies. The backup HFS superblock locations are recorded in the **`/var/adm/sbtab`** file. A damaged HFS superblock may be repaired using the information in these backup superblocks.

© 2004 Hewlett-Packard Development Company, L.P.

HFS Cylinder Groups

Remaining space in an HFS file system is divided among one or more cylinder groups. Larger HFS file systems have more cylinder groups, and smaller file systems have fewer cylinder groups.

Each HFS cylinder group contains a redundant superblock, a portion of the file system's inodes, and a portion of the file system's data blocks.

HFS Inodes

Every file and directory must have an inode. The inode identifies the file's permissions, owner, group, and other attributes.

When you create an HFS file system, you must specify how many inodes you wish to create. The inodes are stored in the new file system's inode table. In an HFS file system, it is possible to run out of inodes in the inode table. Without a free inode, you can't create new files and directories, even if there are free data blocks remaining! You can add additional inodes in one of two ways:

- Allocate additional disk space to the file system. Extending the file system also adds additional inodes in the inode table.

- Backup your data, rebuild the file system with more inodes, and restore your data from tape.

© 2004 Hewlett-Packard Development Company, L.P.

8–8. CHALK TALK: HFS File System Updates

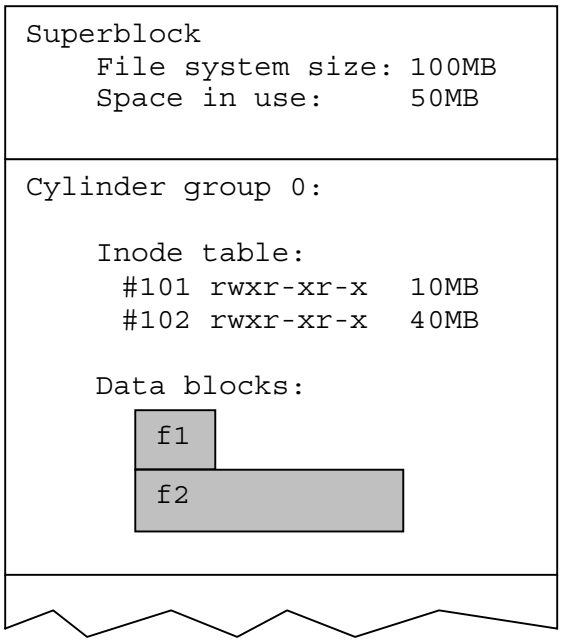
What happens when a file is removed from an HFS file system?

The next few pages are designed to solidify your understanding of the HFS file system structures by describing what happens to those structures when a file is removed.

The diagram at right shows a simple HFS file system containing two files: f1 is a 10MB file, f2 is a 40MB file. The metadata structures in the file system reflect the existence of these two files. The superblock shows 50MB of space in use, the inode table contains inodes for both files, and data blocks appear to have been allocated to the files from cylinder group 0.

When a file system is mounted in HP-UX, the critical file system metadata is copied into memory to improve run-time performance. This "in-core" metadata is also shown in the diagram.

Now the question: what happens to these structures if a user chooses to remove file f1? Turn to the next page to find out!



© 2004 Hewlett-Packard Development Company, L.P.

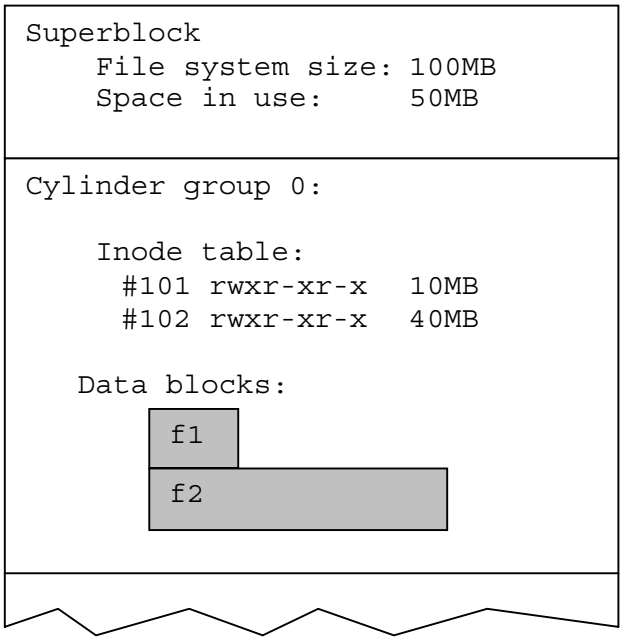
Step 1: HFS updates the in-core metadata

Initially, HFS simply updates the in-core copy of the file system superblock, and inode table, and directory data block. Thus, HFS marks inode 101 as "free," and decrements the space-in-use register in the in-core superblock by 10 MB (since f1 was a 10-MB file).

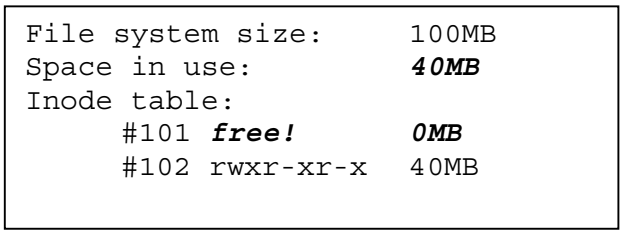
Note that at this point, the on-disk metadata structures remain unchanged!

Question:

At this point, what impact would a system crash have on the file system? Would the file system structures be left in a consistent state?



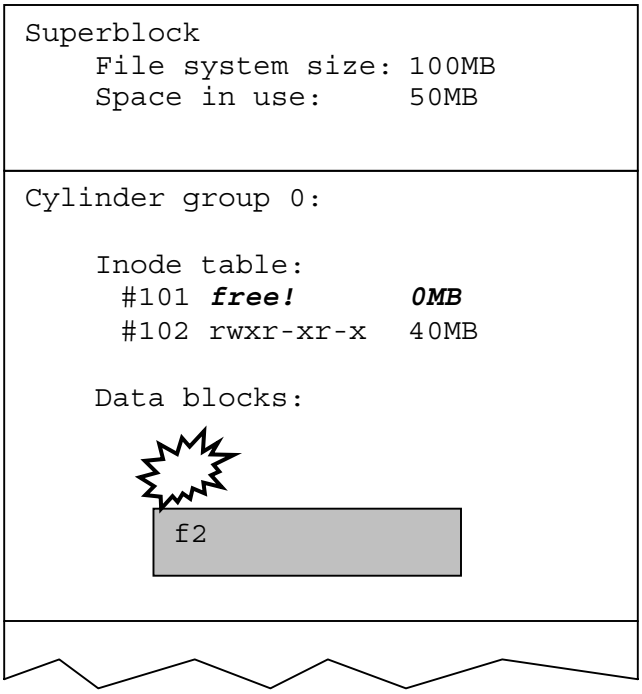
Metadata in-core at start:



© 2004 Hewlett-Packard Development Company, L.P.

Step 2: HFS deallocates f1's inode and data blocks

Next, HFS marks the inode and data blocks that were previously occupied by file f1 as "free". Note, however, that the superblock has not yet been updated!



Question:

At this point, what impact would a system crash have on the file system? Would the file system structures be left in a consistent state?

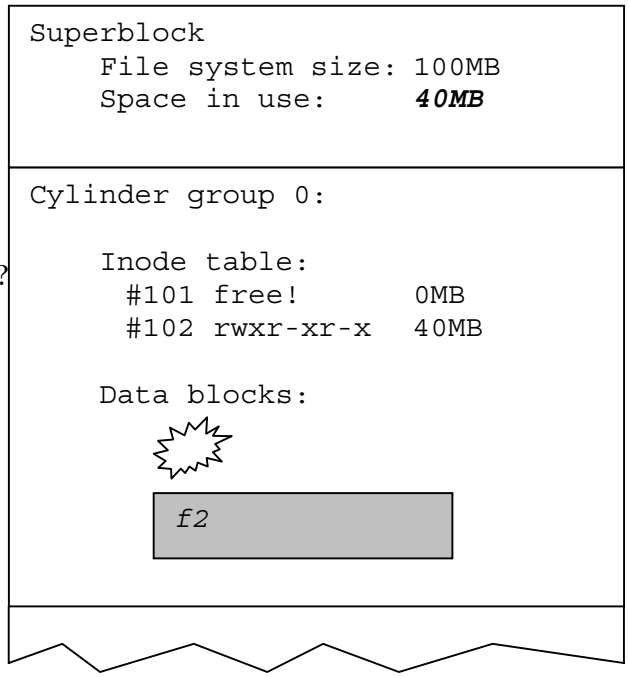
Metadata in-core after step 2:

```
File system size:     100MB
Space in use:         40MB
Inode table:
      #101 free!      0MB
      #102 rwxr-xr-x  40MB
```

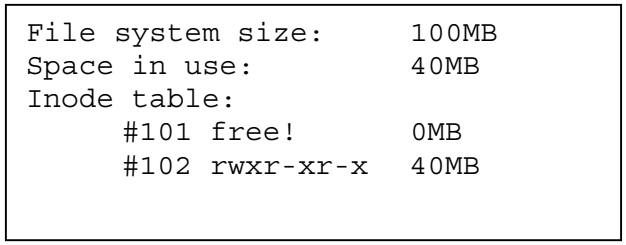

© 2004 Hewlett-Packard Development Company, L.P.

Step 3: HFS updates the superblock

Eventually, HFS updates the file system's superblock as well, thus bringing the file system back to a consistent state.



Metadata in-core after step 3:



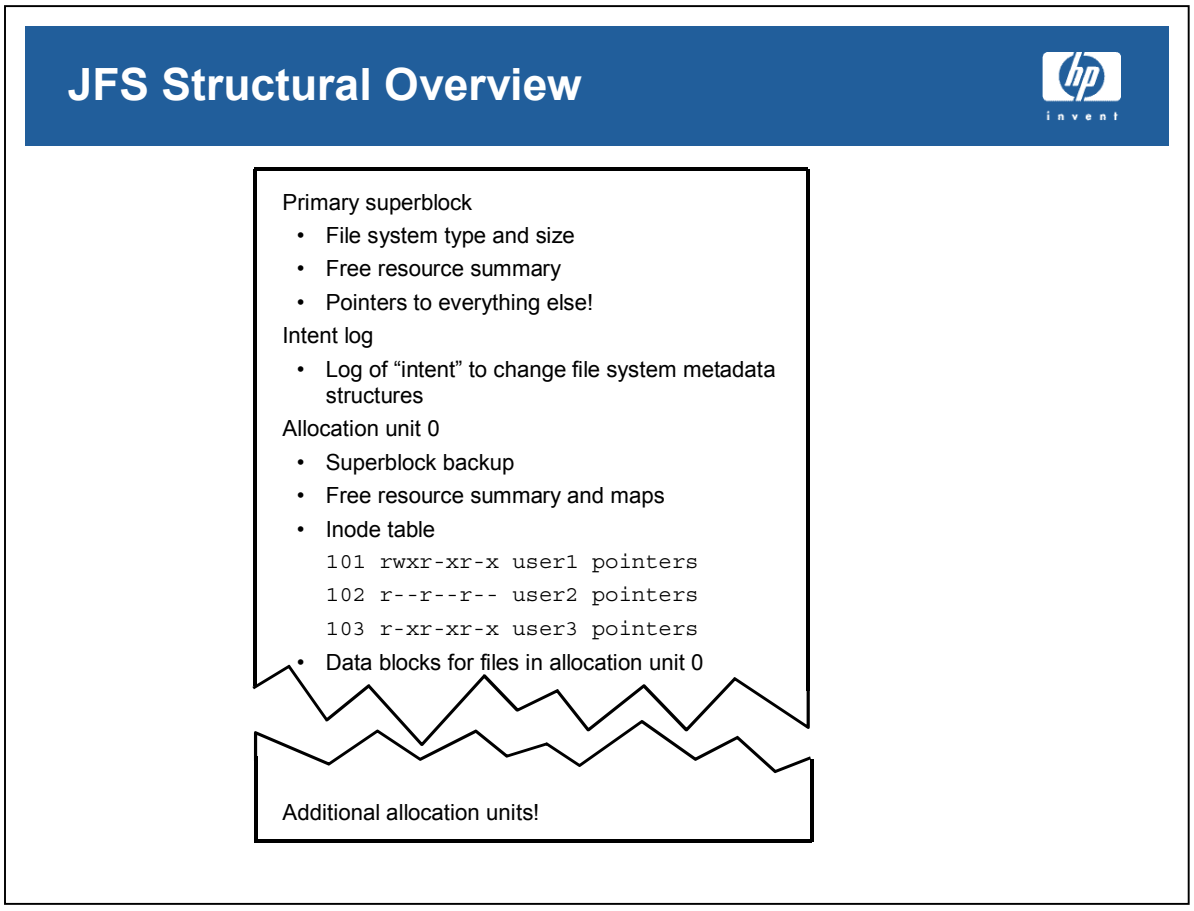
Questions

1. A system crash can cause serious problems in an HFS file system. Why?

2. Conceptually, what must be done to every HFS file system after a system crash?

© 2004 Hewlett-Packard Development Company, L.P.

8–9. SLIDE: JFS Structural Overview



Student Notes

The Journal File System (also known as the Veritas or VxFS file system) was designed to address many of the limitations of HFS. This slide introduces some of the structures underlying JFS, and the next few slides explore the JFS structures in more detail.

JFS Superblocks

Like HFS, JFS has a superblock at the beginning of the file system that describes the file system's size, status, and configuration. The information contained in the superblock is critical! If the primary superblock is corrupted, it may be reconstructed using redundant superblock backups that JFS automatically maintains. Unlike HFS, JFS can find its redundant superblocks automatically; thus, JFS backup superblocks are not listed in `/var/adm/sbtab`.

The JFS Intent Log

A key advantage of JFS is that all metadata changes are recorded in an intent log. This logging mechanism ensures the integrity of the file system, and allows the file system to be recovered quickly in the event of a system crash.

© 2004 Hewlett-Packard Development Company, L.P.

A record of all metadata updates for a given transaction is written with one disk access. This ensures that all the metadata updates are *atomic*. In other words the file system will reflect either a before or after image of an operation, but never an intermediate image. The actual metadata updates to the superblock, inode, maps, and directory can be delayed until a **`sync`** is issued. Once all the metadata updates associated with a transaction have been completed, a *done* record is written to the intent log.

If the system were to crash, the file system can quickly be recovered by applying all changes in the JFS Intent Log. Since only entire transactions are logged, there is no risk of a file change only being partially updated, only some metadata updates related to the transaction being logged, and other metadata updates related to the same transaction not being logged). The logging of only entire transactions prevents the file system from being out-of-sync due to some transaction occurring in the middle of the crash. Either the whole transaction is lost or the whole transaction is logged to the intent log. This allows the JFS intent log to be used in a recovery situation.

After a system crash, the file system metadata can be returned to a consistent state by simply replaying the transactions in the intent log. This can be completed within just 10 to 20 seconds!

JFS Allocation Units

Disk space allocated to a JFS file system is subdivided into one or more 32-MB allocation units. Each allocation unit contains the data blocks. Larger JFS file systems have more allocation units, and smaller JFS file systems have fewer allocation units.

JFS Inodes

Like HFS, JFS stores file and directory attributes in inodes. Unlike HFS, however, JFS creates inodes dynamically. JFS creates several inodes initially, then may create additional inodes as needed. As long as there are free data blocks in the file system, JFS can create more inodes.

Questions

1. What metadata structures does JFS have in common with HFS?

2. What metadata structures and features distinguish JFS from HFS?

Answers are provided in the solutions appendix.

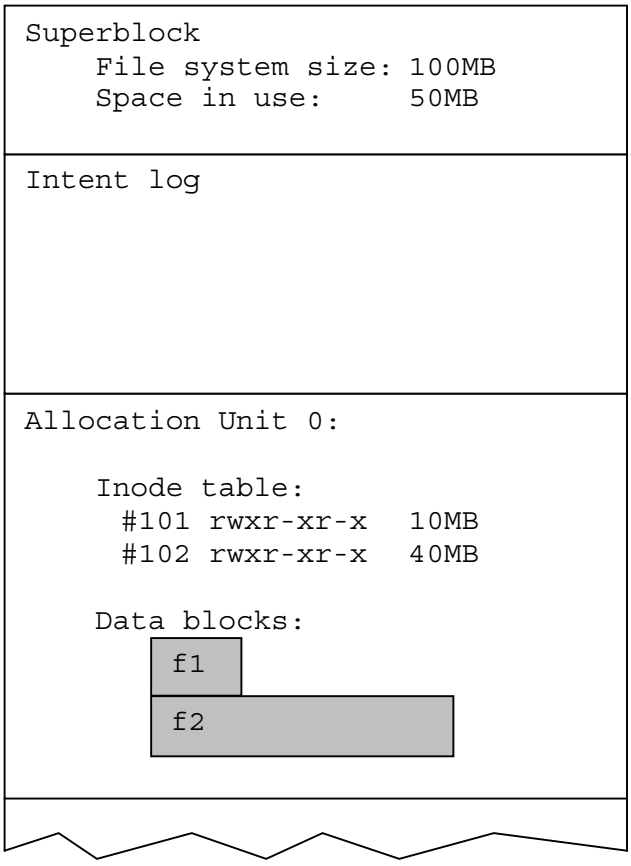
© 2004 Hewlett-Packard Development Company, L.P.

8–10. CHALK TALK: JFS File System Updates

What happens when a file is removed from a JFS file system?

The purpose of this chalk talk is to compare the process HP-UX uses to update a JFS file system to the process HP-UX uses to update a HFS file system. Although the two file systems are similar, they also have some significant differences.

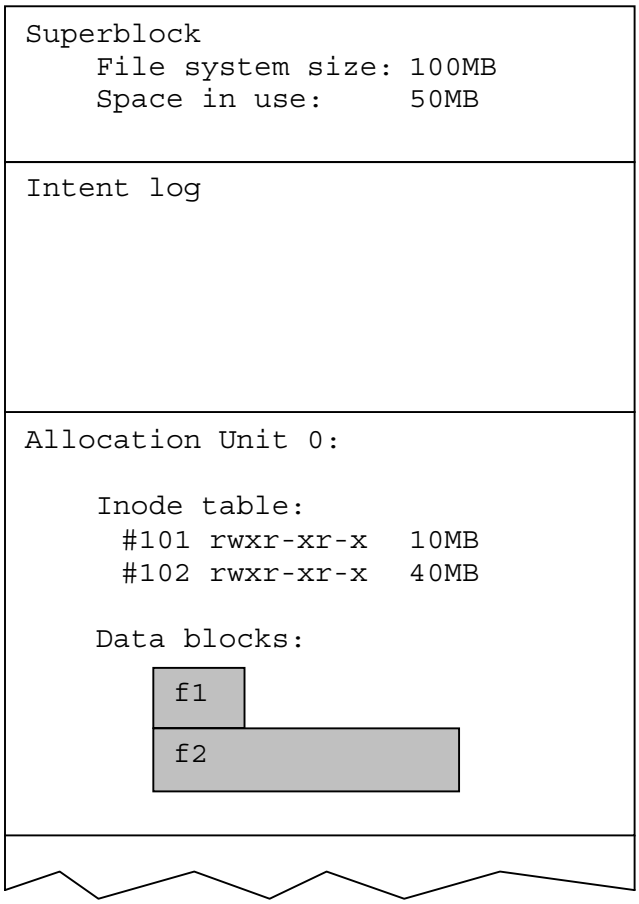
The diagram at right shows a JFS file system containing two files: f1 (10MB) and f2 (40MB). Let's see what happens when file f1 is removed...



© 2004 Hewlett-Packard Development Company, L.P.

Step 1: JFS updates the in-core metadata

JFS, like HFS, starts by updating the in-core metadata to reflect the fact that file f1 has been removed. At this point, JFS does not modify the metadata on disk.



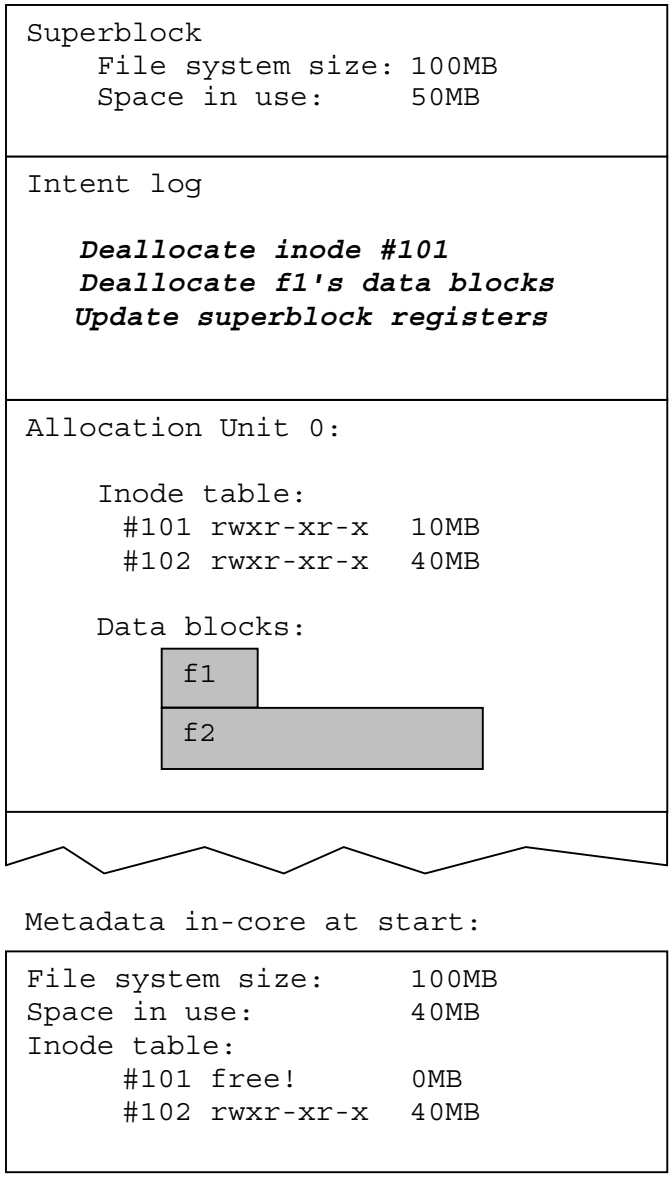
Metadata in-core at start:

```
File system size:     100MB
Space in use:         40MB
Inode table:
      #101 free!      0MB
      #102 rwxr-xr-x  40MB
```


© 2004 Hewlett-Packard Development Company, L.P.

Step 2: JFS creates an intent log entry

Before JFS actually modifies on-disk metadata structures, it records its *intent* to modify the metadata in an on-disk "intent log". Thus, if the update is interrupted by a system crash, JFS has some record of pending changes to the file system metadata. This greatly simplifies recovery from a system crash.



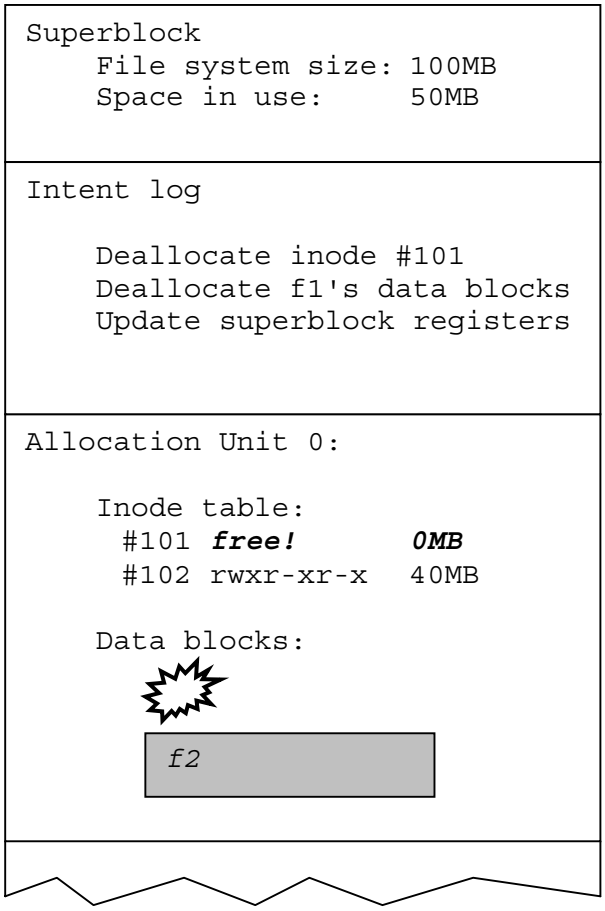
© 2004 Hewlett-Packard Development Company, L.P.

Step 3: JFS de-allocates f1's inode and blocks

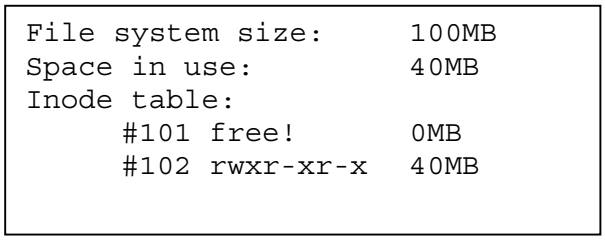
After the intent log record has been written, JFS makes the required changes to the on-disk meta-data. Removing file f1, for example, requires JFS to de-allocate f1' s inode and data blocks.

Question

At this point, what impact would a system crash have on the file system? Would the file system be left in a consistent state?



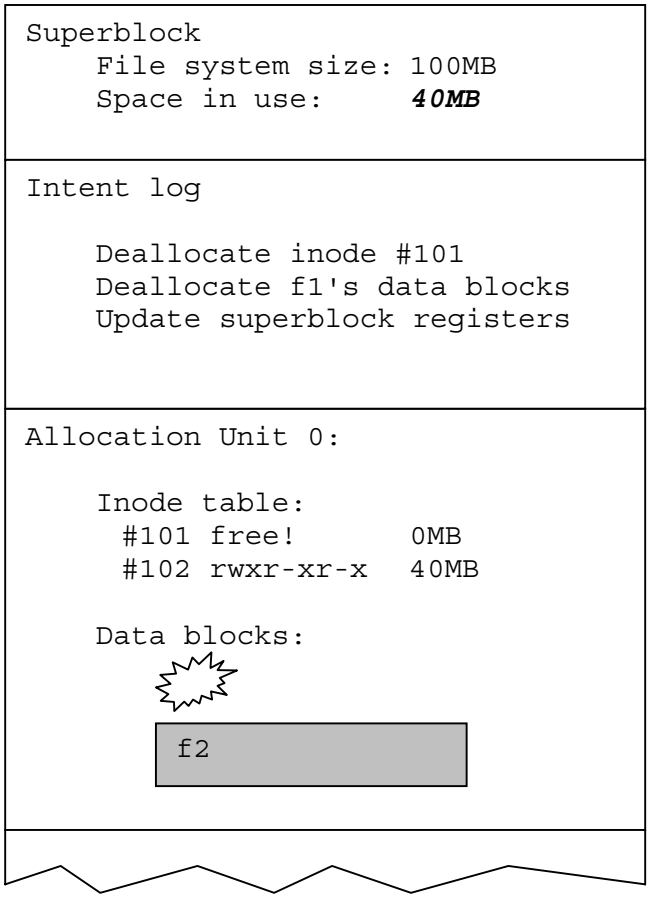
Metadata in-core at start:



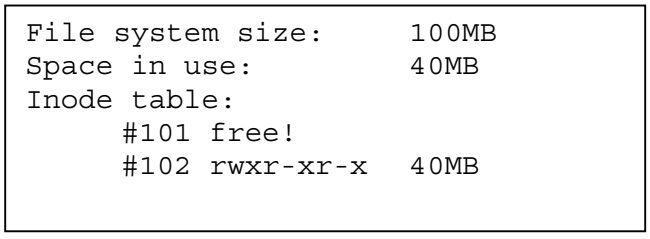
© 2004 Hewlett-Packard Development Company, L.P.

Step 4: JFS updates the superblock

Having updated the inode table and free data block maps, JFS must update the appropriate superblock registers.



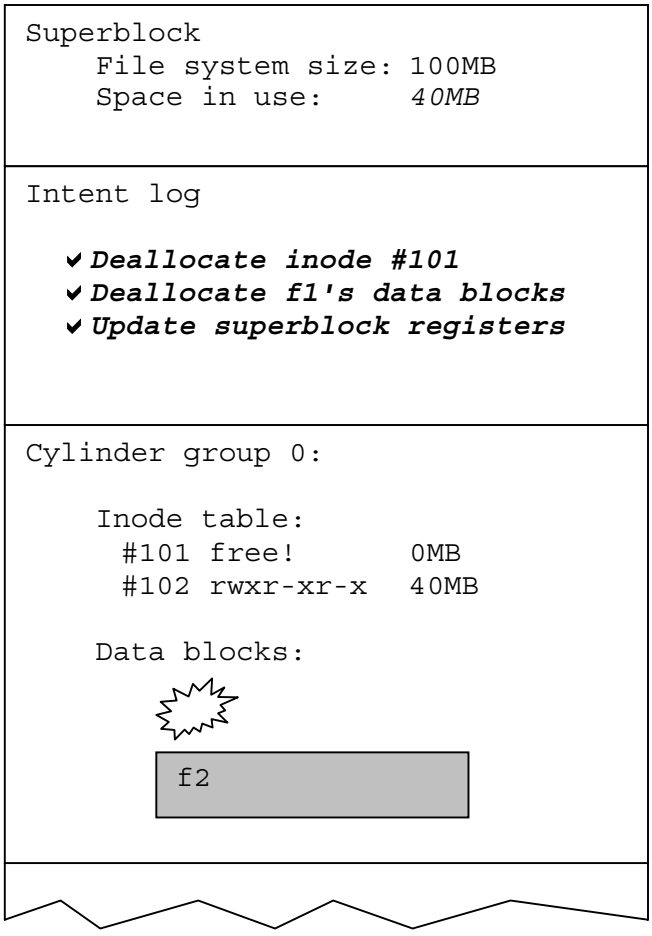
Metadata in-core at start:



© 2004 Hewlett-Packard Development Company, L.P.

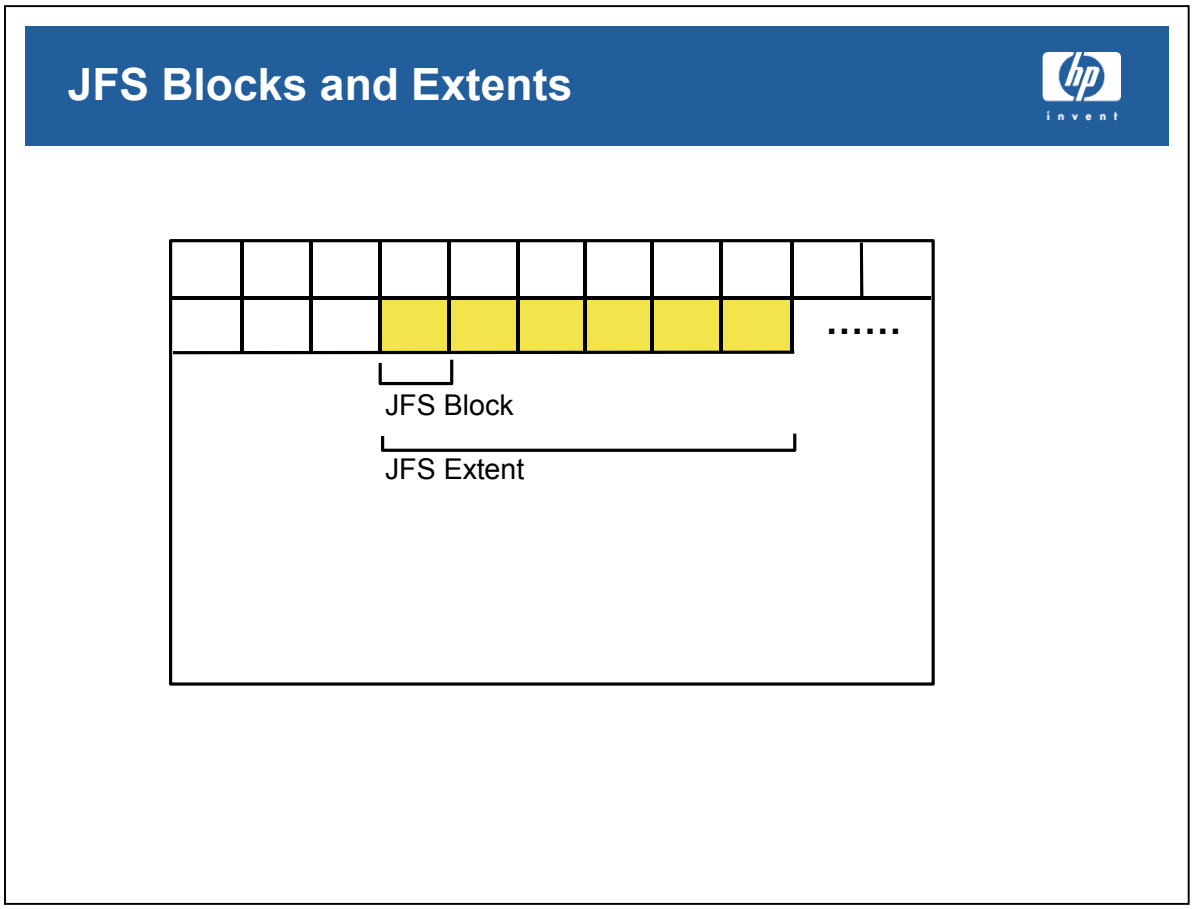
Step 5: JFS marks the intent log entry "done"

As JFS completes changes to the file system metadata, it marks the appropriate intent log entries "done". After an intent log entry is marked "done", JFS reuses that entry's space for newly requested transactions.



© 2004 Hewlett-Packard Development Company, L.P.

8–11. SLIDE: JFS Blocks and Extents



Student Notes

In JFS, a block represents the smallest unit of disk space that can be allocated to a file. Although you can change the block size when creating a JFS file system, the default 1 KB block size is recommended in most situations.

When you create a file, JFS allocates an extent to the file. An extent is a contiguous group of blocks in the file system. As the file grows, JFS tries to increase the size of the extent if contiguous space is available. If contiguous space is not available, JFS uses another extent elsewhere in the file system. In order to optimize performance, JFS tries to allocate the next extent in the same allocation unit if possible.

Since a file's blocks are arranged contiguously on disk in a JFS file system, JFS can read large amounts of data into memory with a single I/O operation.

© 2004 Hewlett-Packard Development Company, L.P.

8–12. SLIDE: File System Comparison

File System Comparison

	HFS	JFS 3.0	JFS 3.3	JFS 3.5
Max File / File System Size?	128GB	1 TB	1 TB	2 TB
May be used for /stand?	Yes	No	No	Yes
Access Control Lists Available?	Yes	No	Yes	Yes
File System Tuning Available?	Yes	No	Yes	Yes
Fast File System Recovery?	No	Yes	Yes	Yes
Online Resizing Available?	No	Yes*	Yes*	Yes*
Defrag Utility Available?	No	Yes*	Yes*	Yes*
Online Backup Available?	No	Yes*	Yes*	Yes*

* Requires the JFS Online product

Student Notes

This chapter introduced the two disk-based file systems available in HP-UX. The slide above summarizes the differences between HFS and JFS, and also notes the differences between JFS versions. JFS product version 3.0 was the default for 10.20, 3.3 was the default for 11i v1.0, and 3.5 is the default for 11i v2.0. See the “Upgrading JFS File Systems” section below for more information.

Some JFS features in the table are followed by an asterisk (*). These features are only available with the "Online" JFS product. See the section below titled "Upgrading Base JFS Systems to Online JFS" for more information.

Max File / File System Size

HFS file systems now support a maximum file/file system size of 128 GB. JFS 3.5 supports file systems and files up to 2 TB in size for customers with very large databases.

Kernel Support

The HP-UX kernel loader on PARISC is unable to read JFS metadata structures. As a result, the **`/stand`** file system must be HFS on PARISC systems. All other file systems may be, and usually are, JFS.

© 2004 Hewlett-Packard Development Company, L.P.

The IPF kernel loader is able to read JFS metadata structures, so on Itanium systems the **/stand** file system is JFS.

Access Control List Support

Standard UNIX permissions only allow three sets of permissions on each file. One set of permissions determines what access rights are granted to the file's owner, one set of permissions determines what access rights are granted to the members of the file's group, and one set of permissions determines what access rights are granted to everyone else.

Access Control Lists make it possible to configure thirteen additional sets of permissions! HFS has supported a proprietary version of Access Control Lists for many years on HFS file systems (see the man pages for **chacl(1)** and **lsacl(1)**). HP's original JFS release didn't support ACLs, but JFS versions 3.3 and beyond do. The HFS and JFS ACL implementations are somewhat different. To learn about JFS ACLs, attend HP's Practical UNIX and Network Security course, or take a look at HP's "HP JFS 3.3 Access Control Lists" white paper on **http://docs.hp.com**..

File System Tuning

HFS supports a **tunefs** command that may be used to tune a variety of file system parameters on a per-file-system basis. JFS first introduced a similar command, **vxtunefs**, at version 3.3. **vxtunefs** is discussed in HP Education's Performance and Tuning class (H4262S) and VERITAS File System class (H7086S).

Fast File System Recovery

After a system crash, it can take minutes or even hours to repair an HFS file system's metadata structures. If an HFS file system's metadata can't be repaired after a crash, it sometimes becomes necessary to restore the entire file system from tape!

JFS metadata, on the other hand, can be repaired after a system crash in a matter of seconds using the JFS intent log. In 24x7 environments the JFS fast crash recovery functionality is invaluable.

Online Resizing

HFS file systems can't be extended while users are still accessing files in the file system; the Administrator must make the file system temporarily inaccessible via the **umount** command. HFS provides no mechanism for *reducing* a file system online *or* offline.

JFS makes it possible to extend, and even reduce, file systems that are still being accessed by users. Note, however, that this feature is only available to users of the Online JFS product; Base JFS users still must **umount** before their file systems can be extended.

Defrag Utility Available?

Over time, as users create, remove, and extend files, file systems tend to become "fragmented" with blocks and fragments scattered across the entire logical volume or disk. Eventually, fragmentation can lead to serious performance problems.

© 2004 Hewlett-Packard Development Company, L.P.

The Online JFS product includes a defragmentation utility that rearranges file system blocks to improve performance. No defragmentation utility is available for HFS.

Online Backup Available?

The Online JFS product includes a file system snapshot capability, which makes it possible to back up a file system, even while files and directories are being modified.

Upgrading HFS File Systems to JFS

In order to migrate an HFS file system to JFS on HP-UX 10.x, Administrators were forced to manually back up the entire file system to tape, overwrite the HFS metadata with JFS metadata structures, then restore the file system data from tape. The sample sequence of commands below might be used to convert the **/myfs2** file system on a 10.20 system to VxFS.

```
# tar -cv /myfs2
# umount /myfs2
# newfs -F vxfs /dev/vg01/rmyfs2
# mount /dev/vg01/myfs2 /myfs2
# tar -xv
```

JFS 3.3 and 3.5 now make it possible to convert HFS file systems to JFS more directly. The sample sequence of commands below might be used to convert the **/myfs2** file system on an 11i system to VxFS (the procedure is described in detail in the **vxfsconvert(1m)** man page.

```
# bdf /myfs2   Verify that the file system has at least 15% free before proceeding.
# umount /myfs2
# /sbin/fs/vxfs/vxfsconvert /dev/vg01/rmyfs2
# fsck -F vxfs -y -o full /dev/vg01/rmyfs2
# mount /dev/vg01/myfs2 /myfs2
```

Upgrading Base JFS Systems to Online JFS

Each version of JFS is available in two flavors. Base JFS provides fast crash recovery functionality and comes standard with all versions of HP-UX.

Online JFS is an extra product that makes it possible to extend, reduce, defragment, back up, and tune JFS file systems on mission critical, 24x7 systems. Online JFS is included in the HP-UX 11i "Enterprise" and "Mission Critical" bundles but must be purchased separately for all other versions of HP-UX.

To determine which version of JFS you have installed, use the **swlist** command:

```
# swlist -l product | grep -i vxfs
```

If you have purchased an Online JFS license, you can install the Online JFS product from your Applications CD with the **swinstall** command:

```
# swinstall
```


© 2004 Hewlett-Packard Development Company, L.P.

After installing the software, you can begin using the Online features immediately; no changes are required to the already existing JFS file systems.

Upgrading JFS File Systems

Each major HP-UX release has introduced a new version of JFS, supporting new features and functions. You must consider two issues when determining which of these features will be available on your system: (a) the version number on the JFS product installed on your system and (b) the file system metadata version associated with your file system. The table below summarizes the JFS versions available for various HP-UX OS versions.

HP-UX OS Version	Default JFS Product Version	Default JFS Layout Version	Comments
10.20	2.0	2	-
11.00	3.0	3	Optional upgrade to JFS 3.3 possible
11i v1.0	3.3	4	
11i v2.0	3.5	4	-

Two steps are required to upgrade HP-UX to a newer version of JFS. First, you must download the new JFS software from **`http://software.hp.com`** and install it with the **`swinstall`** command. The products are called **`JFS`** and **`OnlineJFS`**.

```
# swinstall
```

Next, the metadata structures in your existing file systems must be updated, too, via the **`vxupgrade`** command. This can be done while the file system is in use.

```
# vxupgrade -n 4 /myfs1
/myfs1: vxfs file system version 4 layout
```

Confusingly, file systems created using JFS 3.3 and 3.5 software use the JFS file system layout version 4.0.

Determining Your File System Type and Version

You can determine a logical volume or disk's file system type with the **`fstyp`** command. If the file system is VxFS/JFS, the second line will display the JFS file system layout version number.

```
# fstyp -v /dev/vg01/rmyfs1
vxfs
version: 4
f_bsize: 8192
f_frsize: 1024
f_blocks: 143360
f_bfree: 47970
f_bavail: 45228
f_files: 16632
f_ffree: 11992
f_favail: 11992
f_fsid: 1073741827
f_basetype: vxfs
```


© 2004 Hewlett-Packard Development Company, L.P.

```
f_namemax: 254
f_magic: a501fcf5
f_featurebits: 0
f_flag: 0
f_fsindex: 7
f_size: 143360
```


© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 9 — Creating and Mounting File Systems

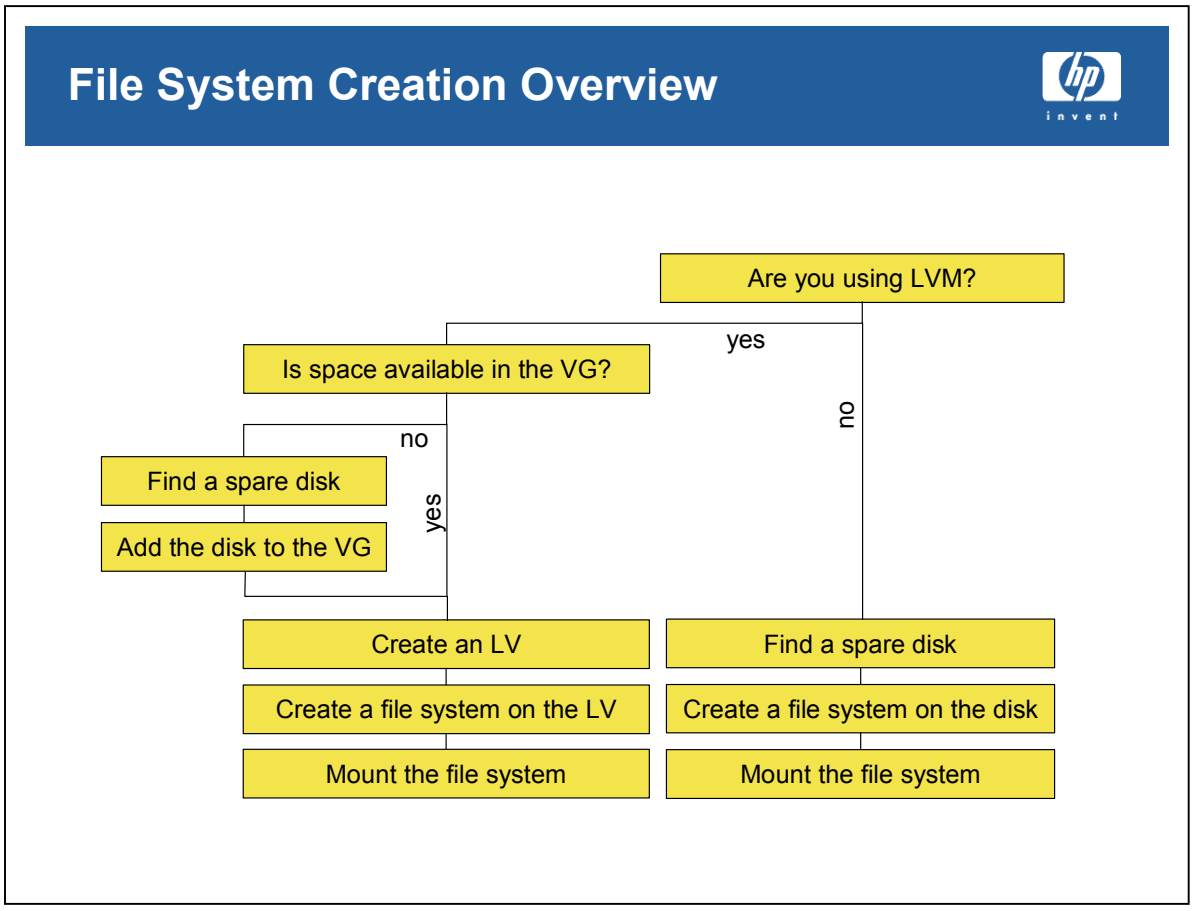
Objectives

Upon completion of this module, you will be able to do the following:

- Create HFS and JFS file systems from the command line and SAM.
- Mount and unmount HFS, JFS, CDFS, and LOFS file systems from the command line.
- Automatically mount file systems via `/etc/fstab`.

© 2004 Hewlett-Packard Development Company, L.P.

9–1. SLIDE: File System Creation Overview



Student Notes

You can expand your file hierarchy's overall capacity by creating new file systems and attaching (mounting) them on the existing file system tree. A file system can be created in either a logical volume or a whole disk. You can use either SAM or a sequence of HP-UX commands to create a file system.

You don't necessarily have to add a new disk drive to create a new file system. If you have free extents in any of your volume groups, you can simply create another logical volume, and then create the file system.

There are many reasons why you may want to add a new file system. Some reasons include:

- You anticipate that your current file system will soon reach maximum capacity.
- Your current file system has already reached maximum capacity.
- You wish to physically separate portions of a file system for a particular reason, such as different uses or different groups of users.

The steps to create a new file system are shown on the slide. We will talk about each step in detail on the following pages.

© 2004 Hewlett-Packard Development Company, L.P.

9–2. SLIDE: File System Creation Commands

File System Creation Commands

1. Create the new file system
   ```
   # newfs -F hfs /dev/vg01/rmyfs1
   ```
2. Create a mount point directory for the file system
   ```
   # mkdir /myfs1
   ```
3. Mount the new file system
   ```
   # mount /dev/vg01/myfs1 /myfs1
   ```
4. Add the file system to the `/etc/fstab` file
   ```
   # vi /etc/fstab
   ```

Student Notes

First, you must create a superblock, inode table, and other file system metadata structures for the new file system. The **`newfs(1m)`** command is the utility of choice for creating both VxFS and HFS file systems. The example on the slide creates a HFS file system in logical volume **`/dev/vg01/rmyfs1`**. The next slide presents more **`newfs`** examples.

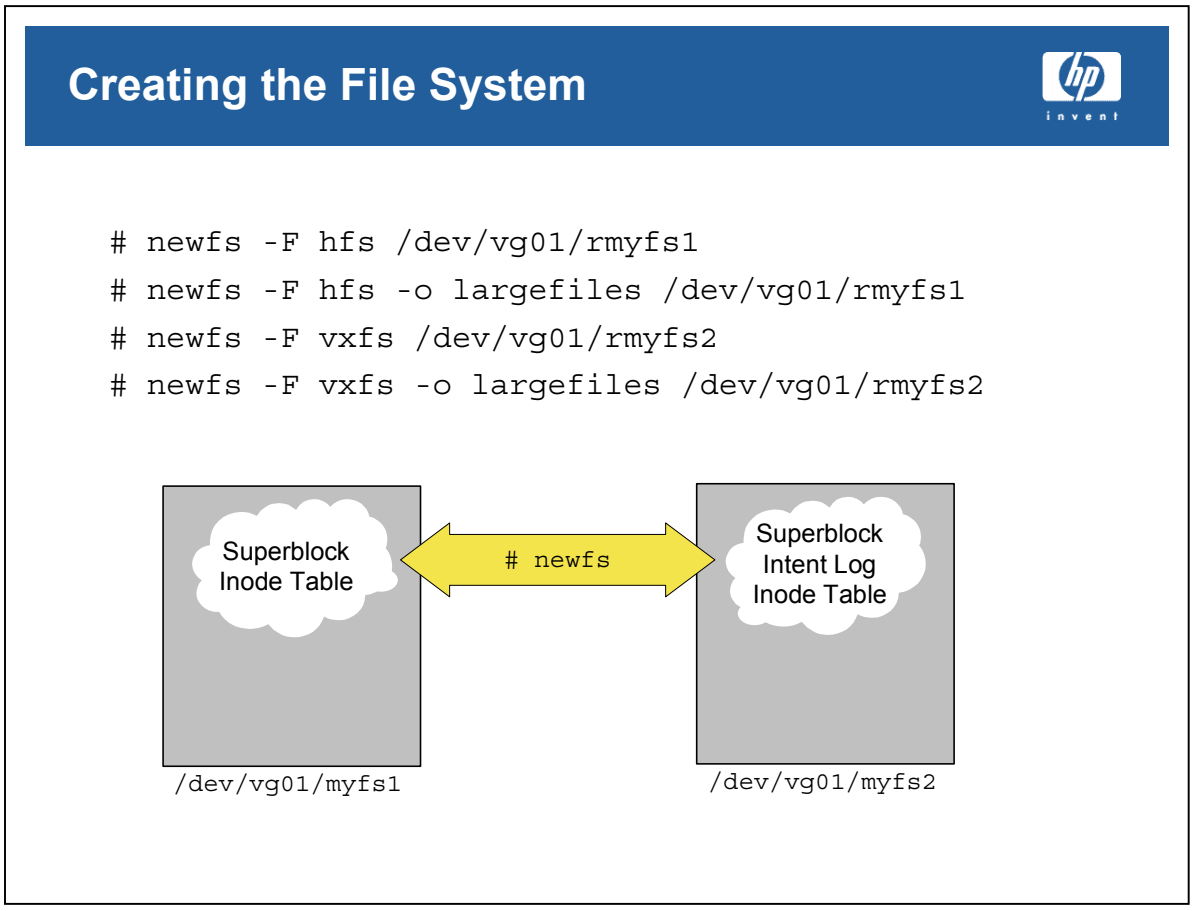
Once the file system metadata structures have been created with **`newfs`**, you must create a mount point directory for the file system, and mount it.

Finally, if you wish to make a file system permanently available, it should be added to the **`/etc/fstab`** file. Every time the system boots, the **`/etc/fstab`** file is consulted to determine which file systems should be automatically mounted. Thus, if a new file system is not added to **`/etc/fstab`**, it will not be automatically mounted after the next system boot.

After creating and mounting a new file system, you may want to issue the **`bdf(1m)`** command or **`mount(1m)`** to ensure that your file system successfully mounted.

© 2004 Hewlett-Packard Development Company, L.P.

9–3. SLIDE: Creating the File System



Student Notes

After a logical volume has been set aside for use as a file system, you must create a superblock, inode table, and other metadata structures before you can create files and directories in the file system. The **`newfs`** command is the utility of choice for creating these structures.

Slide Examples

- The first example on the slide simply creates an HFS file system in a logical volume called **`/dev/vg01/rmyfs1`**. The **`-F hfs`** option specifies that an HFS file system should be created. Also note that the **`newfs`** command requires a raw device file as an argument. The logical volume or disk specified will be overwritten, and any existing data in that logical volume or disk will be destroyed.

  The second example is much like the first. However, it includes the **`largefiles`** option. HP-UX version 10.20 introduced support for large files up to 128 GB in size. Without the **`-o largefiles`** option, however, the largest file size allowed in a file system is 2 GB. JFS versions 3.0 and 3.3 support files up to 1TB. JFS version 3.5 supports files up to 2TB.

- The third example creates a VxFS file system in the **`/dev/vg01/rmyfs2`** logical volume.

© 2004 Hewlett-Packard Development Company, L.P.

- The fourth example creates a VxFS file system with large file support in **/dev/vg01/rmyfs2**.

newfs Options Common to HFS and VxFS

The **newfs** command has many options. Some apply to both file system types, some only apply only to HFS, and some apply only to JFS. The following list is a description of some of the options that apply to both file system types.

-F hfs|vxfs	Defines the desired file system type. If **-F** is not specified, the default file system type is determined from the **/etc/default/fs** file.
-o largefiles	Determines whether or not the file system will allow "large files" over 2 GB in size. "nolargefiles" is the default.
-s size	Specifies the total size of the file system in blocks. By default, **newfs** uses all the space available on the specified logical volume or disk.
-v	Verbose. List all actions performed by **newfs**.

newfs Options Specific to HFS

Some **newfs** options are usually used only when creating HFS file systems:

-L/-S	Early versions of HP-UX supported only "short" file names up to 14 characters in length. You can still enforce this limit if you wish using the "-S" option. By default, however, file systems allow "long" file names up to 256 characters in length.
-b *block-size*	Specifies the file system's HFS block size in bytes. The block size defaults to 8 KB.
-f *frag-size*	Specifies the file system's fragment size in bytes. The fragment size usually defaults to 1 KB.
-m *min-free*	This indicates the percentage of space in the file system reserved for use by root. If the amount of free space in the file system falls below this percentage, the superuser is the only one who can write to the file system. This defaults to 10%. Consider decreasing this value in very large file systems.

newfs Options Specific to VxFS

There are several JFS-specific options on **newfs**. However, with the exception of **-o largefiles** as described previously, it is recommended that you take the default options when creating a JFS file system. See the **newfs_vxfs(1m)** and **mkfs_vxfs(1m)** man pages for details.

© 2004 Hewlett-Packard Development Company, L.P.

`newfs` Options Specific to the Whole-disk Approach

Most administrators today choose to partition disks using LVM. On workstations, however, you may prefer the simplicity of the whole-disk approach.

If you wish to use an entire disk for a file system use a **`newfs`** command similar to one of the following:

```
# newfs -F hfs  /dev/rdsk/c0t2d0   # creates an HFS on disk c0t2d0
# newfs -F vxfs /dev/rdsk/c0t2d0   # creates a JFS on disk c0t2d0
```

The **`-R`** option reserves space at the end of the disk for use as swap space:

```
# newfs -F hfs -R 200 /dev/rdsk/c0t2d0  # HFS, with 200 MB reserved for swap
# newfs -F vxfs -R 200 /dev/rdsk/c0t2d0 # VxFS, with 200 MB reserved for swap
```

You can also create a boot disk using the whole disk approach; see the **`newfs_hfs(1m)`** man page for more information.

Output from `newfs`

After creating the new file system, **`newfs`** displays several lines of information about the new file system:

```
# newfs -F hfs /dev/vg01/rmyfs1
  mkfs (hfs): Warning - 94 sector(s) in the last cylinder are not
  allocated.
  mkfs (hfs): / dev/vg01/rmyfs2 - 16384 sectors in 107 cylinders of
  7 tracks, 22 sectors
  16.8Mb in 7 cyl groups (16 c/g, 2.52Mb/g, 384  i/g)
  Super block backups (for fsck -b) at:
  16,   2504,   4992,   7480,   9968,  12456,  14944

# newfs -F vxfs /dev/vg01/rmyfs2
  version 4 layout
  16384 sectors, 16384 blocks of size 1024, log size 1024 blocks
  unlimited inodes, 16384 data blocks, 15288 free data blocks
  last allocation unit has 16384 data blocks
  first allocation unit starts at block 0
  overhead per allocation unit  is 0 blocks
```

Verifying File System Creation

You can verify that a file system was created properly via the **`mkfs -m`** command. This command uses information from the file system superblock to report which file system parameters were specified at file system creation time.

```
# mkfs -m /dev/vg00/rmyfs2
  mkfs -F vxfs -o ninode=unlimited,bsize=1024,version=4,inosize=256,
  logsize=1024,nolargefiles /dev/vg00/rlvol3 16384
```

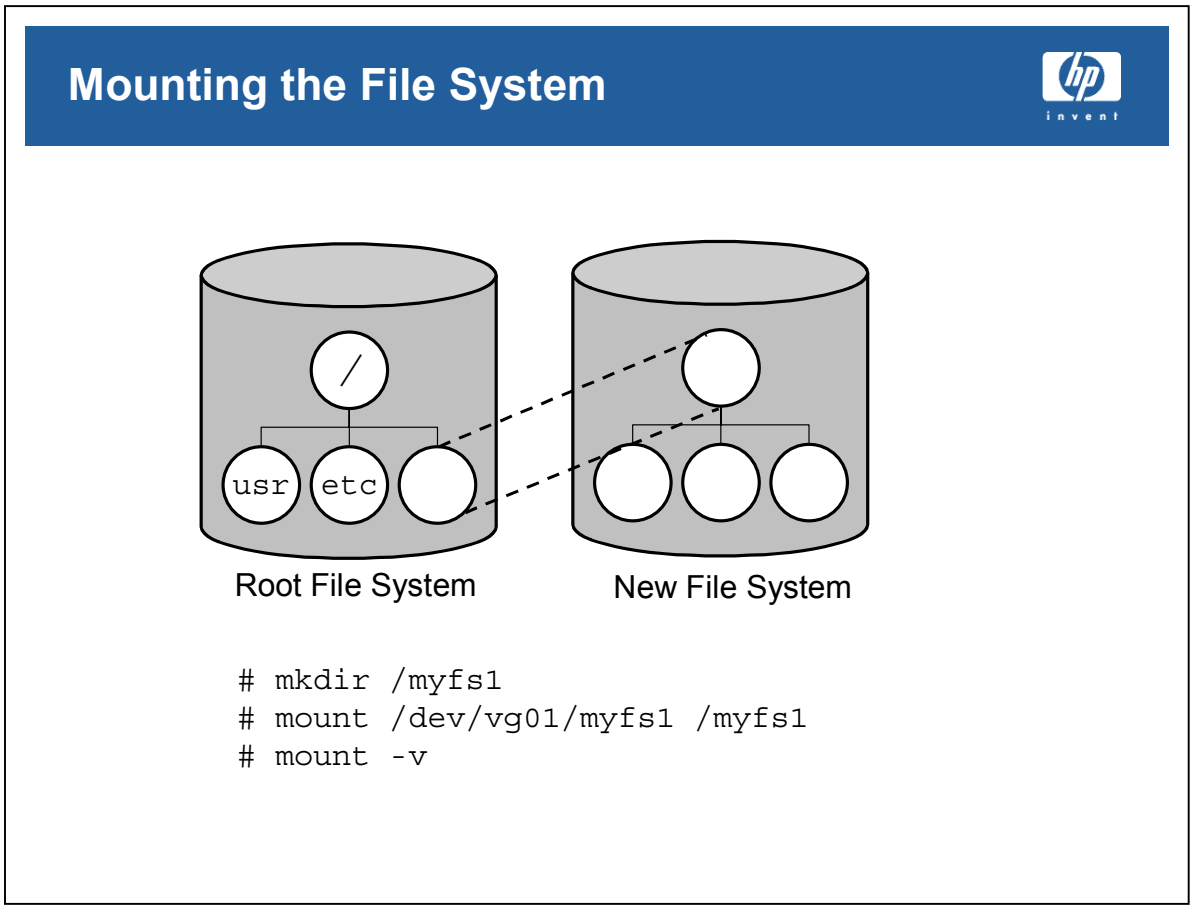

© 2004 Hewlett-Packard Development Company, L.P.

NOTE: There are several man pages for the **`newfs`** command.

- **`newfs(1m)`** describes **`newfs`** options common to all file systems.
- **`newfs_hfs(1m)`** describes HFS-only options
- **`newfs_vxfs(1m)`** describes VxFS-only options.

© 2004 Hewlett-Packard Development Company, L.P.

9–4. SLIDE: Mounting the File System



Student Notes

Mounting a File System

HP-UX cannot use a file system unless it is mounted. After a file system has been created on a logical volume or device, it must be incorporated into the system's file hierarchy by creating a mount point and mounting the file system. Example:

```
# mkdir /myfs1                      # create a mount point
# mount /dev/vg01/myfs1 /myfs1      # mount a file system on the mount point
```

NOTE: The **`mount`** command requires a block device file, while **`newfs`** requires a character device file.

Mounting a file system in this manner logically associates the root directory of the new file system with the mount point directory. Accessing the mount point directory actually references the file system mounted on that directory.

© 2004 Hewlett-Packard Development Company, L.P.

Guidelines for Choosing a Mount Point

Though mount points can be created in any directory, most file systems are mounted on mount points immediately under the **/** directory. **/usr**, **/tmp**, and **/home** are just a few examples of mount point directories you may have on your system.

Also, file systems should only be mounted on empty directories. If a file system is mounted on a directory that already contains files and directories, those files and directories will be hidden until the file system is unmounted!

Finally, note that it is not possible to mount a file system on a directory that is already in use by another user or process. Trying to mount a file system on a directory that is already in use results in a "device busy" message.

Viewing Mounted File Systems

Two commands list your currently mounted file systems.

The **mount -v** command displays a verbose listing showing which file systems were mounted where and when.

```
# mount -v
/dev/vg00/lvol3 on / type vxfs log on Fri Jun 26
/dev/vg00/lvol1 on /stand type hfs defaults on Fri Jun 26
/dev/vg00/lvol8 on /var type vxfs defaults on Fri Jun 26
/dev/vg00/lvol7 on /usr type vxfs defaults on Fri Jun 26
/dev/vg00/lvol6 on /tmp type vxfs defaults on Fri Jun 26
/dev/vg00/lvol5 on /opt type vxfs defaults on Fri Jun 26
/dev/vg00/lvol4 on /home type vxfs defaults on Fri Jun 26
/dev/vg01/myfs1 on /myfs1 type hfs defaults on Fri Jun 26
/dev/vg01/myfs2 on /myfs2 type vxfs defaults on Fri Jun 26
```

The **bdf** command also displays the amount of space in use and available in each mounted file system.

```
# bdf
Filesystem          kbytes    used   avail %used Mounted on
/dev/vg00/lvol3      86016   23302   58855   28% /
/dev/vg00/lvol1      67733   20068   40891   33% /stand
/dev/vg00/lvol8      69632   14138   52387   21% /var
/dev/vg00/lvol7     397312  334233   59179   85% /usr
/dev/vg00/lvol6      49152    1326   44902    3% /tmp
/dev/vg00/lvol5     294912  261523   31352   89% /opt
/dev/vg00/lvol4      24576   19333    4978   80% /home
/dev/vg01/myfs1      15893       9   14294    0% /myfs1
/dev/vg01/myfs2      16384    1109   14328    7% /myfs2
```


© 2004 Hewlett-Packard Development Company, L.P.

`mount` Options Common to HFS and VxFS

The **`mount`** command has many options. Some apply to both file system types, and some only apply to HFS. Following is a list of **`mount`** options common to both file system types.

`-a`	Mount all file systems listed in **`/etc/fstab`**.
`-aF` *`FSType`*	Mount all file systems of the specified type.
`-l`	List all mounted local file systems.
`-v`	List all mounted file systems.
`-o rw`|**`ro`**	Mount file system “read-write” or “read-only”.
`-o suid`|**`nosuid`**	Enable/disable SUID executables.
`-o quota`|**`noquota`**	Enable/disable quota checking.
`-o nolargefiles`	Prevent creation of “large” files, even if the largefiles bit is set.
`-o largefiles`	Allow creation of “large” files, but only if the largefiles bit is set.
`-o defaults`	Mount **`rw,suid,largefiles,noquota`**.

`mount` Options Specific to HFS

There are a few HFS-specific mount options, but they are very infrequently used. Check the **`mount_hfs(1m)`** man page for more information.

`mount` Options Specific to VxFS

VxFS supports numerous mount options, most of which control when and if the intent log and buffer cache are used. Tweaking these options can significantly impact the performance of your JFS file systems. The list below highlights some of the most common combinations of options. In order to use any of these options, you must explicitly state that the file system type is VxFS by using the **`-F vxfs`** option.

© 2004 Hewlett-Packard Development Company, L.P.

Many of the explanations in the list reference two different types of file system write requests. When an application writes data via an "asynchronous" write request, the application waits for confirmation that the data has been recorded in the buffer cache, but may continue execution before the data is physically committed to disk. When an application writes data via a "synchronous" write request, the application waits until the data is physically committed to disk before proceeding. In case of a system crash, data in the buffer cache is lost, while data on disk is preserved. Thus, applications typically use synchronous writes for critical data, but use asynchronous writes to improve performance.

`-o log,mincache=dsync`	Guarantee full integrity for metadata and user data. Logs all metadata changes to the intent log immediately. Sends all write requests (even asynchronous requests!) directly to disk.
`-o log`	Guarantee full integrity for metadata. Logs all metadata changes to the intent log immediately. Sends asynchronous user data write requests through buffer cache, and synchronous requests directly to disk (normal Unix *write* semantics).
`-o delaylog`	Guarantee full integrity for critical metadata. Logs critical metadata changes immediately. Less critical metadata changes (such as time stamp changes) may be lost in case of a system crash. Sends asynchronous user data write requests through buffer cache, and synchronous requests directly to disk (normal UNIX *write* semantics). This is the default mount option at JFS versions 3.5 and beyond.
`-o nolog,convosync=delay`	Best performance. Writes nothing to the intent log. Delays all user data write requests. File systems mounted with these options will likely need to be recreated after a system crash! Only use this option for temporary file systems.

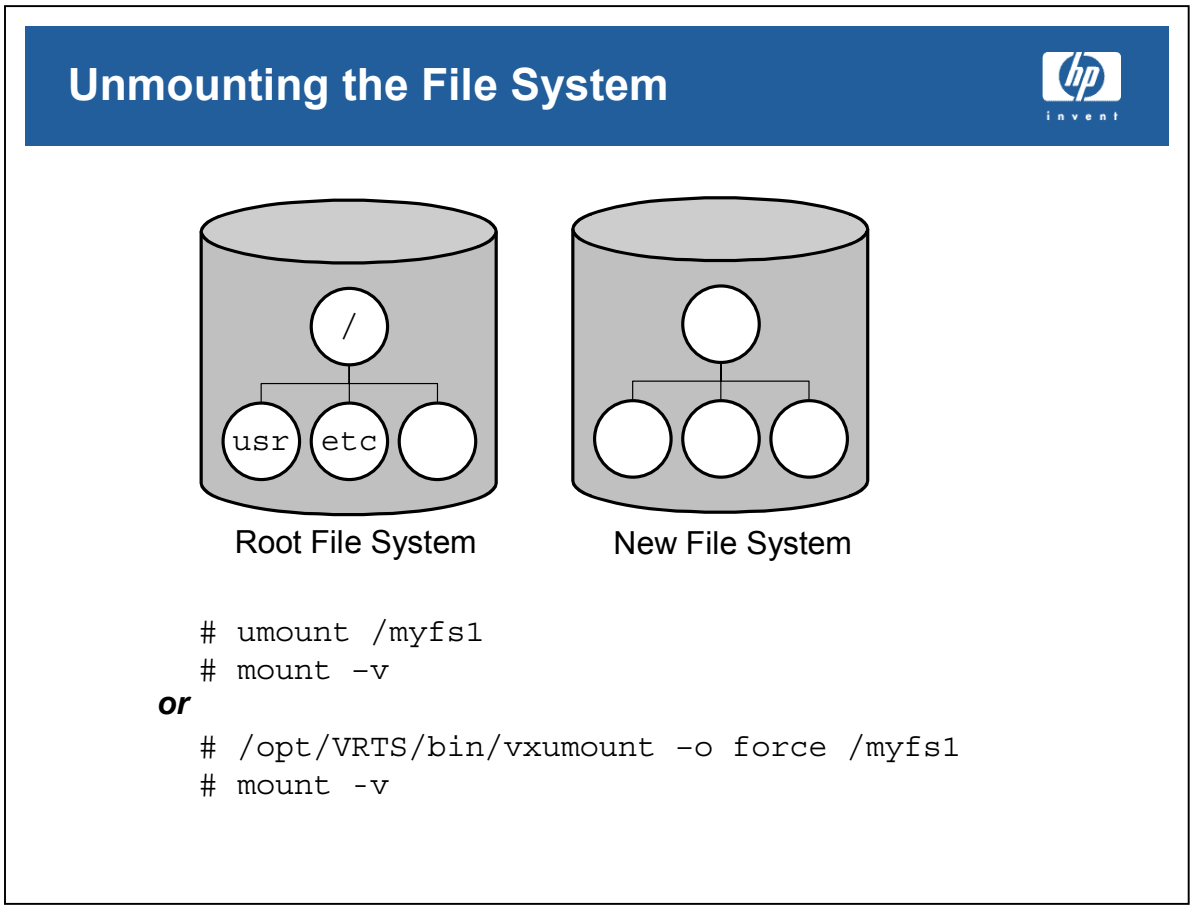
Many other mount option combinations are possible. Investigate the **`mount_vxfs(1m)`** man page, the *HP OnlineJFS 3.3 Veritas File System 3.3 Administrator's Guide* on **`http://docs.hp.com`**, or HP's *HP-UX Performance and Tuning* class for more information.

NOTE: There are several **`man`** pages for the **`mount`** command.

- **`mount(1m)`** describes **`newfs`** options common to all file systems.
- **`mount_hfs(1m)`** describes HFS-only options.
- **`mount_vxfs(1m)`** describes VxFS-only options.

© 2004 Hewlett-Packard Development Company, L.P.

9–5. SLIDE: Unmounting the File System



Student Notes

Now that you know how to mount a new file system, you should also be aware of how to logically disassociate, or unmount, the new file system from the root file system. The command used to unmount the file system is **umount**.

NOTE: The command is **umount**, not "unmount". The command uses the block device file or mount-point directory.

---

umount options include:

-a	umount "all" currently mounted file systems.
-F Fstype	specify a file system type
-v	report the output with the FStype displayed

Instead of using the **umount -a** command, you can also use the **umountall(1M)** command.

© 2004 Hewlett-Packard Development Company, L.P.

Unmounting Busy File Systems

By default, a file system cannot be unmounted if any of the files or directories in the file system are currently in use, or if the file system has been configured for use as file system swap.

You can use the **fuser** command to identify which processes are using a file or file system. You can specify the target file system by device file name or by mount point directory (when using the mount point directory name, also add the **-c** option). With the –k option, fuser will also kill the processes that are using the specified file or file system.

```
# fuser -u /myfs1/data          list processes using a specific file
# fuser -cu /myfs1              list processes using the file system on /myfs1
# fuser -u /dev/vg01/myfs1      list processes using the FS in /dev/vg01/myfs
# fuser -ku /dev/vg01/myfs1     kill processes using the FS in /dev/vg01/myfs
```

OnlineJFS 3.5 now includes a new command that will automatically forcefully unmount a file system, even if processes are currently using the file system.

```
# /opt/VRTS/bin/vxumount –o force /myfs2
```

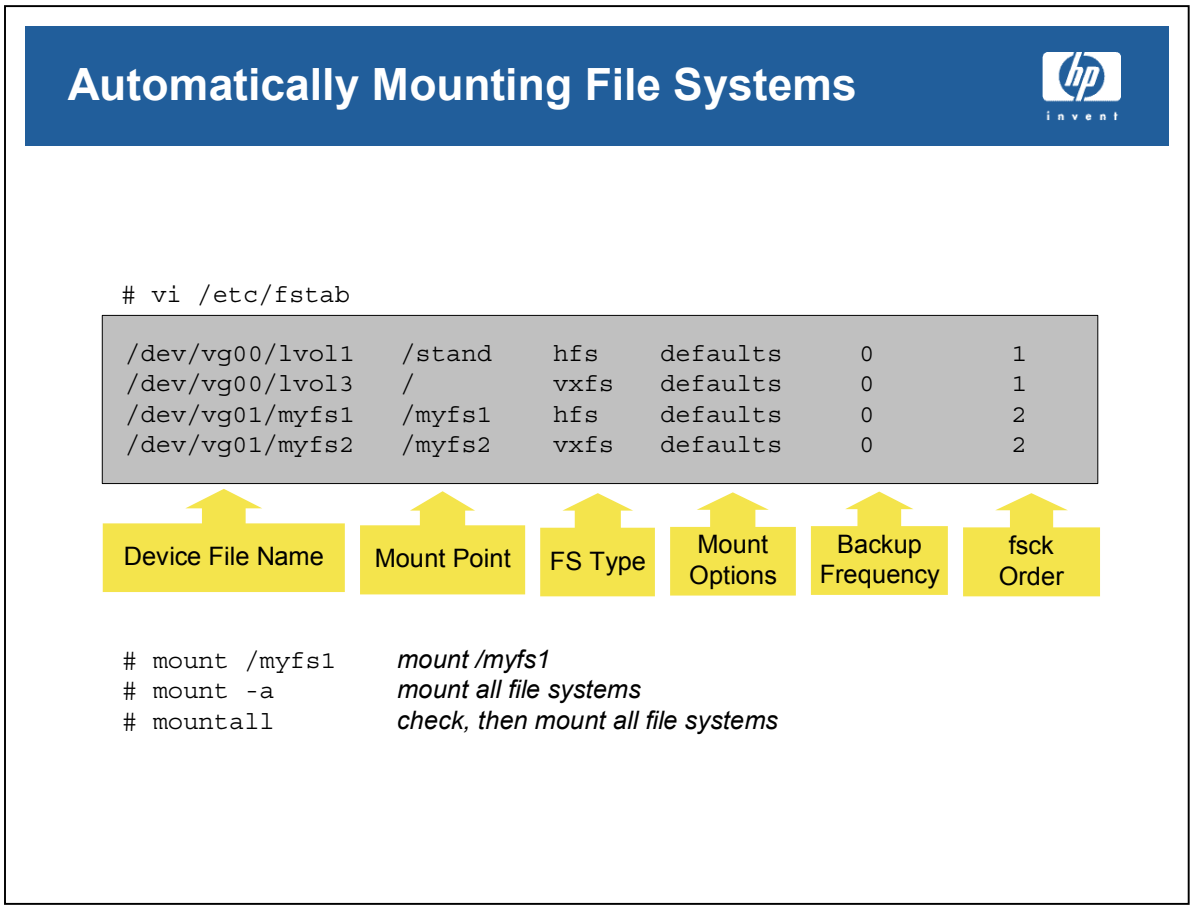
Be very careful when using the **fuser** and **vxumount** commands. If possible, it is much safer to kill application processes gracefully using your vendors' recommended shutdown procedures. Using these utilities to more forcefully kill processes may leave your applications in an unusable state.

Some file systems, such as the **/** (root) file system and **/usr**, can't be unmounted on a running system since critical system daemons can't function properly without them.

The **shutdown** and **reboot** commands automatically unmount *all* file systems during the system shutdown process.

© 2004 Hewlett-Packard Development Company, L.P.

9–6. SLIDE: Automatically Mounting File Systems



Student Notes

All file systems are unmounted during system shutdown. Any file systems that you wish to mount automatically after the next system reboot should be added to the **/etc/fstab** file. During the boot process, the **/sbin/init.d/localmount** script executes the **mount -a** command, which automatically mounts file systems listed in **/etc/fstab**. This configuration file is not automatically maintained by the system; it should be manually edited when file systems are created and removed.

After adding a file system to **/etc/fstab**, you needn't enter the full form of the **mount** command when mounting the new file system. Look at the following examples:

```
# mount -a                  # mount all file systems in /etc/fstab
# mount /myfs1              # mount /myfs -- no need to name the logical volume
# mount /dev/vg01/myfs1     # mount /dev/vg01/myfs1 -- no need to name the
                            mount point
```


© 2004 Hewlett-Packard Development Company, L.P.

Syntax for `/etc/fstab`

Fields in the **`/etc/fstab`** file include the following:

block — the block device file that corresponds to the mounted file system

directory — the directory to which mount mounts the device

type — the file system type. Types include:

- **`cdfs`** — local CD-ROM file system.
- **`hfs`** — high-performance (McKusick) file system.
- **`nfs`** — network or remote file system.
- **`vxfs`** — journaled file system.
- **`swap`** — the device file name is made available as a piece of swap space by the **`swapon`** command.
- **`swapfs`** — the file system which directory resides in is made available as swap space by the **`swapon`** command.
- **`lofs`** — the file system is a loopback file system.
- **`ignore`** — marks unused sections (on multi-file system disks).

options — a comma-delimited list of options used by **`mount(1M)`** and **`swap(1M)`**. Examples are:

- **`defaults`** — Sets **`rw`**, **`suid`** and **`noquota`** mount options. If you use this option, no other options can be specified.
- **`rw`** (default) — read/write
- **`ro`** — read only
- **`suid`** (default) — set user-id allowed
- **`nosuid`** — no set user-id allowed
- **`quota`** — enables checking of disk quota on this file system
- **`noquota`** (default) — no quota checking on this file system

backup-frequency — reserved for possible use by future backup utilities.

pass-number — used by the **`fsck`** command to determine the order in which file system checks are done.

Comment — a comment field (must be preceded by a #)

NOTE: A more detailed description of the **`/etc/fstab`** file syntax can be found in the **`fstab(4)`** man page. Also take a look at the man pages **`mount`**: **`mount(1m)`**, **`mount_hfs(1m)`**, and **`mount_vxfs(1m)`** for file system specific mount options.

© 2004 Hewlett-Packard Development Company, L.P.

9–7. SLIDE: Mounting CDFS File Systems

Mounting CDFS File Systems

CDFS may be used to mount High Sierra, ISO 9600, and ISO 9660/Rock Ridge CDROMs

1. Determine the device file name of your CDROM drive.
   `# ioscan -funC disk`
2. Create a mount point.
   `# mkdir /cdrom`
3. Mount the file system.
   `# mount -o ro,cdcase,rr /dev/dsk/c0t1d0 /cdrom`
4. Add the file system to `/etc/fstab`

```
/dev/vg01/myfs1   /myfs1    hfs    defaults       0  2
/dev/vg01/myfs2   /myfs2    vxfs   defaults       0  2
/dev/dsk/c1t3d0   /cdrom    cdfs   ro,cdcase,rr   0  0
```

Student Notes

CDFS is a kernel subsystem that makes it possible to mount CDROMs and DVDs on HP-UX systems.

Mounting a CDROM requires several steps. First, make sure the CDROM drive has the proper SCSI address configured. Then, shut down the computer and power off (both the computer and the devices on the SCSI channel). Connect the new CDROM, power on the devices on the SCSI channel, and power on the computer. The system will auto-configure the CDROM.

Assuming the proper device driver has been installed, the system should create the necessary device files during the boot process. After booting, create a mount point directory and mount the file system (this example uses **`/cdrom`**). At a minimum, use the **`ro`** mount option to make it clear that the file system is read-only. The **`mount`** example on the slide includes two additional options, which will be described in more detail below.

```
# ioscan -funC disk                      # find the CDROM block device file
# mkdir /cdrom                           # create a mount point directory
# mount -o ro /dev/dsk/c0t1d0 /cdrom     # mount the CD
```


© 2004 Hewlett-Packard Development Company, L.P.

CDFS file systems can be included in the **/etc/fstab** file for automatic mounting at boot much like any other file system. Note that we use **fsck** order number 0 as a place-holder in the file; one would never actually run **fsck** on a CDFS file system. Only add a CDFS file system to **/etc/fstab** if you are sure that the CDROM will always be available in the CDROM drive. If a CDFS file system is included in the **/etc/fstab** file, but the CDROM isn't in the drive at boot time, the boot process will be delayed while the kernel tries to mount the non-existent CD.

```
/dev/dsk/c0t1d0 /cdrom cdfs ro 0 0   # local CDROM drive
```

Once mounted, CDROM file systems can be accessed using the same HP-UX file system commands that are used to navigate an HFS or JFS file system.

Converting File Names using CDFS Mount Options

CDFS supports several common CDROM formats:

- the ISO9600 format
- the High Sierra format
- the ISO9660 Rockridge Extension format, though patches are required:
  on 11.11: PHKL_28025 and PHKL_26269
  on 11.00: PHCO_26449, PHKL_26450, and PHKL_28060

By default, CDFS displays filenames using the standard 8.3;1 filename format that is popular in the PC world:

```
# mount –o ro /dev/dsk/c0t1d0 /cdrom
# ls /cdrom
RUNINSTA.;1
INDEX.HTM;1
```

This format may be inconvenient in an HP-UX environment since the semicolon is used as a command separator in the POSIX shell. You can disable lowercase-to-uppercase filename translation, and suppress the display of version numbers by using the **-o cdcase** option:

```
# mount –o ro,cdcase /dev/dsk/c0t1d0 /cdrom
# ls /cdrom
runinsta.
index.htm
```

If you install the ISO 9660 Rockridge Extension patches mentioned above and mount file systems with the **rr** mount option, CDFS provides even more flexibility. This option is required when dealing with CDs from Oracle and other third party vendors.

```
# mount -F cdfs -o ro,cdcase,rr /dev/dsk/c0t1d0 /cdrom
# ls /cdrom
runInstaller
index.htm
```


© 2004 Hewlett-Packard Development Company, L.P.

Using PFS (Portable File Systems)

The ISO9660 Rockridge Extension patches mentioned above were first were first made available for CDFS in early 2002. Previously, the ISO 9660 Rockridge functionality was only available via a third party alternative to CDFS called the Portable File System (PFS) from Young Minds Inc.

The Portable File System (PFS) provides other conversion options, as well as the capability to export to remote PFS clients. PFS is included with HP-UX, but it is more complicated to configure than CDFS. For more information, see the **`pfs(4)`** man page.

CD-RW CDs

HP now provides limited support for CDRW (re-writable CDs) on some workstation models. Two pre-compiled contributed software applications, **`xcdroast`** and **`cdrecord`**, are included on the install Technical Computing Operating Environment media kit for workstations as well. See the 11i Release Notes on **`http://docs.hp.com`** website and the documentation in **`/opt/OpenSource/cdrtools-1.10/doc/`** for more information. Neither CD-RW drives, nor the **`cdrecord`** software, are supported on servers at this time.

© 2004 Hewlett-Packard Development Company, L.P.

9–8. SLIDE: Mounting LOFS File Systems

Mounting LOFS File Systems

HP's Loopback File System (LOFS) allows the same file hierarchy to appear in multiple places, without creating symbolic links.

1. Create a mount point.
   ```
   # mkdir /opt/data
   ```
2. Mount the file system.
   ```
   # mount -F lofs /data /opt/data
   ```
3. Add the file system to `/etc/fstab`

```
/dev/vg01/myfs1   /myfs1     hfs    defaults      0  2
/dev/vg01/myfs2   /myfs2     vxfs   defaults      0  2
/data             /opt/data  lofs   defaults      0  0
```

Student Notes

The Loopback Filesystem (LOFS) allows the same file hierarchy to appear in multiple places. Traditionally, this has been accomplished via symbolic links:

```
# ln -s /data /opt/data
# ls /data /opt/data

     /data:
     d1   d2   d3

     /opt/data:
     d1   d2   d3
```


© 2004 Hewlett-Packard Development Company, L.P.

HP-UX provides a more elegant solution to this situation via the LOFS file system:

```
# mkdir /opt/data
# mount -F lofs /data /opt/data
# ls /data /opt/data

     /data:
     d1  d2  d3

     /opt/data:
     d1  d2  d3
```

In order to make an LOFS file system accessible after every reboot, add it to the **`/etc/fstab`** file:

```
# vi /etc/fstab
  /data /opt/data lofs defaults 0 0
```


© 2004 Hewlett-Packard Development Company, L.P.

9–9. LAB: Creating File Systems

Directions

Record the commands used to complete the tasks below, and answer all of the questions.

Part 1: Preliminary Steps

1. The exercises in this lab assume that you already have the following three logical volumes:

   ```
   /dev/vg01/data
   /dev/vg01/app
   /dev/vg01/tables
   ```

   If you already have these logical volumes, you can skip ahead to Part 2 of the lab. Otherwise, create the **`vg01`** volume group and all three of the logical volumes listed above. Make each of the logical volumes 12 MB.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Creating File Systems from the Command Line

1. Create an HFS file system in the **data** logical volume. Create a JFS file system in the **app** logical volume. Don't mount your file systems yet.

2. Do a **`mount -v`**. Why don't your new file systems appear at this point?

3. Create a mount point for each of your new file systems. Use **`/data`** as the mount point for the file system in the **`data`** logical volume, and **`/app`** as the mount point for the **`app`** logical volume. Again, do not mount your file systems, yet.

4. Add your new file systems to the **`/etc/fstab`** file so they will be mounted automatically at each system boot. Again, do not mount your file systems, yet.

5. Go ahead and mount your file systems now by doing a **`mount -a`**. Watch the resulting messages carefully. You should see several error messages indicating that **`/dev/vg00/lvol1`** and several other file systems are already mounted. Do the **`mount -a`** output messages offer any indication that your new file systems were successfully mounted?

© 2004 Hewlett-Packard Development Company, L.P.

6. Execute **`mount -a`** a second time and note the output messages again. Why did **`mount -a`** mention your new file systems in its output this time, but not when you did a **`mount -a`** in the previous exercise?

7. Use **`mount -v`** to see what file systems are now mounted. Did your new file systems mount properly? What other information can you glean from the **`mount -v`** output about your mounted file systems? List three fields presented in the **`mount -v`** output.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Experimenting with `mount` and `umount`

1. The exercises in this section will be more meaningful if you have some real files in the **`/data`** and **`/app`** file systems. Towards that end, create a few files in these file systems using the following commands:

```
# cd /data; touch d1 d2 d3
# cd /app;  touch a1 a2 a3
# ls /data /app
# cd /
```

NOTE: You may notice a **`lost+found`** directory in your new file systems. **`newfs`** creates this directory for you automatically. This special directory serves as a home for files that are victims of file system corruption. This directory will be discussed in some detail in a later chapter.

---

2. Can you access files in a file system that is unmounted? Try an experiment to find out: Unmount **`/data`**, then do an **`ls`** of **`/data`**. Does the mount point still exist? Can you access **`/data/d1`** and **`/data/d2`**? Why or why not? Do whatever is necessary to regain access to **`/data/d1`** and **`/data/d2`**.

3. Can you unmount a file system that is still in use? Try an experiment to find out: Open a second window on your screen. In this second window, **`cd`** to the **`/data`** file system. Back in your original window try unmounting the **`/data`** file system. What message do you get? Why?

© 2004 Hewlett-Packard Development Company, L.P.

4. Before you can unmount a file system, you will have to kill all processes accessing the file system. HP-UX provides a command to solve this very problem. Try the following command:

   ```
   # fuser -u /dev/vg01/data
   ```

   Each entry in the **fuser** output lists the PID of a process accessing the file system, a single letter code indicating how the process is using the file system ("**c**" indicates that a user has changed to a directory in the file system), and the name of the user who owns the offending process. **fuser** can also kill all of the offending processes:

   ```
   # fuser -ku /dev/vg01/data
   ```

   The effect of this command should be pretty dramatic. What happens? Now try unmounting **/data**.

5. What happens if you mount a file system on a directory that already contains files? Try an experiment to find out:

   While the **/dev/vg01/data** file system is still unmounted, touch a few files under the **/data** mount point:

   ```
   # touch /data/junk1 /data/junk2
   # ls /data
   ```

   Now mount the **/dev/vg01/data** file system and list the contents of **/data** again:

   ```
   # mount /data
   # ls /data
   ```

   Do you still see **junk1** and **junk2**? Unmount **/data** again and check to see if **junk1** and **junk2** still exist. Why are **junk1** and **junk2** hidden while the data file system is mounted? Explain.

NOTE: File systems should always be mounted on empty mount point directories.

© 2004 Hewlett-Packard Development Company, L.P.

6. Remove the junk files you created in the previous exercise, and remount all of your file systems again.

7. One last experiment: Can you unmount all of your file systems? Try a **`umount -a`** and explain the result.

8. Remount all of your file systems before continuing.

© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Experimenting with `newfs`

1. HP-UX supports two different types of disk-based file systems: HFS and JFS. In all of the examples you have tried so far, you have specified your desired file system type via the **`-F`** option. What happens if you forget the **`-F`**? Try it: do a **`newfs`** on the **`/dev/vg01/tables`** logical volume, but leave off the **`-F`** option. Look at the **`newfs`** output carefully. How does the system determine your default file system type? What is your default file system type?

2. The **`newfs`** command is a powerful tool for creating file systems, but it must also be treated with some respect. What happens if you accidentally **`newfs`** a logical volume or disk that already contains a file system? Try an experiment to find out: What happens if you attempt to create a new **`hfs`** file system on the data logical volume with the following command:

   ```
   # newfs -F hfs /dev/vg01/rdata
   ```

3. What happens if you try to **`newfs`** a logical volume that contains an unmounted file system? Try it.

   ```
   # umount /data
   # newfs -F hfs /dev/vg01/rdata
   # mount /data
   # ls /data
   ```

   Did you encounter any problems? What happened to the d1, d2, d3 files you created earlier? Always use **`newfs`** with care.

© 2004 Hewlett-Packard Development Company, L.P.

Part 5: Using SAM to Create Logical Volumes and File Systems

1. So far, you have created several file systems from the command line. File systems can be created and managed within SAM as well. Open SAM, then do the following:

   a. Select: **`SAM --> Disks and file systems --> Logical volumes`**
   b. Select: **`Actions --> Create`**
   c. Click: **`Select a Volume Group`** (Choose **`vg01`**)
   d. Click: **`Define New Logical Volumes`**
   e. Fill in the blanks in the dialog box that appears:

      LV name: **`app2`**
      LV size (MB): 12
      Usage: File system
      Mount Dir: **`/app2`**

   f. Click: **`Modify FS defaults`**.
   g. Fill in the blanks in the dialog box that appears:

      i. Create a Journaled file system.
      ii. Choose **`Now`** AND **`Every system boot`** options under **`When to mount`**.
      iii. For now, take the defaults for the rest of the screen, and click **`OK`**.

   h. You will be returned to the **`Define new logical volumes`** screen.
   i. Click **`Add`** to add your new logical volume to the list of new LVs to create.
   j. Click **`OK`** to go back to the **`Create new logical volumes`** step menu.
   k. Click **`OK`** to create the new logical volume.

2. Do a **`mount -v`** from the command line. Is the new file system mounted?

3. Look at **`/etc/fstab`**. Did SAM ensure that the new file system will be mounted with every system boot?

© 2004 Hewlett-Packard Development Company, L.P.

Part 6: Cleaning Up

1. Before moving on to the next chapter, use SAM to remove the file systems and logical volumes you created during this lab exercise:

   Select: SAM --> Disks and file systems --> Logical volumes

   While holding down CTRL, select each of the vg01 LVs. Do not remove any of the logical volumes in vg00.

   Select: Actions --> Remove

   When asked for confirmation, answer Yes.

2. What did SAM do on your behalf?

   Are the file systems unmounted?

   Are the file systems still listed in **`/etc/fstab`**?

   If you do a **`vgdisplay`**, do the logical volumes still appear?

   Do the mount points still exist?

3. You should also remove volume group vg01 before continuing on to the next chapter. Again, use SAM:

   Select: **`SAM --> Disks and file systems --> Volume groups`**

   Select **`vg01`**

   Select: **`Actions --> Remove`**

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 10 — Maintaining File Systems

Objectives

Upon completion of this module, you will be able to do the following:

- Monitor space available in the file system with **`bdf`** and **`du`**.
- Clean up the file systems by purging unused files and core files.
- Clean up the **`/var`** file system by trimming log files.
- Extend a volume group from the command line.
- Extend a logical volume from the command line.
- Extend a file system from the command line.
- Repair HFS and JFS file system corruption with **`fsck`**.
- Defragment JFS file systems with **`fsadm`**.

© 2004 Hewlett-Packard Development Company, L.P.

10–1. SLIDE: Monitoring File System Free Space

Monitoring File System Free Space

Monitor available file system space with `bdf`:

```
# bdf
Filesystem          kbytes    used   avail   %used  Mounted on
/dev/vg00/lvol5     294912  261523   31352     89%  /opt
/dev/vg00/lvol4      24576   19333    4978     80%  /home
/dev/vg01/myfs1      16384    1174   14331      8%  /myfs1
/dev/vg01/myfs2      15893   14006     297     98%  /myfs2
```

Determine space used by directory subtrees with `du`:

```
# du -sk /myfs2/*
844       /myfs2/data1
1327      /myfs2/data2
4         /myfs2/lost+found
```

Student Notes

Disk space is often at a premium. The system administrator should monitor disk free space regularly, and take steps to prevent users from encountering "file system full" messages. The **`bdf`** command provides a simple tool for determining the amount of file system free space. The **`bdf`** output has several fields:

`Filesystem`	Block device file of the file system
`Kbytes`	The number of kilobytes of total disk space on the file system
`Used`	The number of kilobytes of disk space used by existing files
`Avail`	The number of kilobytes of available disk space on the file system
`%used`	The percentage of disk space used by files
`Mounted on`	Directory to which the indicated file system is mounted

The **`-i`** option adds three columns to the output that report inode availability, too.

`Iused`	Number of inodes currently in use on the file system
`Ifree`	Number of free inodes on the file system
`%used`	Percentage of inodes used on the file system

© 2004 Hewlett-Packard Development Company, L.P.

When you want to see more details, for example how much space is used beneath a directory, you can do this with the **du** command. By default, **du** shows the amount of space in blocks of 512 bytes. It is recursive, meaning that it starts at the current directory (or *file* specified) and reports on all files and directories from that point on down.

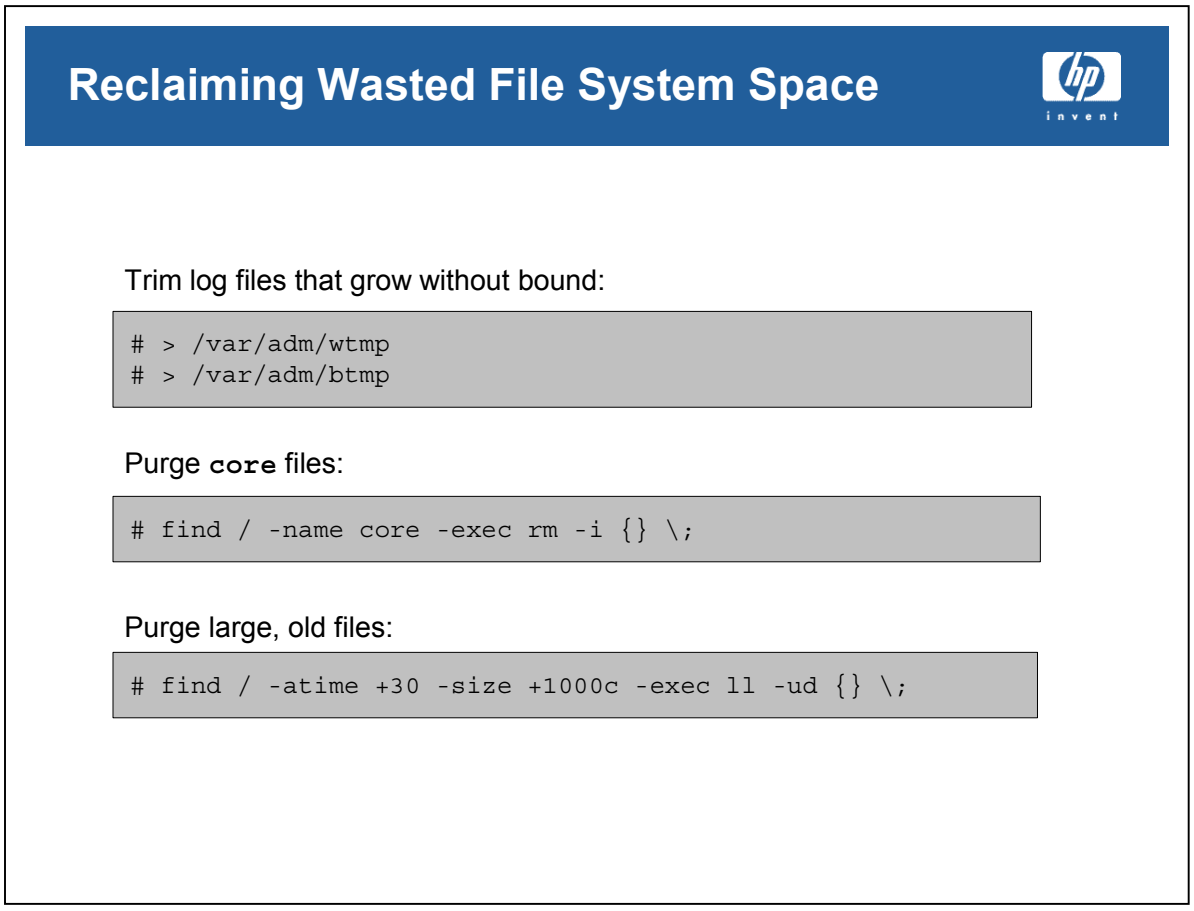
The main options are:

-k	Report output in kilobytes.
-s	Print only the grand total of disk usage for each of the specified directory operands.

For further information refer to the **du**(1) manual entry.

© 2004 Hewlett-Packard Development Company, L.P.

10–2. SLIDE: Reclaiming Wasted File System Space



Student Notes

There are some proactive measures that the System Administrator can take to prevent full file systems.

Trimming Log Files

The **/var** file system is often among the first to cause "file system full" messages. **/var** contains system log files and spool files that can quickly fill the file system if they aren't monitored carefully.

Use the following syntax to purge existing messages from system log files:

```
> logfile
```

This command removes the contents of the log file without changing the file permissions, owner, or group. You may wish to schedule a nightly or weekly batch job to trim your log files automatically on a regular basis. The **cron** job scheduling daemon will be discussed in a future chapter.

© 2004 Hewlett-Packard Development Company, L.P.

SAM has an intuitive interface for interactively trimming log files to reclaim disk space, too:

sam –> Routine Tasks –> System Log Files, select a log file

The SAM interface lets you easily trim selected system log files to a recommended size, a certain number of lines, a percentage of the current size, or size zero. Though SAM initially only lists HP-UX system log files, other application log files can be added to SAM's list as well.

NOTE: Never empty a log file with:

```
# rm logfile
# touch  logfile
```

Doing so may not set the proper permissions for the logfile.

Removing Core Files

A "core" file is occasionally created when a process terminates abnormally as a result of a serious error condition or the QUIT signal. A core file resulting from either of these situations contains a core image of the terminated process that can be used to troubleshoot application bugs.

Core files can be quite large and should be removed from the system when no longer needed. You can easily find and remove core files on the system with the **`find`** command:

```
# find / -name core -exec ll -d {} \;    # list core files
# find / -name core -exec rm -i {} \;    # remove core files
```

You can also use SAM to interactively find and remove core files:

sam –> Routine Tasks –> Selective File Removal

Large, Old Files

Often, users create large files, then forget to remove the files when they are no longer needed. You should periodically search for large files on your system that haven't been recently accessed, and determine if the files are still needed. Removing or archiving large, unneeded files may reclaim a significant amount of disk space.

The **`find`** command can be used for this purpose. The example below finds and lists all files in **`/tmp`** that are over 1000 characters in length, and haven't been accessed in at least 30 days:

```
# find /tmp -atime +30 -size +1000c -exec ll -ud {} \;
```

Print the resulting list of files, then contact the file owners to determine which files can be removed. Note that the **`ll -u`** command lists file access times; without the **`-u`** option, the **`ll`** command lists modification time stamps instead.

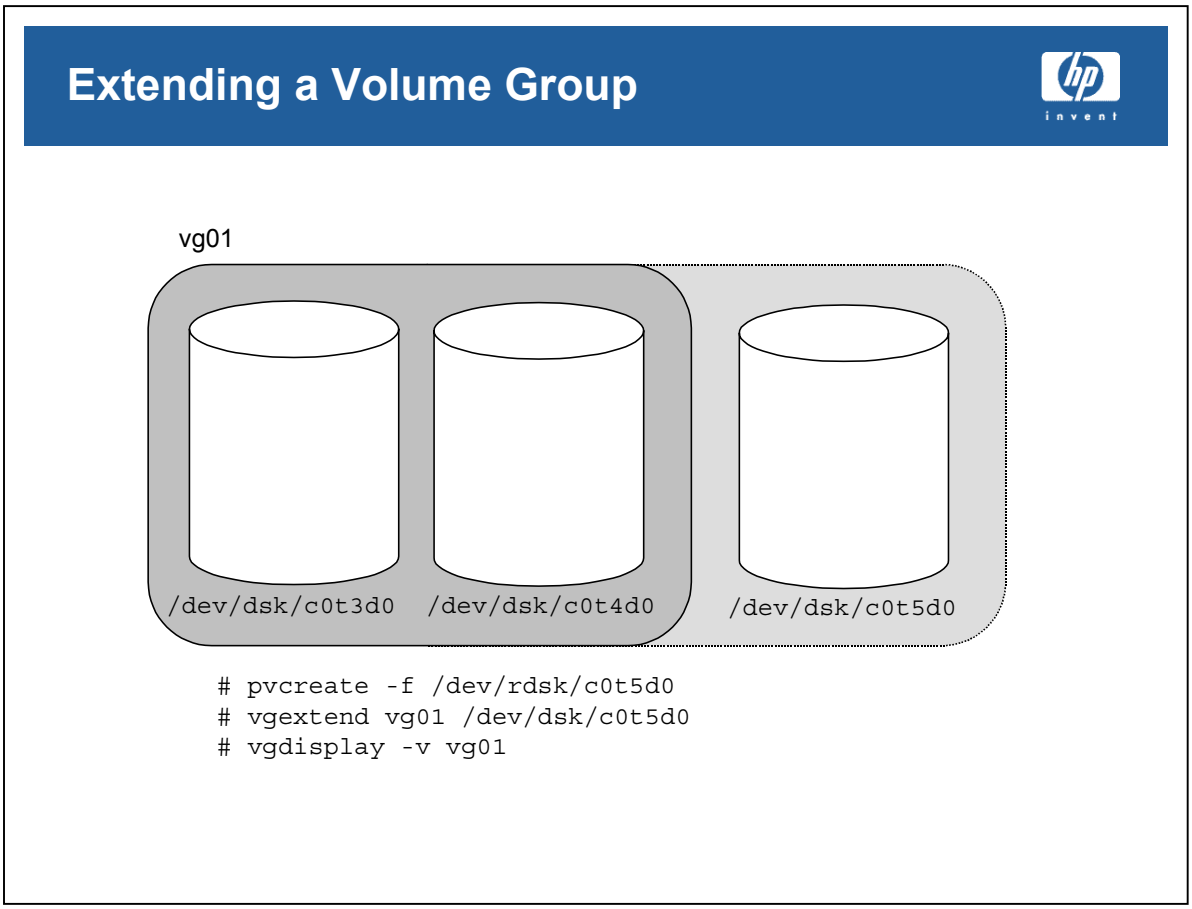
© 2004 Hewlett-Packard Development Company, L.P.

Extending a File System

If a file system reaches 100 percent of capacity and you have already purged all core and large old files, you may need to add additional disk space. The next few slides describe the process required to add a disk to a volume group, extend a logical volume, and extend a file system.

© 2004 Hewlett-Packard Development Company, L.P.

10–3. SLIDE: Extending a Volume Group



Student Notes

In order to extend a file system, it may first be necessary to add a disk to the volume group containing that file system's logical volume.

Adding a Disk to a Volume Group

Adding a disk to a volume group is a two-step process. First, you must **`pvcreate`** the new disk to create the necessary LVM data structures. After creating the necessary LVM data structures on the disk with **`pvcreate`**, you can add the disk to an existing volume group with **`vgextend`**. The example on the slide adds disk **`c0t5d0`** to volume group **`vg01`**:

```
# pvcreate /dev/rdsk/c0t5d0
Physical volume "/dev/rdsk/c0t5d0" has been successfully created.

# vgextend vg01 /dev/dsk/c0t5d0
Volume group "vg01" has been successfully extended.
Volume Group configuration for /dev/vg01 has been saved in
/etc/lvmconf/vg01.conf
```


© 2004 Hewlett-Packard Development Company, L.P.

The **vgextend** command accepts multiple physical volumes as arguments, if you need to add multiple disks to the volume group.

Checking the Volume Group Configuration

You can check to ensure that the disk was successfully added to the volume group with the **pvdisplay** and **vgdisplay** commands. Verify that the new physical volume is included in the **vgdisplay -v** physical volume list and that the number of Free PEs in the volume group has increased. Also note the "VG Name" field in the **pvdisplay** output.

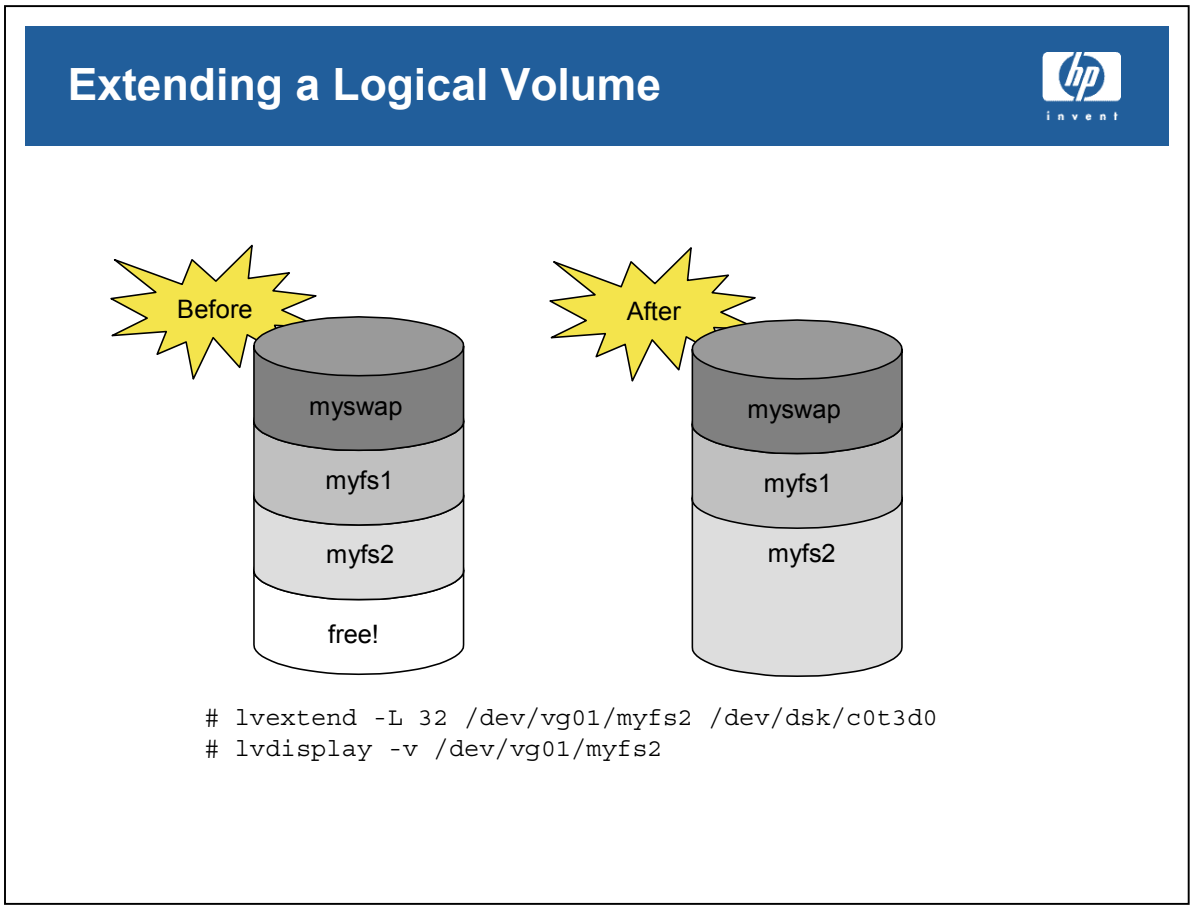
```
# vgdisplay -v vg01
--- Volume groups ---
VG Name                     /dev/vg01
VG Write Access             read/write
VG Status                   available
Max LV                      255
Cur LV                      3
Open LV                     3
Max PV                      16
Cur PV                      3
Act PV                      3
Max PE per PV               1016
VGDA                        6
PE Size (Mbytes)            4
Total PE                    1011
Alloc PE                    12
Free PE                     999
Total PVG                   0
Total Spare PVs             0
Total Spare PVs in use      0

--- Logical volumes ---
 ... LIST OF vg01's LOGICAL VOLUMES ...

--- Physical volumes ---
PV Name                     /dev/dsk/c0t3d0
PV Status                   available
Total PE                    250
Free PE                     238

PV Name                     /dev/dsk/c0t4d0
PV Status                   available
Total PE                    250
Free PE                     250

PV Name                     /dev/dsk/c0t5d0
PV Status                   available
Total PE                    511
Free PE                     511
```


© 2004 Hewlett-Packard Development Company, L.P.

```
# pvdisplay /dev/dsk/c0t5d0
--- Physical volumes ---
PV Name                     /dev/dsk/c0t5d0
VG Name                     /dev/vg01
PV Status                   available
Allocatable                 yes
VGDA                        2
Cur LV                      0
PE Size (Mbytes)            4
Total PE                    511
Free PE                     511
Allocated PE                0
Stale PE                    0
IO Timeout                  default
```


© 2004 Hewlett-Packard Development Company, L.P.

10–4. SLIDE: Extending a Logical Volume



Student Notes

After adding a disk to a volume group, you can allocate the physical extents from the new disk to logical volumes within the volume group. Extending a logical volume requires just one command: **`lvextend`**.

The example on the slide extends the **`/dev/vg01/myfs2`** logical volume from 16 MB to 32 MB. The **`/dev/dsk/c0t3d0`** argument on the end of **`lvextend`** forces LVM to allocate the new physical extents from disk **`c0t3d0`**. If you do not specify where LVM should allocate the new extents, LVM simply uses the first available extents in the volume group. After extending the logical volume, it's a good idea to do an **`lvdisplay`** to see the result.

```
# lvextend -L 32 /dev/vg01/myfs2 /dev/dsk/c0t3d0
Logical volume "/dev/vg01/myfs2" has been successfully extended.
Volume Group configuration for /dev/vg01 has been saved in
/etc/lvmconf/vg01.conf
```


© 2004 Hewlett-Packard Development Company, L.P.

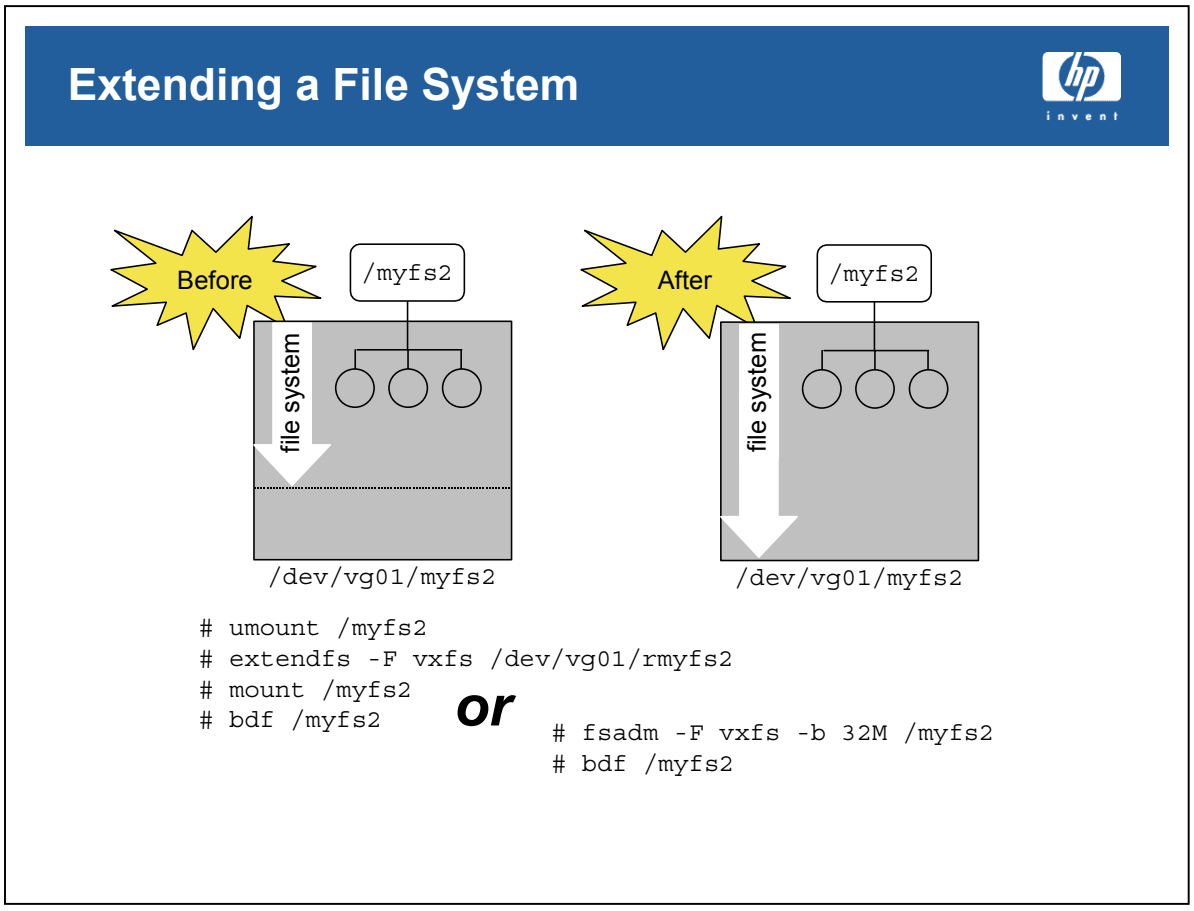
```
# lvdisplay -v /dev/vg01/myfs2
--- Logical volumes ---
LV Name                     /dev/vg01/myfs2
VG Name                     /dev/vg01
LV Permission               read/write
LV Status                   available/syncd
Mirror copies               0
Consistency Recovery        MWC
Schedule                    parallel
LV Size (Mbytes)            32
Current LE                  8
Allocated PE                8
Stripes                     0
Stripe Size (Kbytes)        0
Bad block                   on
Allocation                  strict
IO Timeout (Seconds)        default
    --- Distribution of logical volume ---
    PV Name             LE on PV  PE on PV
    /dev/dsk/c0t3d0     8
      8

    --- Logical extents ---
    LE   PV1                 PE1  Status 1
    0000 /dev/dsk/c0t3d0     0246 current
    0001 /dev/dsk/c0t3d0     0247 current
    0002 /dev/dsk/c0t3d0     0248 current
    0003 /dev/dsk/c0t3d0     0249 current
    0004 /dev/dsk/c0t3d0     0000 current
    0005 /dev/dsk/c0t3d0     0001 current
    0006 /dev/dsk/c0t3d0     0002 current
    0007 /dev/dsk/c0t3d0     0003 current
```

NOTE: A logical volume can span multiple physical volumes, but it cannot span multiple volume groups. A logical volume can only be extended to other disks in the logical volume's volume group.

NOTE: Simply **`lvextend`**ing a logical volume does not extend the file system within a logical volume. Turn to the next slide to learn how to make the new space in the logical volume available to the file system.

© 2004 Hewlett-Packard Development Company, L.P.

10–5. SLIDE: Extending a File System



Student Notes

Simply extending a logical volume does not make the new space available for use by the file system in the logical volume. The new space in the logical volume won't be used until the file system's superblock, and other metadata structures have been notified that the new space is available.

Extending a JFS File System without Online JFS

If you don't have the Online JFS product, you must temporarily unmount a file system any time you wish to extend it. After unmounting the file system, use **`extendfs`** to make additional extents available for use by the file system, then remount and check your file system with **`bdf`**.

Example: Extend the file system in **`/dev/vg01/myfs2`** to fill the 32 MB logical volume.

```
# umount /myfs2
# extendfs -F vxfs /dev/vg01/rmyfs2
# mount /myfs2
# bdf /myfs2
```


© 2004 Hewlett-Packard Development Company, L.P.

The entire **/dev/vg01/myfs2** logical volume is now available for use by the **/myfs2** file system.

Extending a JFS File System with Online JFS

With the Online JFS product, you can extend a file system without unmounting. This proves invaluable in High Availability shops that can't afford downtime.

Example: Extend the file system in **/dev/vg01/myfs2** to fill the 32 MB logical volume.

```
# fsadm -F vxfs -b 32M /myfs2
# bdf
```

Extending an HFS File System

In order to extend an HFS file system, unmount it, and then run **extendfs**.

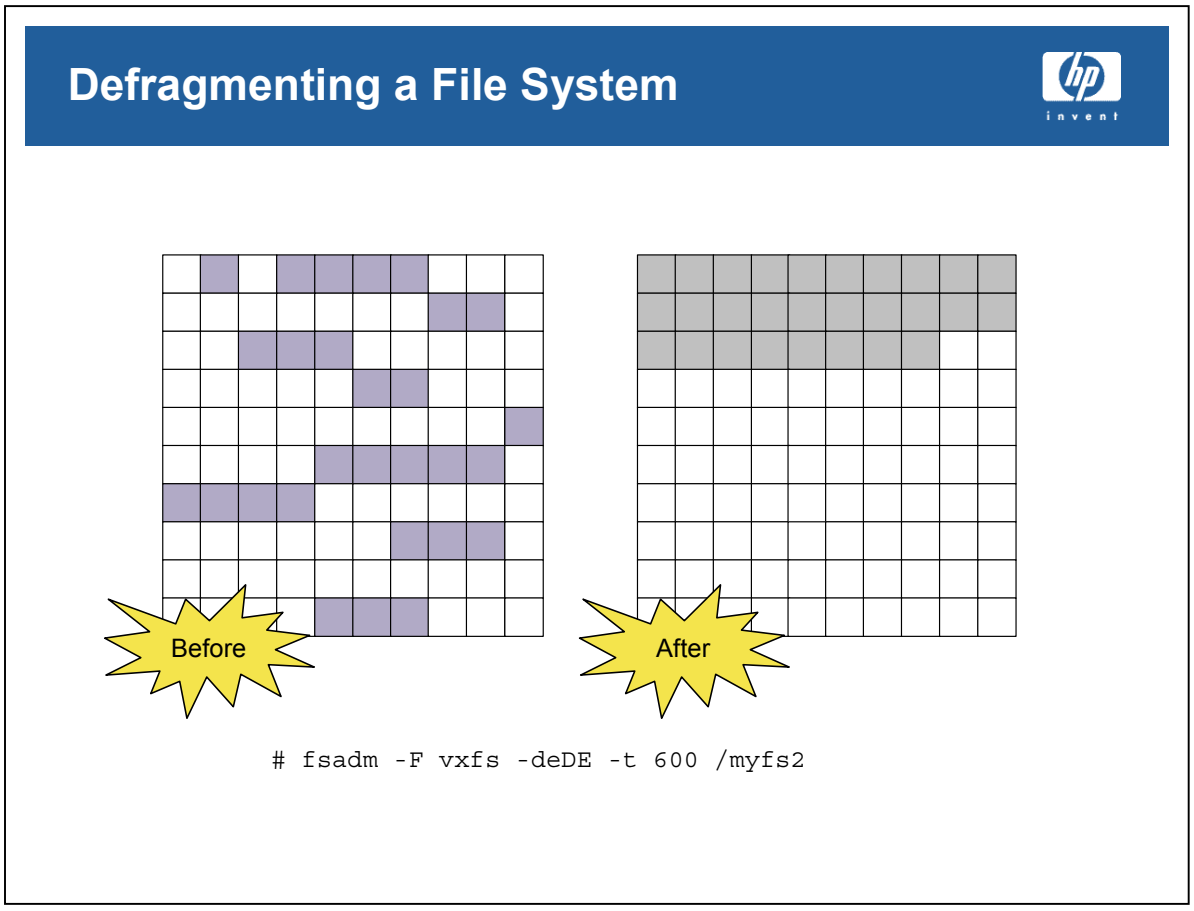
Example: Extend the file system in **/dev/vg01/myfs1** to fill the entire 32 MB logical volume.

```
# umount /myfs1
# extendfs -F hfs /dev/vg01/rmyfs1
# mount /myfs1
# bdf /myfs1
```

The entire **/dev/vg01/myfs1** logical volume is now available for use by the **/myfs1** file system.

© 2004 Hewlett-Packard Development Company, L.P.

10–6. SLIDE: Defragmenting a File System



Student Notes

When a JFS file system is first created, the file system free space is consolidated in a small number of very large extents. However, as files and directories are created, extended, reduced, and removed, the file system gradually becomes "fragmented" into many smaller extents. In a fragmented file system, it is very difficult for JFS to allocate contiguous blocks when existing files are extended, and when large new files are created. Over time, this situation can seriously degrade system performance.

In order to ensure optimal file system performance, HP recommends "defragmenting" JFS file systems on a regular basis using the **`fsadm`** utility. **`fsadm`** attempts to reorganize and consolidate file extents, free extents, and directories to ensure the best possible file system performance.

Defragmenting File Systems with `fsadm`

`fsadm` may be executed several different ways:

```
# fsadm -F vxfs -e /myfs2   re-organize and consolidate extents
# fsadm -F vxfs -d /myfs2   re-organize and optimize directories
# fsadm -F vxfs -de /myfs2  re-organize and optimize directories and extents
```


© 2004 Hewlett-Packard Development Company, L.P.

Note that the file system must be mounted while **fsadm** is running! Users may even be accessing the file system while **fsadm** runs, but performance will be somewhat degraded.

The amount of time required for the defragmentation process to complete varies, depending on the degree of fragmentation, disk speed, and the number of inodes in the file system. If your production schedule allows only a brief period of time for system maintenance activities, you can define a fixed period of time for **fsadm** to execute. **fsadm** then processes as many extents and directories as possible within the given interval, then exits gracefully. The example below defragments the file system for 600 seconds, then terminates:

```
# fsadm -F vxfs -de -t 600 /myfs2
```

To simplify administration and minimize the impact on users, many administrators choose to run **fsadm** automatically as a batch job after hours on a nightly or weekly basis.

Reporting Fragmentation with fsadm

fsadm is able to report, as well as fix, file system fragmentation when executed with two additional options: **-D** reports directory fragmentation, and **-E** reports extent fragmentation. These options may be used independently, or concurrently:

```
# fsadm -F vxfs -E /myfs2      report extent fragmentation
# fsadm -F vxfs -D /myfs2      report directory fragmentation
# fsadm -F vxfs -DE /myfs2     report directory and extent fragmentation
```

Many administrators use **-DE** in conjunction with **-de**:

```
# fsadm -F vxfs -DEde /myfs2
```

This generates a pre-defrag fragmentation report, defragments the file system, then generates a post-defrag report, too. By comparing the pre- and post-defragmentation report values after a day of system activity, a week of system activity, and a month of system activity, you can determine how frequently **fsadm** needs to be executed. If you run **fsadm -DEde** after one week of system activity and see little difference between the pre- and post-values, you may be able to run the utility monthly rather than weekly.

Here's a sample fragmentation report:

```
# fsadm -F vxfs -DE /myfs2
Directory Fragmentation Report
       Dirs        Total      Immed    Immeds   Dirs to   Blocks to
       Searched    Blocks     Dirs     to Add   Reduce    Reduce
       345         812        210      3        6         13

Extent Fragmentation Report
       Total    Average      Average     Total
       Files    File Blks    # Extents   Free Blks
       216      30           1           3973

    blocks used for indirects: 12
    % Free blocks in extents smaller than 64 blks: 29.12
    % Free blocks in extents smaller than  8 blks: 1.33
```


© 2004 Hewlett-Packard Development Company, L.P.

```
       % blks allocated to extents 64 blks or larger: 30.05

   Free Extents By Size
      1:          3            2:          3            4:         11
      8:         24           16:         33           32:         12
     64:          8          128:          6          256:          2
    512:          0         1024:          1         2048:          0
   4096:          0         8192:          0        16384:          0
```

Here are a few of the key fields you should look at:

Immeds to Add: (Smaller is better.)
: Most directories have an inode that records the permissions on the directory and one or more data blocks that record the list of files that are included in the directory. If a directory contains very few files, JFS can actually store the file names in the directory's inode rather than a separate data block. The "Immeds to Add" field reports the number of directories that could be optimized in this manner if **`fsadm`** were executed.

Dirs to Reduce: (Smaller is better.)
: Directory defragmentation attempts to reclaim wasted space in directories. The "Dirs to Reduce" field indicates how many directories might be reduced by directory defragmentation.

Blocks to Reduce: (Smaller is better.)
: Indicates the number of blocks that may be reclaimed as a result of directory defragmentation.

Average # Extents: (Smaller is better, "1" is ideal.)
: Describes the average number of extents occupied by each file.

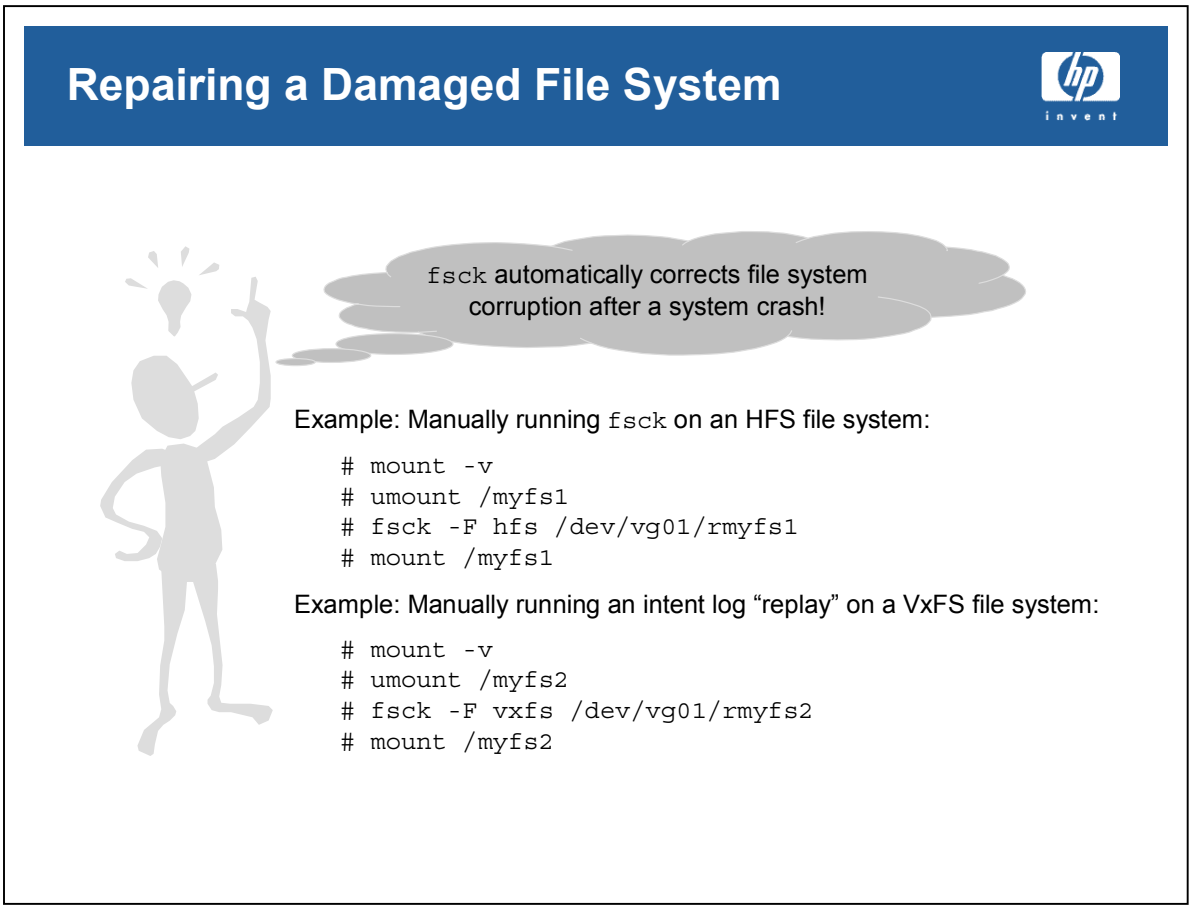
% Free blocks in extents smaller than 64 blks: (Smaller is better and should be less than 50%.)
: JFS prefers to have a few large extents of free blocks rather than many smaller extents of free blocks. Larger free-block extents make it easier for JFS to allocate space for new files.

% Free blocks in extents smaller than 8 blks: (Smaller is better.)
: JFS prefers to have a few large extents of free blocks rather than many smaller extents of free blocks. Larger free-block extents make it easier for JFS to allocate space for new files.

For a more detailed description of JFS fragmentation issues, and the **`fsadm`** command syntax, see the **`fsadm_vxfs(1m)`** man page.

© 2004 Hewlett-Packard Development Company, L.P.

10–7. SLIDE: Repairing a Damaged File System



Student Notes

Occasionally in HP-UX, an improper shutdown or system crash can leave one or more file systems in a corrupted state. During the system boot process, HP-UX automatically checks to ensure that all file systems were properly unmounted during the last system shutdown. If a file system was not properly unmounted, HP-UX automatically runs the **`fsck`** "file system check" utility to identify and correct corrupted metadata structures. In most cases, **`fsck`** will fix the corrupted file systems during the boot process without any interaction with the system administrator. However, if extreme file system corruption occurs, you may be asked to run **`fsck`** manually. The procedure for running **`fsck`** manually is described below:

1. Start by issuing the **`mount -v`** command to determine which file systems are mounted where. Also note which file systems are HFS and which are JFS. You will need to know the file system type when you run **`fsck`**.

   ```
   # mount -v
   ```


© 2004 Hewlett-Packard Development Company, L.P.

2. **fsck** should be run on quiescent file systems; unmount the file system(s) before proceeding.

```
# umount /dev/vg01/myfs1
# umount /dev/vg01/myfs2
```

3. Run **fsck**. You can run **fsck** on both HFS and JFS file systems. Just be sure to specify the proper file system type. For optimal performance, specify the raw device file of the logical volume or disk containing the file system to check.

```
# fsck -F hfs /dev/vg01/rmyfs1
# fsck -F vxfs /dev/vg01/rmyfs2
```

   When run on a JFS file system, **fsck** simply replays the intent log, and completes any pending transactions. See the detailed explanation of JFS options in the notes below for more information. When run on an HFS file system, **fsck** examines the specified file system a number of times, examining a different file system structure with each pass. When **fsck** identifies a file system inconsistency, it reports the problem, and asks if corrective action should be taken.

   If the administrator answers "yes," **fsck** attempts to fix the problem. If the administrator answers "no," **fsck** ignores the inconsistency and continues. There are very few occasions when a "no" response is appropriate.

4. Once **fsck** completes, remount the file system(s).

```
# mount /dev/vg01/myfs1
# mount /dev/vg01/myfs2
```

5. Restore any corrupted files. In order to fix severely corrupted file systems, **fsck** may have to remove one or more files. Watch for "REMOVE" messages in the **fsck** output, and be sure to restore the affected files from tape.

   If you see any "RECONNECT" messages while running **fsck**, you may need to investigate one or more "orphaned" files. Orphaned files are files that have valid inodes and data blocks but no corresponding directory entries. **fsck** places these files in a **lost+found** directory that is created at the root of each file system. Hopefully, by looking at the orphaned files' attributes and contents you will be able to determine their proper filenames. The following commands might be used to investigate the orphaned file in the **/myfs2** file system under inode #1743:

```
# cd /myfs2/lost+found          go to the lost+found directory
# ll \#1743                     determine who owns the orphaned files
# file \#1743                   check the file types
# strings \#1743                check the file contents
# mv \#1743 new_file_name       move & rename the orphaned files as appropriate
```


© 2004 Hewlett-Packard Development Company, L.P.

Some General `fsck` Options

There are several common options that may be used on **`fsck`** when checking either HFS or JFS file systems:

`-n` Assume a "no" response to all questions from **`fsck`**; **`fsck`** checks the file system but doesn't correct inconsistencies. You can use this option to assess the state of a file system, but then be sure to run **`fsck`** again to actually fix any identified problems.

`-y` Assume a "yes" response to all questions. Beware that data may be removed as a result of a "yes" answer. Consider using this option after running **`fsck`** with **`-n`** to assess the state of your file systems.

Some Special HFS `fsck` Options

The following **`fsck`** options only apply to HFS file systems:

`-b` *block#* This option tells **`fsck`** to use the superblock at the specified block# as the superblock for the file system check instead of the default superblock. This is useful if the primary superblock becomes lost or corrupted. All backup superblock locations are written to the **`/var/adm/sbtab`** file. If you can't access this file, try block #16, which always contains the first alternate superblock. If you seem to be getting a lot of strange errors from **`fsck`**, try using this **`-b`** option.

`-f` By default, **`fsck`** gives a warning message and requests confirmation when run against a mounted file system. **`-f`** forces **`fsck`** to run, even if the specified file system is mounted. This option should only be used in single-user mode. After running **`fsck`**, reboot immediately with the **`-n`** option. By default, the **`reboot`** and **`shutdown`** commands flush buffer cache, which may overwrite the corrections made by **`fsck`**. Reboot using **`reboot -n`** to take the system down without flushing buffer cache.

Some Special JFS `fsck` Options

Because of the JFS intent log mechanism, JFS file system metadata should not be corrupted by an improper shutdown. After an improper shutdown, **`fsck`** need only complete any pending intent log transactions to bring a JFS file system to a consistent state. This is known as an intent log replay. **`fsck`** requires minutes or even hours to repair an HFS file system, but can complete a JFS intent log replay in a matter of seconds.

If you wish, you can force **`fsck`** to bypass the intent log and do a full check of all the metadata structures in a JFS file system with the **`-o full,nolog`** option. This may be useful if the intent log area is physically damaged, or other more serious structural problems exist with the file system metadata.

© 2004 Hewlett-Packard Development Company, L.P.

Sample `fsck` Output

```
# mount -v
/dev/vg01/myfs1 on /myfs1 type hfs defaults on Fri Jun 26 15:55:10 1998
/dev/vg01/myfs2 on /myfs2 type hfs delaylog on Fri Jun 26 15:18:55 1998

# umount /myfs1
# umount /myfs2

# fsck -F hfs /dev/vg01/rmyfs1

** /dev/vg01/rmyfs1
** Last Mounted on /myfs1
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
UNALLOCATED  I=397
   OWNER=root MODE=0
SIZE=0 MTIME=Dec 31 18:00 1969
NAME=/data1/last

REMOVE? y
UNALLOCATED  I=397
 OWNER=root MODE=0
SIZE=0 MTIME=Dec 31 18:00 1969
NAME=/data1/lastb
REMOVE? y

** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
FREE INODE COUNT WRONG IN SUPERBLK
FIX? y

** Phase 5 - Check Cyl groups
20 BLK(S) MISSING
BAD CYLINDER GROUPS
FIX? y

** Phase 6 - Salvage Cylinder Groups
110 files, 0 icont, 13987 used, 33842 free (50 frags, 4224 blocks)

***** FILE SYSTEM WAS MODIFIED *****

# fsck -F vxfs /dev/vg01/rmyfs2
file system is clean - log replay is not required

# mount /myfs1
# mount /myfs2
```

NOTE: There are several **`fsck`** man pages:

- **`fsck(1m)`** describes general **`fsck`** features
- **`fsck_hfs(1m)`** describes HFS-specific **`fsck`** features.
- **`fsck_vxfs(1m)`** describes JFS-specific **`fsck`** features

© 2004 Hewlett-Packard Development Company, L.P.

10–8. LAB: File System Management

Directions

Perform the tasks suggested below. Record the commands you use and the answers to all of the questions.

Part 1: Purging and Extending File Systems

1. Occasionally, the **`/tmp`** file system fills up, causing problems on a system. Using SAM or the command line, list all of the files in **`/tmp`** that haven't been accessed within the last 2 days.

2. **`/var`** is another file system which sometimes reaches 100%. Often this is a result of log files that haven't been properly trimmed. Trim the following log files back to size 0:

   ```
   /var/adm/btmp
   /var/adm/wtmp
   ```

3. The questions that follow give you an opportunity to fix a full file system. You should have a script on your system called **`/labs/fillfs.sh`**. The **`fillfs.sh`** script will fill one of your file systems to capacity. Run **`fillfs.sh`**. As the script executes, you may see a number of file system full messages scroll across your screen. Don't worry -- yet!

4. Which file system appears to be full? How many KB were allocated for this file system? What percent of the space in that file system is in use?

© 2004 Hewlett-Packard Development Company, L.P.

5. What happens at this point if a user tries to copy a file to the full file system? Is anything recorded in **`syslog.log`**? Copy a large file (such as **`/stand/vmunix`**) to **`/home`** to find out.

6. List two possible solutions to this full file system problem.

7. Are there any core files in the problem file system that could be removed? If so, remove them. What commands did you use to find the core files?

8. Which directory under **`/home`** is taking the most space? Which command can you use to find out? Mail a message to the culprit asking him or her to purge some files.

© 2004 Hewlett-Packard Development Company, L.P.

9. You could wait for your users to purge some old files, but in many cases, you will eventually need to add some additional space to the full file system. **`pvcreate`** your free disk and add it to vg00. Use **`vgdisplay`** to ensure that the new disk was successfully added to the volume group.

10. Double the size of the logical volume containing **`/home`**. Which disk contains the new extents? What command did you use to find out?

11. Using **`bdf`**, check to see if the **`/home`** file system contained in the logical volume you just extended increased in size. Does extending a logical volume automatically extend the file system within the logical volume?

12. Execute the commands necessary to extend the **`/home`** file system to take advantage of the additional space in the logical volume.

© 2004 Hewlett-Packard Development Company, L.P.

13. Unmount the file system and try doing **`extendfs`** on the logical volume containing **`/home`** again. Explain the resulting message.

14. Before moving on, remove all the "bigfiles" from user5's home directory.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Defragmenting File Systems

1. Run a directory and extent fragmentation report on the **`/var`** file system. At this point, don't do anything to repair the fragmentation. Record the following values:

   a. Directories to reduce:
   b. Average number of extents per file:

2. Defragment the **`/var`** file system.

3. How successful was the defragmentation utility? Run a second fragmentation report and compare the results to the values you recorded above in question 1. Do you see any improvement?

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Repairing HFS File Systems

In the next couple portions of the lab, you will have an opportunity to run **fsck** on a number of corrupted file systems via an **fsck** simulation program. You may run the simulator by typing: **/labs/simfsck**. The **simfsck** simulator contains five file systems:

```
/dev/vg00/lvol1     HFS     /
/dev/vg01/app       HFS     /app
/dev/vg01/cad       HFS     /cad
/dev/vg02/db        HFS     /db
/dev/vg02/data      VXFS    /data
```

The simulator recognizes all of the UNIX commands necessary to run **fsck** on the above file systems. Note that the simulator only recognizes the most common options on each of these commands.

```
bdf
cat
cd
file
fsck
ls
ll
mount
pwd
strings
umount
```

Run the simulations suggested below. Record the commands you use, and answer all the questions. After gaining some experience with **fsck** in the simulator, the last exercise asks you to run **fsck** on some "real" file systems as well.

1. Once in the simulator, choose simulation #1 from the menu. Simulation #1 asks you to run **fsck** on the **/app** file system. You will need to know the name of the logical volume containing the **/app** file system when you run **fsck**, so start by viewing a list of the currently mounted file systems.

2. File systems must be unmounted before running **fsck**. Unmount the **/app** file system.

© 2004 Hewlett-Packard Development Company, L.P.

3. Now run **fsck** on the raw logical volume containing **/app**. Be sure to specify the file system type. **fsck** will check the file system and ask which problems should be corrected.

   Although one could conceivably answer "no" in response to **fsck's** questions, you should generally answer "yes" -- otherwise you will be left with a still-corrupted file system. Take note of the problems identified by **fsck**, but go ahead and answer "yes" to all of the prompts.

4. Once **fsck** terminates, remount the file system.

5. After mounting the file system, you should check the **lost+found** directory to see if **fsck** identified any orphaned files. Are there any orphaned files in this case?

6. Did the **fsck** prompts indicate that **fsck** "REMOVE"d any files from **/app**? If you have a tape backup of the file system, you should restore these files at this point. In the case of our simulator; however, we don't have a tape backup so any removed files are lost forever.

7. Go back to the main simulator by typing: **menu**.

© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Repairing HFS File Systems, Revisited

1. Choose simulation #2, which asks you to run **`fsck`** on the **`/db`** file system.
   Enter choice: 2

2. Following the same procedure used in the previous exercise, **`fsck`** the **`/db`** file system and fix all the corruption identified.

3. Are there any files in **`/db/lost+found`** that need attention? If there are any files in **`lost+found`**, what can you do to find the file's owner and file name?

4. Did **`fsck`** "REMOVE" any files this time?

5. Return to the simulator menu by typing “menu”.

© 2004 Hewlett-Packard Development Company, L.P.

Part 5: Repairing JFS File Systems

1. Run simulation #3, which asks you to run **`fsck`** on the **`/data`** file system. Enter choice: 3

2. What happens if you specify the wrong file system type to **`fsck`**? Unmount the **data** file system, and try running **`fsck -F hfs /dev/vg02/rdata`**. What happens? Despite the worrisome message **`fsck`** offers here, the superblock really isn't damaged. Can you guess why you get this message?

3. Try running **`fsck`** again with the **`-F vxfs`** option. How does running **`fsck`** on a VxFS file system differ from running **`fsck`** on an HFS file system?

4. After running **`fsck`**, remount the file system and return to the main simulator menu.

© 2004 Hewlett-Packard Development Company, L.P.

Part 6: Repairing JFS File Systems, Revisited

1. Run simulation #4, which asks you to run a "full" **`fsck`** on the **`/data`** file system. Menu choice: 4

2. You saw in the previous simulation that, by default, **`fsck`** simply replays the JFS intent log. After a system crash, an intent log replay is all that is required. An intent log replay simply scans the intent log and completes any pending transactions; it does not check the consistency of your superblock, inodes, and allocation units.

   If you suspect more serious file system corruption in a JFS file system, you can perform a full check of the file system. Try the following:

   ```
   Sim: mount -v
   Sim: umount /data
   Sim: fsck -F vxfs -o full /dev/vg02/rdata
   Sim: mount /data
   Sim: ll /data/lost+found
   ```

3. How did this **`fsck`** differ from the **`fsck`** in the previous JFS simulation?

© 2004 Hewlett-Packard Development Company, L.P.

Part 7: Repairing Real File Systems

1. Now that you have had an opportunity to run **`fsck`** in the simulator a few times, try running **`fsck`** on some real file systems. Do a **`mount -v`** to determine which file systems are mounted where on your system.

2. You should have an HFS file system mounted on **`/stand`**, and a JFS file system mounted on **`/home`**. Unmount both of these file systems, and run the appropriate **`fsck`** commands to check them both. Does **`fsck`** identify any problems?

3. If you haven't already remounted your file systems, do so now.

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 11 — File System Backup

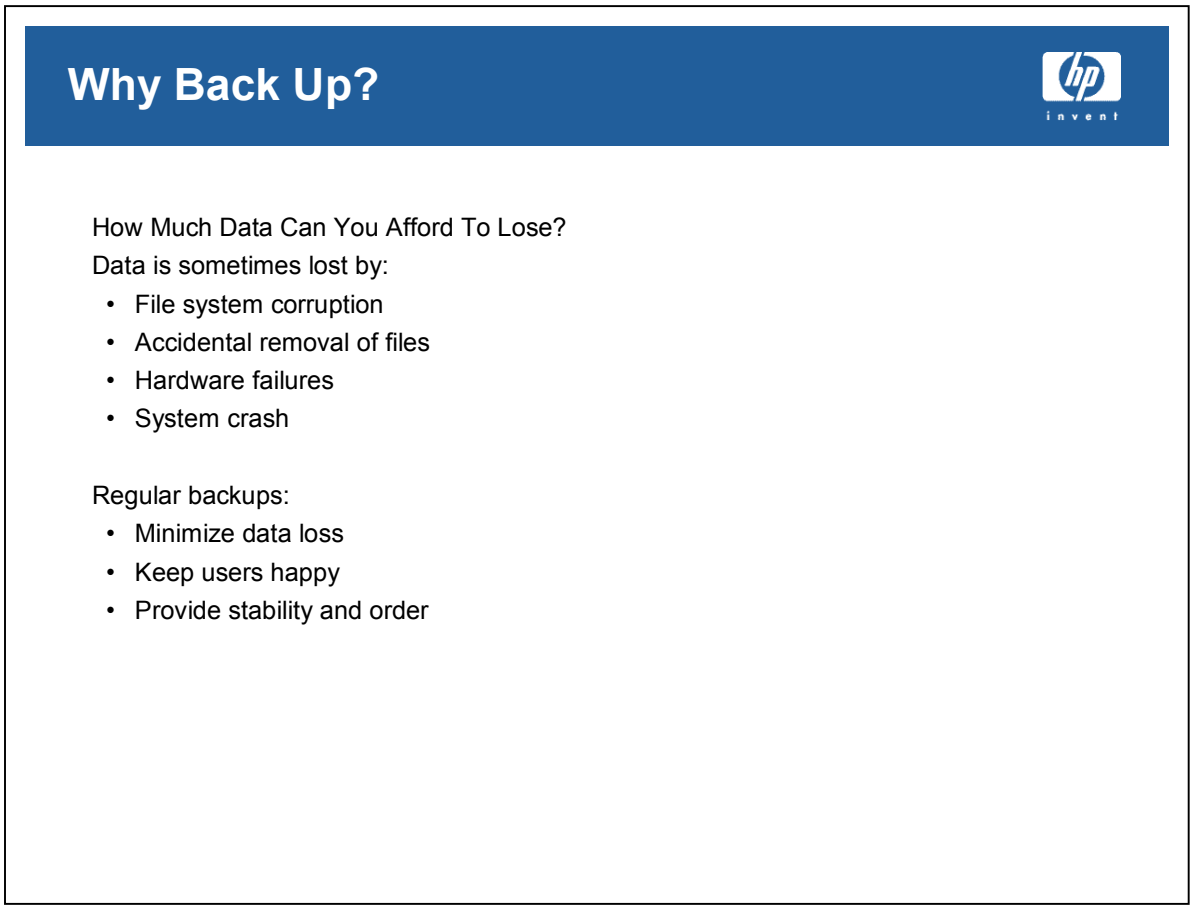
Objectives

Upon completion of this module, you will be able to do the following:

- Explain why backups are necessary.
- Create a graph file to determine which files are included in the system backup.
- Perform a full backup with **`fbackup`**.
- Perform an incremental backup with **`fbackup`**.
- Perform a backup across the network with **`fbackup`**.
- Create a system recovery tape with **`make_tape_recovery`**.
- List steps needed to document the system configuration to ensure a smooth recovery in case of a system crash.

© 2004 Hewlett-Packard Development Company, L.P.

11–1. SLIDE: Why Back Up?



Student Notes

One of the principal responsibilities of a system administrator is preserving the data stored on the system. Unfortunately data is sometimes lost. A piece of hardware may fail, a file may be accidentally removed or overwritten, a command may go astray, or the system may crash. The user community has a reasonable expectation that the administrator has planned and implemented regular backup procedures to minimize data loss.

To further minimize the chance of data loss, all backup media should be stored at a location geographically distant from the system's disk drives. Picture the scenario of a multi-year project under development and a system administrator who has dutifully maintained system backups since the project's inception. However, for ease of retrieval, the backup media are stored in the same room as the computer system. If a fire, flood or other disaster destroyed the computer room, all the work done on the project would be lost.

Consequently, the safest course of action is to store the backup media in a secure environment separate from the computer equipment. In some cases this may mean the next room, a data vault on the premises, or even a data vault at another site.

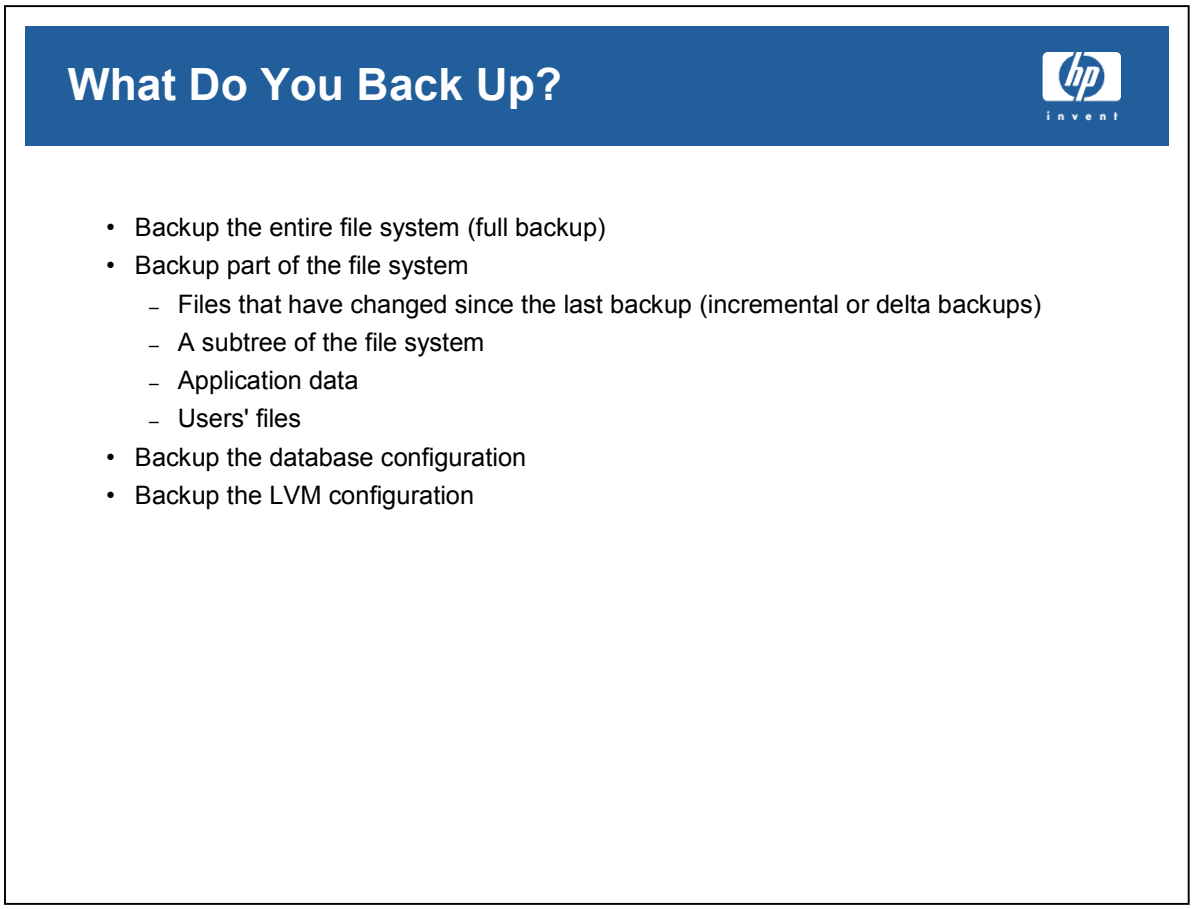
© 2004 Hewlett-Packard Development Company, L.P.

The exact backup procedure employed is determined by a number of factors. A heavily used system, both in terms of number of users and amount of activity, may require some form of backup to be conducted daily. A lightly used system may only need to be backed up weekly or bimonthly. Media used for backups may prove to be expensive. If complete system backups are performed daily, the costs in terms of materials and personnel may grow to be significant. Consequently, a weekly or bimonthly backup could result in minimal costs added to the maintenance of the system.

In this module, we'll look at different backup strategies. Each has its advantages and disadvantages. A backup strategy should be implemented as soon as users begin to work on the system.

© 2004 Hewlett-Packard Development Company, L.P.

11–2. SLIDE: What Do You Back Up?



Student Notes

Before we discuss how to back up the system, we must first discuss what to back up, and when. Doing a full back up of all files and directories under the `/` directory is perhaps the easiest solution.

On larger systems, however, daily full backups may not be practical:

- A full backup may take too long. It may not be possible to complete a full backup overnight before users return the next morning.
- A full backup may consume multiple tapes, and require an operator to swap tapes during the backup. Both of these requirements can be expensive.
- A full backup may not even be necessary. Some file systems, such as `/tmp` and `/cdrom`, don't need to be included in the backup. You can significantly decrease the size and duration of your backups by carefully selecting the directories to include.

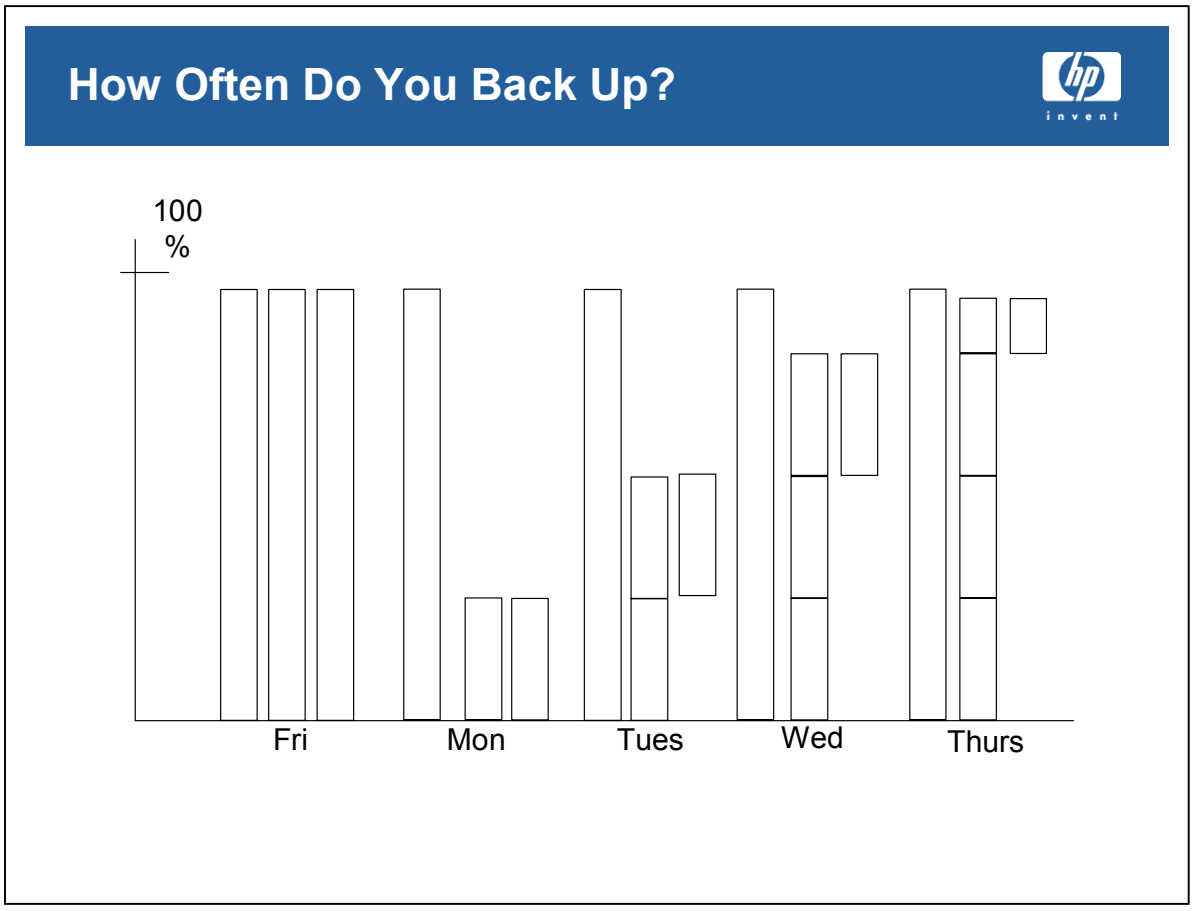
For all of these reasons, many administrators choose to back up only a portion of the file system on a daily basis. The list on the slide suggests several possible approaches. In addition to the files and directories in your file systems, you may need to follow special backup procedures for your database configuration and partitions. Finally, if an LVM disk

© 2004 Hewlett-Packard Development Company, L.P.

has to be replaced, you will need to have a backup of the LVM configuration. Anytime you add, remove, or extend a logical volume, the **`vgcfgbackup`** command automatically writes a backup of the LVM configuration out to the **`/etc/lvmconf`** directory. Make sure **`/etc/lvmconf`** is included in your regular tape backups.

© 2004 Hewlett-Packard Development Company, L.P.

11–3. SLIDE: How Often Do You Back Up?



Student Notes

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 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 the contents of the file?

How often you make backups depends on how much data you can afford to lose. If you can afford to lose a month of data, then you need only back up the system once each month. If you can only afford to lose 6 hours of data, then you must back up every 6 hours. However, backing up every 6 hours can become prohibitive, and other possibilities (such as redundant systems) must be considered. For most applications, full backups once each week and partial backups each night are sufficient.

© 2004 Hewlett-Packard Development Company, L.P.

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. Ideally, 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.

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 has 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, whichever 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. You can define up to ten backup levels (0 - 9). The backup strategy you choose should be determined by 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 the following backup levels:

level 0 full monthly backup

level 1 weekly backup on Friday

level 2 daily backup, except Friday

The following table 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 Week:	Su	M	T	W	Th	F	Sa	Su	M	T	W	Th	F	Sa	Su	…	F
Backup Level:	0	2	2	2	2	1	2	2	2	2	2	2	1	2	2	…	0

If your data became corrupt on Thursday the 12th, then on Friday the 13th you would use the following sequence to restore your system to its Wednesday, the 11th state:

- Restore the monthly full backup from Sunday the 1st.
- Restore the weekly incremental backup from Friday the 6th.
- Restore the incremental backup from Wednesday the 11th.

© 2004 Hewlett-Packard Development Company, L.P.

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. The figure 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 Week:	Su	M	T	W	Th	F	Sa	Su	M	T	W	Th	F	Sa	Su	…	F
Backup Level:	0	1	1	1	1	1	1	0	1	1	1	1	1	1	0	…	0

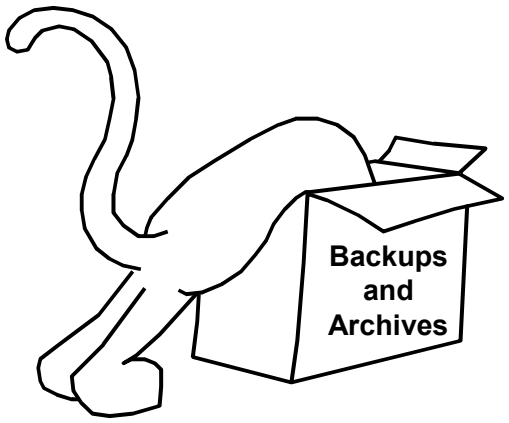
If your data became corrupt on Thursday the 12th, then on Friday the 13th you would use the following sequence to restore your system to its Wednesday the 11th state:

- Restore the full backup from Sunday the 8th.
- Restore the incremental backup from Wednesday the 11th.

© 2004 Hewlett-Packard Development Company, L.P.

11–4. SLIDE: How Do You Perform the Backup?

How Do You Perform the Backup?



- `fbackup`/`frecover` are the most flexible standard HP-UX backup utilities.
- Other utilities are available as well.

Student Notes

There are many commands and utilities you can use to save files to backup media. Some are more flexible than others. Some are limited to only certain types of file systems. The chart on the following page compares the standard HP-UX backup utilities.

Although the utilities listed in the following table are adequate for small to mid-size systems, administrators that are responsible for many gigabytes of data, multiple servers, and heterogeneous environments, often prefer to use HP's purchasable Openview/Omniback II product. The bullets below highlight a few of Omniback's many features:

- Multi-platform support. Omniback II may be used to backup HP-UX, NT, Solaris, Linux, and many other systems.
- Network based. Omniback II can backup multiple network clients in parallel.
- Centralized management. Backups can be configured, scheduled, and monitored remotely from a central backup management station.
- GUI based. Omniback II includes a simple, intuitive GUI interface for configuring and scheduling all backups.
- Extensive device support. Omniback II supports many popular tape libraries and robotic mechanisms.
- Application integration. Omniback II includes integration packages to simplify backups of Oracle, Sybase, Informix, and many other popular applications.

© 2004 Hewlett-Packard Development Company, L.P.

For more information, search for Omniback II on the **http://www.hp.com** website.

Task	Backup Utility				
	fbackup/ frecover	**cpio**	**tar**	**dump/restore** (HFS only)	**vxdump/vxrestore** (2) (**vxfs** only)
Supported on other UNIX platforms	no	yes	yes	yes	yes
Do an incremental backup	Has a powerful multilevel backup	Use find to locate new or modified files	Use the **-u** option to add any new or modified files to the end of the archive	Possible on a single file system only	
Backup/restore large files (>2GB)	Possible	Not possible	Not possible	Not possible	Not possible
Backup/restore across a network	Possible	Possible	Possible	Possible	Possible
Find a particular file	Relatively easy; use **frecover**	Moderate; Wildcards are allowed, but searches the entire tape.	Relatively difficult; Wildcards not allowed; searches the entire tape.	Relatively easy; interactive commands available	Relatively easy; interactive commands available
Multiple, independent backups on a single tape	Not possible (**fbackup** rewinds the tape)	Use mount with no-rewind device to position the tape, then use **cpio**	Use mount with no-rewind device to position the tape, then use **tar**	Use **mount** with no-rewind device to position the tape, then use **dump**	Use **mount** with no-rewind device to position the tape, then use **vxdump**
Verify backup	Use the **-xNv** options	Not possible	Not possible	Not possible	Not possible

© 2004 Hewlett-Packard Development Company, L.P.

Task	Backup Utility				
	`fbackup/ frecover`	`cpio`	`tar`	`dump/restore` (hfs only)	`vxdump/vxrestore` (vxfs only)
List files as they are backed up or restored	Possible; use the `-v` option on the command	Possible; use the `-v` option on the command	Possible; use the `-v` option on the command	Possible (on a restore only)	Possible (on a restore only)
Do a backup based on selected criteria (such as group)	Not possible	Possible; use find	Not possible	Not possible	Not possible
Cross disk or file system boundaries	Use `fbackup -n` to cross NFS boundaries	Possible; use find	Possible	Not possible	Not possible
Restore absolute path names to relative location	Relative to the current directory; use `-X`	Limited; can specify path/name on each file with `cpio -ir`	Not possible	Relative to the current directory; use `restore -r`	Relative to the current directory; use `vxrestore -r`
Interactively decide on files to restore	Not possible	Can specify path or name on each file with `cpio -ir`	"yes" or "no" answer possible using `tar -w`	In interactive mode can specify which files	In interactive mode can specify which files
Use wildcards when restoring	Not possible	Possible	Not possible	Only in interactive mode	Only in interactive mode
Ease of selecting files for backup from numerous directories	High	Medium	Low	Not possible	Not possible
Backup a snapshot file system	Not possible; (`fbackup` gives error messages when backing up read-only files	Possible	Possible	Possible	Possible
Backup/restore extent attributes	Possible	Not possible	Not possible	Not possible	Possible

© 2004 Hewlett-Packard Development Company, L.P.

The utilities covered in the chart all backup files. The following utility performs a byte for byte backup:

`dd` The **`dd`** command is useful in some limited situations, but is technically *not* a backup command. The **`dd`** command is a general purpose physical file copy utility. This is different from all the above utilities in that **`dd`** copies no file names or file attributes to a backup media; it simply copies everything, bit for bit. Thus no selective restore is possible. For these reasons, **`dd`** is *not* recommended for regular system backups. It is generally used for two purposes:

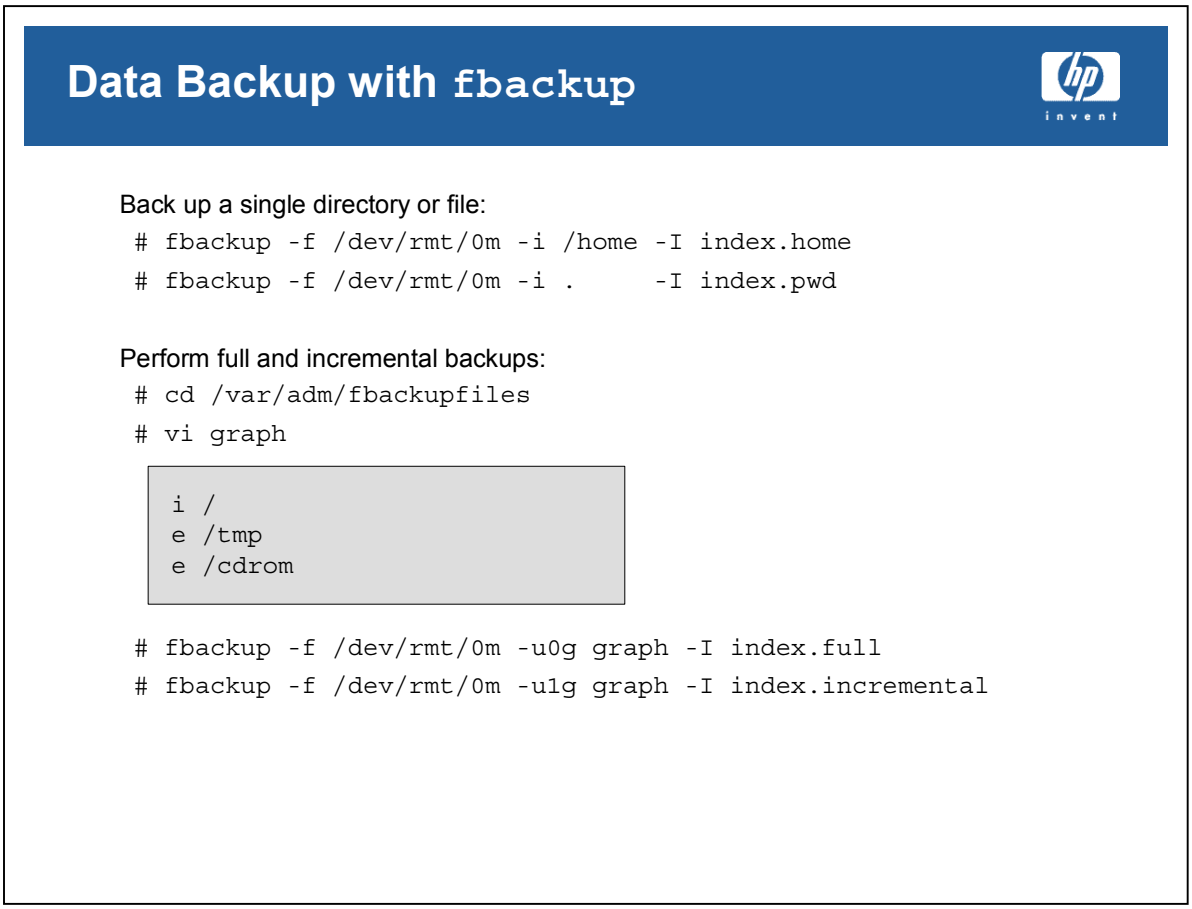
- Make a duplicate copy of a disk quickly. This assumes a destination disk the same size or larger than the source.

- Read or translate a foreign 9-track magnetic tape. For example, **`dd`** has the capability for reading backup media with user-defined record sizes, ASCII or EBCDIC translation, byte switching, and other useful options.

NOTE: Since there are many commands available, and many options to each of these commands, it is strongly recommended that you write the command you use to create your backup on the label of your backup media.

© 2004 Hewlett-Packard Development Company, L.P.

11–5. SLIDE: Data Backup with `fbackup`



Student Notes

`fbackup` is the tool of choice in HP-UX for creating both full and incremental backups. The slide above and notes below describe the most commonly used **`fbackup`** options.

`fbackup` Options

`-f` *`device`* Determines the destination device. Most system administrators write their backups to DDS or DLT tape drives. In this case, specify the appropriate **`/dev/rmt`** device file. Alternatively, you may backup to a file on disk (just specify the desired pathname), or standard out (specify **`-`**).

`-i` *`path`* Identifies a file or directory to include in the backup.

`-e` *`path`* Identifies a file or directory to exclude from the backup.

© 2004 Hewlett-Packard Development Company, L.P.

-g ***graph***	File that contains a list of files and directories to be included or excluded from the backup. When backing up just one or two files or directories, use **-i** and **-e**. For more complicated backups, use a graph file. Graph files are described in detail below.
-0-9	Backup level, used for incremental backups. Backup levels are described in greater detail below. The default backup level is 0. (Only useful in conjunction with **-g**)
-u	Update the **/var/adm/fbackupfiles/dates** file, which records which graph files were used when. (Only useful in conjunction with **-g**)
-I ***path***	Create an index file at the designated path. Index files are described in detail below.

`fbackup` Backup Levels

The **[0-9]** option allows the user to define and use “backup levels”. Recall that an incremental backup strategy only makes backup copies of files that have changed. The key question to answer is "changed since when"? The backup level option answers this question.

Level 0	Is predefined to mean "the beginning of time" by the **fbackup** command. If invoked without a backup level option, **fbackup** performs a level 0 backup that results in a full backup.
Levels 1-9	Are used for incremental backups. Each time **fbackup** is invoked with a backup level option, **fbackup** makes copies of files that have changed since the last time a backup was made at a lower level. For example, suppose **fbackup** is run at level 0 on Monday, level 1 on Tuesday, and level 2 on Wednesday. A full backup would be made on Monday, an incremental backup of files that had changed since Monday would be made on Tuesday, and an incremental backup of files that had changed since Tuesday would be made on Wednesday.

It is up to the system administrator to assign a meaning to each backup level. For example, a backup strategy that calls for full backups on Monday and incremental backups dating back to the most recent full backup on Tuesday, Wednesday, Thursday, and Friday could be implemented by making a level 0 backup on Monday and level 5 backups on Tuesday, Wednesday, Thursday, and Friday. Selecting level 5 for the incremental backups provides some flexibility in the backup scheme, should changes be required.

© 2004 Hewlett-Packard Development Company, L.P.

`fbackup` Graph Files

HP-UX file system trees can get very large. Multiple physical disks can be mounted under a single file system to produce a logical file system tree that is enormous. For this reason, it is desirable to have a mechanism to specify only parts of the file system to be backed up. It may also be the case that part of a file system is entirely static and may require only occasional backup (for example, monthly or semiannually). Simply put, you may wish to specifically include or exclude parts of the file system during backup.

One approach is to specify which files to include or exclude via **`-i`** and **`-e`** options on the command line. More typically, though, administrators create an ASCII text include/exclude "graph" file. Each line in the file contains a directory path that is to be included or excluded. Lines that begin with an "**`i`**" indicate a directory path that should be included. Lines that begin with an "**`e`**" indicate files that will be excluded. The graph file may be stored anywhere on the system, using any file name. Here's a sample graph file that backs up all the files and directories under **`/`** except **`/cdrom`** and **`/tmp`**.

```
i /
e /cdrom
e /tmp
```

The combination of backup levels and graph files provides significant flexibility. Recall that **`fbackup`** uses backup levels as a mechanism to identify files that have changed since the most recent backup that was made at a lower backup level. When invoked with a backup level and a graph file, **`fbackup`** makes a backup of files that changed since the most recent backup of that graph at a lower level.

`fbackup` Backup Levels and Graph Files

In order to do incremental backups, **`fbackup`** must be able to determine which graph files have been backed up at which backup levels when. **`fbackup`** logs this information in the **`/var/adm/fbackupfiles/dates`** file. The **`dates`** file contains a log entry for the most recent backup of each graph file at each backup level. Each **`dates`** file entry includes the following fields:

- 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

The **`dates`** file is automatically updated anytime **`fbackup`** is executed with the –u option.

Creating an `fbackup` Index File

The **`[ -I`** *path* **`]`** option creates an index of files included in the backup. The index is written to the file specified in the path argument. Beware that a full system backup may generate a very large index file.

© 2004 Hewlett-Packard Development Company, L.P.

Explanation of the Slide Examples

The following example writes a backup of all files and directories under **/home** to the tape in **/dev/rmt/0m**. **fbackup** also writes an index of the files included in the backup to **index.home**.

```
# fbackup -f /dev/rmt/0m -i /home -I index.home
```

The next example creates a backup of all files and subdirectories under the present working directory.

```
# fbackup -f /dev/rmt/0m -i . -I index.pwd
```

The next couple of commands lay the groundwork for running full and incremental backups by creating a graph file in **/var/adm/fbackupfiles**. The graph file includes the **/** directory, but excludes **/cdrom**. and **/tmp**.

```
# cd /var/adm/fbackupfiles
# vi graph
   i /
   e /tmp
   e /cdrom
```

After creating a graph file, you can do a full, level 0 backup of the files specified in the graph using the next command on the slide. **-u** ensures that the dates log file is updated with the results of the backup, and **-I** creates an index of the files included in the backup.

```
# fbackup -f /dev/rmt/0m -u0g graph -I index.full
```

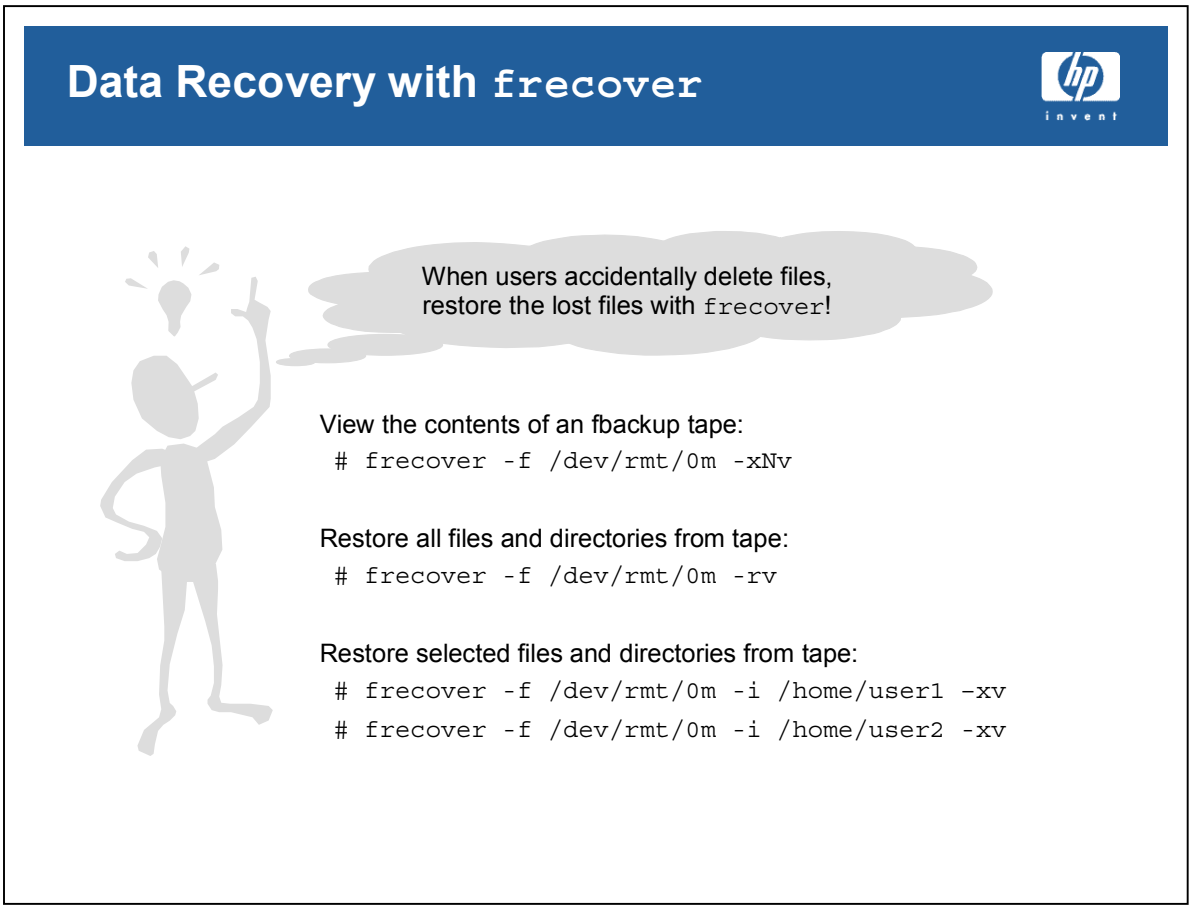
The last example does an incremental backup of the graph file, updates the "dates" log file, and creates an index.

```
# fbackup -f /dev/rmt/0m -u1g graph -I index.incremental
```

Many other examples and options are described in the **fbackup(1m)** man page.

© 2004 Hewlett-Packard Development Company, L.P.

11–6. SLIDE: Data Recovery with `frecover`



Student Notes

The **`frecover`** command is the partner to the **`fbackup`** command. It is designed to retrieve files from backups that were created with **`fbackup`**. Like **`fbackup`**, **`frecover`** is flexible and has many options that modify its default mode of operation. Only a few key options are explained here. The **`frecover(1M)`** manual page contains a full description of all the options.

There are three basic modes of operation for **`frecover`**:

`frecover -r`	Recover everything on a backup volume.
`frecover -x`	Extract selected files from a backup volume. Files to be extracted must be specified with the **`-i`** option.

© 2004 Hewlett-Packard Development Company, L.P.

frecover -r and **frecover -x** share several common options. Some of the most frequently used options are described below:

-f Identify the source to restore from. The restore source may be a tape device file, a pathname for a file on disk, or "-" if you wish to read from stdin. If not specified, frecover restores from **/dev/rmt/0m**.

-v Verbose. List all files and directories as they are restored.

-h Recovers only directories, but not the files in those directories.

-F Recovers files, but not directories. If **/usr/bin/vi** and **/bin/sh** were recovered from tape using the **-F** option while in **/home/root**, the resulting files would be **/home/root/vi** and **/home/root/sh**.

-o Used to force **frecover** to overwrite a newer file with an older one. Normally, frecover will not overwrite an existing disk file with an older version of the file.

-X Restores files and directories relative to the current working directory. Suppose the current working directory **/home/root** and the file **/usr/bin/vi** were being recovered. With the **-X** option, the file would be deposited in **/home/root/usr/bin/vi**. This option can be very useful when you are unsure about the directory and files that might result from a **frecover** session.

-N Prevent **frecover** from actually recovering any files onto disk, but read the backup as if it was, in fact, recovering the data from the backup, producing the same output that it would on a normal recovery. This option is useful for verifying backup media contents.

-g *graph* Extract selected files and directories, as defined in the specified graph file. The format of the **frecover** graph file is identical to the format of the **fbackup** graph file. Lines that begin with an **i** identify pathnames of files to include in the recovery. Lines that begin with an **e** indicate a path that is to be excluded from the recovery.

© 2004 Hewlett-Packard Development Company, L.P.

Slide Examples

View the contents of an **fbackup** tape. The **-N** option stands for no extract. The **–v** verbose option displays file permissions and attributes in addition to the file names. This combination of options is sometimes used to verify the contents of an **fbackup** tape.

```
# frecover -f /dev/rmt/0m -xNv
```

Restore all files and directories from tape.

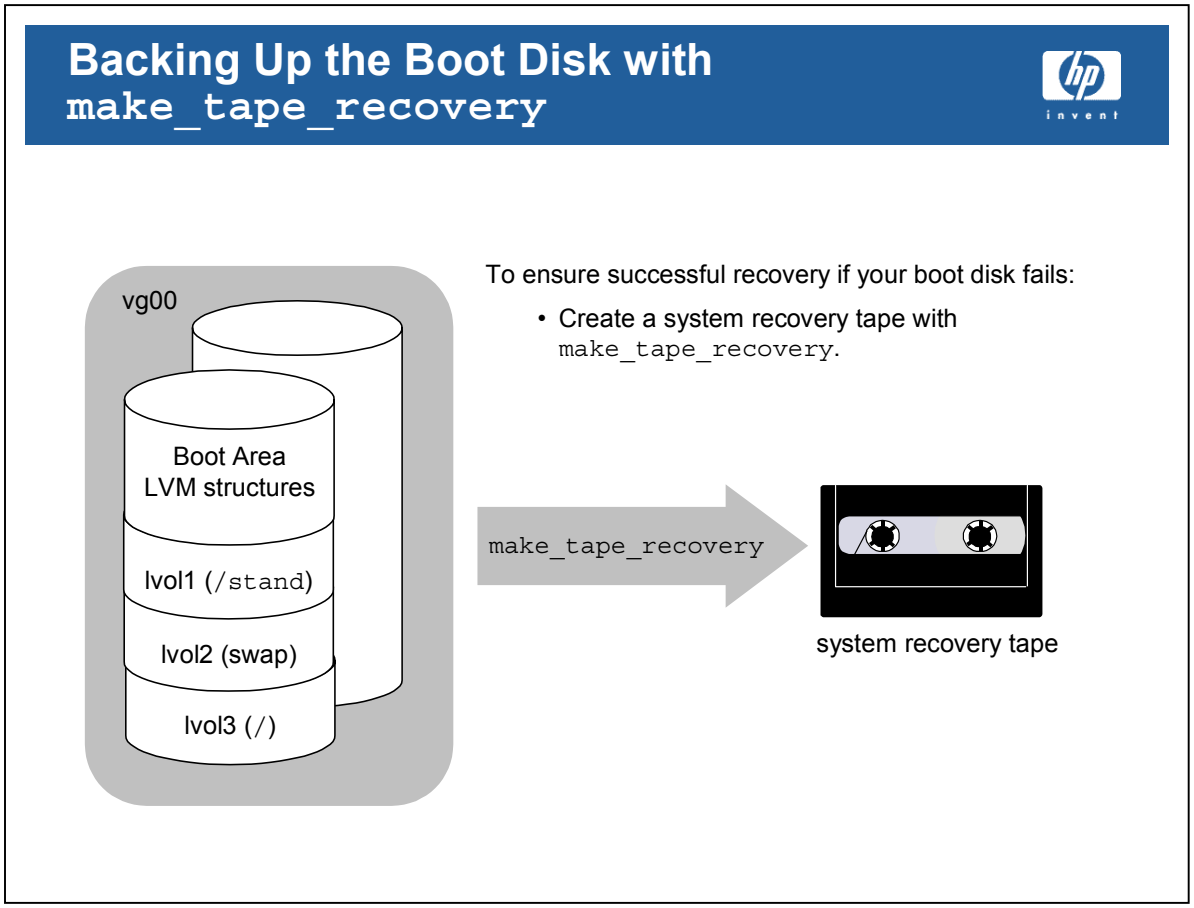
```
# frecover -f /dev/rmt/0m -rv
```

The next examples extract **/home/user1** and **/home/user2** respectively, from the **/dev/rmt/0m** archive:

```
# frecover -f /dev/rmt/0m -i /home/user1 -xv
# frecover -f /dev/rmt/0m -i /home/user2 –xv
```


© 2004 Hewlett-Packard Development Company, L.P.

11–7. SLIDE: Backing up the Boot Disk with `make_tape_recovery`



Student Notes

`frecover` is adequate for recovering some files and directories. However, if the root disk becomes corrupted, it will no longer be possible to boot the kernel, let alone run **`frecover`** to restore files and directories.

The **`make_tape_recovery`** command is designed precisely for this situation. **`make_tape_recovery`** creates a bootable tape image of the system boot disk. If the boot disk becomes corrupt, the administrator can bring the system to a bootable state by booting from the recovery tape.

Booting from the recovery tape will:

- Rebuild the boot area on the boot disk.
- Recreate the logical volumes and file systems in **`vg00`**.
- Restore the critical files and directories in those file systems.

© 2004 Hewlett-Packard Development Company, L.P.

After restoring the system to a minimal state using the recovery tape, the administrator can use **frecover** or another backup/recovery utility to restore the remaining files and directories from the most recent tape backup.

What's Included on the make_tape_recovery Tape?

The contents of the resulting system recovery tape include:

- A boot image
- LVM configuration information for the root volume group.
- An archive of the critical files and directories contained in the file systems in **vg00**.

By default, the archive portion of **make_tape_recovery** will only include the "Minimum Core-OS" files and directories from **vg00** that are required to boot the system:

```
/.profile
/.rhosts
/dev
/etc
/opt/dce
/opt/ignite/bin/print_manifest,
/opt/ignite/share/man/man1m.Z/print_manifest.1m,
/opt/upgrade
/sbin
/stand
/usr            (selected subdirectories)
/var            (selected subdirectories)
```

You may wish to include additional files in the recovery tape archive. If, for instance, you plan to use HP Omniback II for your data backup and recovery, you will probably want to include the Omniback II executables, libraries, and other critical files in the recovery tape. This guarantees that after restoring from the recovery tape, you will be able to successfully run Omniback to restore the remainder of your system.

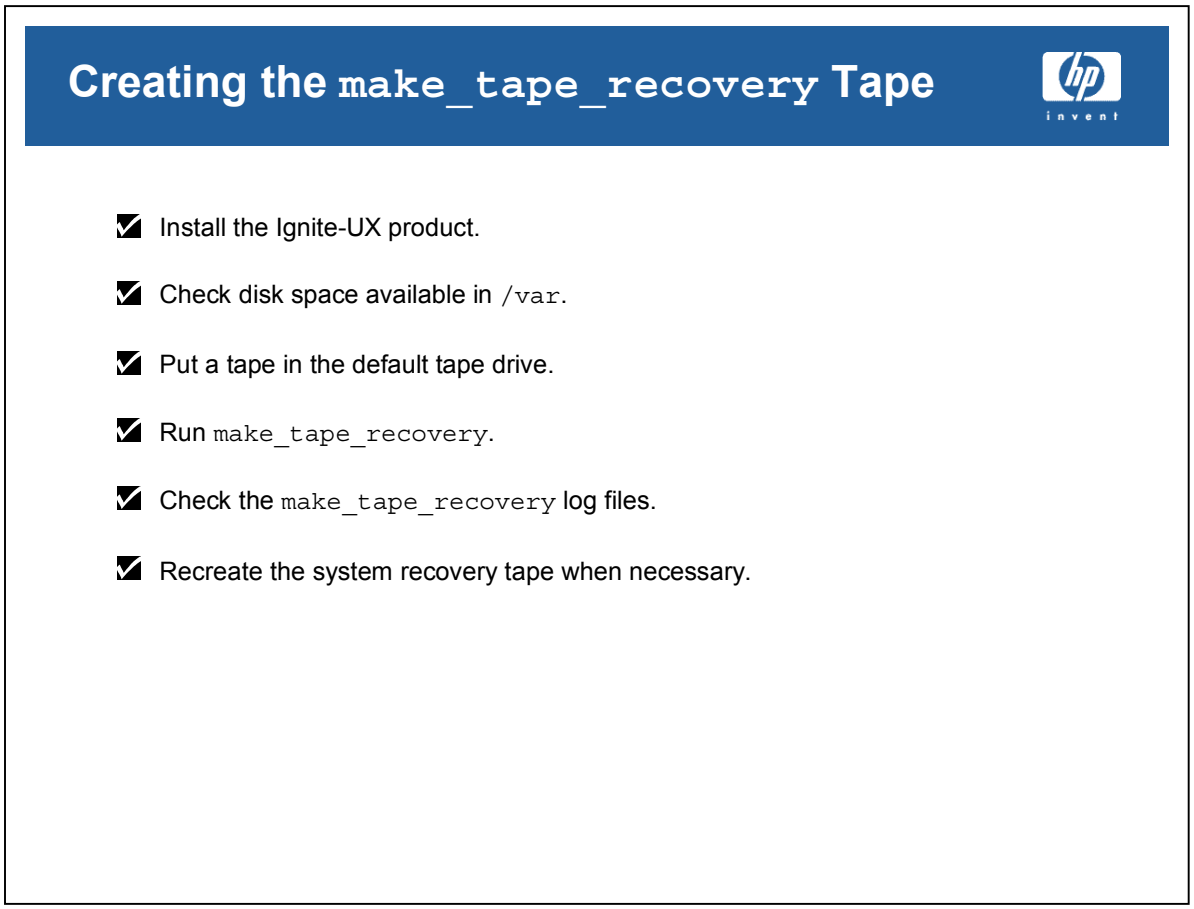
If you wish to include *all* of the vg00 files and directories in your archive, then execute **make_tape_recovery** with the **-A** option.

make_tape_recovery provides a great deal of flexibility. These notes just cover the most common options. See the man pages for more options and examples.

NOTE:	A recovery tape created with **make_tape_recovery** supplements, but does not replace, a regular system backup scheme! You should continue to do regular full and incremental backups of your system using **fbackup**, HP Omniback, or some other backup utility.

© 2004 Hewlett-Packard Development Company, L.P.

11–8. SLIDE: Creating the `make_tape_recovery` Tape



Student Notes

Step 1 — Install the Ignite-UX Product

`make_tape_recovery` is a part of the Ignite-UX product, which is not included in a standard load of HP-UX. Ignite is included on the Applications CD for HP-UX 11.x. The latest version may also be downloaded from HP software web site (**`http://www.software.hp.com`**). You can verify that Ignite is installed using **`swlist`**.

```
# swlist -l product Ignite-UX
```

Step 2 — Check Disk Space Available in `/var`

`make_tape_recovery` requires a minimum of 32 MB of disk space in the **`/var`** file system. Use **`bdf`** to ensure that you have enough free disk space.

```
# bdf /var
```


© 2004 Hewlett-Packard Development Company, L.P.

Step 3 — Put a Tape in the Default Tape Drive

The default tape drive in HP-UX is **`/dev/rmt/0m`**. Insert a non-write protected tape in the drive before proceeding any further. You may choose to use a different tape drive by using the **`-d`** ***`tape-device-file`*** option on **`make_tape_recovery`**.

Step 4 — Run `make_tape_recovery`

Run **`make_tape_recovery`** to create the recovery tape. Be patient: creating the recovery tape may take thirty minutes or more.

Step 5—Check the `make_tape_recovery` Log File

`make_tape_recovery` creates a log file that you should check once the program terminates:

```
/var/opt/ignite/recovery/<datetime>/recovery.log
```

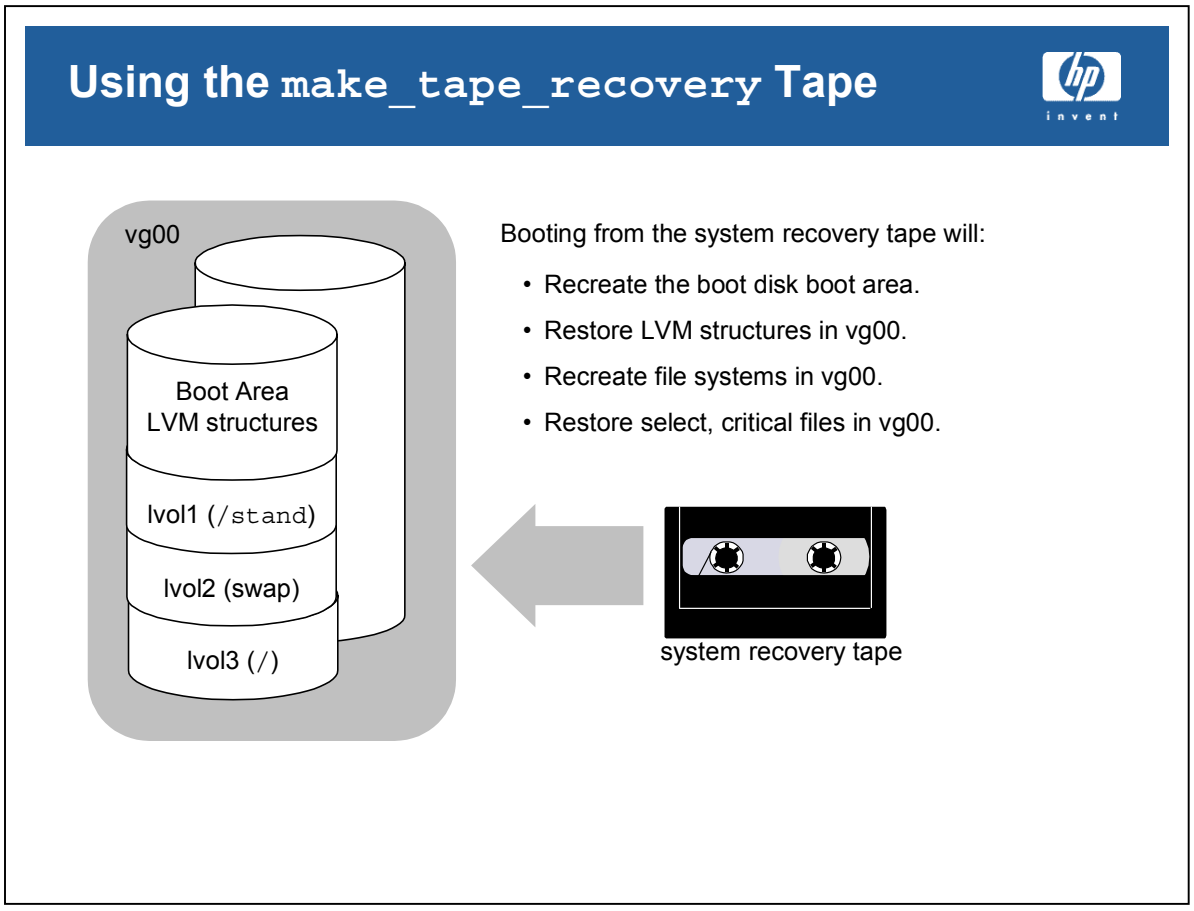
Step 6 -— Update the System Recovery Tape when Necessary

The system recovery tape needs to be recreated periodically to reflect new hardware, software, and patches. Some administrators create a new recovery tape on a daily or weekly basis as part of their normal system backup procedures. Other administrators only choose to recreate the recovery tape on an as-needed basis. To create a new recovery tape, simply repeat the steps shown on the slide.

NOTE:	Since different models require different drivers and different configurations, a **`make_tape_recovery`** tape should generally only be restored on the system on which it was created

© 2004 Hewlett-Packard Development Company, L.P.

11–9. SLIDE: Using the `make_tape_recovery` Tape



Student Notes

The tape created by **`make_tape_recovery`** is called the “system recovery boot tape”. This tape can be used to boot your system and reinstall the created image automatically if your boot disk becomes corrupted.

To recover a non-bootable system:

- Power up the system.
- Interrupt the boot.
- Load the recovery tape into the tape drive.
- Search for bootable devices.
- Boot from the recovery tape.
- Let the system restore vg00 automatically.
- Verify the recovery.

© 2004 Hewlett-Packard Development Company, L.P.

- Use your preferred backup/recovery software to restore your most recent backup.

Non-Interactive vs. Interactive Recoveries

A **make_tape_recovery** tape may be used to restore a system in either of two modes. Booting from the recovery tape "Non-Interactively" restores the configuration and files found on the recovery tape verbatim. This is the normal mode that should be used when recovering a corrupted boot disk.

Alternatively, you may choose to interrupt the recovery procedure to tweak the system configuration. This "Interactive" mode is sometimes used to:

- Convert file systems from HFS to VxFS.
- Change the size of the root file system.
- Change the size of the primary swap logical volume.
- Change the system's network configuration.

If you wish to do an interactive recovery, watch for the message that says "**Press <Return> within 10 seconds to cancel batch-mode installation**"

Sample Console Output from a D220 Booting from a System Recovery Tape

The next few pages show sample output from a D220 booting from a system recovery tape created via **make_tape_recovery -AC**. Note that the console messages and menus on other models may be slightly different.

power

```
Processor is booting from first available device.
        To discontinue, press any key within 10 seconds.
```

esc

```
 Boot terminated.

------ Main Menu ---------------------------------------------------

  Command                          Description
  -------                          -----------
  Boot [PRI|ALT|>]                 Boot from specified path
  Path [PRI|ALT|CON|KEY] [<path>]  Display or modify a path
  SEArch [Display|IPL] [<path>]    Search for boot devices
  Configuration [<command>]        Access Configuration menu/commands
  Information [<command>]          Access Information menu/commands
  SERvice [<command>]              Access Service menu/commands
  Display                          Redisplay the current menu
  Help [<menu>|<command>]          Display help for menu or command
  RESET                            Restart the system
-----------------------------------------------------------------------

Main Menu: Enter command > search ipl

Searching for device(s) with bootable media ...
This may take several minutes.
```


© 2004 Hewlett-Packard Development Company, L.P.

```
To discontinue search, press any key (termination may not be immediate).

  Path Number    Device Path (dec)    Device Type and Utilities
  -----------    -----------------    -------------------------
  P0              8/16/5.6            Random access media
                                      IPL
  P1              8/16/5.5            Random access media
                                      IPL
  P2              8/16/5.0            Sequential access media
                                      IPL
  P3              8/16/6.0            LAN Module

Main Menu: Enter command > boot p2
Interact with IPL (Y, N, or Cancel)?> n

Booting ...
Boot IO Dependent Code (IODC) revision 144

SOFT Booted.
ISL Revision A.00.38 OCT 26, 1994
ISL booting hpux (;0):INSTALL

Boot
: tape(8/16/5.0.0.0.0.0.0;0):INSTALL

...(all of the rest of the boot messages)...

======= 06/22/98 15:31:20 EDT HP-UX Installation Initialization.
(Mon Jun 22 15:31:20 EDT 1998)

@(#) Ignite-UX Revision 1.45
@(#) install/init (opt) $Revision: 10.133 $

* EISA configuration has completed. Following the completion of a
       successful HP-UX installation, please check the
       "/etc/eisa/config.err" file for any EISA configuration messages.
* Scanning system for IO devices...

WARNING: The configuration information calls for a non-interactive
installation.

Press <Return> within 10 seconds to cancel batch-mode installation:

* Using client directory: /var/opt/ignite/clients/0x0060B0A37808
* Checking configuration for consistency...
WARNING: The disk at: 8/16/5.6.0 (SEAGATE_ST34572N) appears to contain a
file system and boot area. Continuing the installation will destroy any
existing data on this disk.

Press > within 10 seconds to cancel batch-mode installation:

* Continuing despite above warnings.
* Attempting a non-interactive installation.
```


© 2004 Hewlett-Packard Development Company, L.P.

```
======= 06/22/98 15:32:11 EDT Starting system configuration...

* Configure_Disks: Begin
* Will install B.11.00 onto this system.
* Creating LVM physical volume "/dev/rdsk/c1t6d0" (8/16/5.6.0).
* Creating volume group "vg00".
* Creating logical volume "vg00/lvol1" (/stand).
* Creating logical volume "vg00/lvol2" (swap_dump).
* Creating logical volume "vg00/lvol3" (/).
* Creating logical volume "vg00/lvol4" (/home).
* Creating logical volume "vg00/lvol5" (/opt).
* Creating logical volume "vg00/lvol6" (/tmp).
* Creating logical volume "vg00/lvol7" (/usr).
* Creating logical volume "vg00/lvol8" (/var).
* Extending logical volume "vg00/lvol1" (/stand).
* Extending logical volume "vg00/lvol2" (swap_dump).
* Extending logical volume "vg00/lvol3" (/).
* Extending logical volume "vg00/lvol4" (/home).
* Extending logical volume "vg00/lvol5" (/opt).
* Extending logical volume "vg00/lvol6" (/tmp).
* Extending logical volume "vg00/lvol7" (/usr).
* Extending logical volume "vg00/lvol8" (/var).
* Making HFS filesystem for "/stand", (/dev/vg00/rlvol1).
* Making VxFS filesystem for "/", (/dev/vg00/rlvol3).
* Making VxFS filesystem for "/home", (/dev/vg00/rlvol4).
* Making VxFS filesystem for "/opt", (/dev/vg00/rlvol5).
* Making VxFS filesystem for "/tmp", (/dev/vg00/rlvol6).
* Making VxFS filesystem for "/usr", (/dev/vg00/rlvol7).
* Making VxFS filesystem for "/var", (/dev/vg00/rlvol8).
* Setting rotational delay to 0 for "/stand".

* Configure_Disks: Complete
* Download_mini-system: Begin
x ./sbin/fs/hfs/mkfs, 237568 bytes, 464 tape blocks
x ./sbin/fs/hfs/newfs, 114688 bytes, 224 tape blocks

(extraction of the rest of the essential rebuild tools...)

* Download_mini-system: Complete
* Loading_software: Begin
* Installing boot area on disk.
* Enabling swap areas.
* Backing up LVM configuration for "vg00".
* Processing the archive source (recovery).
* Mon Jun 22 15:36:03 EDT 1998: Starting archive load of the source
 (Recovery Archive).
* Positioning the tape (/dev/rmt/0mn).
* Archive extraction from tape is beginning. Please wait.
* Mon Jun 22 16:00:56 EDT 1998:Completed archive load of the source
 (Recovery Archive).
```


© 2004 Hewlett-Packard Development Company, L.P.

```
* Executing user specified script:
"/opt/ignite/data/scripts/os_arch_post_1".
Running in recovery mode.
NOTE: Could not save /etc/resolv.conf from archive: file not found
NOTE: Could not save /etc/eisa/system.sci from archive: file not found

* Running the ioinit command ("/sbin/ioinit -c")
NOTE: tlinstall is searching filesystem - please be patient
NOTE: Successfully completed
* Setting primary boot path to "8/16/5.6.0".
* Executing user specified commands.
* Loading_software: Complete
* Build_Kernel: Begin

NOTE: Since the /stand/vmunix kernel is already in place, the kernel
will not be re-built. Note that no mod_kernel directives will be
processed.

* Build_Kernel: Complete
* Boot_From_Client_Disk: Begin
* Rebooting machine as expected.
NOTE: Rebooting system.
* Running the ioinit command ("/sbin/ioinit -c")
* Boot_From_Client_Disk: Complete
* Run_SD_Configure_Scripts: Begin
* Run_SD_Configure_Scripts: Complete
* Run_Postconfigure_Scripts: Begin
* Applying the networking information.
* Executing user specified script:
"/opt/ignite/data/scripts/os_arch_post_c".
* Running in recovery mode.
* Run_Postconfigure_Scripts: Complete

======= 06/22/98 16:03:36 EDT Installation complete: Successful
```

After Booting from the Recovery Tape

Remember that the recovery tape is only intended to bring your system to a bootable state. You will still need to restore your most recent full and incremental backups using **`frecover`**, HP Omniback, or another restore utility.

© 2004 Hewlett-Packard Development Company, L.P.

11–10. TEXT PAGE: System Recovery Checklist

To ensure that you are able to recover from a crashed disk, we recommend that you keep each item on this checklist at hand. With help from your HP Support Representative and this list, you stand a good chance of timely recovery.

- Recent full and incremental system backup tapes, containing backups of your data.
- Recent **`make_tape_recovery`** tape, which could be used to rebuild vg00.
- Tape backup of **`/etc/lvmconf`**, which contains a backups of your LVM configuration.
- Printed **`print_manifest`** output, summarizing your basic system configuration.
- Printed **`vgdisplay -v`** output for each Volume Group.
- Printed **`lvdisplay -v`** output for each Logical Volume.
- Printed **`pvdisplay -v`** for each physical volume.
- Printed output of **`lvlnboot -v`**, which describes your LVM boot disk configuration.
- Printed **`/etc/fstab`**, which lists your file system
- Printed copy of output from a recent **`bdf(1m)`**.
- Printed output of **`swapinfo(1m)`**.
- Printed **`ioscan -fun`**.
- Printed **`ioscan -kf`**.
- Printed information from any software or database packages that use logical volumes for raw data storage. The backup and recovery utilities discussed on this chapter only do not backup the contents of raw logical volumes! See your application documentation to learn how to obtain this information.

© 2004 Hewlett-Packard Development Company, L.P.

11–11. LAB: Backup and Recovery

Directions

Perform the tasks suggested in the questions that follow. Record the commands you use, and answer all the questions.

Since your class machine may not have a tape drive, perform the backups to a series of fake tape files called tape1, tape2, etc. Note that on production machines you would more commonly use a true tape device file (e.g. **`/dev/rmt/0m`**).

Part 1: Preliminary Steps

1. If you don't already have a **`/var/adm/fbackupfiles`** directory on your machine, create it. By default, this is the directory where **`fbackup`** maintains the "dates" log file of backups completed on your system. Make this directory.

2. Run the **`/labs/corp1.sh`** script, which will create a few directories that you can practice backing up. Note the directories that are created by the script.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Full and Incremental Backups

1. Create a graph file that includes everything in the **/corp** directory.

2. Perform a full backup of **/corp** using the graph file you just created. Write the backup to a file called **tape1**, write an index of the files included in the backup to **index1**, and use the **-u** option to ensure that the **dates** file records the time stamp of your backup.

3. What was recorded in the **dates** file as a result of your **fbackup**?
   Were all of the files under **/corp** backed up?
   Look at the **index1** file that was created, and explain what you see.

4. Run **/labs/corp2.sh**. This script should create a few new files under the **/corp** directory. Note the changes.

5. Do a level 1 backup of **/corp** using your graph file. Write the backup to **tape2**, create an index of your backup in **index2**, and use the **-u** option to ensure that the **dates** file is updated again.

© 2004 Hewlett-Packard Development Company, L.P.

6. What was recorded in your **dates** file as a result of your backup?
   Were all of the files under **/corp** backed up?
   Look at the **index2** file and explain what you see.

7. Run the **/labs/corp3.sh** script. Note the newly created files.

8. Run another level 1 backup. Write the backup to **tape3**. Ensure that the **dates** file is updated and create an index file called **index3**.

9. What changed in your **dates** file as a result of your backup? Explain.
   Which files and directories under **/corp** backed up? Explain.

10. Run the **/labs/corp4.sh** script. Note that this script doesn't touch any new or existing files.

11. Now try running a level 2 backup of **/corp** using your **graph** file. Ensure that **fbackup** updates the **dates** file and creates an index file called **index4**. What files and directories are included in this backup? Explain.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Restoring Files with `frecover`

1. Run the **`/labs/corp5.sh`** script, and note the effect this script has on your **`/corp`** directory.

2. You did a total of four backups in the previous part of this lab. Is it necessary to restore all four tapes to fully recover your system? Explain.

3. Do whatever is necessary to fully restore **`/corp`**.

4. Occasionally, users accidentally remove a file and expect the administrator to be able to restore the missing file from tape. Remove **`/corp/reporta`**. How can you restore this single file? Do it.

5. Will **`frecover`** overwrite newer data in a file with older data from a tape archive? Try it and find out.

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 12 — Scheduling `cron` Jobs

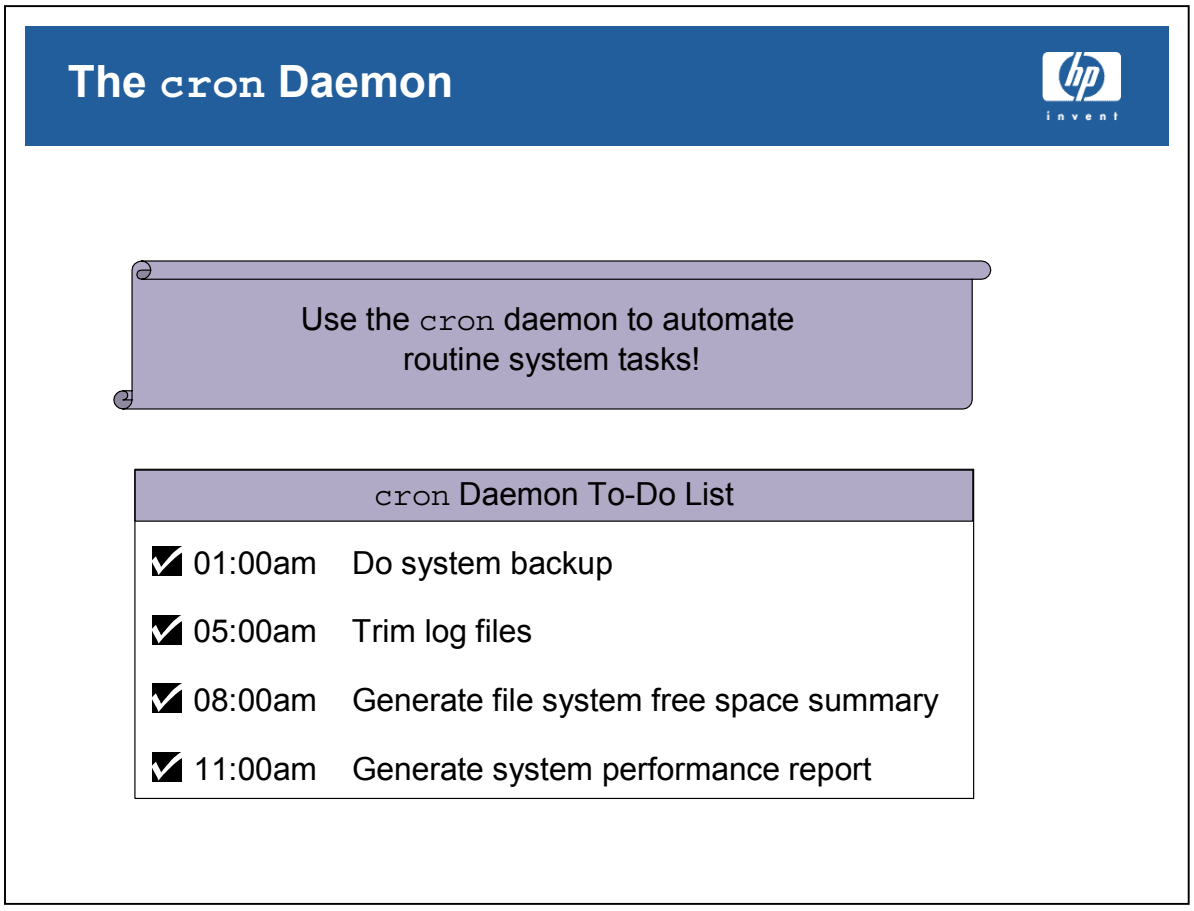
Objectives

Upon completion of this module, you will be able to do the following:

- Submit, list, and remove time-scheduled jobs with **`cron`**.
- Schedule full and incremental backups to run automatically.
- Grant non-root users access to **`cron`**.

© 2004 Hewlett-Packard Development Company, L.P.

12–1. SLIDE: The `cron` Daemon



Student Notes

System administrators often need to run backups and other processes on a daily, weekly, or monthly basis. Although such processes can be started manually, it is often desirable to schedule them to run automatically. HP-UX provides the **`cron`** daemon for just this purpose.

The **`cron`** daemon is started during the system boot process and executes time scheduled jobs submitted by the system administrator and other users.

You can check to see if the **`cron`** daemon is running with the **`ps`** command:

```
# ps -ef | grep cron      # is cron running?
# cron                    # start the cron daemon, if it isn't currently running
```

Regular users, as well as **`root`**, can utilize **`cron`** for repetitive execution of programs. Jobs are submitted to **`cron`** with the **`crontab`** command. **`Root`** controls who can use **`crontab`** through the **`/var/adm/cron/cron.allow`** file. Users are permitted to use the **`crontab`** command if their names appear in the **`cron.allow`** file. If **`cron.allow`** does not exist, then **`/var/adm/cron/cron.deny`** is checked to determine if the user should be denied

© 2004 Hewlett-Packard Development Company, L.P.

access. If both exist, **`cron.allow`** takes precedence. If neither file exists, only root is allowed to submit a job. An empty **`cron.deny`** file allows all to use **`crontab.`**

Table 12-1: Who Can Use `cron`?

***.allow**	***.deny**	**Who Can Use?**
—	—	Superuser
Exists	Ignored	Everybody in *.allow
—	Exists	Everybody who is not in *.deny
Empty	Ignored	Nobody
—	Empty	Everybody

© 2004 Hewlett-Packard Development Company, L.P.

12–2. SLIDE: Understanding the `cron` File

Understanding the `cron` File

- ☑ The `cron` daemon is configured via one or more `cron` files.
- ☑ Each `cron` file contains one line for each scheduled job.

```
# min    hour   date   month   day     command
# 0-59   0-23   1-31   1-12    0-6     be sure to redirect output!
  0      *      *      *       *       /usr/bin/date >/dev/console
  0      6      1,15   *       *       >/var/adm/btmp
  0      5      *      *       1-5     /usr/bin/who | /usr/bin/lp
```

Student Notes

Each user that is using the **`cron`** daemon should have a **`cron`** configuration file in the **`/var/spool/cron/crontabs`** directory. The **`cron`** daemon consults the files in this directory to determine which jobs should be executed when. The **`cron`** file contains a one-line entry for each time-scheduled job.

Each entry in the **`cron`** file has six fields, separated by white space or tabs. The first five fields contain integers that represent the date and time that a command is to be executed. Each of these fields may contain a single integer, a comma-separated list of integers, a range of integers defined by a dash, or an asterisk, which represents all legal values for a field.

The last field specifies the command to be executed at the specified time. A percent character in this field (unless escaped by \) is translated as a new-line character.

© 2004 Hewlett-Packard Development Company, L.P.

NOTE: You must redirect the standard output and standard error of your commands. If you do not, any output generated will be mailed to you.

Always use full path names for your commands and file names, since **`cron`** uses only the standard environment of **`/usr/bin/sh`** and does not know about your environment, for example, your current directory, variables and PATH.

Explanation of the Examples

The first example on the slide executes the date command on the system console at the top of every hour.

The second example trims the **`/var/adm/btmp`** file to zero bytes at 6am, on the 1st and 15th of every month.

The third example prints a log of users logged onto the system every weekday at 5am.

Questions

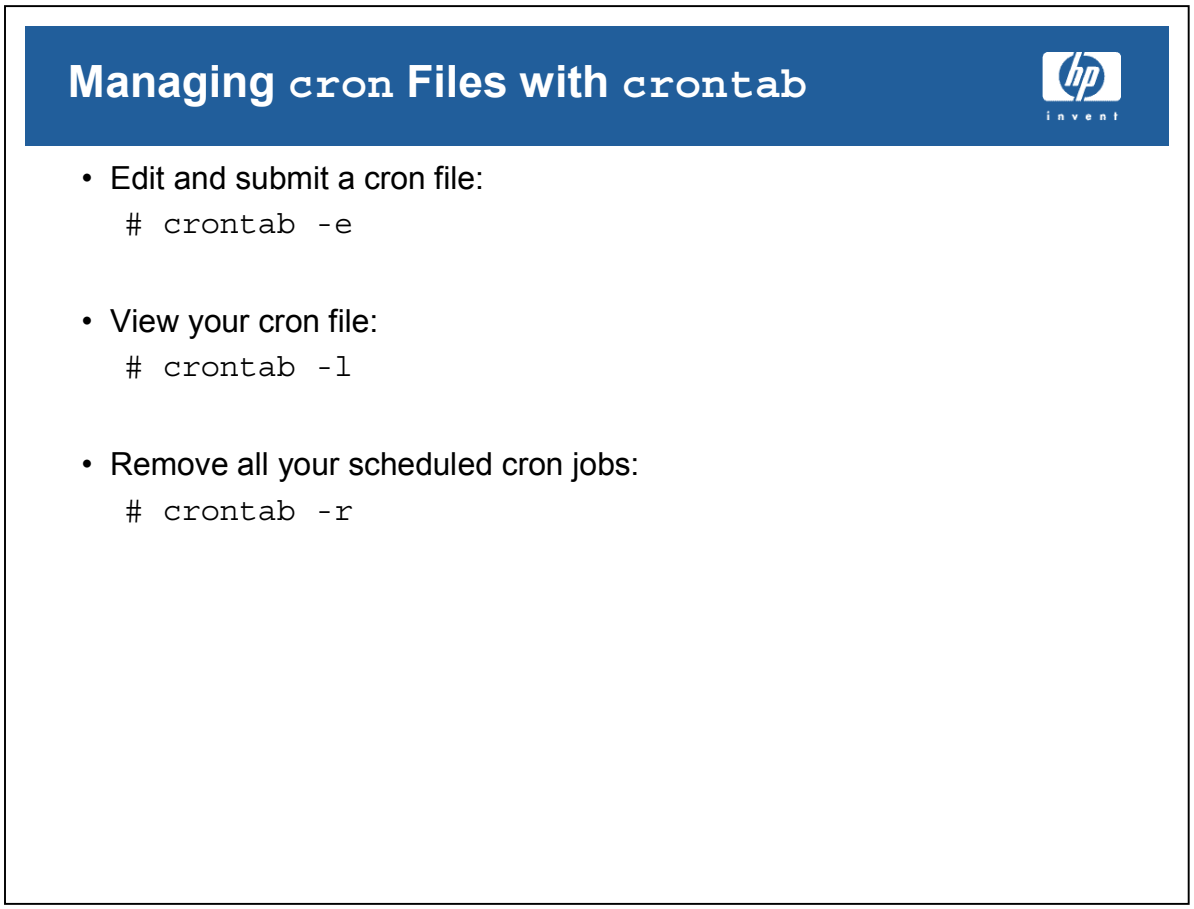
What values would you include in the **`cron`** file time fields if you wanted to execute a command:

1. Every day at 6:30 pm?
2. Every weekday at 6:30 pm?
3. Every Monday, Wednesday, and Friday at 6:30 pm?
4. Every 10 minutes, every hour, every day?

Answers are provided in the solutions appendix.

© 2004 Hewlett-Packard Development Company, L.P.

12–3. SLIDE: Managing `cron` Files with `crontab`



Student Notes

Users may create, submit, and modify their **`cron`** files via the **`crontab`** command:

`# crontab -e` Automatically copies your currently scheduled **`cron`** jobs to a temporary **`cron`** file, then launches **`vi`** so that you can add, remove, and change your **`cron`** entries. When you save your changes and quit out of **`vi`**, the **`cron`** daemon recognizes your changes immediately.

`# crontab -r` Removes all of your currently scheduled **`cron`** jobs.

`# crontab -l` Lists all of your currently scheduled **`cron`** jobs.

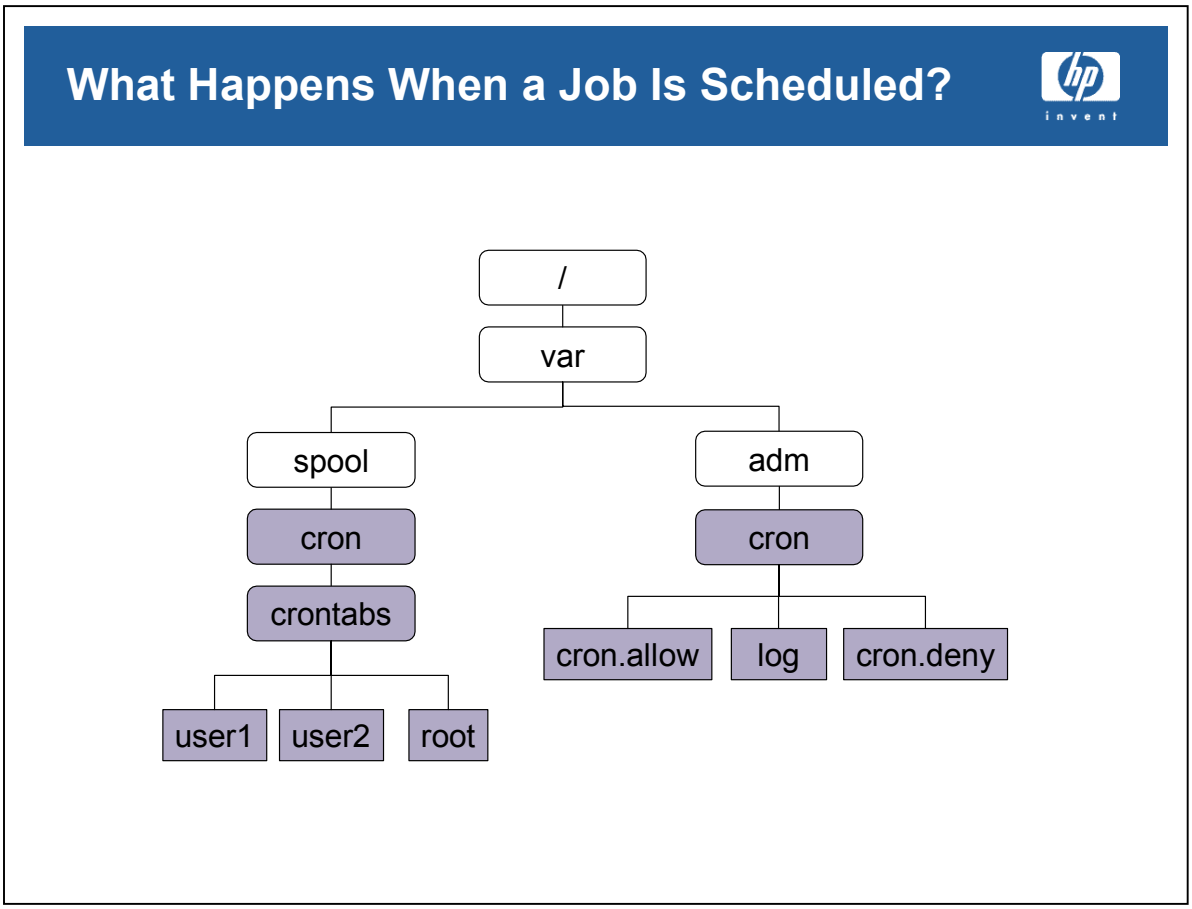
By default, **`crontab`** opens the user's own **`cron`** file. However, in HP-UX 11.x, the system administrator can modify any user's **`cron`** file by specifying the user's username as an argument to the **`crontab`** command:

```
# crontab -e user1
```

NOTE: Never modify a user's **`cron`** file in **`/var/spool/cron/crontabs`** directly. Always use the **`crontab`** command.

© 2004 Hewlett-Packard Development Company, L.P.

12–4. SLIDE: What Happens When a Job Is Scheduled?



Student Notes

By default, only the system administrator can submit jobs to the **`cron`** daemon. If you wish to allow other users access to the **`cron`** daemon, add their names to the **`/var/adm/cron/cron.allow`**. Alternatively, you can define which users *cannot* use the **`cron`** daemon in **`/var/adm/cron/cron.deny`**.

Each user authorized to schedule **`cron`** jobs has a **`cron`** file in the **`/var/spool/cron/crontabs`** directory. The **`cron`** daemon consults these files to determine which jobs should be executed at which time.

When you use the **`crontab`** command to create and submit a **`cron`** file to the **`cron`** daemon, **`crontab`** automatically copies your **`cron`** file to the **`/var/spool/cron/crontabs`** directory. The copy of your cron file is named with your username.

When the scheduled time arrives, **`cron`** starts your job and logs a record to the **`/var/adm/cron/log`** file. The **`cron`** log grows without bound, and should be checked and emptied periodically.

© 2004 Hewlett-Packard Development Company, L.P.

NOTE:	Although the files in **`/var/spool/cron/crontabs`** can be viewed with **`ls`** and **`cat`**, they should never be directly edited or removed. Doing so can leave the **`cron`** daemon in an undefined state. Always use the **`crontab`** command when making changes to your scheduled **`cron`** jobs.

© 2004 Hewlett-Packard Development Company, L.P.

12–5. LAB: `cron`

Directions

In the exercises in Part 1, you will schedule a simple **`cron`** job that repeatedly displays the output from the date command in the system console window. To ensure that you see these messages, open a console window by typing:

```
# dtterm -C -name console &
```

Any output sent to **`/dev/console`** now appears in this window.

Part 1: Managing Scheduled cron Jobs

1. **`cron`** should start automatically during the boot process. Check to ensure that the **`cron`** daemon is running on your machine.

2. Create and submit a **`cron`** job to display the current time and date to the console every minute.

3. Add another **`cron`** job that sends the output from the **`who`** command to the console every 10 minutes.

4. List your scheduled **`cron`** jobs. Are they both there?

© 2004 Hewlett-Packard Development Company, L.P.

5. Edit your **`cron`** file again. See what happens if you precede the **`/usr/bin/date`** line with a **#** sign. Save your changes. What effect does the **#** sign have?

6. Why would commenting a line out of the **`cron`** file be preferable to simply removing the line?

7. How can you remove ALL of your scheduled **`cron`** jobs? Do it.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: (Optional) Scheduling Backups with `cron`

Your goal in this part of the exercise is to schedule full and incremental backups of your home file system to occur automatically via **`cron`**. If you have a tape drive on your student system, write your backups to the tape drive device file. If you don't have a tape drive, write your backups to **`/tmp/tape`**.

1. Create a **`graph`** file that includes **`/home`**, but nothing else. Name your graph file **`/home.graph`**.

2. Schedule a full backup of **`/home.graph`** to occur every Sunday night, and an incremental backup to occur every weekday night. Schedule both backups to run at 11 pm. Redirect the error messages from your backups to a file called **`/home.err`**.

3. List your **`cron`** jobs to ensure they are properly scheduled.

4. If you have a tape drive on your system, leave your scheduled **`cron`** jobs in place, and check your **`/home.err`** file tomorrow morning to see if your backup succeeded. If you don't have a tape drive on your system, remove all of your scheduled cron jobs to avoid filling the **`/tmp`** file system.

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 13 — Managing Swap Space

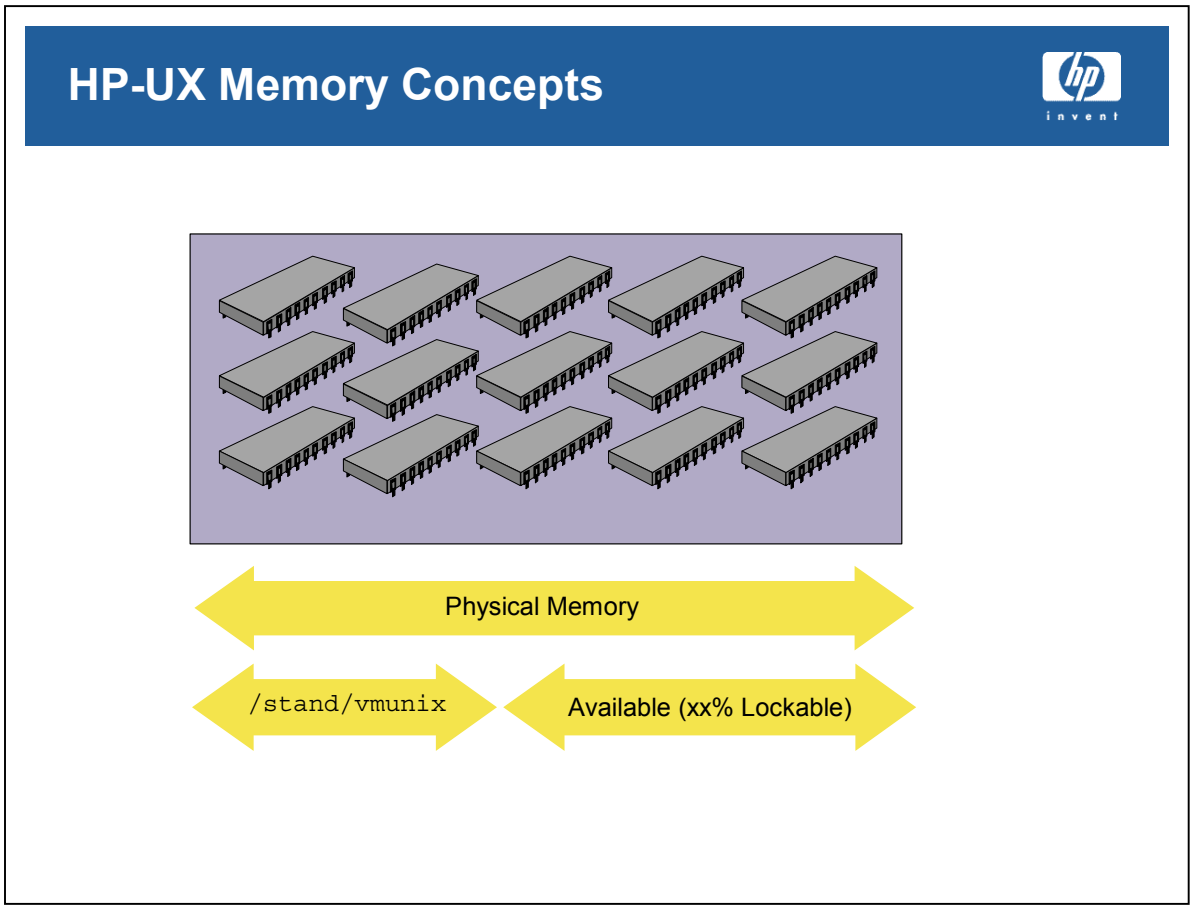
Objectives

Upon completion of this module, you will be able to do the following:

- Explain the concept of **demand paging**.
- Define **physical**, **available**, and **lockable** memory.
- Define **device swap**, **file system swap**, and **pseudoswap**.
- Configure device swap from the command line.
- Configure file system swap from the command line.
- Determine the amount of configured physical, available, and lockable memory.
- Determine the amount of swap space currently configured and in use.
- Disable swap space.
- List considerations for selecting appropriate file system and device swap areas.

© 2004 Hewlett-Packard Development Company, L.P.

13–1. SLIDE: HP-UX Memory Concepts



Student Notes

Physical memory is the random access memory (RAM) installed in your computer. At system startup, the system console displays the amount of physical memory installed:

```
Physical: xxxxxxx Kbytes
```

Not all physical memory is available to HP-UX processes. Some memory is reserved for kernel code and data structures. The remaining memory is referred to as *available* memory. Available memory may be used by processes and applications on your system. During system startup, the system console displays the amount of available memory:

```
Available: xxxxxxx Kbytes
```

Physical memory may be supplemented by specially configured space on disk known as *swap*. When a system is under a heavy load, the virtual memory subsystem may temporarily move data from physical memory out to the system's configured swap areas. When a process requests data that has been sent to swap, the virtual memory subsystem brings that data back into physical memory. This allows HP-UX to run more concurrent processes than would otherwise be possible.

© 2004 Hewlett-Packard Development Company, L.P.

All or part of available memory can be *locked* by a subsystem or by user processes. Locked memory cannot be swapped out to disk. Typically, locked memory holds frequently accessed programs or data structures. By keeping them memory-resident, process performance improves. If most of the available memory is locked, the system may deadlock. Some non-lockable memory must be available to prevent deadlock.

During system startup, the system console displays the amount of memory that can be locked:

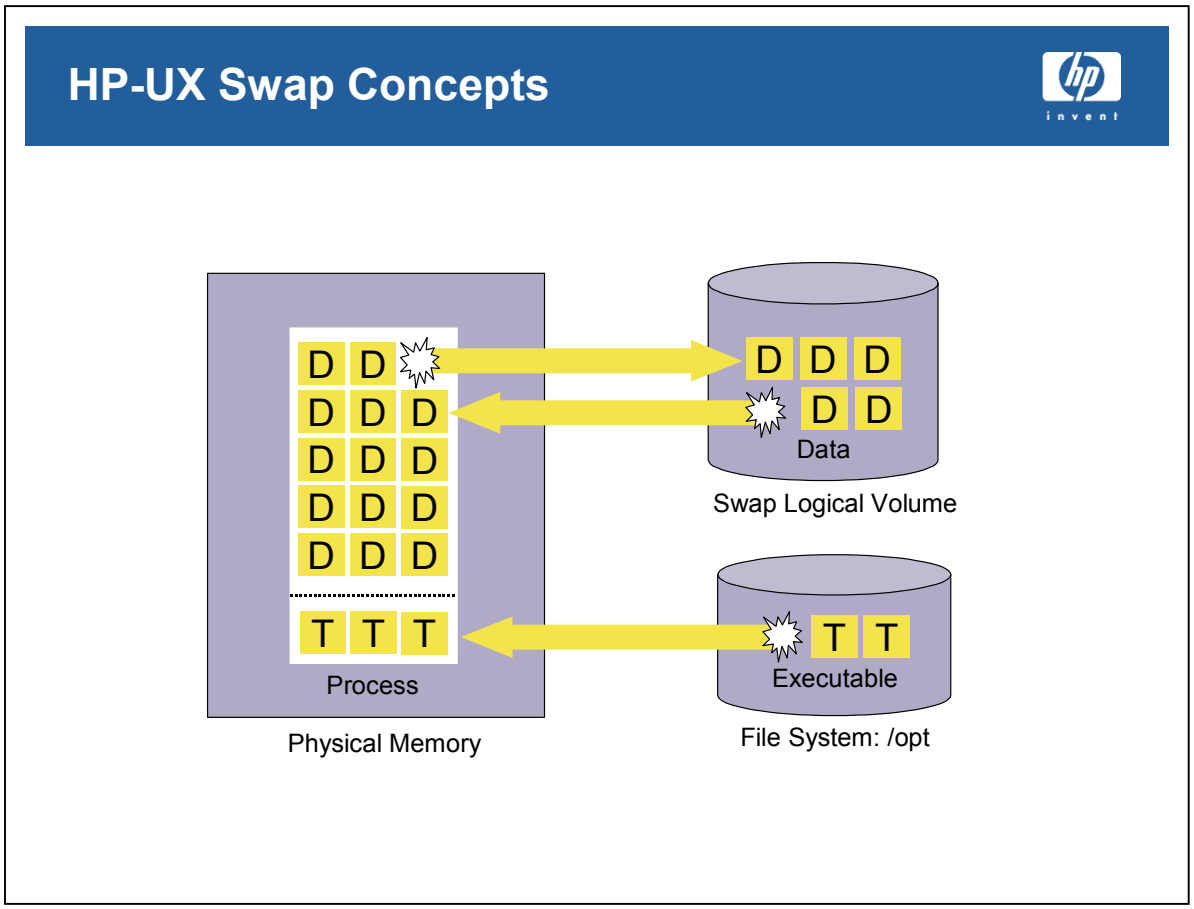
```
lockable: xxxxxxx Kbytes
```

Available memory minus the memory locked by subsystems or user processes is the memory that is actually usable for virtual memory demand paging. The **unlockable_mem** kernel parameter determines the amount of memory that cannot be locked.

The **dmesg** command will show selected messages that have appeared on the system console. These messages include the amounts for physical, available, and lockable memory in kilobytes (KB).

© 2004 Hewlett-Packard Development Company, L.P.

13–2. SLIDE: HP-UX Swap Concepts



Student Notes

Every process running on an HP-UX system requires space in memory. Space is allocated to processes in units known as “pages”. Some of a process’s pages are used to store the process executable (“text”), while other pages are used to store the process’s “data.”

Physical memory is a finite resource on any computer, and oftentimes there aren’t enough pages in physical memory for all the processes your users wish to run. The HP-UX virtual memory subsystem allows the total size of all user processes to exceed the size of physical memory by using an approach termed *demand paging*. Using demand paging, HP-UX brings portions of a process into memory as needed (on demand!), and pushes pages that haven’t been recently referenced out to disk.

Since executables are static, the text portion of a process may be loaded in physical memory on-demand from the on-disk copy of a program’s executable.

A process’s data, on the other hand, is dynamic. Data pages that languish in physical memory unreferenced may be sent out to a *swap area* on disk. When the data is accessed again, the virtual memory subsystem copies the needed pages from swap back into physical memory.

Two daemons manage the virtual memory subsystem.

© 2004 Hewlett-Packard Development Company, L.P.

`vhand`

The kernel always tries to maintain a threshold of free pages in order to keep the system running efficiently. As long as this threshold, referred to as *lotsfree*, is maintained, no paging occurs. When the number of free pages drops below this threshold, a daemon known as **`vhand`** selects pages that haven't been recently referenced, copies them out to swap, then adds the pages to the free page list. This is referred to as a *page-out*. A *page fault* occurs when a process tries to access a page that is not currently in memory. The page will then be copied into RAM via a *page-in*, either from the swap space or from the executable on disk.

On systems with very demanding memory needs (for example, systems that run many large processes), the paging daemons can become so busy swapping pages in and out, that the system spends all of its time paging, and not enough time running processes. When this happens, system performance degrades rapidly. At this point, the system is said to be *thrashing*. If you suspect that your system may be thrashing, check the **`ps`** output to see how much CPU time the **`vhand`** process is using. If **`vhand`** is consuming a large amount of CPU time, consider purchasing more physical memory!

`swapper`

The term *swap* dates back to early UNIX implementations that managed physical memory resources by moving entire processes between main memory and secondary storage. Most modern virtual memory systems today no longer swap entire processes, because this method causes the system to spend most of its time processing I/O instead of doing real work. Swapping has been replaced by a *deactivation* scheme, which allows pages to be pushed out over time by a paging mechanism. Paging is a more efficient memory resource management mechanism for virtual memory.

When the system begins thrashing, or when free memory falls below another threshold, known as *minfree*, the **`swapper`** becomes active. The **`swapper`** deactivates processes, which prevents them from running, and thus reduces the rate at which new pages are accessed. This gives **`vhand`** an opportunity to send the unreferenced pages out to swap, and free pages in physical memory. When **`swapper`** detects that available memory has risen above the minfree threshold and the system is no longer thrashing, it will reactivate the deactivated processes.

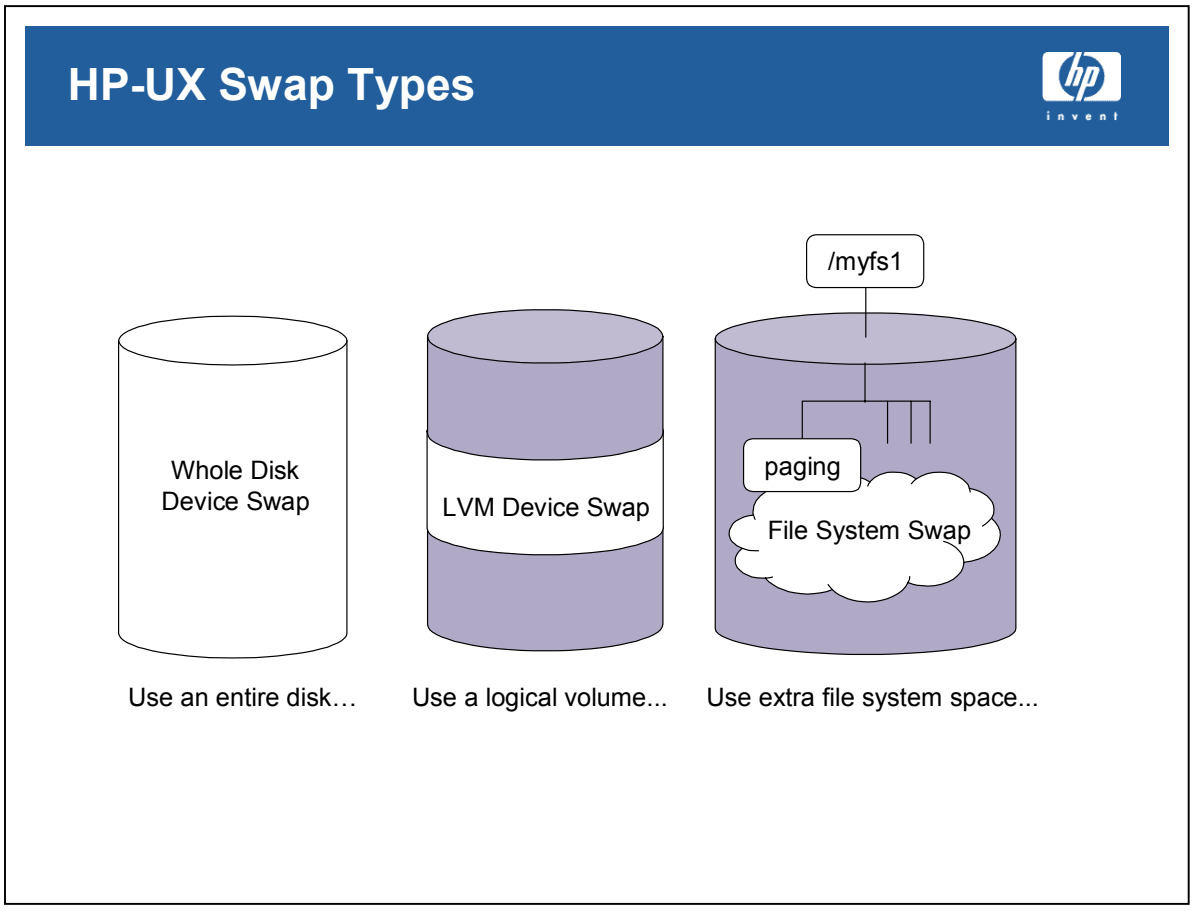
Evaluating Swap Space Needs

Before you install your system, try to guess how much swap space you will need. Most application programs need a minimum amount of swap space to operate properly. This figure is usually contained in the documentation that comes with the application.

After initially installing the system, monitor swap space usage and make adjustments as necessary. We will see how to do this later in this module.

© 2004 Hewlett-Packard Development Company, L.P.

13–3. SLIDE: HP-UX Swap Types



Student Notes

HP-UX supports several different types of swap space.

Whole Disk or LVM Device Swap

Swap space that resides on a dedicated disk, partition, or logical volume is known as *device swap*. Logical volumes or disks configured as device swap don't contain file systems; they are managed and accessed exclusively by the virtual memory subsystem.

Using the *whole disk approach*, you can either use an entire disk for swap, or reserve space at the end of the disk for swap by using the **`-R`** option on **`newfs`**. For example, the following command creates a file system on a disk and reserves 200 megabytes (MB) at the end of the disk for swap:

```
# newfs -R 200 /dev/rdsk/c0t2d0
```


© 2004 Hewlett-Packard Development Company, L.P.

Using the *LVM approach*, you create a separate logical volume for device swap. The following is an example of an **lvcreate** command used to create a 200-MB device swap logical volume:

```
# lvcreate -L 16 -n myswap /dev/vg01
```

After a logical volume is enabled for use as device swap, the logical volume may not be reduced or removed, nor may it be configured for use as a file system.

File System Swap

File system swap is a form of secondary swap. It can be configured dynamically. File system swap allows a process to use an existing file system if it needs more than the designated device swap space. File system swap is used only when device swap space is insufficient to meet demand-paging needs. File system swap consumes a variable amount of space because it only uses that portion of a file system that it needs. You can limit file system swap to a fixed size to prevent it from consuming too much space.

When file system swap is enabled in a file system, a directory called **/paging** is created under the root directory of the file system. A file is created for each swapchunk used in the file system. By default, a swapchunk is 2 megabytes.

Note that once a file system has been enabled for file system swap, it isn't possible to unmount that file system until the swap is disabled at the next system reboot.

Primary versus Secondary Swap

Your system must have at least one device swap area available for use during the boot process. This area is known as *primary* swap. By default, primary swap is located on the same disk as the root file system.

If you are using LVM, the location of the primary swap area is stored in the boot data reserved area (BDRA) in the LVM structures on disk. The **lvlnboot** command may be used to view the name of the primary swap logical volume:

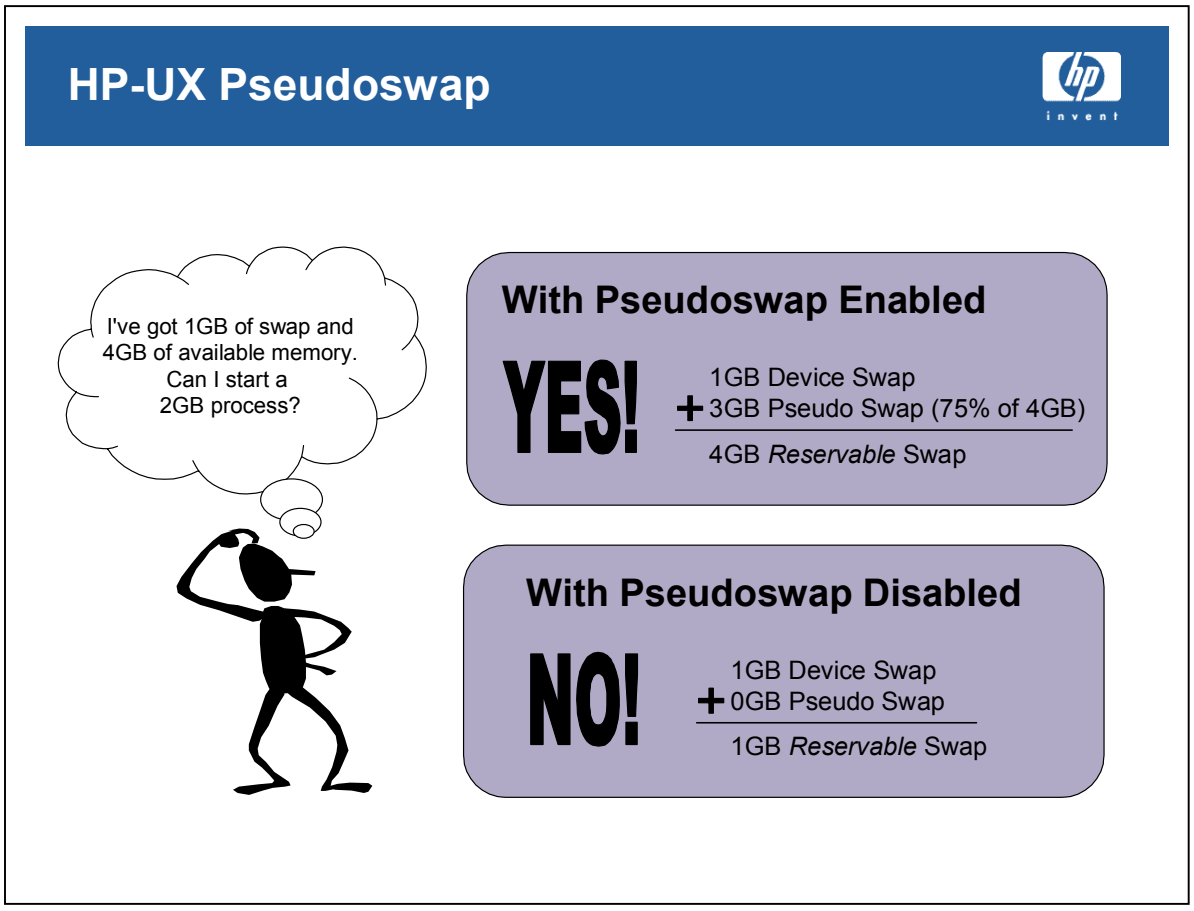
```
# lvlnboot –v
```

Multiple *secondary swap* areas may be configured to supplement primary swap. These secondary swap areas may be configured either as device or file system swap.

Secondary swap can be enabled automatically at boot time or it can be dynamically added while the system is running.

© 2004 Hewlett-Packard Development Company, L.P.

13–4. SLIDE: HP-UX Pseudoswap



Student Notes

Swap Reservation

The swap subsystem *reserves* swap space at process creation time. However, swap space isn't actually *allocated* to a process until the process is paged out to disk. Reserving swap at process creation protects the virtual memory subsystem from running out of swap space.

When the system can't reserve enough swap space for a new process, it won't allow the process to start

While swap reservation prevents the system from running out of swap space, it also prevents processes from being created before all the swap space is actually allocated.

Relaxing the Swap Reservation Policy with Pseudo-Swap

In the early days of HP-UX, the swap reservation policy described above was strictly enforced: a process could only be started if the system could reserve enough swap space such that the entire process could be sent out to swap if necessary. However, since **`vhand`** will never page everything in RAM out to swap simultaneously, a large portion of your swap space may be reserved, but never used.

© 2004 Hewlett-Packard Development Company, L.P.

In order to more efficiently use disk and memory resources, HP-UX now uses a concept known as *pseudo-swap* to relax the swap reservation restrictions. If pseudo-swap is enabled, then HP-UX allows processes to reserve more swap space (up to 75% of available memory) than has actually been configured on the system.

Consider a system with 4 GB of available physical memory, and 1 GB of swap. With pseudo-swap disabled, this system could only run 1 GB worth of processes. With pseudo-swap enabled, processes on this system could reserve up to 4GB of swap (1 GB + 75% of 4 GB).

NOTE:	Note that pages will never actually be "swapped out" to physical memory: pseudo-swap simply relaxes the swap reservation restrictions! This is possible since it will never be necessary to page the entire contents of physical memory out to swap.

Pseudo-swap is enabled by default at HP-UX 10.x and 11.x, but may be disabled by changing the **`swapmem_on`** kernel parameter to 0.

© 2004 Hewlett-Packard Development Company, L.P.

13–5. SLIDE: Enabling Swap from the Command Line

Enabling Swap from the Command Line

Examples:

```
①   # swapon /dev/vg01/myswap
②   # swapon -e /dev/dsk/c0t1d0
③   # swapon /dev/dsk/c0t2d0
④   # swapon -p 4 -l 4M /myfs1
```

Explanation:

① Swap on a logical volume
② Swap on the end of a whole disk device
③ Swap on a whole disk
④ Swap on a file system

Student Notes

Both file system and device swap can be enabled from the command line using the **`swapon(1m)`** command. The examples on the slide show several common uses of **`swapon`**:

- The first example enables the **`/dev/vg01/myswap`** logical volume for use as device swap. The entire logical volume will be claimed as swap, so it will no longer be available for use as a file system. If the logical volume contained a file system in the past, you may need to use the **`-f`** option to "force" an overwrite of the old file system structures.

- The second example enables a file system and swap to coexist on a disk configured using the whole disk approach. The file system must be using **`newfs`**, and the **`-R`** option to reserve space on the end of the disk for swap:

  ```
  # newfs -R 200 /dev/rdsk/c0t1d0
  ```

  Executing the **`swapon`** command enables **`vhand`** to use the reserved space on the end of the disk. The **`-e`** option on **`swapon`** should only be used on disks configured using the whole disk approach.

© 2004 Hewlett-Packard Development Company, L.P.

- The third example enables device swap on the whole disk **/dev/dsk/c0t2d0**. If the disk contains a file system that was created with **newfs -R 200 /dev/rdsk/c0t2d0**, you can preserve the file system and simply enable swap on the available space reserved at the end of the disk by including the **-e** option on **swapon**. If you wish to overwrite the file system on the disk, use the **-f** force option instead.

- The fourth example enables the file system mounted on **/myfs2** for use as file system swap. The **-p** option sets the priority for this swap area to 4, and the **-l** ensures that **vhand** can take no more than 4 MB from the file system for use as swap.

NOTE:	You cannot unmount a file system while the file system is activated for use as file system swap.

Enabling Swap Using SAM

To add swap using the System Administration Manager (SAM) choose **Disks and File systems**, from the SAM functional area launcher. Next choose **Swap**.

SAM will display your current swap areas. Choose either **Add Device Swap** or **Add File System Swap** from the **Actions** menu. Fill out the menu. SAM will allow you to enable the swap now and at every system boot. Selecting at every system boot causes SAM to update the file **/etc/fstab**. This file is used at boot time to activate swap areas.

Enabling Swap Using `swapon`

Syntax for Adding Device Swap

```
/usr/sbin/swapon  [-p priority] [-e |-f] device
```

priority	Indicates the order in which space is taken from the file systems and devices used for swapping. 0 is the highest and 10 is the lowest (default is 1).
`-e`	Uses space after the end of the file system on block device for paging. This option cannot be used with the **-f** option. Do not confuse this with swapping to a file system; this option is to be used with the "whole disk" approach, when a disk has both file system and swap space on it (**newfs -R swap**).
`-f`	Forcibly enables device swapping to a device where an unmounted file system exists (*destroys the file system!*).
device	Block special file which is to be used for paging.

© 2004 Hewlett-Packard Development Company, L.P.

Syntax for Adding File System Swap

```
/usr/sbin/swapon [-m min][-l limit][-r reserve][-p priority] directory
```

-m *min*	*min* specifies the amount of paging space the paging system will initially take from the file system. *min* can be specified in units of kilobytes (k suffix), megabytes (M suffix), or file system blocks (no suffix).
-l *limit*	*limit* specifies the maximum space the swap system is allowed to take from the file system. *limit* can be specified in units of kilobytes (k suffix), megabytes (M suffix), or file system blocks (no suffix). (The default is no limit.)
-r *reserve*	*reserve* specifies the space, in addition to the space currently occupied by the file system, that is reserved for file system use only, making it unavailable to the paging system. This reserved space is in addition to the minimum free space specified by the administrator when the file system was created.
-p *priority*	Same as for device swap.
directory	The directory where the file system swapping occurs. The system creates a directory named **paging** under the root of the file system for swap.

NOTE: You cannot unmount a file system when file system swap is active.

Swap Size Limitations

Several kernel tunable parameters limit the amount of swap that can be made available.

- The default maximum amount of swap space you can configure, for both device swap and file system swap combined, is approximately 512 MB. The tunable system parameter, **maxswapchunks**, controls this maximum. This parameter (default value of 256) limits the number of swap space chunks. The default size of each chunk of swap space is 2 MB. The size of a **swapchunk** can be modified with the **swchunk** kernel tunable parameter.

- The system parameter **nswapdev** in **/stand/system** sets the maximum number of dynamically configured swap devices at ten (default). The maximum is 25. More than **nswapdev** swap devices would require a kernel regeneration.

- The system parameter, **nswapfs**, determines the maximum number of file systems you can enable for swap. The default is 10 and the maximum is 25.

© 2004 Hewlett-Packard Development Company, L.P.

13–6. SLIDE: Enabling Swap via `/etc/fstab`

Enabling Swap via `/etc/fstab`

```
# vi /etc/fstab
    /dev/vg01/myfs1    /myfs1   hfs     defaults      0 2
    /dev/vg01/myfs2    /myfs2   vxfs    delaylog      0 2
①   /dev/vg01/myswap   .        swap    defaults      0 0
②   /dev/dsk/c0t1d0    .        swap    end           0 0
③   /dev/dsk/c0t2d0    .        swap    defaults      0 0
④   /dev/vg01/myfs1    /myfs1   swapfs  pri=4,lim=4M  0 0
# swapon -a
```

Explanation:

① Swap on a logical volume
② Swap on the end of a whole disk device
③ Swap on a whole disk
④ Swap on a file system

Student Notes

In order to ensure that a swap area is enabled at every system boot, define the swap area in the **`/etc/fstab`** file. During the boot process, the **`swapon -a`** command enables all **`/etc/fstab`** swap entries. Fields in the **`/etc/fstab`** file are described below:

block device	The block special file name.
directory	The name of the root of the mounted file system, if there is one. If *type* is **`swapfs`**, it can be the name of any directory.
type	Can be **`swap`**, **`swapfs`**, or **`ignore`** (other values are available, for mounting file systems). If the *type* field is **`swap`**, *directory*, *backup-frequency*, and *pass_number* are ignored. If the *type* field is **`swapfs`**, *block device*, *backup-frequency*, and *pass_number* are ignored.
options	Options to the **`swapon`** command, if the *type* is **`swap`** or **`swapfs`**.
min= *min*	Amount of paging space the paging system will initially take from the file system. Same as **`swapon -m`** option.

© 2004 Hewlett-Packard Development Company, L.P.

lim= *limit*	Maximum space the paging system can take from the file system. Same as **`swapon -l`** option.
res= *reserve*	Space reserved for files in the file system. Same as **`swapon -r`** option.
pri= *priority*	Swap priority. Same as **`swapon -p`** option.
end	Use space after end of file system. Same as **`swapon -e`** option.
backup frequency	Reserved for future use.
pass number	Unused with **`swap`** and **`swapfs`** (used by the **`fsck`** command to determine the order in which file system checks are done).
comment	Optional field that starts with **`#`**.

By listing swap devices in **`/etc/fstab`** you ensure that the swap device is automatically enabled when the system is rebooted. **`swapon -a`** executes as part of the **`/sbin/init.d/swap_start`** script. The **`-a`** option instructs the system to read **`/etc/fstab`** for swap information, and to activate all swap areas.

NOTE: After making changes to **`/etc/fstab`** swap entries, run **`swapon -a`**. This forces the system to reread the **`fstab`** file and enable any newly-defined swap entries, as well as identify possible syntax errors.

© 2004 Hewlett-Packard Development Company, L.P.

13–7. SLIDE: Monitoring Swap Space

Monitoring Swap Space

Examples:

```
①  # swapinfo
②  # swapinfo -f
③  # swapinfo -d
④  # swapinfo -tm
```

Explanation:

① Report usage of all swap areas
② Report only on file system swap
③ Report only on device swap
④ Report values in MB, with a total line

Student Notes

After configuring one or more swap areas on your system, monitor the use of those swap areas over time using **`swapinfo(1m)`**. **`swapinfo`** lists the configured swap areas and reports what percentage of each is currently in use. If you are running low on swap space, you may need to configure an additional swap area to ensure that your users are able to run the applications they need.

Selected **`swapinfo`** options:

`-t` Add a totals line with a type of *tot*.

`-m` Display information in megabytes instead of kilobytes.

`-d` Show information about device swap areas only.

`-f` Show information about file system swap areas only.

`-q` Quiet mode. Print only a total Kb AVAIL.

© 2004 Hewlett-Packard Development Company, L.P.

Sample **swapinfo** output:

```
# swapinfo

             Kb      Kb      Kb   PCT  START/      Kb
TYPE      AVAIL    USED    FREE  USED   LIMIT RESERVE  PRI  NAME
dev      204800       0  204800    0%       0       -    1  /dev/vg00/lvol2
dev       16384       0   16384    0%       0       -    1  /dev/vg01/myswap
dev      303104       0  303104    0%       0       -    1  /dev/dsk/c0t1d0
dev     1025730       0 1025730    0%       0       -    1  /dev/dsk/c0t2d0
localfs    4096       0    4096    0%    4096       0    4  /myfs2/paging
reserve       -   81264  -81264
memory    93564   55308   38256   59%
```

Explanation of output:

TYPE **dev** Device swap.

localfs File system paging space on a file system residing on a local disk.

network File system paging space on a file system residing on another machine. This file system will have been mounted on the local machine via NFS.

reserve Paging space on reserve. This is the amount of paging space that may be needed by processes that are currently running, but that has not yet been allocated from one of the above paging areas.

memory Memory paging area (also known as pseudo-swap). This line appears only if memory paging is enabled.

Kb AVAIL The total available space from the paging area including any paging space already in use. For file system paging areas the value is not necessarily constant. It is the current space allocated for paging (even if not currently used), plus the free blocks available on the file system to ordinary users, minus RESERVE (but never less than zero).

Kb Used The current number of 1-Kbyte blocks used for paging in the paging area. For the memory paging area, this count also includes memory used for other purposes and thus unavailable for paging.

Kb FREE The amount of space that can be used for future paging.

PCT USED The percentage of capacity in use, based on Kb USED divided by Kb AVAIL; 100% if Kb AVAIL is zero.

START/LIMIT For device paging areas, START is the block address on the mass storage device of the start of the paging area. The value is normally 0 for devices dedicated to paging. For file system paging areas, LIMIT is the maximum number of Kbs that will be used for paging, the same as the limit value given to **swapon**.

© 2004 Hewlett-Packard Development Company, L.P.

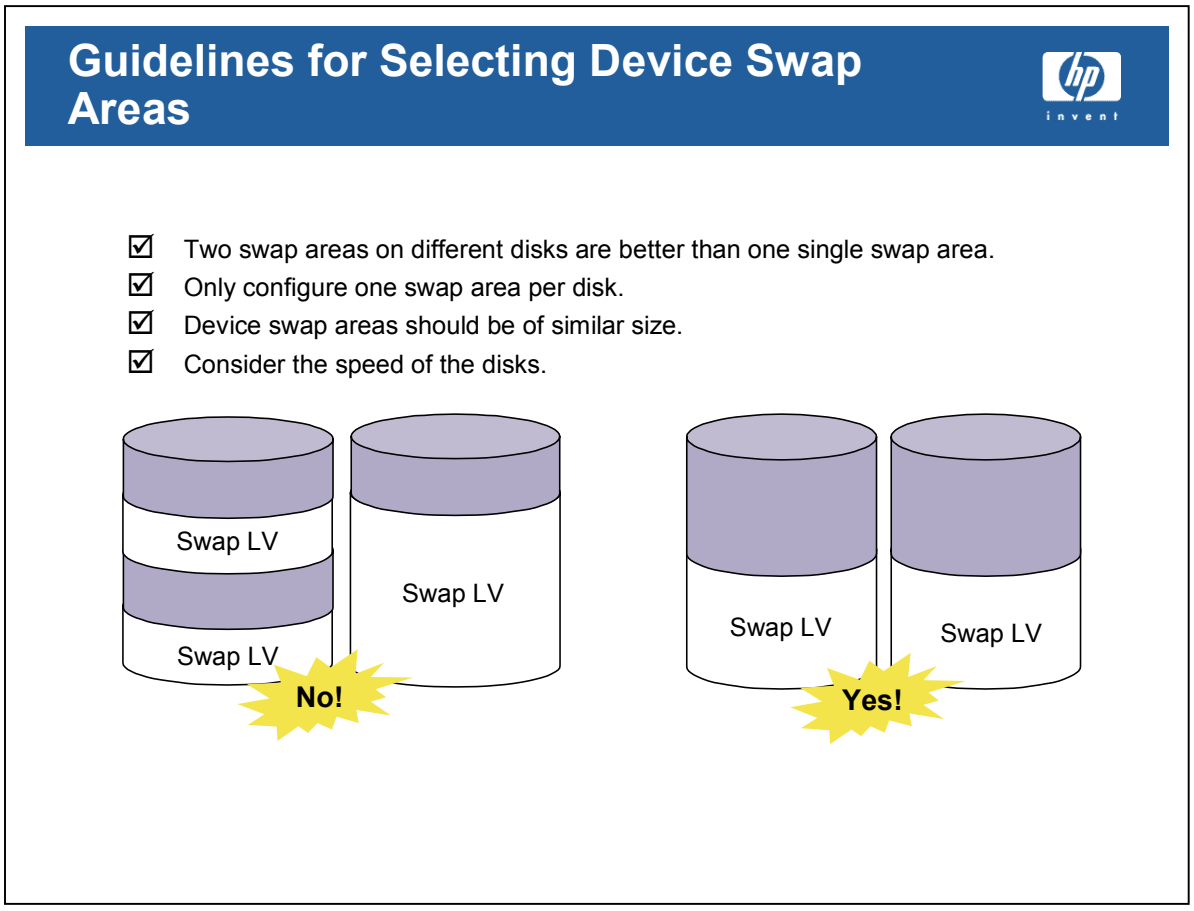
RESERVE — For device paging areas, this value is always -. For file system paging areas, this value is the number of Kbs reserved for file system use by ordinary users, the same as the reserve value given to **`swapon`**.

PRI — The same as the priority value given to **`swapon`**. This value indicates the order in which space is taken from the devices and file systems used for paging. Space is taken from areas with lower priority values first. *priority* can have a value between 0 and 10.

NAME — For device paging areas, the block special file name whose major and minor numbers match the device's ID. For file system swap areas, NAME is the name of a directory on the file system in which the paging files are stored.

© 2004 Hewlett-Packard Development Company, L.P.

13–8. SLIDE: Guidelines for Selecting Device Swap Areas



Student Notes

To ensure optimal performance, you should follow the guidelines shown on the slide.

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. Also for performance reasons, multiple swap sections on the same disk should *not* be used.

Multiple swap areas are used in an interleaved fashion. *Interleaved swapping* means that space from one swap device is used, and then space from another space device. If the swap areas are on different disks, the swap areas can be written to concurrently, in order to avoid disk head contention for multiple writes to the same disk.

Don't create a separate device swap area on the disk containing the primary swap area, because this causes excessive head movement on that disk and slows the system down.

If you are using LVM, set up multiple device swap areas in logical volumes that are on different disks (physical volumes).

Device swap areas should be of similar sizes for best performance. Otherwise, when all the space in the smaller device swap area is used, interleaving is no longer possible.

© 2004 Hewlett-Packard Development Company, L.P.

How Swap Space Is Prioritized

All swap devices and file systems enabled for swap are assigned a priority ranging in value from 0 to 10. The priority number determines which swap areas will be used first. The priority is set by swap activation with the `-p` option of **`swapon`**. The default priority number is 1. Pages are sent out to your configured swap areas according to the following rules:

- Start at the highest priority swap device or file system. The lower the number, the higher the priority; that is, space is taken from a system with priority "0" before it is taken from a system with priority "1".

- If multiple devices have the same priority, swap space is allocated from the devices in a round-robin fashion. Thus, to interleave swap requests among a number of devices, the devices should be assigned the same priority. Similarly, if multiple file systems have the same priority, requests for swap are interleaved among the file systems.

It is recommended that you assign the same swapping priority to most swap devices, unless a device is significantly slower than the rest. Assigning equal priorities limits disk head movement, which improves swapping performance.

© 2004 Hewlett-Packard Development Company, L.P.

13–9. SLIDE: Guidelines for Selecting File System Swap Areas

Guidelines for Selecting File System Swap Areas

- ☑ Avoid file system swap altogether if possible.
- ☑ Avoid using busy file systems such as the root file system.
- ☑ Avoid using file systems already near capacity.
- ☑ Avoid using file systems on disks that already have swap configured.
- ☑ Set priorities appropriately.
  - Choose faster devices over slower devices.
  - Choose infrequently-used file systems over busier file systems.

Student Notes

When you need more swap space and you have no disk space available for additional device swap, you can dynamically configure file system swap. File system swap is significantly slower than device swap, though, so it should only be configured as a last resort.

These guidelines are provided to help the system administrator decide which file system sections should be used for swap space. Once again, most of these rules stem from performance issues.

When you add swap areas, you can assign a priority to each swap area. Priorities range from 0 (the highest) to 10 (the lowest). The system uses the swap areas with higher priority before using the lower priority swap areas. If you assign the same priority to two different swap areas, the system will use each of them on an alternating basis.

In general, it is best to assign highest priorities to the swap areas that afford the fastest performance. Therefore, give device swap areas priority over file system swap areas; give faster devices priority over slower devices; give lower use file systems priority over higher use file systems. Swap devices of the same type, (e.g. all device swap on disks of approximately the same speed) should all be assigned the same priority to take advantage of interleaved swap.

© 2004 Hewlett-Packard Development Company, L.P.

Enabling two file systems on the same disk (or, physical volume), can result in excessive head movement and will result in slower system performance.

NOTE: Device swap offers better performance than file system swap. If given a choice, use device swap.

© 2004 Hewlett-Packard Development Company, L.P.

13–10. LAB: Swap

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. How much physical memory does your training system have?
   How much available memory?
   How much lockable memory?

2. Is pseudoswap enabled on your system? Use the **`kmtune`** command (11i v1) or **`kctune`** command (11i v2) to view your kernel parameters and find out!

   ```
   # kmtune -q swapmem_on    (11i v1)
   # kctune swapmem_on       (11i v2)
   ```

3. How much device swap is currently configured on your system?
   How much file system swap is currently configured?
   How much swap space has been reserved by processes on your system?

4. As you add new applications on your system, your application vendor may ask you to configure additional swap space. Create an additional 100 MB logical volume in vg00 and enable it for use as device swap.

© 2004 Hewlett-Packard Development Company, L.P.

5. In crunch periods, you may need to configure file system swap to supplement already configured device swap. Enable file system swap on **`/home`**. Set the priority level to 4 to ensure that this file system swap is only used when your device swap areas are full. Also ensure that no more than half of the file system will be used as swap.

6. Execute **`swapinfo`** again to verify that your new swap areas are configured properly.

7. Do whatever is necessary to ensure that your new swap areas are automatically enabled after the next reboot.

8. Can you unmount a file system that has file system swap enabled? Try it.

9. You should have discovered in the previous question that once swap is enabled in a file system, it is impossible to unmount that file system. Is this a problem? Explain.

© 2004 Hewlett-Packard Development Company, L.P.

10. Try to view the man page for **swapoff**. What happens?

NOTE:	There is no **swapoff** command in HP-UX! To disable swap, edit the **/etc/fstab** file; comment out or delete the lines for the swap areas you wish to disable; and reboot. When the system reboots, the commented lines in **fstab** will be ignored. There is no way to disable swap without rebooting.

11. You need to disable both of your new swap areas before moving on to the next section. Remove your swap entries from **/etc/fstab**, then reboot using the command shown below:

```
# vi /etc/fstab            remove all swap entries
# shutdown -ry 0
```


© 2004 Hewlett-Packard Development Company, L.P.

Module 14 — Managing Printers

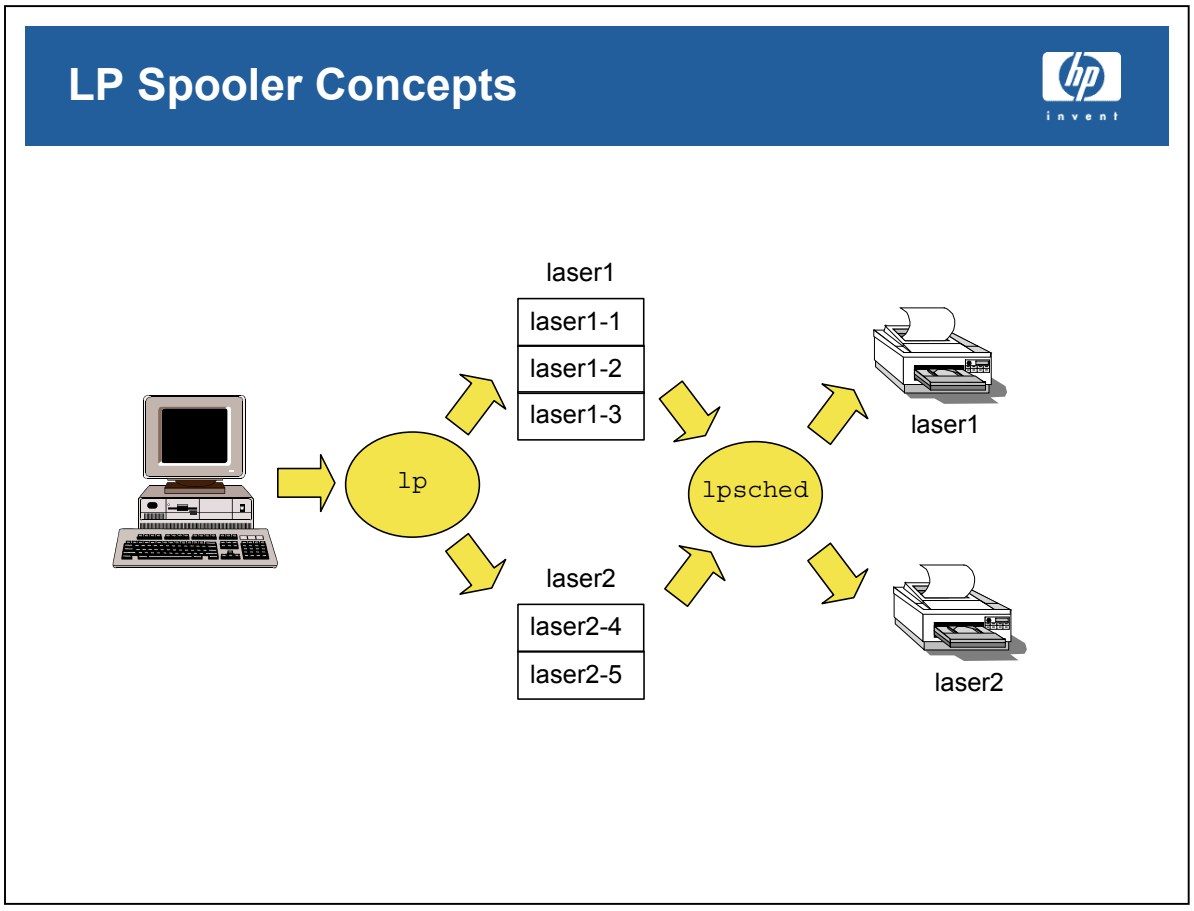
Objectives

Upon completion of this module, you will be able to do the following:

- Distinguish between local, remote, and network printers.
- Add and remove local, remote, and network printers using system administration Manager (SAM).
- Start and stop the LP spooler from the command line.
- Check LP spooler status from the command line.
- Manage print queues from the command line.

© 2004 Hewlett-Packard Development Company, L.P.

14–1. SLIDE: LP Spooler Concepts



Student Notes

Standard HP-UX provides two different spooling systems for managing print requests: the HP Distributed Print Service (HPDPS) and a variation on the original AT&T LP spooler. Although HPDPS provide much more flexibility than the LP spooler, it is significantly more complex to administer. Most HP-UX administrators use the LP spooler.

Users submit requests to the spooling system via the **`lp`** command. **`lp`** assigns a unique request ID to the users print request, then chooses a queue for the submitted job. Users may explicitly specify a destination printer, or simply allow **`lp`** to enter the job in the default printer's queue as defined by the administrator.

Each job entered in the queue is assigned a priority between 0 and 7 that will eventually determine which job prints first. The user may specify a priority level, or simply accept the printer's default priority as defined by the administrator.

When the destination printer becomes available, the **`lpsched`** scheduling daemon checks the printer's queue, selects the highest priority job and invokes an appropriate interface script to pass the request to the destination printer.

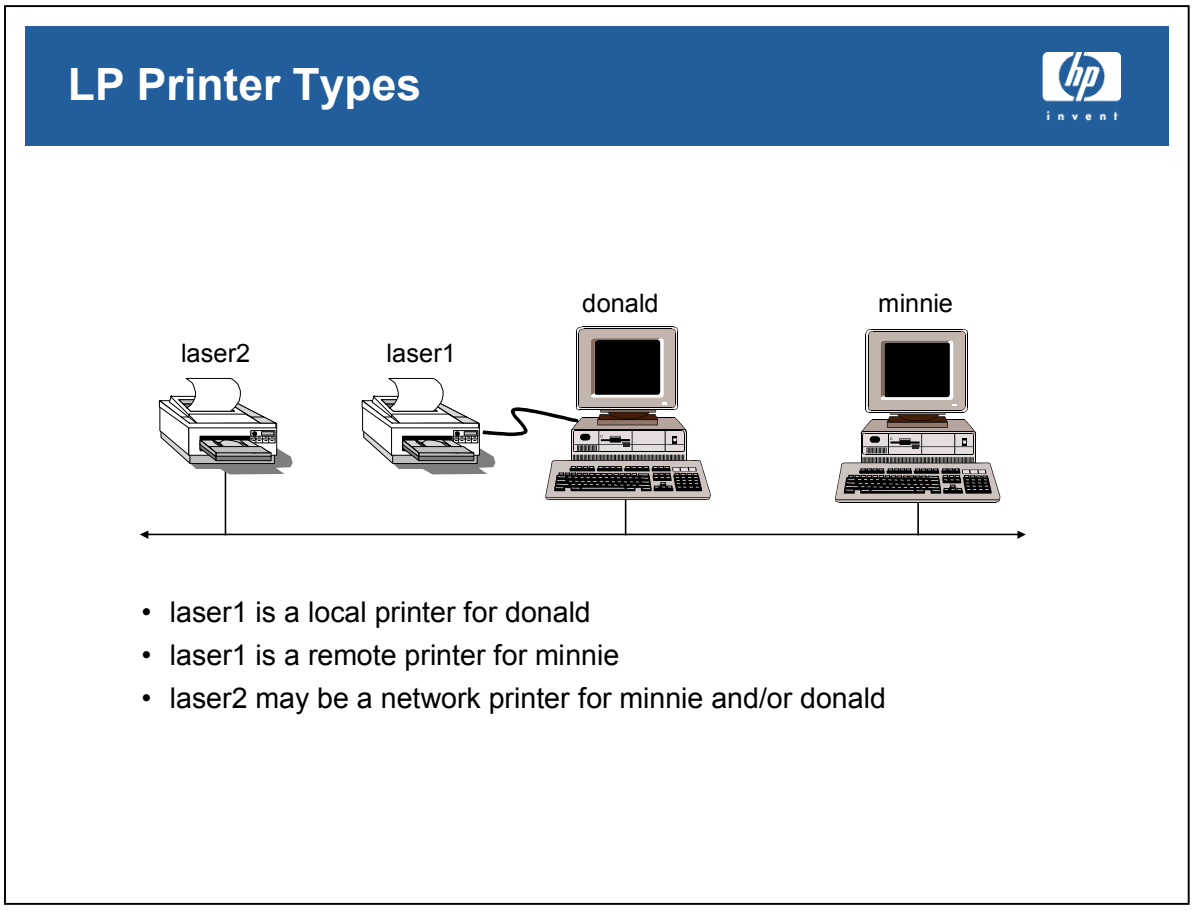
© 2004 Hewlett-Packard Development Company, L.P.

The administrator has a significant amount of control over this entire process. Using command line utilities or SAM, you may:

- Add or remove queues in the spooling system.
- Temporarily disable downed printers.
- Temporarily reject additional requests in a queue.
- Cancel requests.
- Re-prioritize already submitted requests.
- Move requests from queue to queue.
- And much more!

© 2004 Hewlett-Packard Development Company, L.P.

14–2. SLIDE: LP Printer Types



Student Notes

The LP spooler recognizes three different categories of printers.

Any printer that is directly connected to your system by a serial or parallel interface is considered to be a local printer. In the example on the slide, laser1 appears to be connected to *donald* via a serial or parallel cable. Thus, laser1 is *donald's* local printer.

A network printer is printer that has its own network interface card, and its own connection to the local area network. The printer can then be shared by multiple hosts on the LAN, even if those systems are running different operating systems. In the example on the slide, laser2 appears to be connected directly to the LAN. Hosts *donald* and *minnie* could both configure laser2 as a network printer

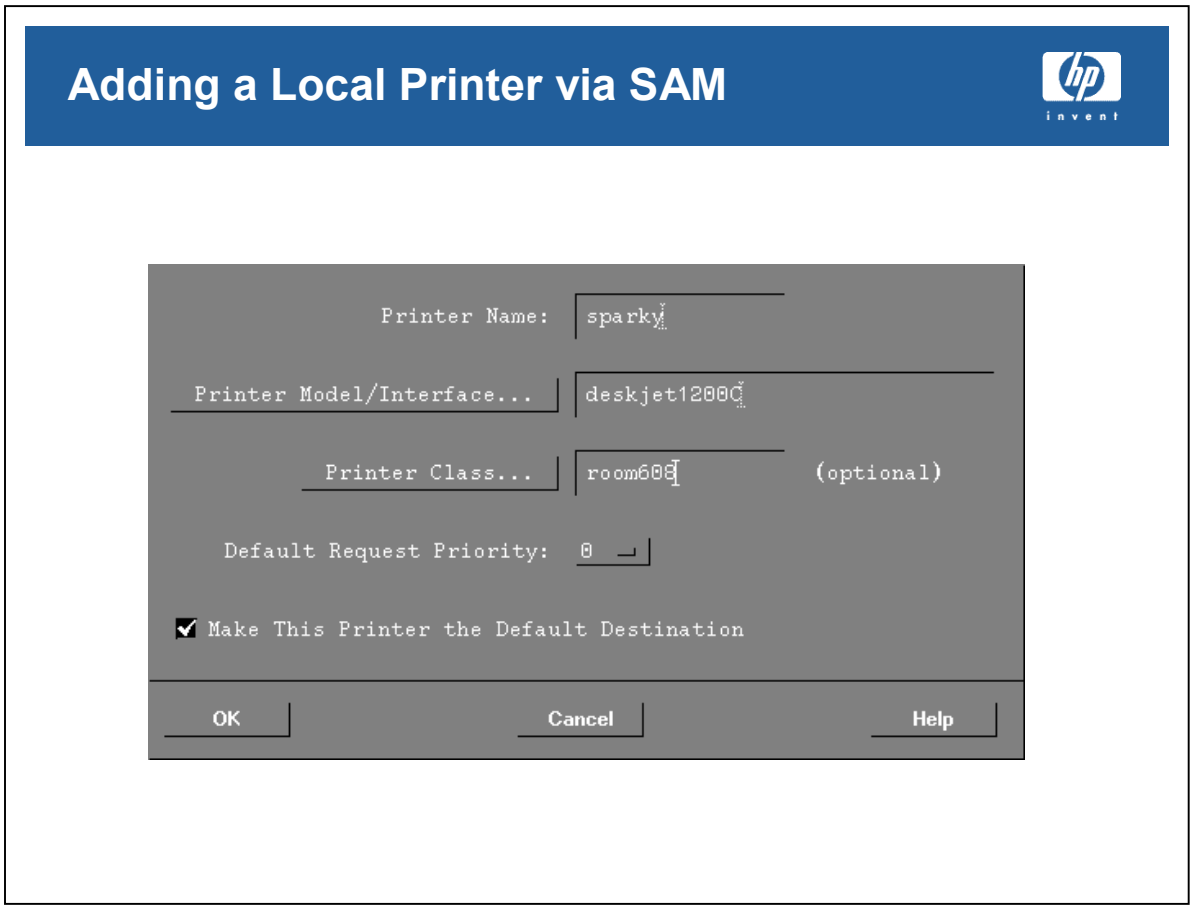
A remote printer is simply a printer that is configured on a system other than the one you are logged into when you submit a print request. In the example on the slide, *minnie* doesn't have a direct connection to laser1. However, *minnie* could configure laser1 as a remote printer, via host *donald*. All requests to laser1 must first be passed to *donald*.

© 2004 Hewlett-Packard Development Company, L.P.

Some administrators choose to funnel all requests to their network printers through a single host configured as a print server. In the example on the slide, for instance, laser2 could be configured as a network printer for donald. *minnie* might then choose to configure laser2 as a remote printer via *donald*. This provides a single point of control for laser2: all requests to laser2 would have to funnel through the queue on *donald*. This gives the administrator a chance to monitor and prioritize print requests on a central print server.

© 2004 Hewlett-Packard Development Company, L.P.

14–3. SLIDE: Adding a Local Printer via SAM



Student Notes

When adding a local printer you must be able to answer the following questions:

- What is the name you are giving to this printer?
- What is the appropriate model script?
- Do you want to set a default printer priority?
- Do you want to specify a printer class?
- Will this be the system default printer?

In addition, you must be sure the printer driver is in the kernel. You can then add the local printer either using SAM or the **`lpadmin`** command. If you are using **`lpadmin`**, you must also know the device file name of the printer.

To add a local printer using SAM, select **`Printers and Plotters`** from the functional area launcher. Select **`lp Spooler`** from the sub menu. Select **`Printers and Plotters`** from the next sub menu, then select **`Add Local Printer/Plotter`** from the **`Action`** menu. SAM will present a list of types of local printers from which to choose. The list

© 2004 Hewlett-Packard Development Company, L.P.

includes Parallel and Serial. If the print driver is not in the kernel, SAM will prompt you to reconfigure the kernel before proceeding. SAM will scan the hardware for devices of the appropriate type. For example if you choose Parallel as the type of printer, SAM will display all the parallel ports. Highlight the device you wish to configure as a printer and press **OK**. SAM will then display a menu to add the printer.

For each local printer you wish to add you must specify the following:

Printer Name

This is the name being assigned for use whenever this particular printer or plotter is accessed. This is the name that accompanies the **-d** option with the **lp** command. It can be any unique combination of up to 14 alphanumeric characters and underscores.

Printer Model

A printer model is a model of an interface script that is shipped with the print/Interface spooler software. Each printer has an interface program that tells the print spooler how to access the special features of the printer. The interface models can be used without modification, or you can create customized interface programs or modify copies of the model interfaces. The interface model scripts are in the directory **/usr/lib/lp/model**. To see a list of the printer models available on the system, select the **Printer Model/Interface** button. If your printer does not match any of these, use the **dumb** model. When you display the list of model scripts you will notice that there are two types of interface scripts available for many printers. One interface script is the standard HP model script, the other is for use with **SharedPrint/UX. SharedPrint/UX** enhances the basic capabilities of the **lp** spooling system, using a client-server implementation. **SharedPrint** is not activated by default. It requires either Network Computing System (NCS) or HP DCE/9000. One of these must be configured in order to use **SharedPrint.**

We will not be covering **SharedPrint/UX** configuration in this course. For more information refer to: *SharedPrint/UX User and Administrator's Guide*, Part Number B1171-90124.

Printer Class

A class is a group of related printers and plotters. When print requests are sent to a class, they are serviced by the first available printer or plotter that is a member of that class. A class name can be any unique combination of up to 14 alphanumeric characters and underscores. To create a new class, simply enter a new name.

Default Request Priority

This defines the default priority of print requests submitted to this printer. Any job submitted to the printer uses this priority unless the priority is explicitly set to a different value by a **-p** option in the lp command used to send the request to the spooler. Priorities range from 0 to 7 with 0 being the lowest priority and 7 being the highest.

© 2004 Hewlett-Packard Development Company, L.P.

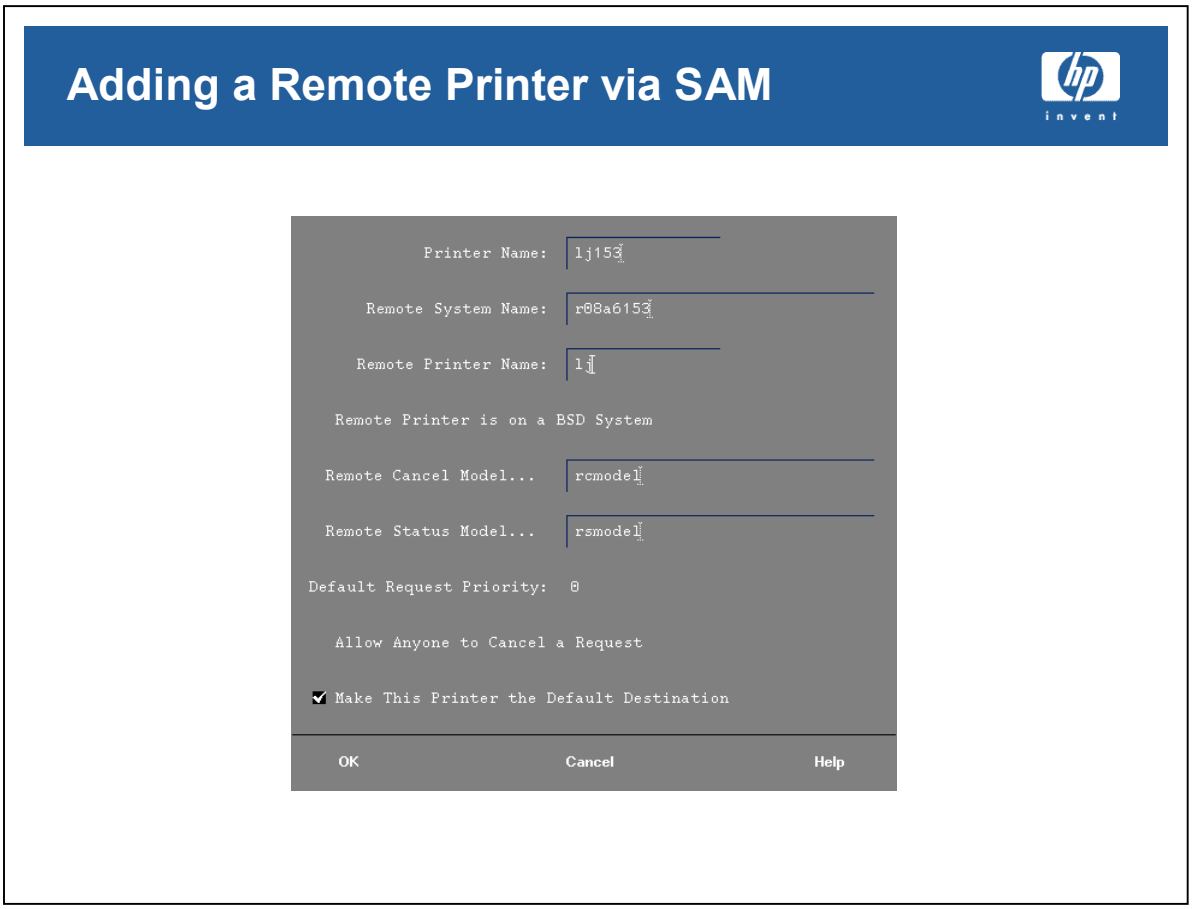
Default Destination	All print requests sent by the **lp** command without a **-d** option are sent to the default destination, unless the LPDEST variable is set in the user's shell environment.

When you have finished filling out the menu press **OK**. SAM will give you the option of adding the printer to the CDE front panel. SAM will temporarily shut down the **lp** spooler to add the printer and then restart the **lp** spooler.

NOTE:	Anytime a printer is added to the lp spooler, SAM stops and restarts the scheduler. This causes all currently printing jobs scheduled by your host to restart at page 1.

© 2004 Hewlett-Packard Development Company, L.P.

14–4. SLIDE: Adding a Remote Printer via SAM



Student Notes

A remote printer is a printer that has already been configured on another system. To configure a remote printer you must be able to answer the following questions:

- What is the name you are giving to this printer?
- What is the remote system name?
- What is the remote printer name?
- Will this be the system default printer?
- Will you allow anyone to cancel a request, or only root?
- Is the remote printer on a BSD system?

To add a remote printer select **`Add Remote Printer/Plotter`** from the **`Actions`** menu of the Printers/Plotters area. SAM will present you with a menu. You must specify the following:

© 2004 Hewlett-Packard Development Company, L.P.

Printer Name	This is the name assigned for use whenever this particular printer or plotter is accessed. It is the name that accompanies the **-d** option with the **lp** command. It can be any unique combination of up to 14 alphanumeric characters and underscores. This name does not have to match the name used on the system to which the printer is connected.
Remote System Name	This is the Internet host name of the system that print requests for this remote printer are sent to. This hostname must be able to be resolved either in the **/etc/hosts** file or by using a naming service such as DNS or NIS.
Remote Printer Name	This is the name the remote system uses to access the printer.
BSD System	If your print requests are intended to go to (or pass through) a Berkeley Software Distribution (BSD) system, you must check this box. Berkeley systems use a different spooler.
Remote Cancel Model	The system uses an interface program to cancel print requests on a remote system. For remote printers, use the cancel model script rcmodel. This script is found in the **/usr/lib/lp/cmodel** directory.
Remote Status Model	The system uses an interface program to obtain the status of print requests on a remote system. For remote printers, use the status model script rsmodel. This script is found in the directory **/usr/lib/lp/smodel**.
Default Request Priority	This defines the default priority of print requests submitted to this printer. Any job submitted to the printer uses this priority unless the priority is explicitly set to a different value by a **-p** option in the **lp** command used to send the request to the spooler. Priorities range from 0 to 7 with 0 being the lowest priority and 7 being the highest.
Allow Anyone to Cancel?	Toggle to allow users to cancel other users' print requests. The default (toggle not set) is that a user can cancel only his or her own print requests. This option is not available for local printers.
Default Destination?	All print requests sent by the **lp** command in which no **-d** option was specified, are sent to the default destination, unless the LPDEST variable is set in the user's shell environment.

Note, you do not specify a model interface script. The model interface script name **rmodel** is used for remote printers. This script will issue the **rlp** command to send LP line printer requests to a remote system.

© 2004 Hewlett-Packard Development Company, L.P.

Tasks Performed on the Remote System

If you are having trouble printing on the remote system, make sure that the remote system is configured properly:

- Edit the file **`/etc/services`** and, if needed, uncomment the line beginning with the word **`#printer`** by removing the **`#`**.

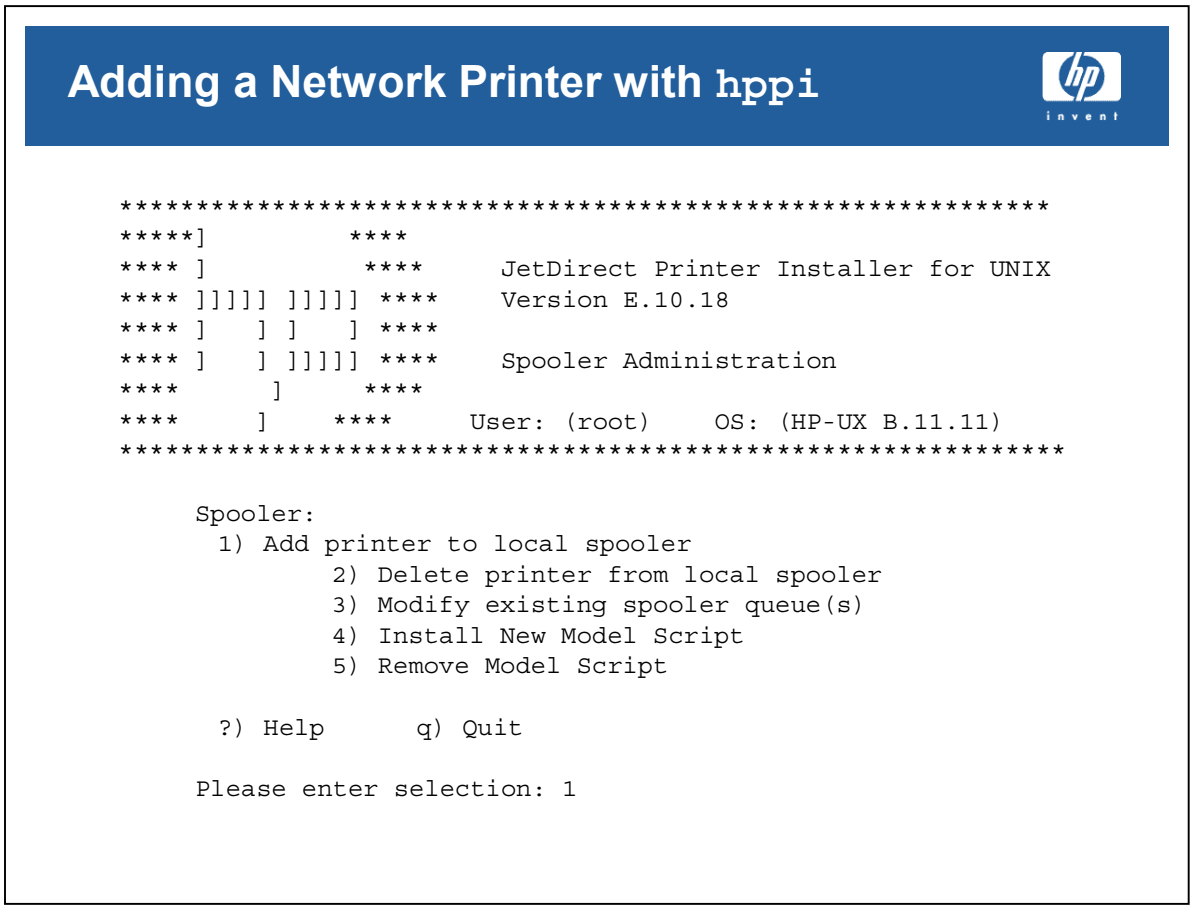
- Ensure that no systems are disallowed access via **`/var/adm/inetd.sec`**.

- **`rlpdaemon`** must be running on the remote system so that it can accept remote print requests. Edit the file **`/etc/inetd.conf`** and, if needed, uncomment the line beginning with the word **`#printer`**. Then invoke the command **`inetd -c`**.

- Ensure the **`/etc/hosts`** files on both systems are configured.

© 2004 Hewlett-Packard Development Company, L.P.

14–5. SLIDE: Adding a Network Printer with `hppi`



Student Notes

Many printers today have their own network interface cards and connect directly to a Local Area Network rather than a parallel or serial port on a server or workstation. Any host on the network, then, can submit requests directly to the printer. Adding a network printer to your local print spooler requires special vendor-specific software. Printers that use HP's "JetDirect" interface cards are configured using HP's "HP Printer Installer" (**`hppi`**) utility. **`hppi`** is part of the HP Network Printer Library (HPNPL) product which can be downloaded and installed free of charge from **`http://www.hp.com`**.

After you have installed HPNPL according to the instructions on the web site, the procedure for adding a network printer to your print spooler is fairly straightforward:

1. Run **`hppi`**. (If you choose "Actions -> Add Network Printer" from the SAM Printers and Plotters object list, SAM simply opens a terminal emulator window and runs this same command).

   ```
   # /opt/hpnpl/bin/hppi
   ```


© 2004 Hewlett-Packard Development Company, L.P.

2. **hppi** is a fairly intuitive menu-based utility. From the main menu, choose option 1, **Spooler Administration**.

```
# /opt/hpnpl/bin/hppi
  -> 1) Spooler Administration
```

3. The "Spooler Administration" option should display the menu shown on the slide. From this screen, you can add, modify, or remove network printers. Choose option 1, "Add printer to local spooler".

```
-> 1) Add printer to local spooler
```

4. You will be prompted for the printer hostname or network IP address. You can obtain this information from your IT department or by generating a printer test page.

```
Enter the network printer name or IP address (q - quit):
laser
```

or

```
Enter the network printer name or IP address (q - quit):
128.1.1.1
```

After you enter your printer's network name, **hppi** attempts to verify that the specified printer is actually reachable on the network. If the printer is unreachable, contact your networking department to help troubleshoot the problem.

5. When asked if you wish to enable realtime model script, accept the default response: **Disable**.

```
Do you want to Enable Realtime Model Script Update (0-Enable,
default-Disable)?
```

6. Next, you have an opportunity to modify several printer parameters.

```
The following is a list of suggested parameter values for this
queue. You may change any settings by selecting the corresponding
non-zero numbers. The values will be used to configure this queue
when '0' is selected. To abort the operation, press 'q'.

Configurable Parameters:                Current Settings
------------------------                ----------------

1) Lp destination (queue) name:          [laser]
2) Status Log                            [(No Log)]
3) Queue Class                           [(Not assigned)]
4) Default Queue                         [NO]
5) Additional printer configuration...

Select an item for change, or '0' to configure (q - quit): 5
```


© 2004 Hewlett-Packard Development Company, L.P.

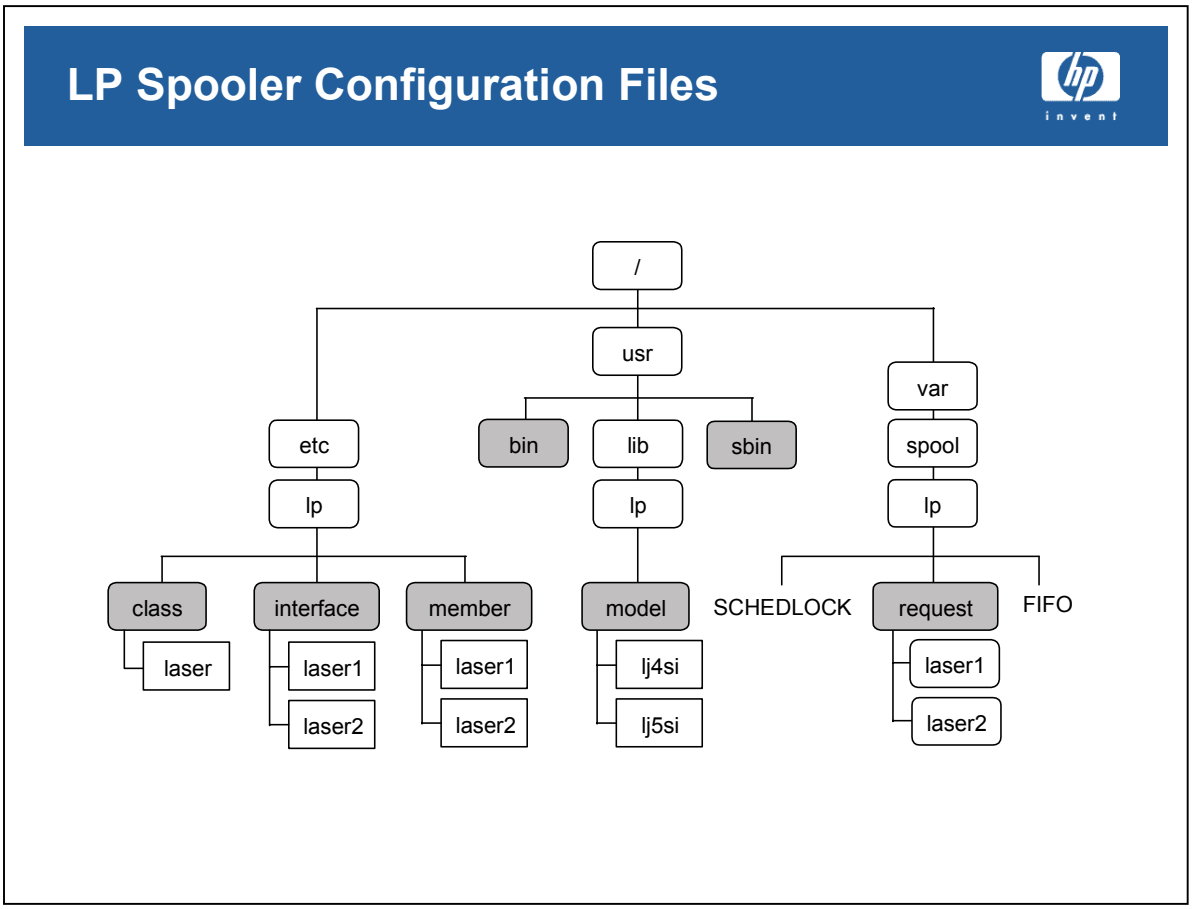
```
The following is a list of suggested parameter values for this
printer.  To abort this operation, press 'q'.

Configurable Parameters:              Current Settings
------------------------              ----------------

1) Model Script:                       [net_lj4x]
2) Default Printing Language           [AUTO]
3) Job Recovery                        [ON]
4) True End-of-Job                     [ON]
5) Banner Page                         [ON]
6) PostScript Level                    [Level 2]
7) Banner Tray                         [default]
8) Banner Paper Size                   [default]
9) Duplex Print                        [default]

Select an item for change, or q when done: q
```

Set `these` items as you wish, then press "0" to finish adding the printer to your local spooler.

The same **`hppi`** utility can be used to remove, or modify network printers, too: simply follow the menu prompts!

© 2004 Hewlett-Packard Development Company, L.P.

14–6. SLIDE: LP Spooler Configuration Files



Student Notes

The LP spooler system is configured via a collection of directories, configuration files, and executables under the **`/etc`**, **`/usr`**, and **`/var`** directories. An understanding of these directories often proves useful when configuring the spooler, and troubleshooting spooling problems. The most important components of the spooling system are described below:

The following is a road map of the directories in the spooling system.

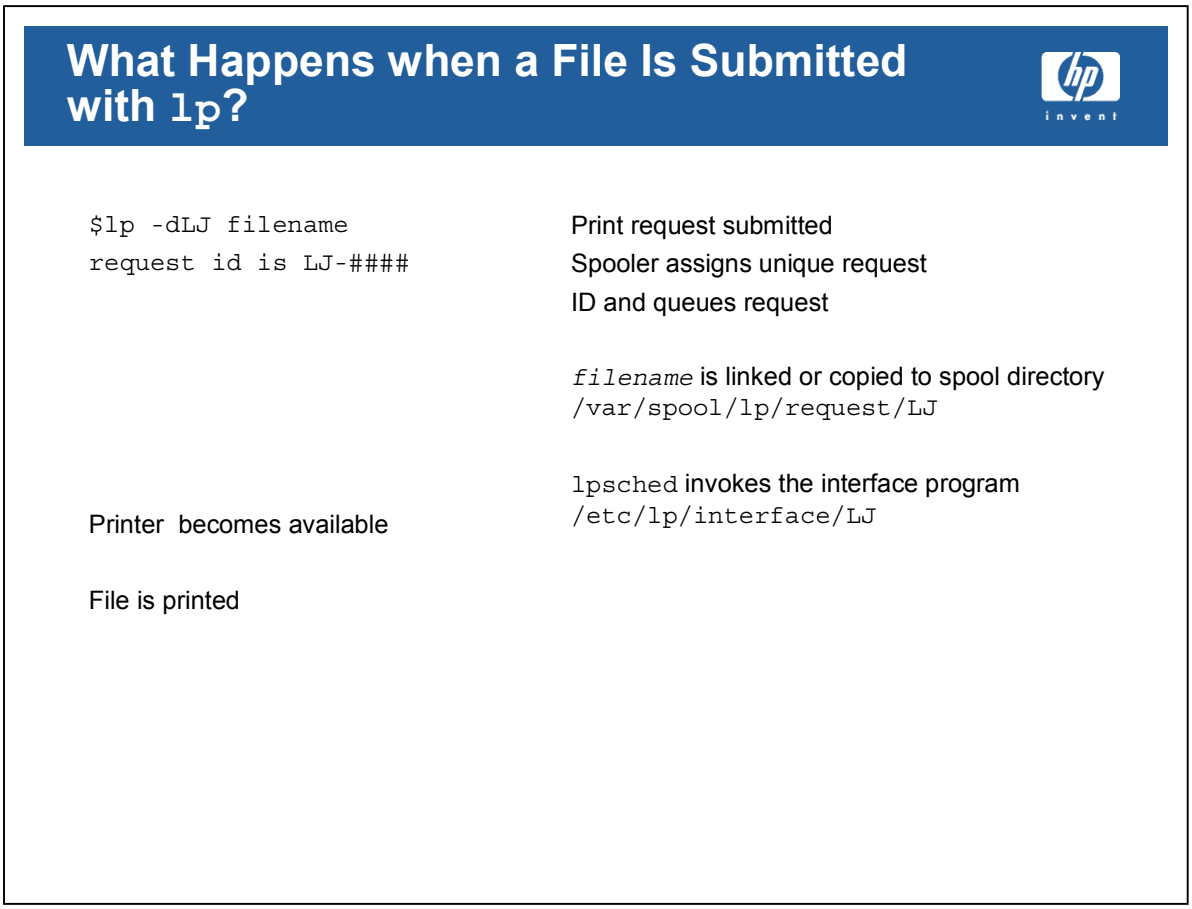
`/var/spool/lp`	lp spooler system parent directory. All information about the setup and printing queues is located here.
`/etc/lp/class`	You may choose to pool several printers together in a printer class. Users then have the option of submitting requests to the printer class, or an individual printer. Jobs submitted to a printer class are automatically entered in the queue of the printer in the class that has the fewest pending requests. **`/etc/lp/class`** contains one configuration file for each class defined on the system. .

© 2004 Hewlett-Packard Development Company, L.P.

/usr/lib/lp/model	System-supplied interface programs. This directory contains the model shell scripts designed to submit requests to various printer models. The model scripts are responsible for generating banner pages and setting default fonts and other options. When adding a printer to the spooling system you will select a model script from this directory. The model script is copied into the **/etc/lp/interface** directory and renamed to match the name of the printer.
/etc/lp/interface	Interface programs in use on your system. This has shell scripts from **/usr/lib/lp/model** that may be modified for particular printers. For example, if you do not want a printer to produce the banner page, modify the interface program.
/var/spool/lp/request	Destination queues. This is where all **lp** requests are queued. It usually contains a subdirectory for each printer configured on the system.
/usr/bin	Contains user-executable commands, including **lp** spooler commands that general users can execute.
/usr/sbin	Contains administrator commands, including the **lp** commands that only root or the lp user can execute.
/etc/lp/member	Lists all configured printers, one file per printer. The contents of each printer's member file is the assigned device file.
/var/spool/lp/SCHEDLOCK	This file is created when the lp scheduler is started with the **lpsched** command. It ensures that only one copy of the scheduler can run. The SCHEDLOCK file is removed when the scheduler is properly shut down with the **lpshut** command. If the scheduler is not properly shut down, you must remove the SCHEDLOCK file manually before restarting the scheduler. It is done automatically at boot time.
/var/spool/lp/FIFO	This file is used internally by the lp scheduler. Like SCHEDLOCK, it is created automatically by the scheduler and removed when the scheduler is stopped.
/usr/lib/lp/fonts	Contains fonts for LaserJet printers.
/usr/spool/lp/log	Contains spooler log file.

© 2004 Hewlett-Packard Development Company, L.P.

14–7. SLIDE: What Happens When a File Is Submitted with `lp`?



Student Notes

When the **`lp`** command is invoked, a print destination is determined and a print request is submitted to the spooling system. A unique request ID number is assigned to each print request.

The print destination (in our example, LJ) is a logical name known to the spooling system and related to a spool directory. Usually, the destination is the name of a particular printer. However, several printers can be combined in a class of printers, so a class may also be a destination. If the destination of a print request is the name of a printer, the request will go to that printer's queue only; however, if the destination is a class of printers, the request will be spooled on to that class's queue and will be printed by the first printer that is available within that class. This helps avoid backups in a particular queue.

All print requests are queued in their destination spool directory strictly in priority order, that is the job with the highest priority will print first. If two jobs have the same priority they will be printed in FIFO (first-in-first-out) order.

© 2004 Hewlett-Packard Development Company, L.P.

Before a print request is actually printed, an interface script is invoked by the spooling system. An *interface script* is the link between the scheduler and the device (that is, its device file). It takes arguments from the scheduler and sends header and configuration information to the device. Each device has its own interface script which can be modified by the administrator.

Why Modify Interface Scripts?

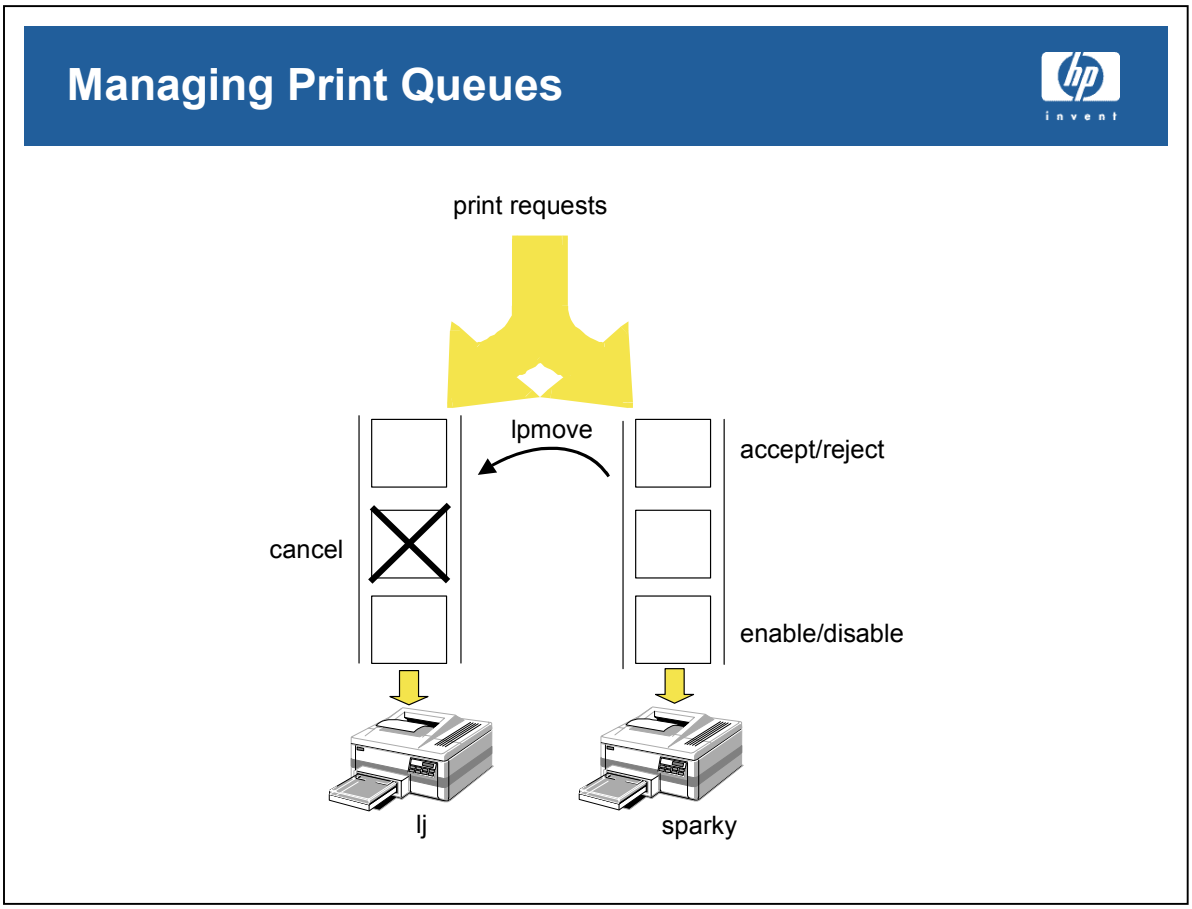
You may want to modify the speed of the serial line to which the printer is attached. The default speed is 9600 bits per second (BAUD), but many printers are able to work at much higher speed, thus decreasing the time to print a file.

Another example would be if you generally don't want to waste paper with the banner page. You can comment out the **`do_banner`** function in the interface scripts.

You must be a bit more familiar with shell scripts to modify the interface scripts, but for these two examples the changes are easy. Search for "9600" or "do_banner" in the file and do the modifications.

© 2004 Hewlett-Packard Development Company, L.P.

14–8. SLIDE: Managing Print Queues



Student Notes

When a print request is made it is put on a print queue. The queue may be for a specific printer or for a class of printers.

Accept/Reject

The **`lp`** command cannot place a request on a queue unless the destination is accepting requests. When a destination is **`rejecting requests`**, any attempt by the **`lp`** command to queue a request to the destination will fail. Destinations are configured to accept or reject requests either with SAM or the **`accept`** and **`reject`** commands.

Example:

```
# reject -r"Sparky is down. Use the destination lj" sparky
# accept sparky
```


© 2004 Hewlett-Packard Development Company, L.P.

Enable/Disable

While **accept** and **reject** control what can be placed on the queue, **enable** and **disable** control whether the printer will process requests from the queue. The **enable** command activates the named printers, enabling them to print requests taken by **lp**.
disable deactivates the named printers, disabling them from printing requests taken by **lp**. By default, any requests that are currently printing on the designated printers are reprinted in their entirety either on the same printer or on another member of the same class. This can be useful if a print job has begun printing on the wrong type of paper. Printers can also be enabled and disabled using SAM. Any user can enable and disable printers.

Examples:

```
# disable sparky
# enable sparky
```

Moving Requests

Requests can be moved from one queue to another using either SAM, **lpmove(1m)** or **lpalt(1)**. You can either move all the requests in the queue or move individual requests. Before using **lpmove**, you must first shut down the scheduler. Be aware that this restarts all currently printing requests.

Examples:

`# lpshut`	*Must stop the scheduler before* **lpmove***ing requests*
`# lpmove sparky lj`	*Move all requests from sparky to lj*
`# lpmove sparky-123 lj`	*Move only one request to lj*
`# lpsched`	*Restart the scheduler*
`# lpalt sparky-123 -dlj`	*Move only one request. This can be used by an ordinary user*

Cancel

The **cancel** command cancels requests that were made with the **lp** command, even if they are currently printing. At least one ID or printer must be specified. Print requests can also be cancelled using SAM.

Examples:

`# cancel sparky-123`	*Cancel request sparky-123*
`# cancel sparky`	*Cancel current request printing on sparky*

By default any user can cancel any other user's requests.

Display Status

The **lpstat** command displays the output of the line printer spooling system. It displays the status of the user's print requests when used with no options.

There are many more options which display various areas of the LP spooler. The **-t** option displays all status information.

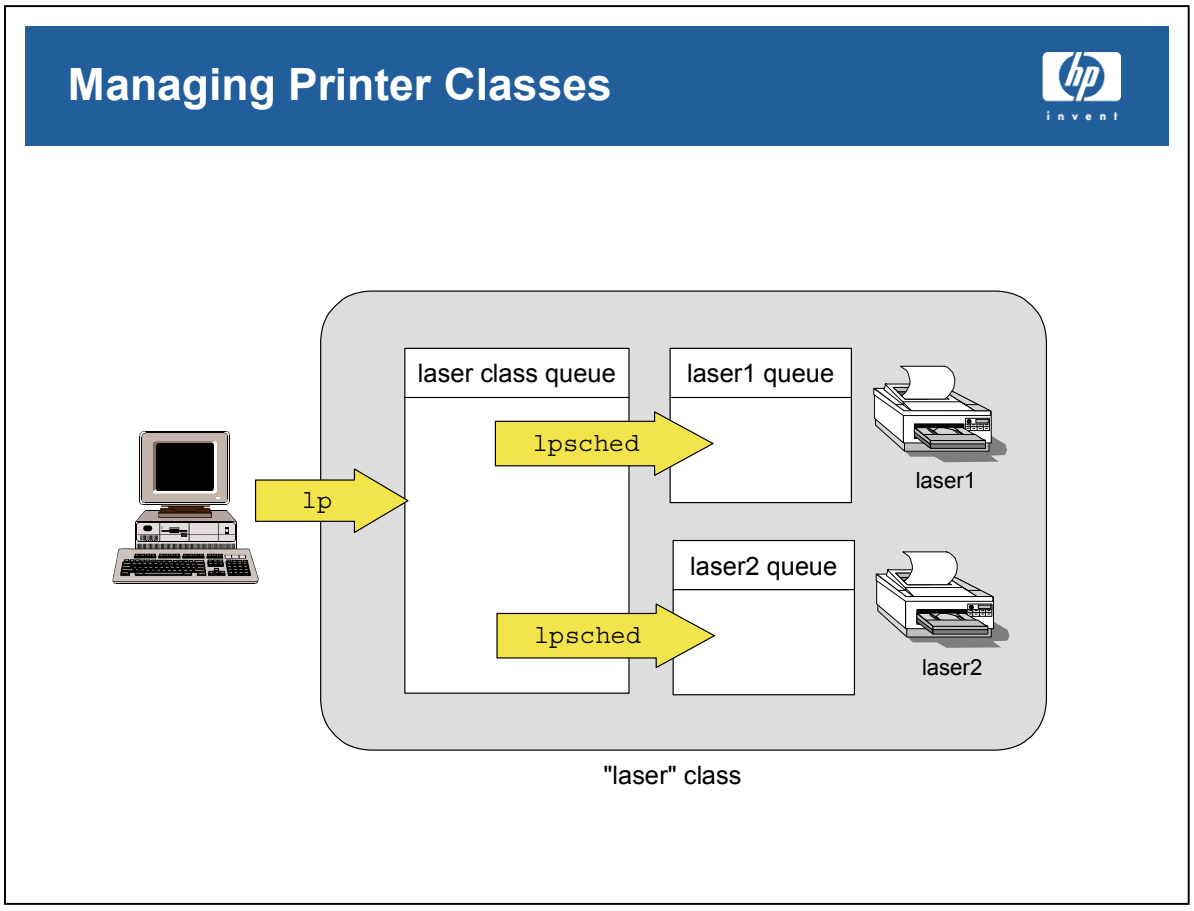
© 2004 Hewlett-Packard Development Company, L.P.

Example:

```
# lpstat -t
scheduler is running
system default destination: lj
device for lj: /dev/null
  remote to: lj_id on hpsfecid
device for sparky: /dev/c1t0d0_lp
sparky accepting requests since Dec 19 08:18
lj accepting requests since Dec 19 08:54
printer sparky disabled since Dec 19 10:47 -
        reason unknown
        fence priority : 0
printer lj is idle.  enabled since Dec 19 08:54
        fence priority : 0
sparky-18           root           priority 0  Dec 19 10:47
        hosts                                   37410 bytes
printer queue for lj
no entries
```


© 2004 Hewlett-Packard Development Company, L.P.

14–9. SLIDE: Managing Printer Classes



Student Notes

When several similar model printers are located in a single physical location, it may make sense to configure those printers as a printer class, with a common class queue. As requests are sent to the class's queue, the **`lpsched`** daemon automatically forwards each request to the class member that currently has the fewest pending requests. Using this configuration, users can send their requests to the class queue to ensure that their requests print as soon as possible, on the printer with the shortest pending queue.

Administering a Printer Class

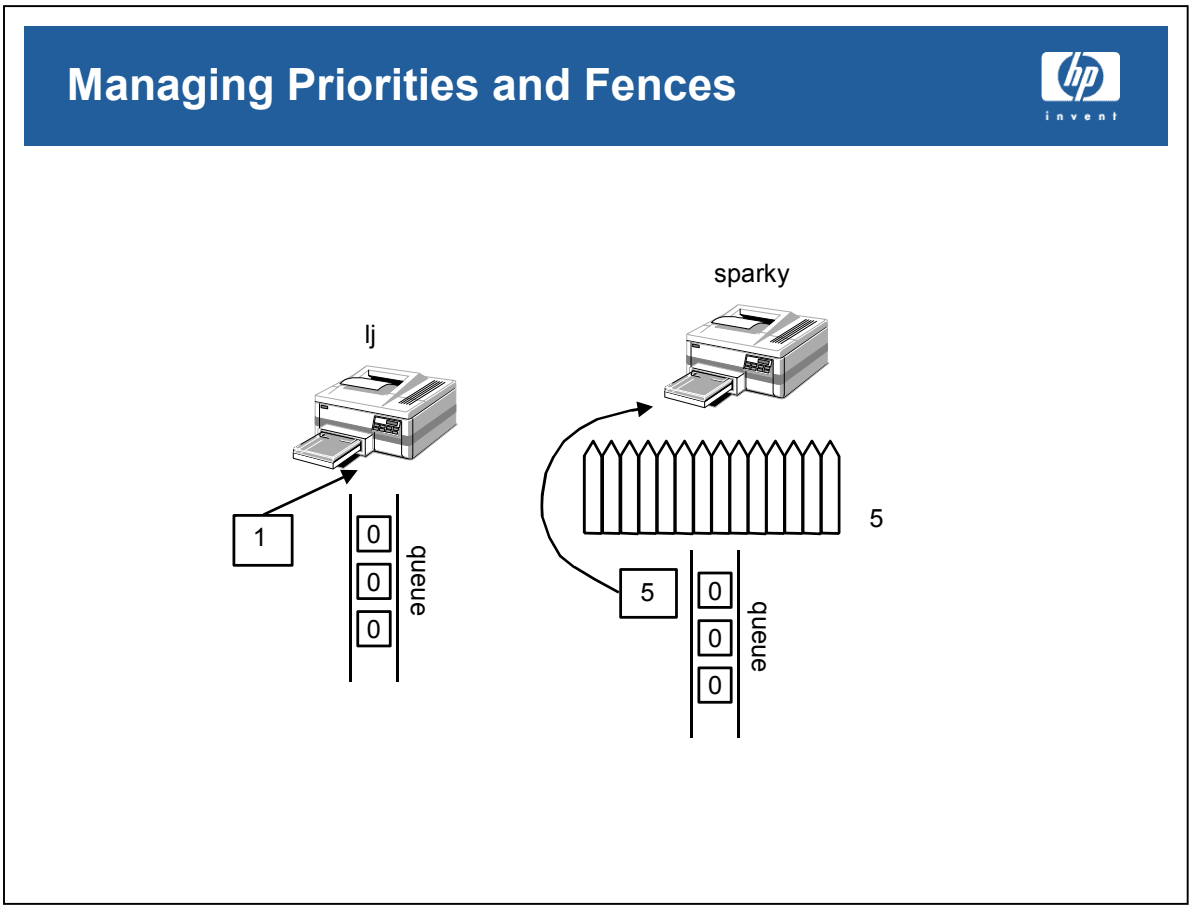
Printer classes are most easily managed from the command line:

```
# lpshut                        Shutdown the scheduler before modifying classes
# lpadmin -plaser1 -claser      Create a new class
# lpadmin -plaser2 -claser      Add a printer to an existing class
# lpsched                       Restart the scheduler
# lp -dlaser myfile             Submit a request to a printer class
```

The **`cancel`**, **`lpstat`**, **`disable`**, **`enable`**, **`accept`**, **`reject`**, and **`lpmove`** commands can be used to manage printer classes just as they are used to manage individual printer queues.

© 2004 Hewlett-Packard Development Company, L.P.

14–10. SLIDE: Managing Priorities and Fences



Student Notes

Priorities are set both for printers and print jobs. The printer priority is set during printer setup, either with SAM or with the **`lpadmin`** command. The printer priority can be changed later. If a priority is not specified when issuing the **`lp(1)`** command, the request will be sent with the printers default priority. Priorities range from 0 to 7, with 0 being the lowest priority.

All print requests are queued in their destination spool directory strictly in priority order, that is, the job with the highest priority prints first. If two jobs have the same priority they are printed in FIFO (first-in-first-out) order.

In the example shown on the slide, the default priority for **`lj`** is 0. All **`lp`** requests made without the **`-p`** ***`priority`*** option are queued with priority 0. The request made by issuing the command: **`lp -p1 myfile`** will jump to the front of the queue. Any user can use this option.

© 2004 Hewlett-Packard Development Company, L.P.

Fence Priority

A printer's fence level defines the minimum required priority for the spooled file to be printed and is configured with SAM or the **lpfence** command. A print request sent with a priority lower than the printer's fence level will sit in the printer's queue until the print job's priority is raised or the fence level is lowered.

In the example on the slide, the printer **sparky** has the fence level set at 5. The requests in the queue with priority 0 will not print, even if the printer is not busy. A new request made with a priority of 5 meets the fence requirement and will print.

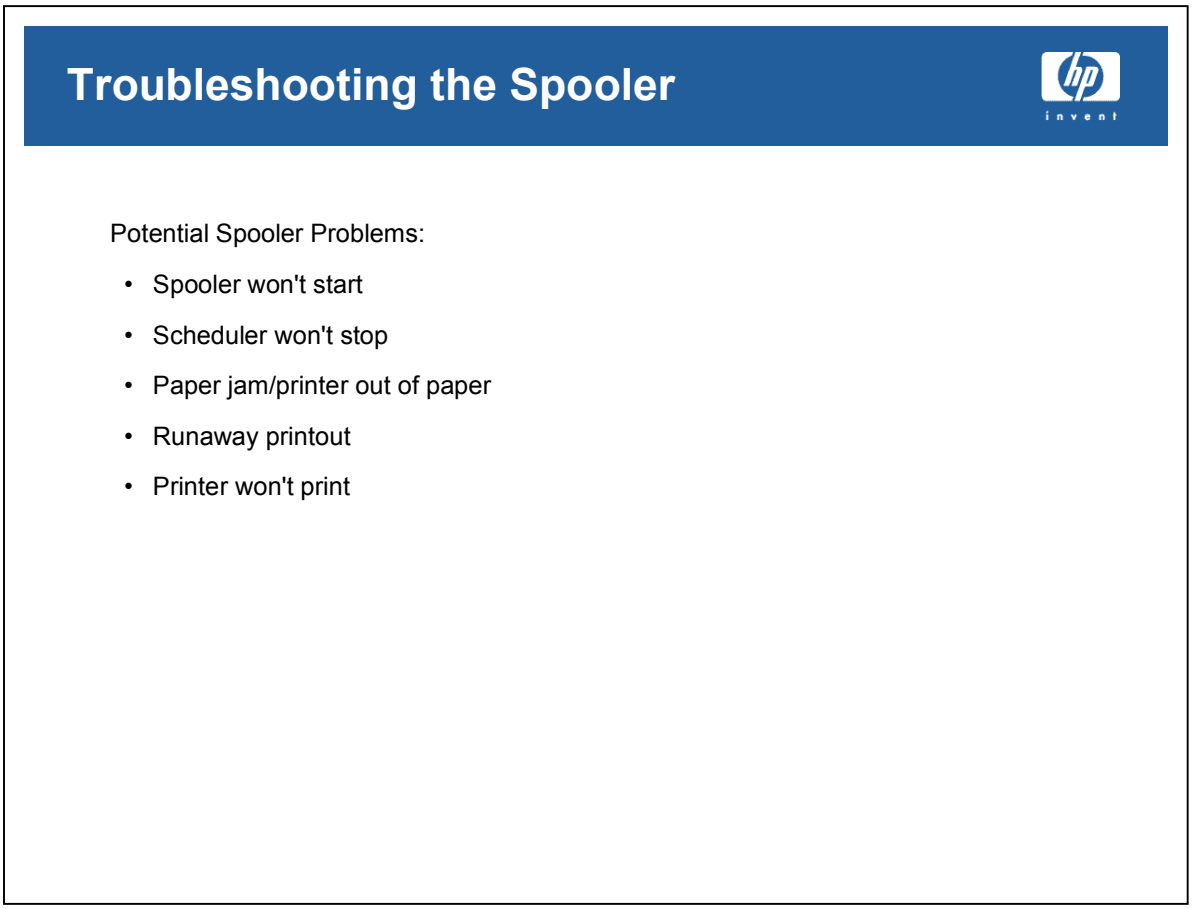
Examples

```
# lpshut                    Shutdown the scheduler
# lpfence sparky 5          Set the fence level
# lpsched                   Restart the scheduler

# lp -p5 -dsparky myfile    Submit a job with priority 5
# lpalt sparky-1 -p5        Change an existing job's priority

# lpshut                    Shut down the scheduler
# lpfence sparky 0          Remove the fence
# lpsched                   Restart the scheduler
```

The **lpalt** command can be issued by ordinary users. **lpfence** is restricted.

© 2004 Hewlett-Packard Development Company, L.P.

14–11. SLIDE: Troubleshooting the Spooler



Student Notes

Scheduler

If the scheduler won't start up when you execute the **`lpsched`** command, check to see if the **`SCHEDLOCK`** file exists in the **`/var/spool/lp`** directory. If it exists, remove it with this command:

```
# rm -f /var/spool/lp/SCHEDLOCK
```

Then try again to start the scheduler with **`lpsched`**.

If the scheduler won't stop using **`lpshut:`**

```
# kill -15 lpsched_pid
```


© 2004 Hewlett-Packard Development Company, L.P.

Paper Jam/Paper Out

When a paper jam occurs or the paper runs out, you may wish to either print the output again from the beginning, or restart printing from where it stopped.

Restart from the Beginning

1. Put printer off-line.
2. **`disable`** the printer.
3. Clear the jam or load paper.
4. Put printer on line.
5. **`enable`** the printer.

   Printing begins from the beginning.

Restart from Stopping Point

1. Clear the jam or load paper.
2. Put printer on line.
3. You may need to **`enable`** the printer.

Runaway Printout

1. Determine the request-id of the output.

   `# lpstat -u`

2. Cancel the request.

   `# cancel` *`request-id`*

© 2004 Hewlett-Packard Development Company, L.P.

Printer Won't Print

1. Move its print requests to another printer.

```
# lpshut                                              Shut down the LP scheduler
# lpmove source_printer_destination_printer           Move the print jobs
# lpsched                                             Restart the LP scheduler
```

2. Ask questions.

— Do other printers print?
— Has this printer printed before?
— Were there any error messages?
— Can other users print on this printer?

3. Take action based on the answers to questions.

If the problem affects only one printer, check physical connections (power, cables, etc). Make sure the printer is on line. Use **`lpstat -s`** to make sure the printer is correctly defined in the spooling system. Is the correct device file associated with the printer? Try

```
# sleep 200 < /dev/ttynpn &
```

This opens the port.

```
# stty 9600 CS8 -istrip -parenb opost onlcr < /dev/ttynpn
```

This configures the printer.

```
# cat /etc/motd > /dev/ttynpn
```

If the problem affects all the printers on the system, make sure the scheduler is running

```
# lpstat –r
```

If other users can print to this printer, check the printer's priority fence. If your print request has a lower priority than its printer's priority fence, it will not print. Also, make sure the user **`lp`** has permission to access the file you are trying to print.

SAM's Help with Spooling Problems

SAM has the ability to save and restore the actual spooling system configuration. This can be very helpful if someone (e.g. **`root`**?) has removed a job by deleting the jobs file in **`/var/spool/lp/request`**. Since the spooler state is kept in a few more (binary) files, you won't be able to restore normal spooling operation except by going through every file and directory and delete file contents and directory files. This is most time consuming.

You should always save the current spooler state with SAM after adding or deleting a printer.

© 2004 Hewlett-Packard Development Company, L.P.

14–12. LAB: Hands-On Adding Printers

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. If your system has a locally attached printer available, add that printer to your local spooler using SAM. If your system does not have a locally attached printer, run the **`/labs/fakeprinter.sh`** script. This script creates a terminal window at the bottom right of your CDE desktop that may be configured as a pseudo-printer.

   From the SAM functional area launcher, choose **`Printers and Plotters --> LP Spooler --> Printers and Plotters`**. Then, select **`Actions --> Add Local Printer --> With Non-Standard Device File`**. When prompted for the device filename, use "**`/dev/fakeprinter`**". When prompted for the Model Interface Script to use, choose "**`dumb`**". Configure your fake printer as your default printer.

2. List the status of the LP scheduler. Is the scheduler running? What is your system default destination? Is your printer enabled? Is your printer accepting requests?

3. Try to print to your printer. Issue the command:

   `#banner success|lp -d`*`your_destination`*

   Try to print to the destination **`class1`**.

© 2004 Hewlett-Packard Development Company, L.P.

4. Set the fence priority on your printer to a priority higher than your default priority. Issue the **lp** command without the **-p** option. Did the request print? Display the status of the scheduler. Do you see your request on the queue? Raise the priority on your request to make it print.

5. Configure your neighbor's local printer as a remote printer for your machine. Print a copy of **/etc/passwd** on your new remote printer and see what happens.

6. Use SAM to remove your printer and exit SAM when finished.

7. If the equipment is available in the classroom, configure a network printer.

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 15 — Accessing the System Console via the Management Processor

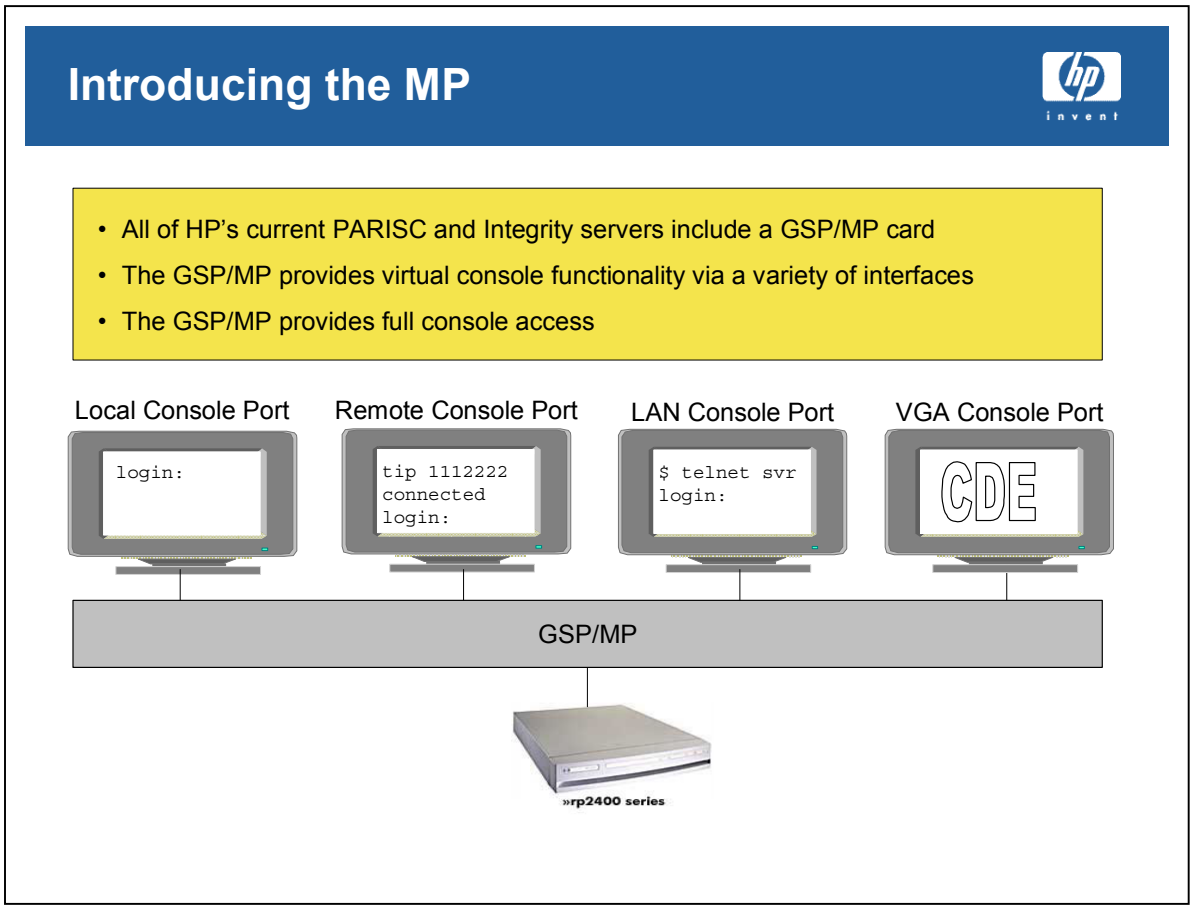
Objectives

Upon completion of this module, you will be able to do the following:

- Describe the console connectivity options for HP-UX servers and workstations.
- Describe the purpose and features of the Secure Web Console.
- Describe the purpose of the Guardian Service Processor / Management Processor.
- Connect and configure the MP serial port.
- Connect and configure the MP LAN console port.
- Configure MP user accounts and access levels.
- Navigate the MP menu interface.
- Navigate the MP help menus.
- Access Virtual Front Panel (VFP) interfaces via the MP.
- Access Console and Chassis Logs via the MP.
- Access nPar partition consoles via the MP.
- Access vPar partition consoles via the MP.
- Reset, TOC, power-up, and power-down systems via the MP.

© 2004 Hewlett-Packard Company

15–1. SLIDE: Introducing the MP



Student Notes

One of the most important hardware components you will connect to your HP-UX system is the system console. The system console is the terminal used by the administrator during system shutdown and startup, and is the only terminal on which some critical administrative tasks may be performed. HP supports several different console options.

Using the Local Serial Console Port

Older HP-UX servers included a serial console port on the server core I/O card. Most HP-UX workstations provide a core I/O serial console port, too. In either case, simply run an RS-232 cable from the console/serial port to an HP ASCII terminal, and power-on the terminal!

Newer HP-UX servers provide serial console connectivity via a serial port on the Guardian Service or Management Processor card, which is described later in the chapter.

The default "out of the box" parity, baud rate, and other terminal configuration settings on HP terminals should be compatible with your HP-UX server without modification. If you wish to use a non-HP serial terminal as your system console, see the *Configuring HP-UX for Peripherals* manual on **`http://www.docs.hp.com`** for more information.

© 2004 Hewlett-Packard Company

The local console port is by far the easiest console solution to configure, and is the preferred console solution if you have easy physical access to your server.

Using a Graphics Console

If you wish to run the SAM GUI, or other graphics-based administrative utilities on the system console, then you might choose to use a graphical, rather than serial, console.

All HP-UX workstations include a VGA port on the core I/O board. Add-on graphics console support is available for some servers, too.

Since different graphics monitors support different resolutions and refresh rates, your graphics adapter must be configured to match the resolution/frequency parameters for your graphics monitor. The procedure required to make this adjustment varies somewhat from model to model; see your system and monitor owner manuals for more information.

Using the Remote Console Port

If you have a server, it may also be possible to gain console level access to your system via a "remote access/support modem". Enabling the remote support modem makes it possible to gain console level access to your system from a remote location via a modem.

Some models have a built-in internal modem that may be used for this purpose, while others have a serial port that may be used to connect an external modem for remote console access. After physically connecting the remote support modem, the modem must be configured, enabled, and password protected. The procedure required to do this varies somewhat from model to model. See your system owner's guide for more information.

Using the LAN Console / Secure Web Console Port

All of HP's current server models now offer a LAN Console Port. The LAN console port has an IP network address that is independent of the server's standard LAN interface. Any authorized administrator that has LAN connectivity to the server's console LAN port can **`telnet`** to that port's IP address and obtain console level access. The **`telnet`** server functionality is built into the console LAN console interface firmware; thus, the administrator can **`telnet`** into the console port no matter what state the server is in.

This LAN console port is an ideal console connectivity solution for administrators in data center environments with multiple servers. The administrator can configure one central management PC or workstation, then simply **`telnet`** to the LAN console port of any server in the data center.

Newer servers' LAN console ports also support HTTPS access via a web-server built into the LAN console firmware. This makes it possible to access your server console over a secure SSL connection from any web browser on your network. This feature, too, is available even when the server is in a power-down state.

Controlling Console Access via the GSP/MP

All of the current HP-UX server models provide console access control via a Guardian Service Processor (GSP) or Management Processor (MP) included in the system chassis. The GSP/MP board includes:

© 2004 Hewlett-Packard Company

- An RS232 port, which can be used to connect an ASCII terminal console, remote support modem, or a UPS unit,

- A LAN console port, for remote **`telnet`** and/or HTTPS access via a private management LAN, and on some models,

- A VGA port on the MP interface card which may be used to connect a VGA graphic monitor.

After connecting to the MP via any one of these mechanisms, the administrator is prompted for a special MP login name and password before console access is provided. If two administrators access the ports simultaneously, every line of console input and output will be echoed to both console sessions.

From the MP, it is possible to shutdown, reboot, or install the system, view the system's "Virtual Front Panel", or perform a variety of MP administrative tasks. The MP even has an independent power supply, so you can retain console connectivity while power-cycling your machine.

The MP menus and commands will be described in detail later in this chapter.

NOTE:	Earlier servers included a Guardian Service Processor (GSP) card. Newer servers include a Management Processor (MP) card instead. The MP/GSP menu structure varies somewhat depending on your interface card and GSP/MP firmware revision, but the core features are similar across platforms. Most screenshots in this chapter were captured on an rp7405.

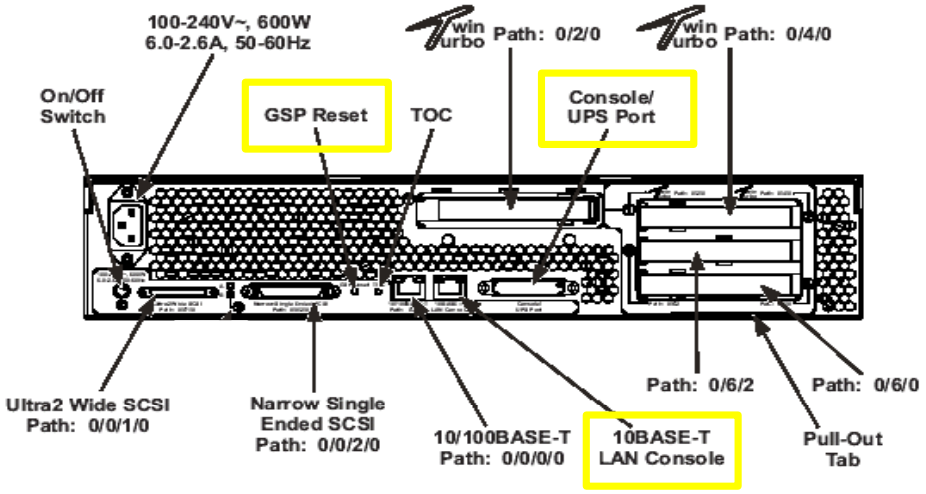
NOTE:	HP's V-Class servers use an entirely different console solution. V-Class servers require a dedicated workstation, connected via a private network, to serve as the server console. For more information on this configuration, please see the *HP9000 V-Class Operator's Guide* on **`http://docs.hp.com`**.

© 2004 Hewlett-Packard Company

15–2. SLIDE: Viewing the MP Backplane

Viewing the MP Backplane

- The MP is accessible via several ports on the system backplane
- The MP offers a LAN console port, and a 25-pin serial port
- The MP reset button returns the MP to its factory default configuration



Student Notes

On most current server models, the MP is located on the server backplane. Although the physical arrangement of the ports varies somewhat from model to model, at a minimum, you should find the following:

- A 25-pin serial port, which may be connected via a "W" cable to a serial VT100 or terminal or terminal emulator. This will be described in greater detail on the next slide. Some servers have a separate DB9 serial port that may also be used to connect a serial terminal if you don't have a "W" cable and don't intend to connect a remote support modem or UPS unit.

- An MP Ethernet LAN console interface, which provides `telnet` access to the MP interface via a `telnet` server embedded in the MP firmware. You can connect this port via Cat5 twisted pair cable to your datacenter network. Note that this LAN interface is independent from the core I/O LAN interface that clients would typically use to access your application server daemons.

- An MP hard reset button. If you misconfigure the MP LAN or user accounts, the reset button may be used to reset the MP to its factory default configuration. Some servers have an MP soft reset, too, which simply power-cycles the MP but maintains the

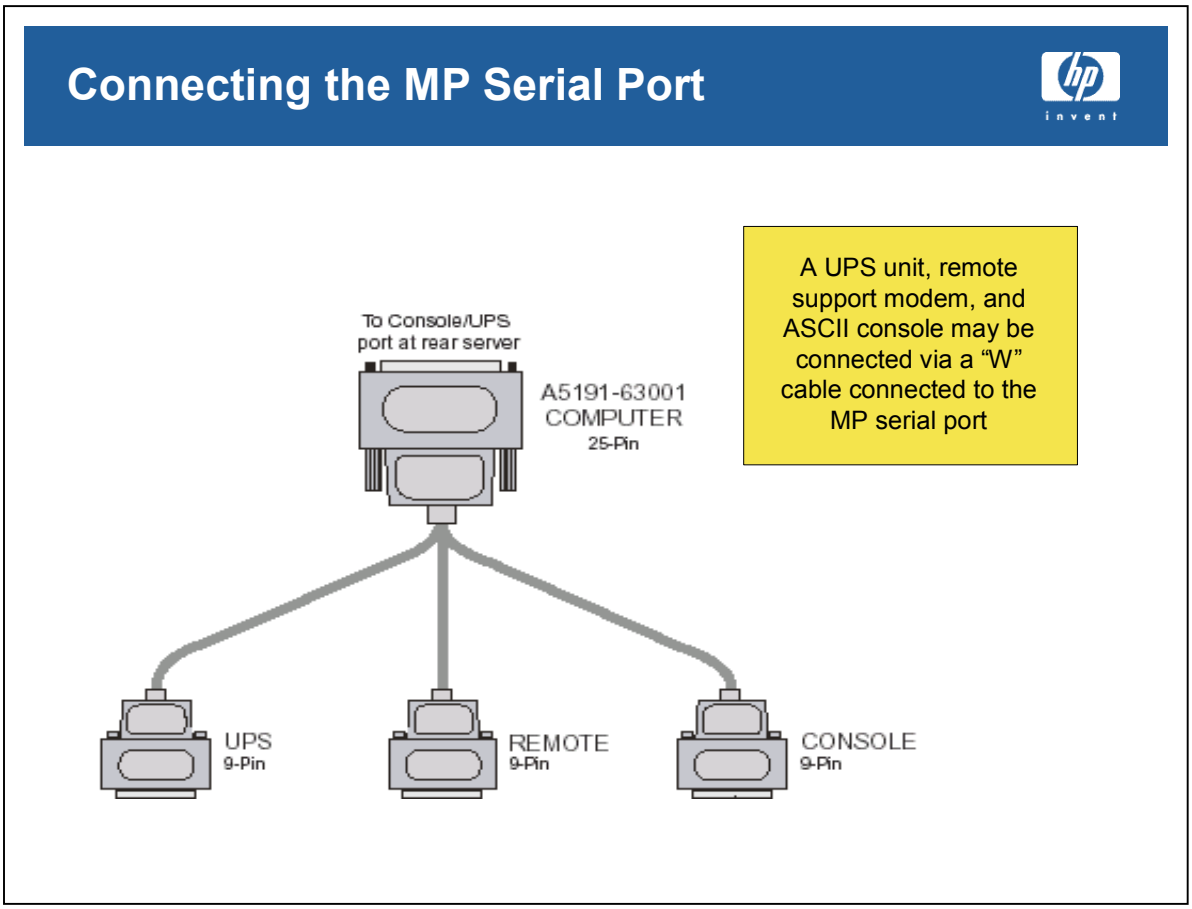
© 2004 Hewlett-Packard Company

configuration. Neither reset button should have any impact on the server itself; the server remains functional and clients remain connected throughout the MP reset process.

Some servers may have a MP VGA port in addition to the ports described above.

© 2004 Hewlett-Packard Company

15–3. SLIDE: Connecting the MP Serial Port



Student Notes

The MP interface card has a single 25-pin female serial port.

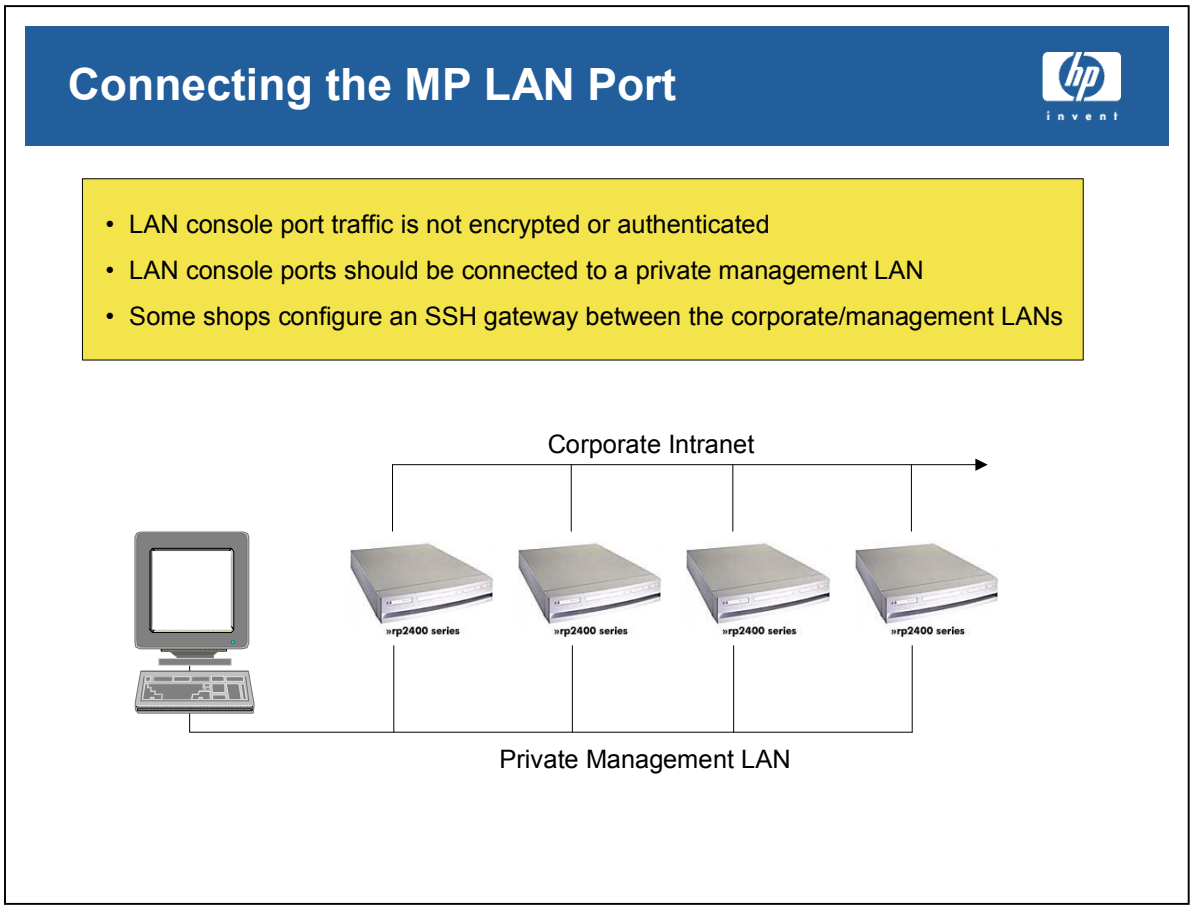
If you wish to connect a local serial console to your server, connect a "W-cable" (HP PN#5191-63001) to the 25-pin port on the MP card. Then connect the W-cable's 9-pin "console" connector to the 9-pin serial port on your PC. If you choose to connect an HP 700 series ASCII terminal, you will need HP's 25542G 9-pin to 25-pin RS-232 adapter. In either case, be sure to set your terminal to vt100 emulation mode

The other two 9-pin connectors on the W-cable may be used to connect a UPS unit or an external remote support modem.

After you connect a serial console, additional MP configuration may be required. This will be discussed later in the chapter.

© 2004 Hewlett-Packard Company

15–4. SLIDE: Connecting the MP LAN Port



Student Notes

Although the MP may be accessed via a remote support modem or a local serial console, many administrators choose to access the MP via the LAN console port.

If you choose to enable **`telnet`** access to the LAN console port, note that **`telnet`** traffic is neither encrypted nor authenticated. Thus, if a hacker were able to gain access to a LAN port on your LAN console port's network, she or he would very likely be able to gain console access on your system.

For this reason, the LAN console port should NOT be connected to the Internet or your company's intranet. For improved security, your datacenter servers' LAN console ports should be connected to a secure, private management network that is only accessible within your datacenter.

Some administrators configure an SSH (Secure Shell) gateway server with connectivity to both the corporate intranet and the datacenter private management network. This allows administrators to access the gateway via the secure SSH protocol, then **`telnet`** from the gateway to any LAN console port in the datacenter. SSH configuration is described in HP Education's HP-UX Security course, course number H3541S.

© 2004 Hewlett-Packard Company

15–5. SLIDE: MP User Accounts

MP User Accounts

- Users must provide a username and password to access the MP
- The “Admin” user (password “Admin”) is predefined
- Up to nineteen MP users may be defined by the MP administrator
- A user’s access level determines which MP features and partitions s/he can access
- Factory default MP password: Admin/Admin

Access Level	Commands Available	Partitions Available
Administrator	All	All
Operator	Some	All
Single-partition user	Some	Some

Student Notes

In order to access the MP, users must enter a valid MP username and password. Up to twenty MP users may be defined on a system. The first MP user, by default, is the administrator. The MP administrator has full access to all MP menus and commands and all partitions. The username for this account is initially set to “Admin” with password “Admin”. For security reasons, this should be changed as soon as possible.

The Admin user (or any other user with Administrator access rights) may configure the nineteen remaining MP users as desired. Each user is assigned a username, password, and access level. The user’s access level determines which GSP features and node partitions the user can access. Three access levels are available on the MP:

Administrator Users: Administrator users have access to all MP commands, and all partitions on the system.

Operator Users: Operators have access to all commands necessary to access and manage all partitions, but don’t have the ability to create additional MP users or modify the configuration of the MP itself.

Single-Partition Users: Single-partition users have access to a limited number of MP commands that are required to access, boot, and reboot a specified

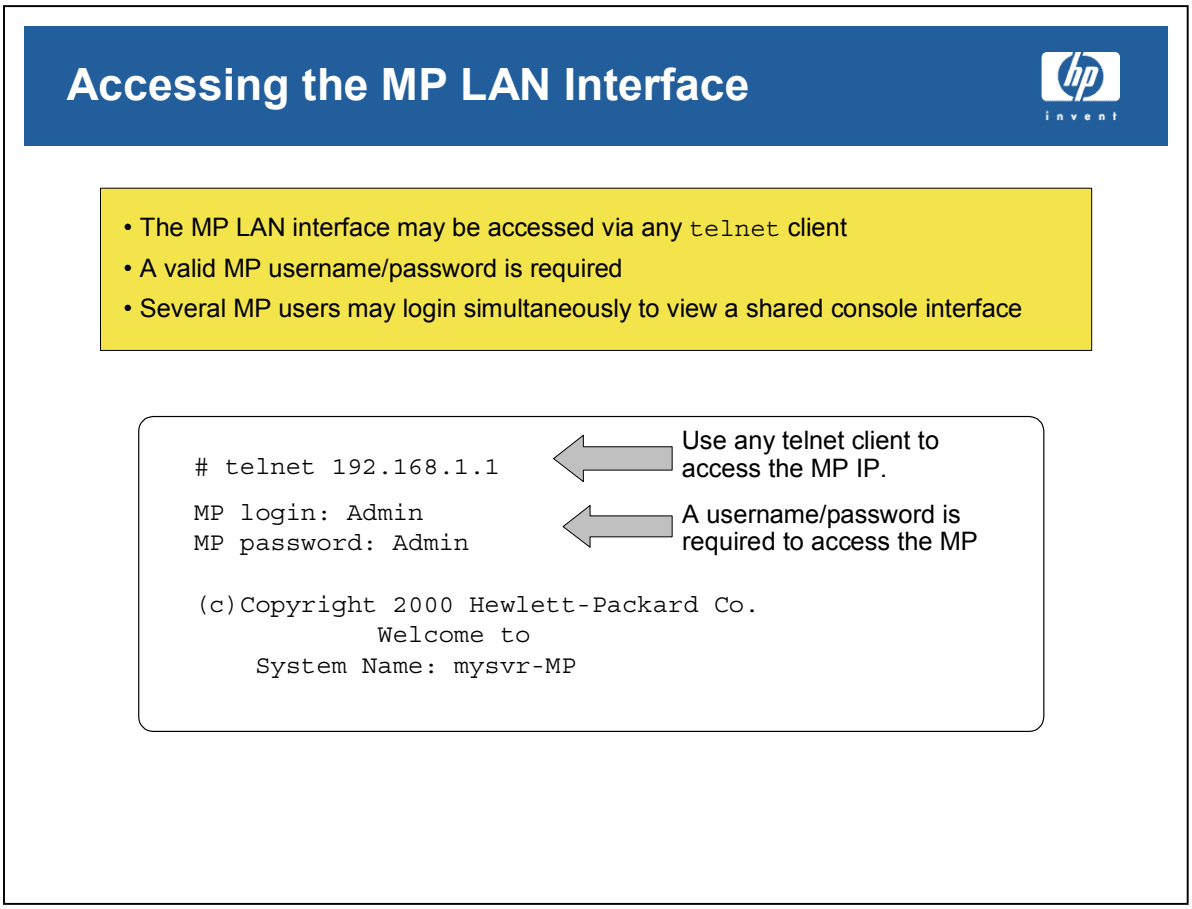
© 2004 Hewlett-Packard Company

partition. The single-partition access level is only available on systems that support HP's nPar functionality.

Note that MP user accounts have no relationship to the user accounts defined in the HP-UX **`/etc/passwd`** file. The HP-UX **`/etc/passwd`** file is stored on disk and determines who can login on a partition running HP-UX. The MP password database is stored in the MP's non-volatile RAM and determines which users can access the MP menu interface.

© 2004 Hewlett-Packard Company

15–6. SLIDE: Accessing the MP LAN Interface



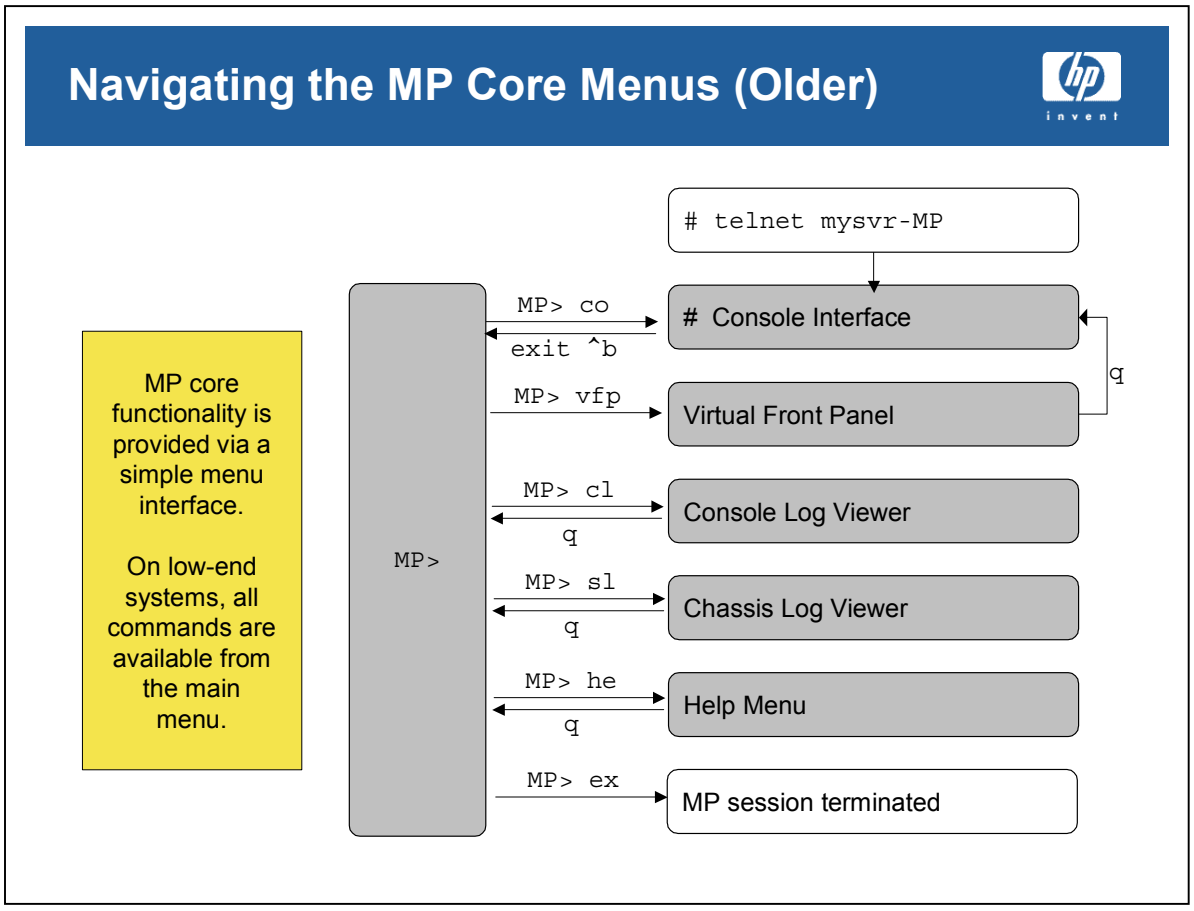
Student Notes

The first time you access a system via the MP, it will likely be necessary to connect via a local console so you can configure an appropriate MP IP address and enable the LAN console functionality. Login using predefined username “Admin” and password “Admin”, then refer to the LAN Configuration slide later in this chapter to learn how to change the MP LAN IP, and the User Configuration slide to learn how to change the Admin password.

After the initial configuration, many administrators prefer to access the MP remotely via the MP LAN interface. Simply use your favorite **`telnet`** client application to connect to the MP LAN IP address, and enter any valid MP username and password.

© 2004 Hewlett-Packard Company

15–7. SLIDE: Navigating the MP Core Menus (Older)



Student Notes

The MP provides a menu-based interface with several dozen commands for accessing and managing your system. This slide introduces the general MP menu structure used on some older systems. The next slide describes the somewhat different menu structure that is used on many newer systems.

On these older systems, when you first login on the MP, you will be immediately placed in a console interface in read-only mode. Recall that multiple users can connect to the MP console simultaneously. In read-only mode, you should be able to view the commands being typed by other MP users, but you won't be able to execute any commands yourself. In order to take control of the console, press **^e** followed by **cf** and **[return]**, and login on the console using any HP-UX user name and password. This console interface may be used to shutdown, reboot, or perform any other administrative chores that would normally be done on the system console.

If you wish to execute other MP commands, logout of your HP-UX session by typing **exit**, and press **^b** to return to the **MP>** prompt.

Several core commands are available at the **MP>** prompt.

© 2004 Hewlett-Packard Company

- The **co** command returns to the console interface. To return to the main MP menu, exit out of your login session and press **^b**.

- The **vfp** command provides access to a "Virtual Front Panel". The VFP may be used to determine the system's current status, particularly in the early stages of the boot process when the system is running hardware self-checks. Press **q** to exit the VFP and access the console interface. From the console interface, you can press **^b** to return to the **MP>** prompt.

- The **cl** command displays recent console messages and commands. This is useful if you want to see which commands were recently executed, or if you want to review error or status messages that may have been sent to the console during the last shutdown or reboot. To return to the main MP menu, press **q**.

- The **sl** command displays low level chassis log messages generated by the system. These would primarily be of interest to HP Engineers. To return to the main MP menu, press **q**.

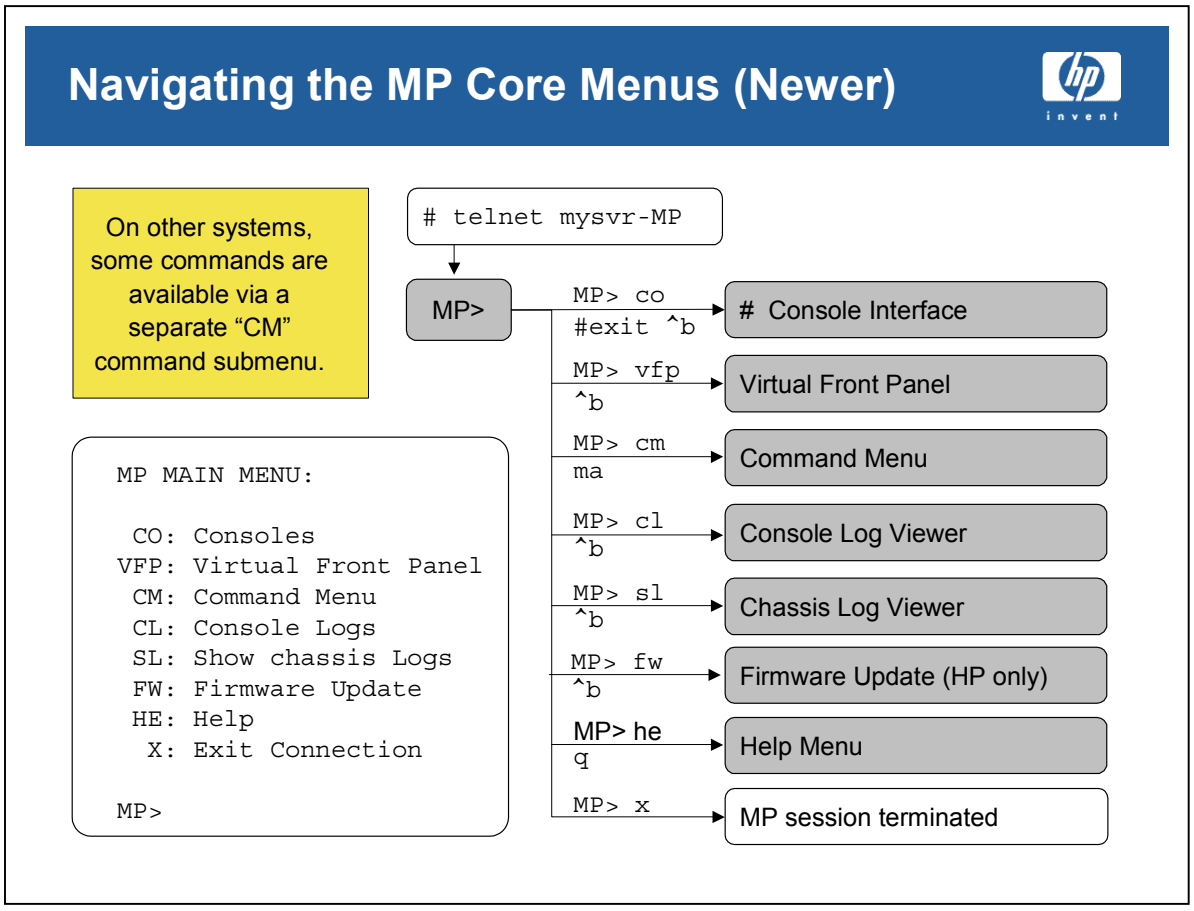
- The **he** command provides access to an interactive help menu that describes the features and commands available in the MP. To return to the main MP menu, press **q**.

- The **ex** command terminates your MP connection.

MP commands are not case-sensitive. Since the MP firmware varies somewhat from model to model, this course only covers the most common commands that are available on all models. See the MP help menus on your system for a complete list of commands on your specific system model.

© 2004 Hewlett-Packard Company

15–8. SLIDE: Navigating the MP Core Menus (Newer)



Student Notes

Newer servers offer a somewhat different MP menu structure. When you first login on the MP, you will be placed in the MP main menu rather than the system console interface. The MP main menu supports several commands.

- The **co** command provides access to an interactive console for each nPar partition on the system. These consoles may be used to shutdown, reboot, or perform other administrative chores that would normally be done on the system console. To return to the main MP menu, exit out of your login session and press **^b**.

- The **vfp** command provides access to a "Virtual Front Panel" for each nPar partition on the system. These VFPs may be used to determine a partition's current status, particularly in the early stages of the boot process when the system is running hardware self-checks. To return to the main MP menu, press **^b**.

- The **cm** command takes you to a MP command menu. The command menu may be used to configure the MP LAN interface, add or remove MP users, reset or boot a partition, and perform many other MP tasks. Later slides in the chapter will discuss some common **CM** commands in greater detail. To return to the main MP menu, type **ma**. Note that some

© 2004 Hewlett-Packard Company

servers don't have a separate **CM** menu; on these systems, all MP commands are available from the initial MP prompt.

- The **cl** command displays recent console messages and commands for each nPar partition on the system. This is useful if you want to see what commands were recently executed on an nPar's console, or if you want to review error or status messages that may have been sent to the console during the last shutdown or reboot. To return to the main MP menu, press **^q**.

- The **sl** command displays low level chassis log messages generated by the system. These would primarily be of interest to HP Engineers. To return to the main MP menu, press **^q**.

- The **fw** command provides a mechanism for updating the MP firmware. This function should only be used by HP support personnel.

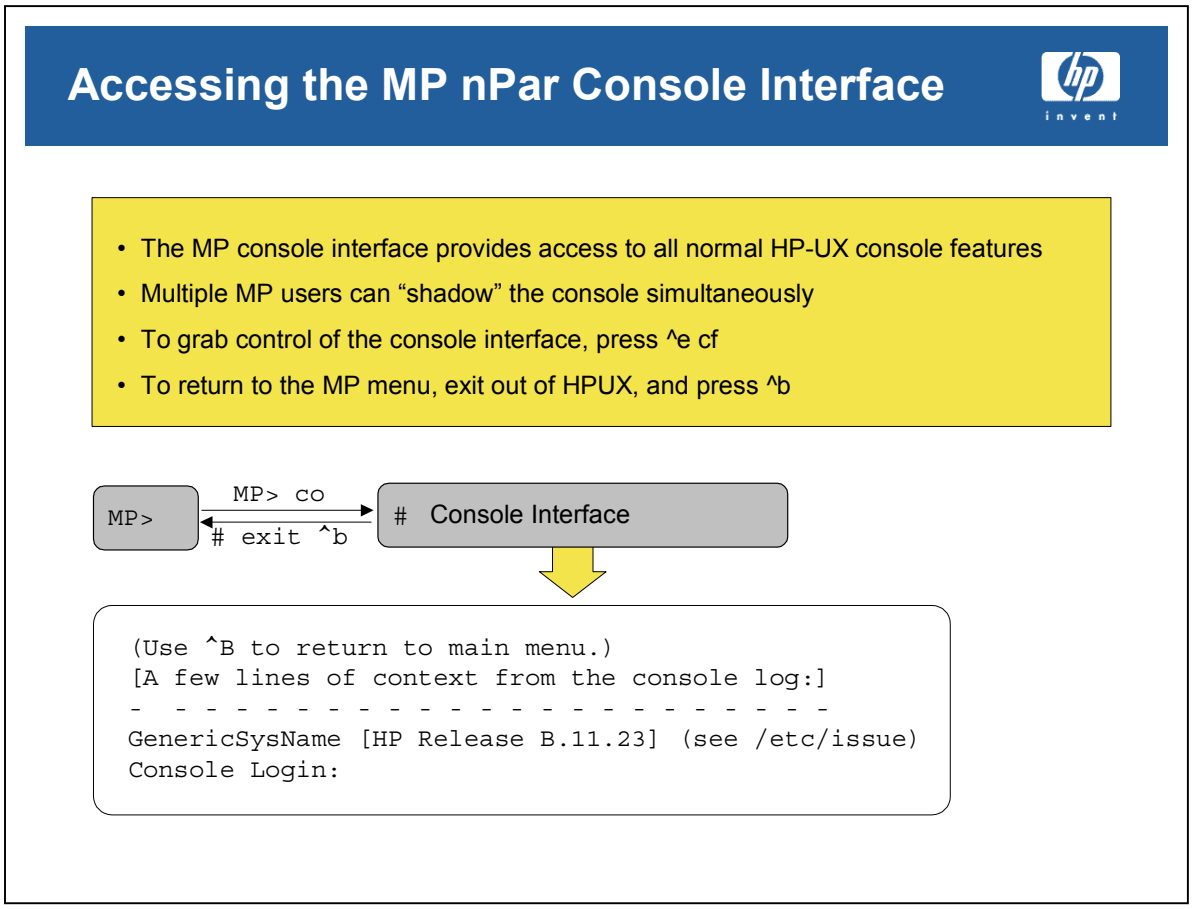
- The **he** command provides access to an interactive help menu that describes the features and commands available in the MP. To return to the main MP menu, press **^q**.

- The **x** command terminates your MP connection.

MP commands are not case-sensitive. Since the MP firmware varies somewhat from model to model, this course only covers the most common commands that are available on all models. See the MP help menus on your system for a complete list of commands on your specific system model.

© 2004 Hewlett-Packard Company

15–9. SLIDE: Accessing the MP nPar Console Interface



Student Notes

The MP console interface provides full console access. This makes it possible to remotely shutdown and reboot a system, install the OS, or even power on/off a system without missing key console messages. On partitionable systems, the MP **co** command is the only mechanism available to gain console access on nPars.

On partitionable systems, after typing the **co** command, MP operators and administrators are asked which partition they wish to access. Single-partition users, and administrators on systems that don’t support nPars, won’t be prompted for a partition number.

```
MP> co

  Partitions available:

  Part#  Name
  -----  ----
     0)  Partition 0
     1)  Partition 1
     Q)  Quit
```


© 2004 Hewlett-Packard Company

```
  Please select partition number: 0
       Connecting to Console: Partition 0

        (Use ^B to return to main menu.)

        [A few lines of context from the console log:]

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

# exit
logout root

GenericSysName [HP Release B.11.11] (see /etc/issue)
Console Login:

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

GenericSysName [HP Release B.11.11] (see /etc/issue)
Console Login:
```

Several users can access the MP functionality, and can even concurrently execute different MP commands. One might be viewing the VFP, while another views the chassis log, while yet another executes command from the **MP:CM>** prompt.

Several users can even access an nPar console interface concurrently. However, only one MP user has keyboard control within the console session of a specific nPar. Other MP users can view the commands executed within the console interface, but only one can provide keyboard input. If you join a MP console session and wish to take control of the console keyboard, press **^e**, then **cf**.

The **MP:CM> who** command displays a list of currently logged-in MP users and indicates which one has keyboard control.

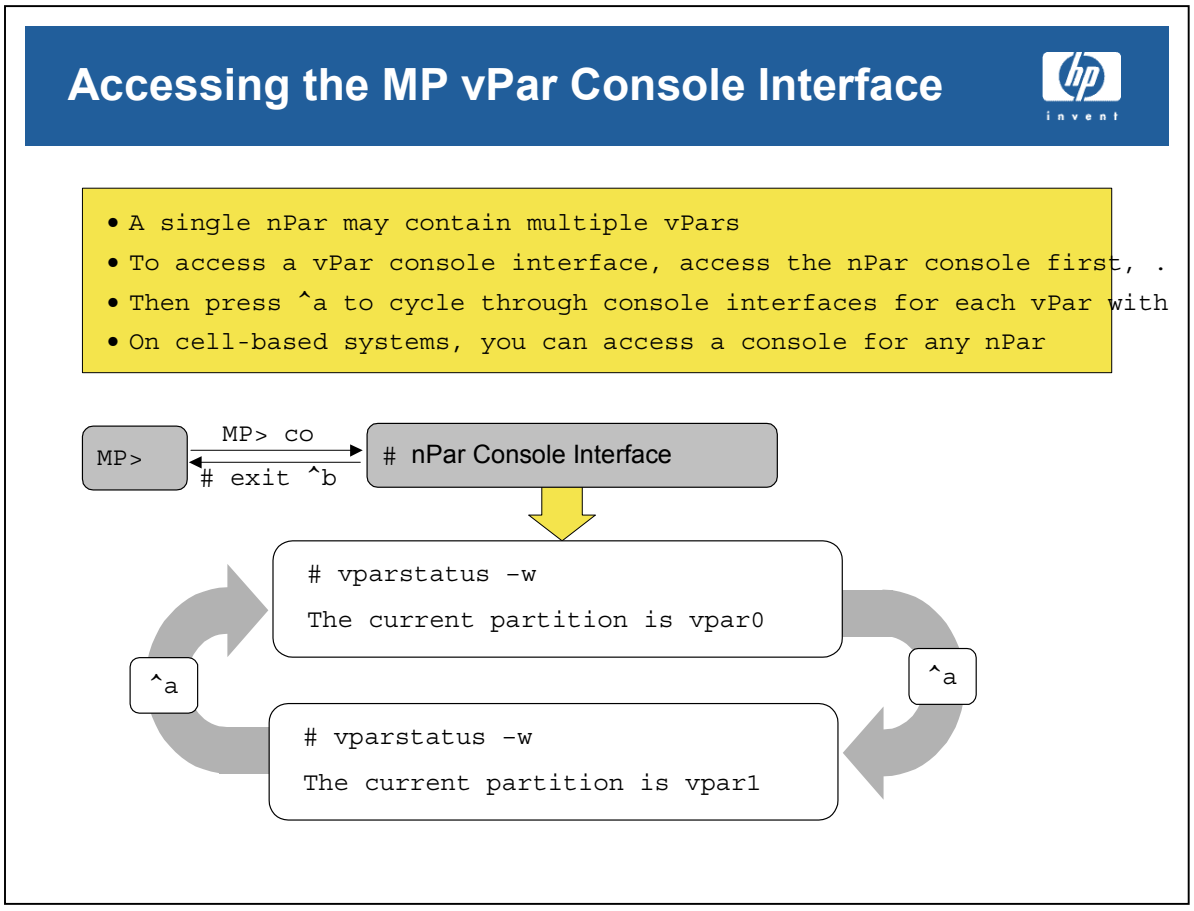
After logging in, you can determine which nPar console you are currently accessing by typing **parstatus -w**. This command only works on node-partitionable systems.

```
# parstatus -w
The local partition number is 0.
```

When you are finished with a MP console session, be sure to exit out of your HP-UX shell, then press **^b** to return to the **MP>** main menu. If you fail to terminate your HP-UX login session before returning to the **MP>** menu, the next user that accesses the MP console for that partition will be able to access the console without entering an HP-UX username and password.

© 2004 Hewlett-Packard Company

15–10. SLIDE: Accessing the MP vPar Console Interface



Student Notes

An nPar partitioned system may be subdivided into one or more node partitions. Each nPar partition has a separate console interface that is accessible via the MP. On a system running HP's vPar software, each nPar may in turn contain one or more vPars. Accessing a console interface for vPars within nPars presents some challenges.

The first step is to access the console interface for the appropriate nPar via the MP as described earlier in the chapter.

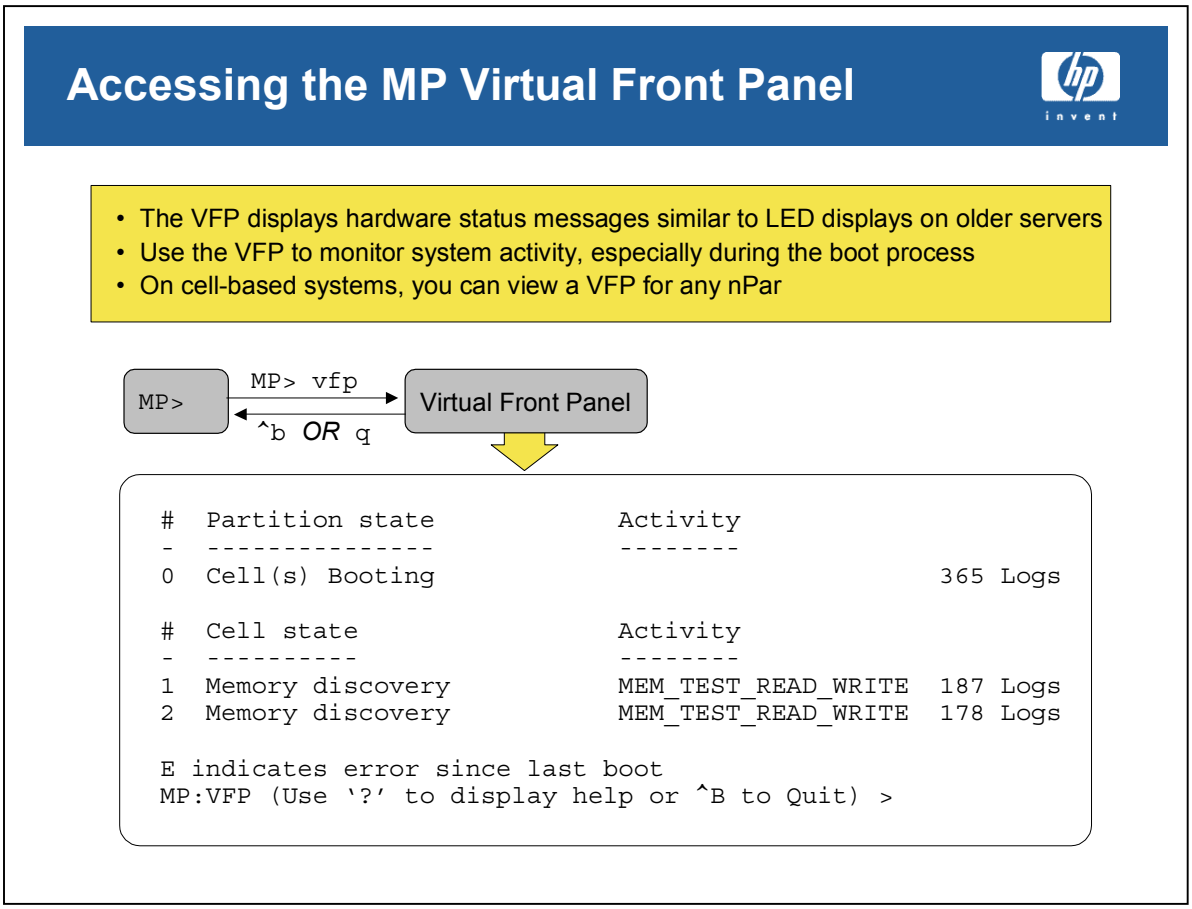
After accessing the console of the appropriate nPar, you can cycle through the console interfaces for each of the vPars within that nPar by pressing **^a**.

From HP-UX, you can determine which vPar console you are currently accessing by typing **vparstatus -w**.

```
# vparstatus -w
The current partition is vpar0
```


© 2004 Hewlett-Packard Company

15–11. SLIDE: Accessing the MP Virtual Front Panel



Student Notes

Older servers traditionally had an LED front panel on the server chassis that displayed status codes that could be used to monitor the system status and troubleshoot system problems. Since many administrators now manage servers remotely, all the newer servers have "Virtual Front Panels" (VFPs) that provide similar information remotely via the MP interface.

The format of the information reported on the VFP varies from model to model. The sample output on the slide was taken from an rx7620 nPar partition. On cell-based systems, after initially typing the **vfp** command at the **MP>** prompt, the **MP:VFP>** prompt expects the administrator to specify which partition's VFP should be displayed. In the example on the slide, the resulting output for the selected partition shows that the partition has two cell boards, and is in the process of completing a sequence of memory self-checks during the boot process.

```
MP> vfp

  Partitions available:

  Part#  Name
  -----  ----
     0)  Partition 0
```


© 2004 Hewlett-Packard Company

```
       1)  Partition 1
       Q)  Quit

     Please select partition number: 0

#  Partition state           Activity
-  ---------------           --------
0  Cell(s) Booting                                365 Logs

#  Cell state                Activity
-  ----------                --------
1  Memory discovery          MEM_TEST_READ_WRITE  187 Logs
2  Memory discovery          MEM_TEST_READ_WRITE  178 Logs

E indicates error since last boot
MP:VFP (Use '?' to display help or ^B to Quit) >
```

The **Partition state** field reports the status of the selected partition. The **Cell state** field reports the status of the cells included in the selected partition. The **Activity** field reports what the cell or partition is currently doing. The **Logs** field reports the number of chassis log entries generated by the selected partition or cell. Most chassis log messages are informational. If there's an **E** in the first column of any line, before either a partition or cell number, it means that the current partition/cell status is the result of an error. If you see an error, use the MP's console and chassis log viewers to determine what caused the problem.

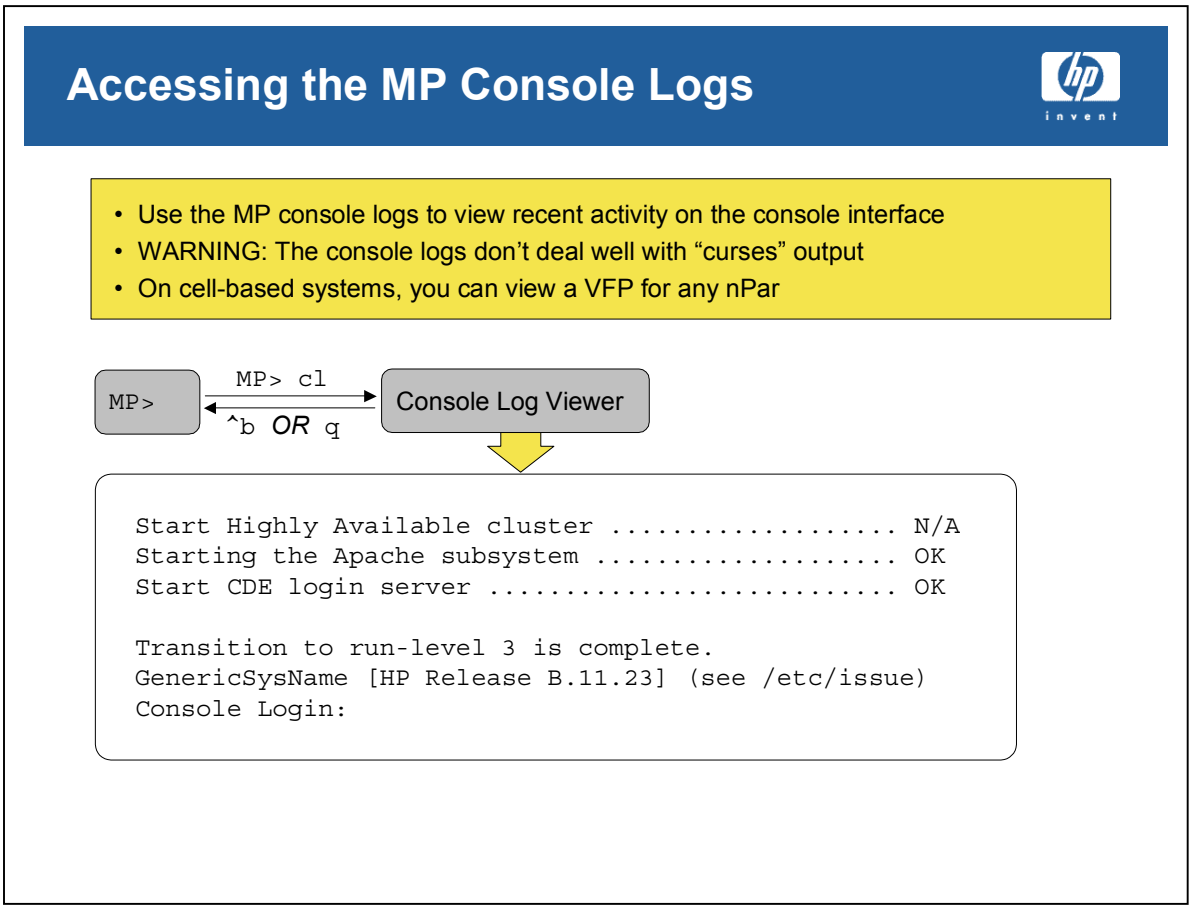
At the end of the boot process, you should see a blinking OS heartbeat asterisk in the **Partition state** field. This indicates that the **/sbin/init** deamon is running.

```
#  Partition state           Activity
-  ---------------           --------
0  OS heartbeat: *                        365 Logs

#  Cell state                Activity
-  ----------                --------
1  Cell has joined partition             187 Logs
2  Cell has joined partition             178 Logs

E indicates error since last boot
MP:VFP (Use '?' to display help or ^B to Quit) >
```

At any time, you can press **^b** or **q** to leave the VFP and return to the **MP>** main menu.

© 2004 Hewlett-Packard Company

15–12. SLIDE: Accessing the MP Console Logs



Student Notes

In order to prevent the administrator from missing critical console messages, the MP records all keystrokes and output on the system console in a 20KB console log buffer. This is particularly useful after a system reboot, as it allows you to review boot console messages after the boot process is complete.

Use the **`MP> cl`** command view the contents of the console log. On non-partitionable systems, this will display the console log immediately. On partitionable systems, the MP asks which partition's console log you wish to view.

```
MP> cl

Partition Console Logs available:
Part#  Name
-----  ----
   0)  Partition 0
   1)  Partition 1
   C)  Clear a partition's console log.
   Q)  Quit

MP:VW> 0
```


© 2004 Hewlett-Packard Company

```
Start Highly Available cluster ..................... N/A
Starting the Apache subsystem ...................... OK
Start CDE login server ............................. OK

Transition to run-level 3 is complete.
GenericSysName [HP Release B.11.23] (see /etc/issue)
Console Login:

(N)ext or <cr>, (P)revious, ^B to exit to menu
```

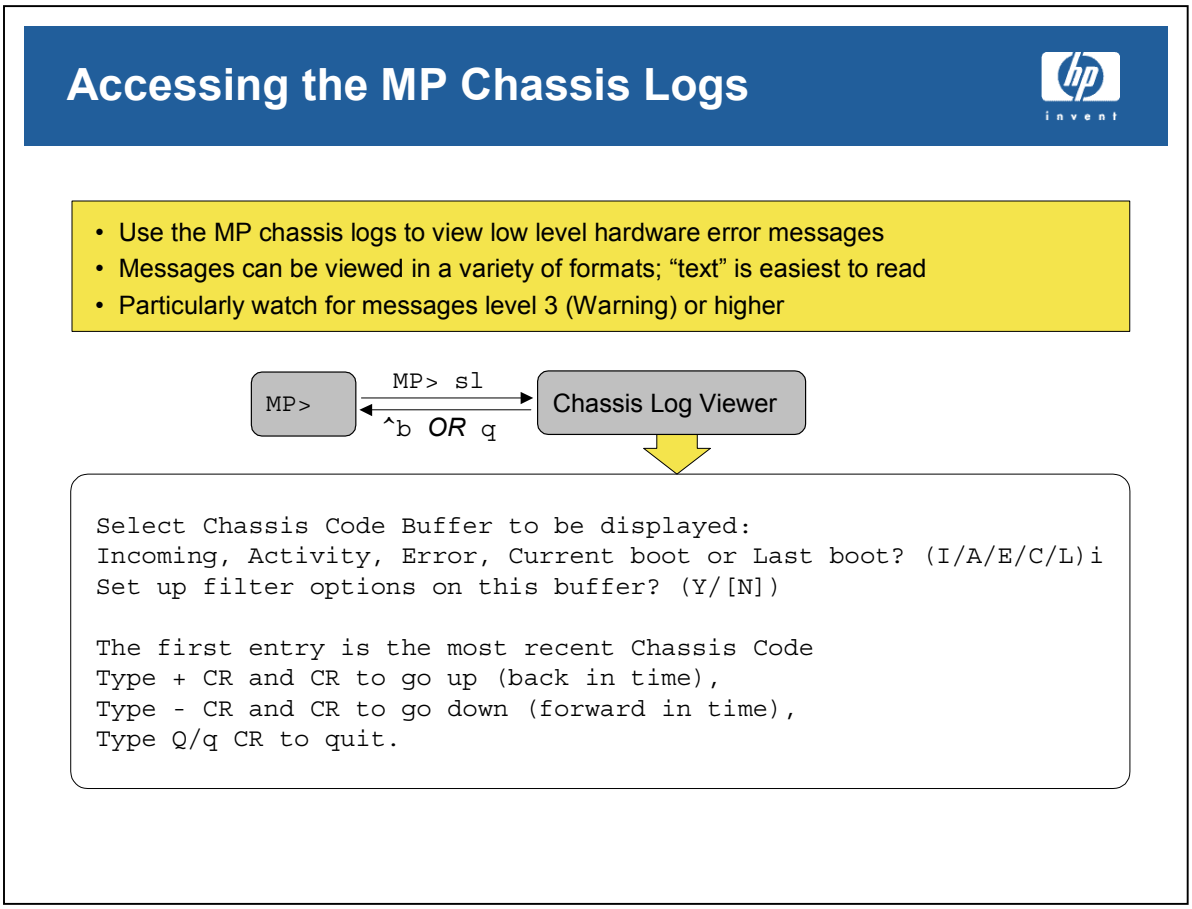
The commands required to navigate the console logs varies somewhat depending on your platform and MP firmware revision. Press **?** to view the navigation commands for your system. Here is a sample list of navigation commands supported on many of the new systems:

```
Console Log Navigation Command Help
      +       Go to next page
      -       Go to previous page
      <CR>    Continue to next or previous page
      C       Clear console log
      D       Dump the entire log
      F       Go to first page of history (oldest)
      L       Go to last page of history (most recent)
      ?       Print this help menu
      Ctrl-B  Exit command, and return to the Main Menu



MP:CL (+, -, <CR>, C, D, F, L, ? for help, or Ctrl-B to Quit){Pg 1 of
57} >
```

Use **^b** or **q** to return to the **MP>** main menu.

© 2004 Hewlett-Packard Company

15–13. SLIDE: Accessing the MP Chassis Logs



Student Notes

The **`MP> sl`** command displays system chassis logs, which may be useful when troubleshooting hardware issues. This is particularly useful when you discover that an attention light is blinking, if your system fails to boot, or if an **`E`** appears on the VFP indicating that an error occurred. If you suspect that you have a hardware problem, but can't determine the meaning of a message in the chassis log, consider calling the HP Response Center to troubleshoot the problem.

You can view past activity logs, past error logs, and live chassis logs (log messages reported in real time) in a variety of formats.

By default, the logs are reported in hexadecimal with cryptic keyword explanations. Use the **`t`** (text) command to view the logs in a somewhat more readable format.

You may also wish to filter the logs by severity level by pressing the **`a`** key. The table below describes the various alert levels (note that alert levels 4 and 6 are currently unused):

© 2004 Hewlett-Packard Company

Alert Level	Description
0	Minor Forward Progress
1	Major Forward Progress
2	Informational
3	Warning
5	Critical
7	Fatal

The system generates many log messages at levels 0-2 as part of the normal system boot process. Log messages at levels 3, 5, and 7. May deserve extra attention.

Some sample **sl** output is included below:

```
MP> sl

Chassis Logs available:

     (A)ctivity Log
     (E)rror Log
     (L)ive Chassis Logs
     (C)lear All Chassis Logs
     (Q)uit

MP:VW> e

To Select Entry:
     (<CR> or <space>) View next or previous block
     (+) View next block (forwards in time)
     (-) View previous block (backwards in time)
     (D)ump entire log for capture and analysis
     (F)irst entry
     (L)ast entry
     (J)ump to entry number
     (V)iew Mode Select
     (H)elp to repeat this menu
     ^B to exit

MP:VWR (<CR>,<sp>,+,-,D,F,L,J,V,H,^B) > v
Select Mode:
     (K)eyword
     (T)ext
     (R)aw hex

Text Viewer Mode Selected.  Please press return to continue.
MP:VWR (<CR>,<sp>,+,-,D,F,L,J,V,H,^B) >
26   PDC  0,1,0 *12 DNA_2_DNA_ERROR_ACCESSING_GROUP_A  08/26/2003
16:54:58
Log Entry 26: 08/26/2003 16:54:58
Alert Level 12: Software failure; Keyword:
DNA_2_DNA_ERROR_ACCESSING_GROUP_A
Crossbar chip 0 ; Status: 14
Logged by system firmware 4 during configuration subActivity 97
Actual Data: 0x00000079ff81fee0
0x180046c05100297e 0x00000079ff81fee0
```

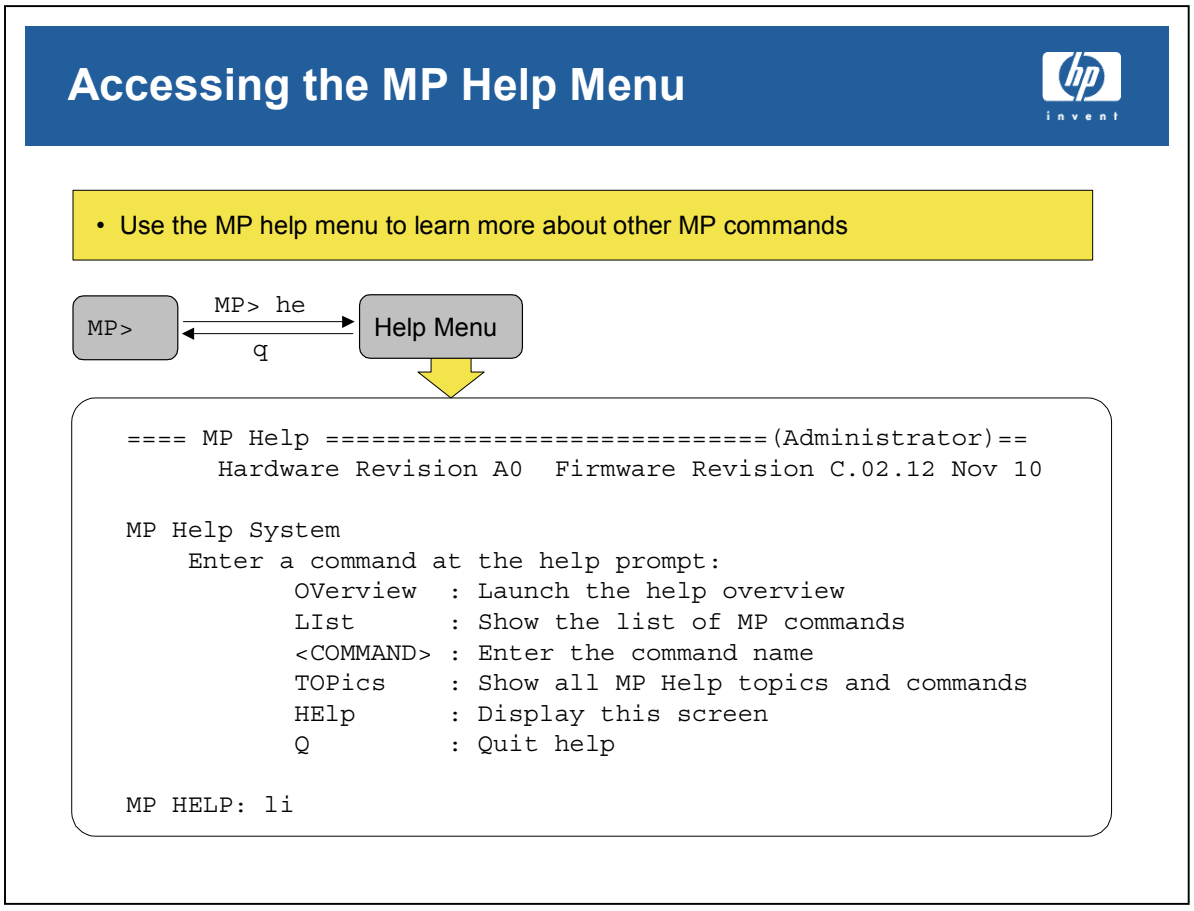

© 2004 Hewlett-Packard Company

```
0x58004e0000002970 0x000067071a10363a

25   PDC  0,1,0 *12 DNA_2_DNA_ERROR_ACCESSING_GROUP_A  08/26/2003
16:54:58
Log Entry 25: 08/26/2003 16:54:58
Alert Level 12: Software failure; Keyword:
DNA_2_DNA_ERROR_ACCESSING_GROUP_A
Crossbar chip 0 ; Status: 14
Logged by system firmware 4 during configuration subActivity 97
Actual Data: 0x000000ffff5f00d0
0x180045c05100297e 0x000000ffff5f00d0
0x58004d0000002970 0x000067071a10363a

MP:VWR (<CR>,<sp>,+,-,D,F,L,J,V,H,^B) > ^b
```

Press **^b** or **q** to return to the **MP>** main menu.

© 2004 Hewlett-Packard Company

15–14. SLIDE: Accessing the MP Help Menu



Student Notes

If you need additional information about any of the features described in this chapter, take advantage of the MP help interface, which includes comprehensive descriptions of all of the MP commands.

First, type the **he** command at the main menu to access the interactive MP help sub-menu.

```
MP> he
==== MP Help ================================================

MP Help System

Enter a command at the help prompt:
OVerview  : Overview of MP capabilities and concepts
TOPics    : Show all MP Help topics and commands
<TOPIC>   : Enter the topic name for help on a particular topic
LIst      : Show the list of MP commands
<COMMAND> : Enter the command name for help on individual command
HElp      : Help on help
Q         : Quit help
```


© 2004 Hewlett-Packard Company

```
MP:HELP (HE for main help, enter command name, or Q to quit)>
```

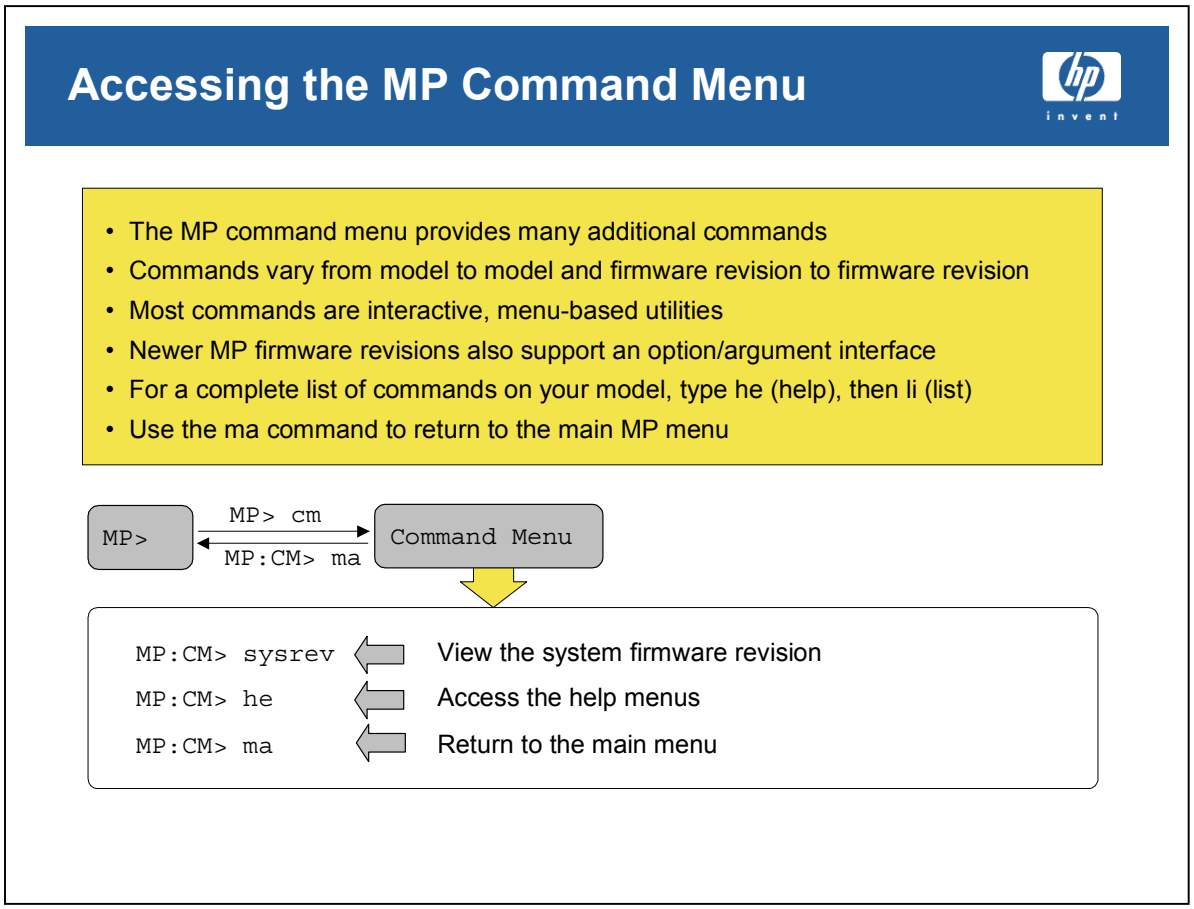
From the interactive help utility, you can view a number of non-command-specific help screens. Type **top** for a complete list of these additional topics, or **li** for a list of all of the supported MP commands. To view information about a specific command, simply type the command name.

```
MP:HELP (HE for main help, enter command name, or Q to quit)> top

==== TOPics ================================================
HELP TOPICS

The following topics can be entered for general information:

BOOTing           MP overview         PASSwords
CHASsis logs      HPterm and VT100    PORTS and connecting
COMMand menu      LIst of commands    POWER display and control
COMPlex profiles  MAIN menu           RESET, TOC, boot
CONSole           MODEM management    VFP
FW                MODES               WEB

MP:HELP (HE for main help, enter command name, or Q to quit)> who

==== MP Help: Connection ===========================

WHO : Display a list of MP connected users

This command displays the login name of the connected console client
users and the port on which they are connected.  For the LAN console
clients the remote IP address is also displayed.

SEE ALSO:  TE  (TEll)
```

Press **q** to return to the **MP>** main menu.

© 2004 Hewlett-Packard Company

15–15. SLIDE: Accessing the MP Command Menu



Student Notes

The commands discussed thus far represent the core MP functionality for accessing and viewing the system console interface. There is also a rich collection of tools available for managing the MP itself. Some of these will be described over the next couple slides. On some systems, these commands are available directly from the **`MP>`** main menu. On newer systems, however, these commands are only available from the **`MP:CM>`** command sub-menu. To get to the command menu, execute the **`MP> cm`** command.

```
MP> cm
MP:CM>
```

The commands that are available from this menu are model and firmware dependent. To determine your current firmware revision, type the **`sysrev`** command at the **`MP:CM>`** prompt. This command may not be supported on older models.

```
MP:CM> sysrev
SYSREV
Current firmware revisions
 MP FW      : E.02.26
 BMC FW     : 01.50
```


© 2004 Hewlett-Packard Company

```
 EFI FW    : 01.22
 System FW : 02.21
```

Most of the MP commands are interactive, menu-base utilities. Newer MP firmware revisions also provide a UNIX-like option/argument interface. You can access the help interface described on the previous page directly from the command menu on most models by typing **he**. To return to the command menu, press **q** at the help prompt. The sample output below shows the **lc** command description on one of the newer Integrity servers.

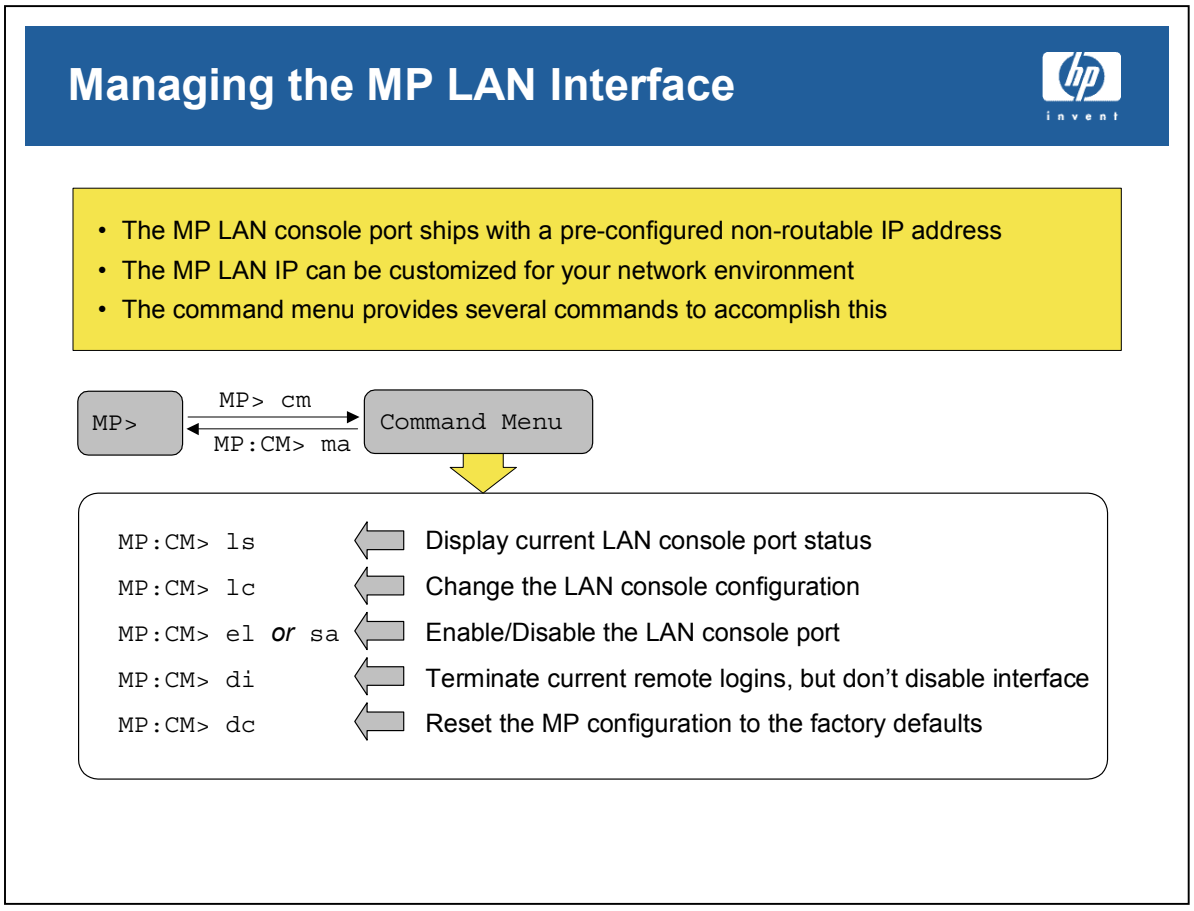
```
MP:CM> he
MP:HE> lc

==== MP Help: Port Configuration
============================(Admin)============

  LC  : LAN Configuration usage (IP address, etc.)

This command modifies the LAN Configuration.  Configurable parameters:
MP IP Address, MP host name, subnet mask, gateway, web access port
number, LAN speed, duplex, and autonegotiation. MP Host Name set in
this command is displayed at the MP command interface prompt.
Typically the DNS name for the LAN IP is entered.  This field can be
programmed to any useful name or phrase.  For clarity,
it is useful to enter: "MPNAME-on-SYSTEM" as the MP Host name, so
both names show up in the prompt (limit 19 chars, no spaces allowed.)

Command line usage:
LC [-ip <ipaddr>][-host <text>][-subnet <subnet>]
   [-web <n>][-gateway <ipaddr>][-link <auto|T(10baseT)>][ -nc ]

(ADMINISTRATOR level command.)
SEE ALSO:  LS, SA  (LAN Status, Set Access)
====
(HE for main help, enter command name, or Q to quit)
MP:HE> q
MP:CM>
```

To return from the command menu to the main MP menu, type **ma** at the **MP:CM>** prompt.

```
MP:CM> ma

MP MAIN MENU:

         CO: Consoles
        VFP: Virtual Front Panel (partition status)
         CM: Command Menu
         CL: Console Logs
         SL: Show chassis Logs
         HE: Help
          X: Exit Connection
MP>
```


© 2004 Hewlett-Packard Company

15–16. SLIDE: Managing the MP LAN Interface



Student Notes

Initially, the LAN console port is pre-configured with a non-routable IP address, which varies from model to model. Check your owner's manual for details. For security reasons, this IP should be changed as soon as possible. Several MP commands may be used to do this.

Viewing the current configuration with `ls`

From the **`CM`** menu, you can use the **`ls`** (LAN status) command to view the current configuration.

```
MP:CM> ls

Current configuration of MP LAN interface
  MAC address    : 00:30:6e:49:ff:9a
  IP address     : 192.168.1.1
  Hostname       : myhost-mp
  Subnet mask    : 255.255.255.0
  Gateway        : 192.168.1.1
  Status         : UP and RUNNING
  AutoNegotiate : Enabled
  Data Rate      : 100 Mb/s
```


© 2004 Hewlett-Packard Company

```
  Duplex         : Half
  Error Count    : 0
  Last Error     : none
```

Most servers have just one MP LAN port, but Superdomes have two:

- A "customer" LAN connection, which is intended for connection to the customer's LAN, for normal administrative access to partition consoles.
- A "private" LAN connection, which is intended for support and diagnostic access only over a separate local network with limited access.

Changing the Current Configuration with `lc`

You can change your LAN console port configuration via the **`lc`** (LAN configure) command:

```
MP:CM> lc

This command modifies the LAN parameters.

Current configuration of MP LAN interface
  MAC address    : 00:30:6e:49:ff:9a
  IP address     : 192.168.1.1
  Hostname       : myhost-mp
  Subnet mask    : 255.255.255.0
  Gateway        : 192.168.1.1
  Status         : UP and RUNNING
  AutoNegotiate : Enabled
  Data Rate      : 100 Mb/s
  Duplex         : Half
  Error Count    : 0
  Last Error     : none

Do you want to modify the configuration for the customer LAN? (Y/[N]) n
```

Answering **`y`** at the prompt will generate a series of interactive prompts that allow you to change your LAN configuration.

Disabling and Enabling the LAN console port with `el/dl/sa`

If you decide not to use the MP LAN console functionality, the feature can be disabled. Some MP cards support an explicit **`dl`** (disable LAN) command. Others simply use the **`el`** (enable LAN) command in a toggle fashion to turn the remote LAN console port functionality on and off.

```
MP:CM> el
Enable telnet access? (Y/[N]) y
-> Telnet access enabled.

MP:CM> dl
Disable telnet access and close open telnet connections? (Y/[N]) y
WARNING: Answering yes will close this connection.
Are you sure? (Y/[N]) y
```


© 2004 Hewlett-Packard Company

Newer systems provide an **sa** (system access) command that may be used to enable/disable the modem port, LAN port, and other access methods, too. The output below was captured on an Integrity Superdome.

```
MP:CM> sa

This command displays and allows modification of access parameters.

    T - Telnet access          : Enabled
    M - Modem access           : Enabled
    N - Network Diagnostics    : Enabled
    D - DIAG Menu              : Disabled
    I - IPMI Lan access        : Enabled

    Select access mode to change : t
    Are you sure you want to disable Telnet access (Y/N) ? : y
```

If you disable the remote console interface while logged into the MP via a **telnet** session, note that your session will be terminated, and you will only be able to re-enable the port via a remote support modem or a local console.

Disconnecting Remote Console Users with `di`

You can immediately terminate all remotely connected console users via the **di** command. If you yourself are logged into the MP remotely, note that your session will be terminated, too! Before you execute this command, you may wish to use the **who** command to determine which other users are logged into the MP, and the **te** (tell) command to warn them that they may be disconnected.

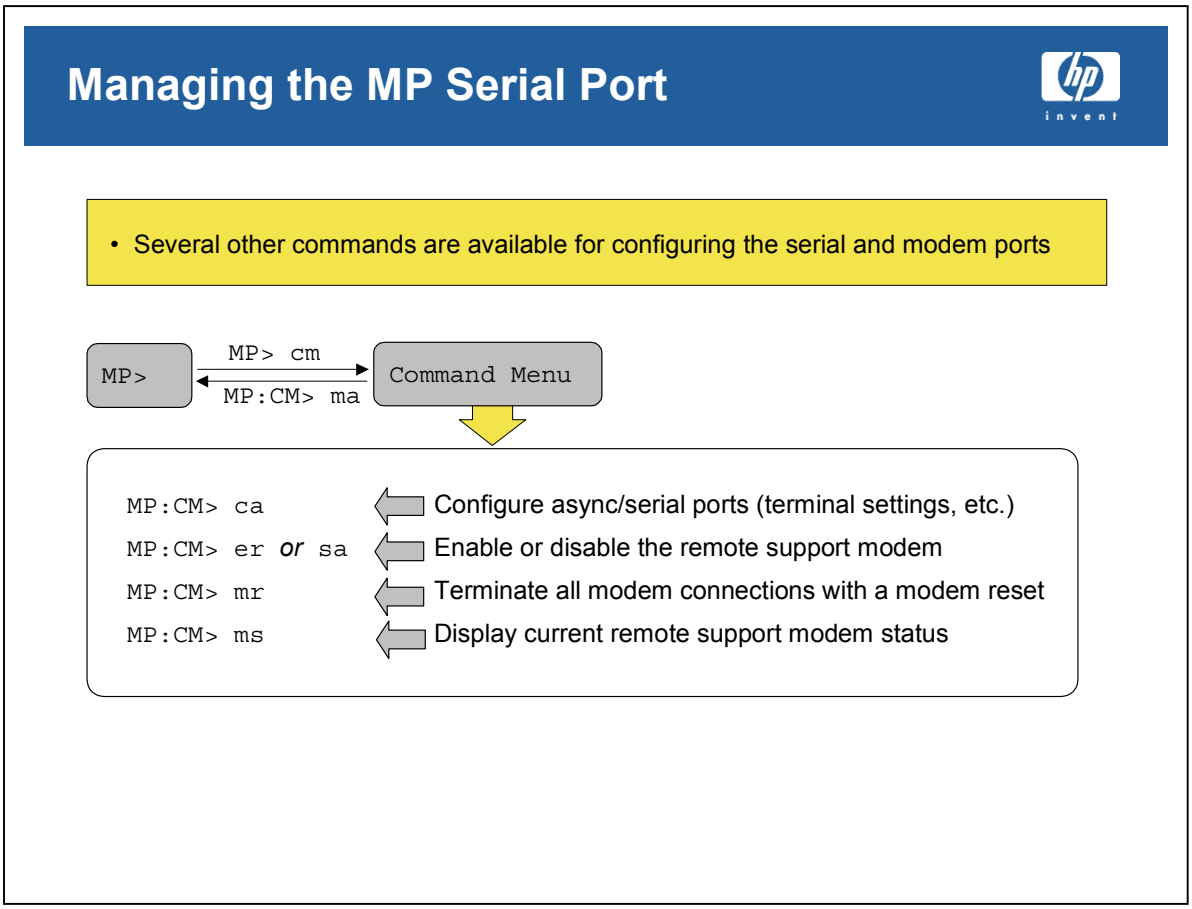
```
MP:CM> di

This command disconnects console connections.

    Remote console:  DISCONNECTED
    LAN console:     CONNECTED (1)

    Disconnect Remote console? (Y/[N])  y
    Disconnect LAN console? (Y/[N])  y

    Remote console:  DISCONNECTED
    LAN console:     DISCONNECTED

WARNING: Answering yes will close this console.
Are you sure? (Y/[N]) y

<Your 'TELNET' connection has terminated>
```

Resetting the Default MP Configuration with `dc`

If you mis-configure the MP LAN port, you can return to the MP factory defaults via the **dc** command, or by hitting the MP reset button (see your hardware owner's manual for details). Be very cautious with this command, as it also removes all defined MP user accounts!

© 2004 Hewlett-Packard Company

15–17. SLIDE: Managing the MP Serial Port



Student Notes

The MP command menu also makes it possible to view and modify the configuration of your local serial console and remote support modem ports.

Configuring the Serial/Modem Ports with `ca`

The **`ca`** command allows the MP user to set the communications parameters of the local and remote RS232 ports.

```
MP:CM> ca

This command allows modification of the asynchronous parameters.

    WARNING: Changing the asynchronous parameters requires
             rebooting the MP for proper local operation.

Do you want to modify the local rs232 port parameters? (Y/[N]) y

    Current Asynchronous Baud Rate is: 9600 bits/s
    Do you want to modify it? (Y/[N])
```


© 2004 Hewlett-Packard Company

```
    Current Flow Control is: Software
    Do you want to modify it? (Y/[N])

Do you want to modify the modem parameters? (Y/[N]) y

    Current Modem Protocol is: CCITT
    Do you want to modify it? (Y/[N])

    Current Modem Baud Rate is: 9600 bits/s
    Do you want to modify it? (Y/[N])

    Current Flow Control is: Software
    Do you want to modify it? (Y/[N])
```

Enabling and Disabling the Remote Support Modem via `er/dr/sa`

Once you've defined your remote support modem parameters, the port must be enabled before it will accept incoming calls. Use the **`er`** (enable remote support) command to enable the port. On some models, the same **`er`** command may be used to toggle the modem off again. Other models have a dedicated **`dr`** (disable remote support) command. The sample output below was taken from an rp7405.

```
MP:CM> er

    Current remote/modem port access: MP
    Do you want to modify this configuration? (Y/[N]) y

    Please indicate the new access mode for the remote/modem port.
    L - Lock out all dial-in access.
    G - Dial-in results in an MP login.
                 Choose one of (L or G): l
    New remote/modem port access settings will be: Locked-out
    Please confirm (Y/[N]) y

WARNING: Command execution may take 25 seconds.
```

Newer systems provide an **`sa`** (system access) command that may be used to enable/disable the modem port, LAN port, and other access methods, too. The following output was taken from an Integrity Superdome.

```
MP:CM> sa

This command displays and allows modification of access parameters.

    T - Telnet access          : Enabled
    M - Modem access           : Enabled
    N - Network Diagnostics    : Enabled
    D - DIAG Menu              : Disabled
    I - IPMI Lan access        : Enabled

    Select access mode to change : m
    Are you sure you want to disable Modem access (Y/N) ? : y
```


© 2004 Hewlett-Packard Company

Disconnecting Current Modem Users via `mr`

You can disconnect remote support modem users at any time via the **`mr`** (modem reset) command. This command makes the MP send an AT Z command to the modem, which will reset it. Any modem connections will be lost. The initialization results can be viewed via the **`ms`** command.

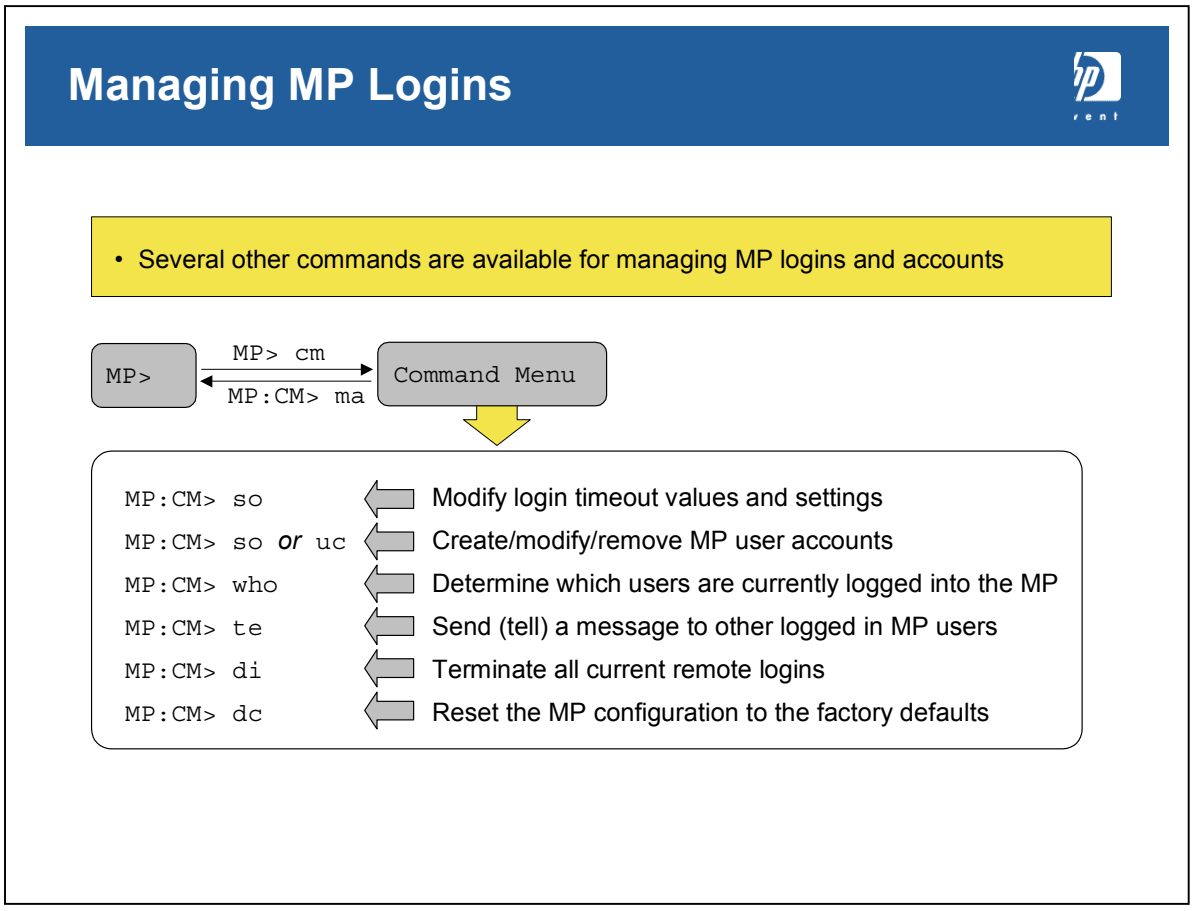
```
MP:CM> mr

    Do you want to reset the modem? (Y/[N]) y

WARNING: Command execution may take 25 seconds.
```

Viewing the Modem Status via `ms`

The **`ms`** (modem status) command displays the state of the modem lines connected to the remote/modem serial port. The display can be updated by pressing the **`[return]`** key. The current state of the status signals DCD, CTS, DSR, RI and the last state of the control signals DTR, RTS set by the firmware are displayed. The text under "modem state" is an interpretation of these signals.

© 2004 Hewlett-Packard Company

15–18. SLIDE: Managing MP Logins



Student Notes

In order to access the MP, users must enter a valid MP username and password. Several users can login on the MP simultaneously, though only one user at a time is allowed to access any single partition's system console. The MP provides several tools for managing user logins and sessions.

Configuring Login Parameters and User Accounts with `so`

The **`MP:CM> so`** (security options) command may be used to modify MP login timeout parameters and a number of other security parameters. The sample output below shows the security options that can be configured on an Integrity Superdome. For now, we'll focus on the first option. The next slide discusses user parameters. The IPMI password allows administrators on cell-based systems to modify the partition configuration remotely.

```
MP:CM> so

    1. MP wide parameters
    2. User parameters
    3. IPMI password
       Which do you wish to modify? ([1]/2/3) 1
```


© 2004 Hewlett-Packard Company

```
MP wide parameters are:
Login Timeout : 3 minutes.
Number of Password Faults allowed : 3
Flow Control Timeout : 5 minutes.

Current Login Timeout is: 3 minutes.
Do you want to modify it? (Y/[N])

Current Number of Password Faults allowed is: 3
Do you want to modify it? (Y/[N])

Current Flow Control Timeout is: 5 minutes.
Do you want to modify it? (Y/[N])
```

Creating and Modifying MP User Accounts via `so` and `uc`

On some platforms, the **`so`** command may be used to configure and manage MP user accounts. Other platforms use the **`uc`** (user configure) command to manage MP user accounts. Both commands will be described in detail on the next slide.

Determining who is Logged in via `who`

Recall that several users can login on the MP simultaneously. The **`who`** command reports which users are currently logged in on the MP, and where they logged in from.

```
MP:CM> who

User Login          Port Name       IP Address
Admin               LAN             10.1.1.1
Oper                LAN             10.1.1.2
```

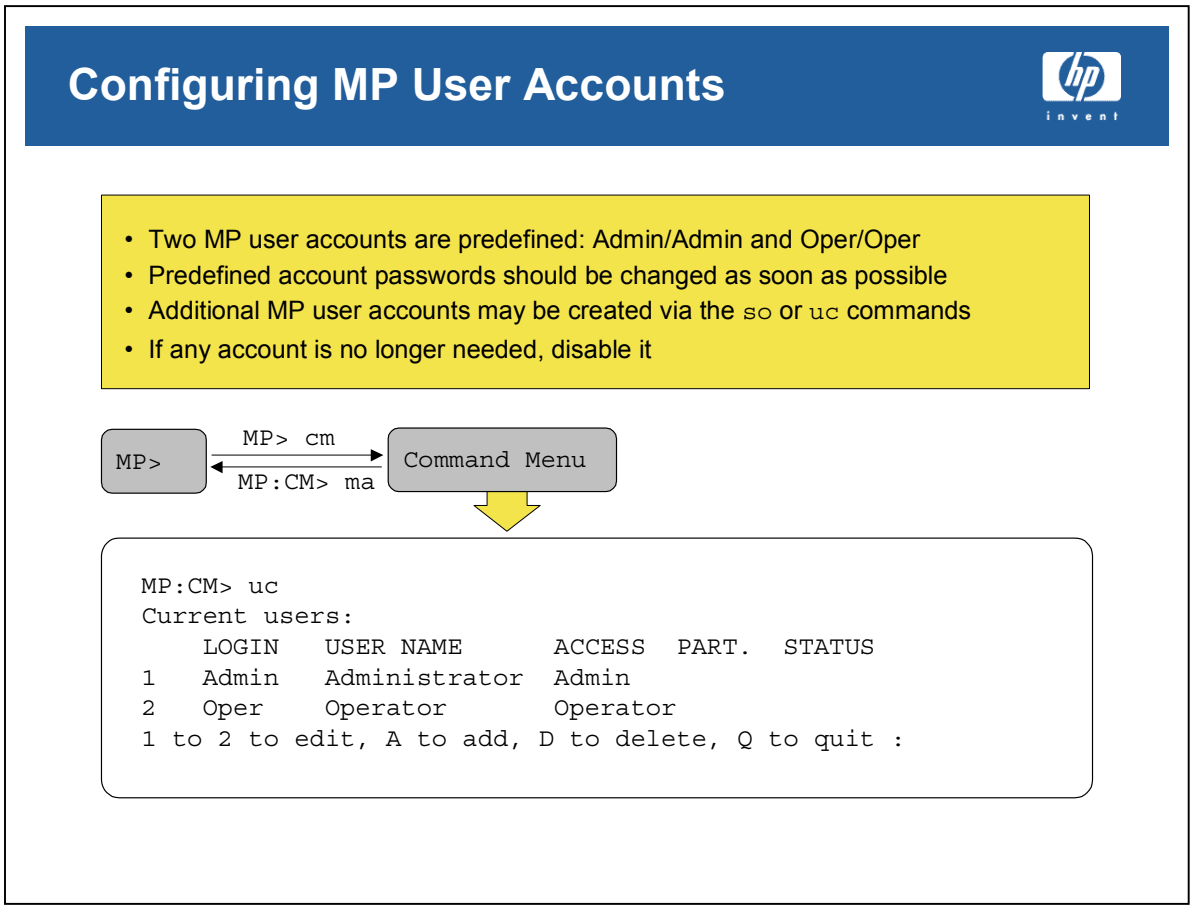
Communicating with Other MP Users via `te`

You can use the **`te`** (tell) command to display a message to all users that are currently logged into the MP. This is particularly useful if you intend to reboot a partition, change the MP configuration, or perform other tasks that might affect other MP users.

```
MP:CM> te
Enter message to broadcast (must be less than 80 characters).
Starting MP firmware update in 5 minutes.  Please logout.
```


© 2004 Hewlett-Packard Company

15–19. SLIDE: Configuring MP User Accounts



Student Notes

Up to twenty MP user accounts may be defined on a system. The first MP user is predefined, and has full access to all MP menus and commands and partitions. The username for this account is initially set to “Admin” with password “Admin”. This should be changed as soon as possible.

The Admin user (or any other user with Administrator access rights) may configure the nineteen remaining MP users as desired. Each user is assigned a username, password, and access level. Three access levels are available on the MP:

Administrator Users:	Administrator users have access to all MP commands, and all partitions on the system.
Operators Users:	Operators have access to all commands necessary to access and manage all partitions, but don’t have the ability to modify the configuration of the MP itself.
Single-Partition Users:	Single-partition users have access to a limited number of MP commands that are required to access, boot, and reboot a specified partition.

© 2004 Hewlett-Packard Company

Note that MP users have no relationship to the user accounts defined in the HP-UX **/etc/passwd** file. Entries in the HP-UX **/etc/passwd** file determine who can login on a partition running HP-UX, while the MP password database determines which users can access the MP menu interface.

Creating and Managing MP Users via `so` or `uc`

On some systems, you can add/remove/edit MP users via the **MP:CM> so** (security options) command, which can also be used to set login timeout values and a variety of other parameters. The output below shows the procedure required to configure a MP user account on an Integrity Superdome:

```
MP:CM> so

    1. MP wide parameters
    2. User parameters
    3. IPMI password
       Which do you wish to modify? ([1]/2/3) 2

Current users:

    LOGIN   USER NAME                ACCESS        PART.  STATUS

1   Admin   Administrator            Admin
2   Oper    Operator                 Operator         Disabled

1 to 5 to edit, A to add, D to delete, Q to quit : a

Enter Login : student

Enter Name : HPES student

Enter Organization : HPES

Valid Access Levels:  Administrator, Operator, Single Partition User
Enter Access Level (A/O/[S]) : o

Valid Modes:  Single Use, Multiple Use
Enter Mode (S/[M]) : m

Valid States:  Disabled, Enabled
Enter State (D/[E]) : e

Enable Dialback ? (Y/[N]) n

Enter Password :
Re-Enter Password :

New User parameters are:
Login             : student
Name              : HPES student
Organization      : HPES
Access Level      : Operator
Mode              : Multiple Use
State             : Enabled
```


© 2004 Hewlett-Packard Company

```
Default Partition :
Dialback          : (disabled)

Changes do not take affect until the command has finished.
Save changes to user number 3? (Y/[N]) y
```

Other systems supply an **MP:CM> uc** command (user configure) command specifically for configuring user accounts. The output below was taken from an rx2600.

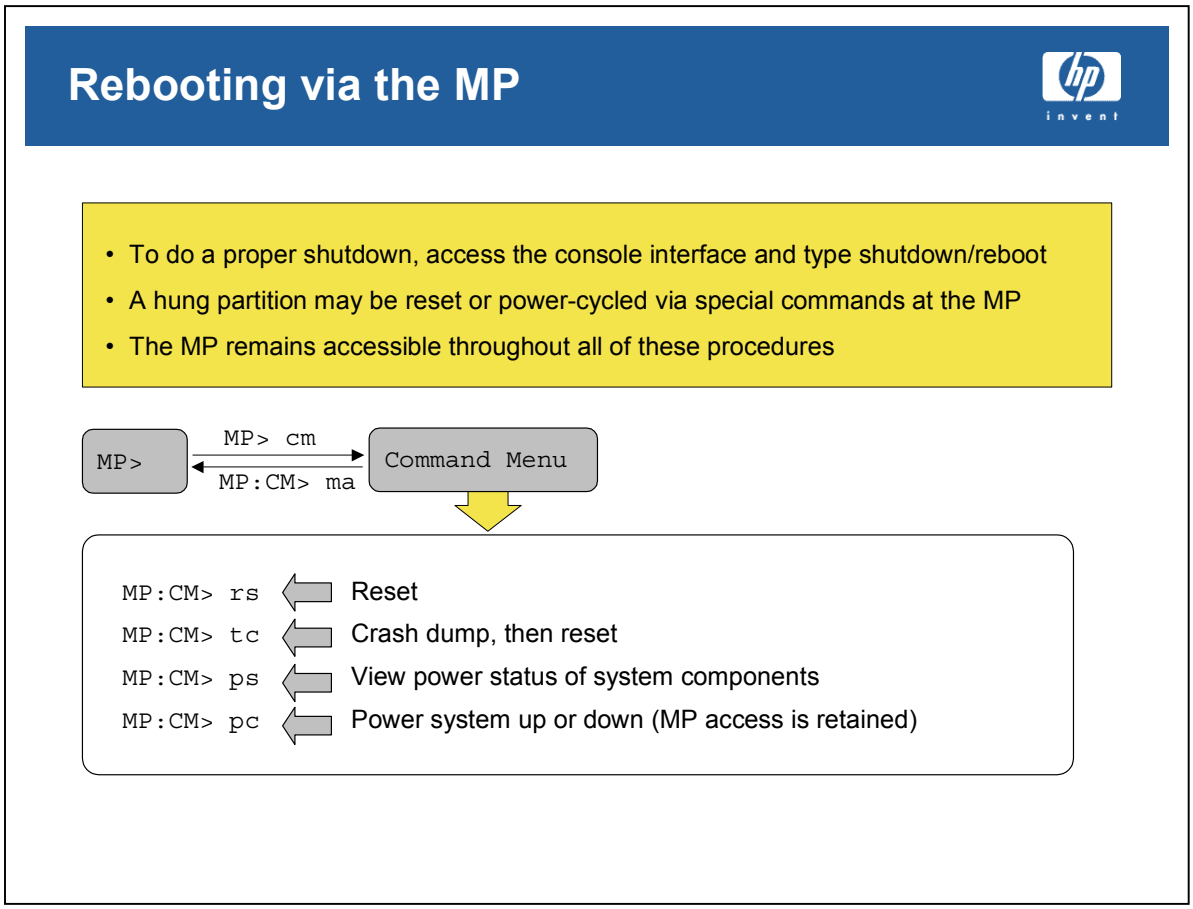
```
MP:CM> uc

This command allows you to modify the user configuration.
User Configuration Menu:
     L - List current users
     N - Add a New user
     C - Change a current user
     D - Delete a current user

Enter menu item or [Q] to Quit: l
Current User Configuration:
    Login ID     User Name         Access Enabled
1 - admin        Administrator     Admin  Yes
2 - oper         Operator          Oper   Yes

Enter number for more detail, or [Q] to Quit: q
```

What if You Forget the Admin Password?

If you forget the MP Admin user's password, the MP security configuration may be reset by executing the **dc** command or by pressing the reset button on the MP card. Doing so removes all of the MP users, disables the remote support modem, and resets the LAN console interface parameters to their factory default values.

© 2004 Hewlett-Packard Company

15–20. SLIDE: Rebooting via the MP



Student Notes

Normally, you should use the standard HP-UX **`shutdown`** and **`reboot`** commands to shutdown and/or reboot your system. These commands will be described in detail elsewhere in the course.

On a hung system, or otherwise non-responsive system, though, it may not be possible to access an HP-UX prompt that would allow you to shutdown properly. In this situation, it may be necessary to reboot using **`MP:CM>`** commands.

The **`MP:CM>`** menu offers several alternatives.

`rs` This command performs an immediate partition reset. The command irrecoverably halts all processing and I/O activity, and restarts the partition. The effect of this command is very similar to cycling the system power - the OS is not notified, and data corruption may result. On partitionable systems, you will be asked which partition you wish to reset. If possible, you should always halt HP-UX via **`shutdown`** or **`reboot`** before executing this command.

© 2004 Hewlett-Packard Company

tc Resets the partition via the INIT or TOC (Transfer of Control) signal. Execution of this command irrecoverably halts all system processing and I/O activity and restarts the partition. Unlike the **rs** command, the **tc** command also signals the processors to "dump state" before restarting. This results in a system core dump under the **/var/adm/crash/** directory that may be analyzed afterwards if the system hung due to an application or OS error. On partitionable systems, you will be asked which partition you wish to TOC. On all systems, this command should be used with caution: it will very likely cause data corruption.

pe or **pc** The **pe** (power entity) and **pc** (power control) commands may be used to control power to your system. On partitionable systems, the **pe** command may be used to toggle power to individual entities such as cell boards, I/O chassis, and entire cabinets. On non-partitionable systems, the **pc** command may be used to toggle power for the entire system. Neither command affects power for the MP interface.

ps The **ps** command reports power status for various system components. On partitionable systems, the **ps** command allows the MP user to select an entity of interest, and displays detailed status information for the specified entity. On non-partitionable systems, the command displays a single screen summary of the system power status. The sample output below was generated on an rp7405. Output on other models may vary. If possible, you should always halt HP-UX via **shutdown** or **reboot** before executing this command.

```
MP:CM> ps

Display detailed status of the selected MP bus device.

The following MP bus devices were found:
+---+----+----+-----+-----+-------+-----+
|   |    |    |     |     |       |     |
|   |    |LAN\| Sys |     |  IO   |     |
|Cab| MP |SCSI|Bkpln|Cells|Chassis| BPS |
| # |M  S|0  1|     | 0 1 | 0   1 | 0 1 |
+---+----+----+-----+-----+-------+-----|
| 0 |*  *|*  *|  *  | * * | *   * | * * |
+---+----+----+-----+-----+-------+-----+

You may display detailed power and hardware status for the
following items:

    T - Cabinet
    S - System Backplane
    G - MP (Core I/O)
    P - IO Chassis
    C - Cell
        Select Device: t

HW status for rp7410 cabinet : NO FAILURE DETECTED

Power switch is on
Right Door is closed
Top Door is closed
Left Door is closed
```


© 2004 Hewlett-Packard Company

```
Total Power Available 7200 VA
Total Power Needed 2215 VA
Power Redundancy : redundant
Power Viability : viable

 Power Status
---------------+-----+-----+-------+-----+
               |     |     |       |     |
               | Sys |     |  IO   |     |
               |Bkpln|Cells|Chassis| BPS |
               |     | 0 1 | 0   1 | 0 1 |
---------------+-----+-----+-------+-----|
 Populated     |  *  | * * | *   * | * * |
 Enabled       |  *  | * * | *   * | * * |
 Power OK      |  *  | * * | *   * | * * |
 Warning/Fault |     |     |       |     |
 Attention LED |     |     |       |     |


 AC Line status:
  Line A0 Present
  Line A1 NOT PRESENT
  Line B0 Present
  Line B1 NOT PRESENT

              -- Press <CR> to continue, or 'Q' to Quit --

Front Fan Speed   : normal
Rear Fan speed    : normal
I/O Bay Fan Speed : normal
Temperature state : normal
Fan Redundancy    : redundant
Overtemp Shutdown Enabled

            | BPS |     PCI     |
            | Fans|     Fans    |
            | 0 1 | 0 1 2 3 4 5 |
+-----------+-----+-------------+
  Populated | * * | * * * * * * |
  Failing   |     |             |
  Failed    |     |             |

            | Standby/Main Fans |
            |                   |
            | 0 1 2 3           |
+-----------+-------------------+
  Populated | * * * *           |
  Failing   |                   |
  Failed    |                   |

            | Cell Fans |
            | 0 1 2 3 4 |
+-----------+-----------+
 Cell 0     |           |
  Populated | * * * * * |
  Failing   |           |
```


© 2004 Hewlett-Packard Company

```
 Failed    |           |
Cell 1     |           |
 Populated | * * * * * |
 Failing   |           |
 Failed    |           |
```

Other Boot Related MP Commands

Two other boot related MP commands are required during the partition modification process on node-partitionable systems: **rr** and **bo**. These commands are described in detail in HP's partition management class, U5075S.

© 2004 Hewlett-Packard Company

15–21. SLIDE: For Further Study

For Further Study

- To learn more about the GSP/MP, have a look at the *HP System Partitions Guide Administration for nPartitions* on http://docs.hp.com.
- Also look for your particular system model's hardware manual on http://docs.hp.com.
- Finally, quite a bit of information is available through via the GSP/MP help menus.

Student Notes

© 2004 Hewlett-Packard Company

15–22. LAB: Interacting with the MP

Directions

Many critical, low-level administration tasks may only be accomplished via the system console. On all of HP's current server models, a virtual system console is accessible via the Management Processor (or "Guardian Service Processor" on older systems). This lab provides an opportunity for you to explore the MP's features and commands.

Your instructor should have given you the connectivity information for your server's Management Processor. Record the username and network information below:

MP IP address: _____

MP hostname: _____

MP admin username: _____

MP admin password: _____

Assigned partition: _____ (only needed on nPar systems)

Some sites may prohibit student admin access to the MP. In this case, your instructor may simply demonstrate some portions of the lab.

© 2004 Hewlett-Packard Company

Part 1: Accessing and Exploring the MP

This portion of the lab shows you how to connect to your system's MP. Operator access is sufficient to complete this portion of the lab.

1. Open a **`telnet`** session to your server's MP.

2. Login using the username and password suggested by your instructor.

3. Some versions of the MP firmware immediately display the MP main menu. Others immediately place MP users in the MP console interface. If your MP sent you directly to the MP console interface, press **`^b`** to return to the MP main menu.

4. You should see several commands on the MP main menu. However, there are many other commands that don't appear in the main menu. Which command can you use to learn more about the features of the MP? Use this command to generate a list of supported MP commands and fill-in the blanks below. After you answer the questions, return to the MP main menu.

   a. The ____ command may be used to access the system console.
   b. The ____ command may be used to access the virtual front panel.
   c. The ____ command may be used to reset the system.
   d. The ____ command may be used to configure MP user accounts.
   e. The ____ command may be used to view the MP LAN configuration.
   f. The ____ command may be used to view the system's power status.
   g. The ____ command may be used to view a list of users currently using the MP.

© 2004 Hewlett-Packard Company

Part 2: Accessing the System Console via the MP

Perhaps the most common reasons for accessing the MP is to access the system console. This is your chance to explore this functionality! Operator access is sufficient to complete this part of the lab.

1. Use the MP to access your system console.

2. Login as HP-UX user root, and run the hostname command to verify that you accessed the right console.

3. From the console interface, you can return to the MP main menu at any time by pressing **`^b.`** Without logging out from HP-UX, press **`^b`**.

4. Access the console interface again. Are you prompted for a Unix username and password this time?

5. Logout from HP-UX, and press **`^b`** to return to the MP.

© 2004 Hewlett-Packard Company

6. Access the console interface again. What happens?

7. Why is it important to logout of HP-UX before using `^b` to return to the MP main menu?

© 2004 Hewlett-Packard Company

Part 3: Rebooting the System via the MP Console

When you reboot the system, it is important to do so via the MP console interface to ensure that you don't miss any important messages. Try it! This portion of the lab only requires MP operator privileges.

1. If you aren't already logged in as root on the MP console, login now.

2. While logged in as root, type **`shutdown -hy 0`** to halt your system or nPar. Watch the resulting shutdown messages.

   ```
   # shutdown -hy 0
   ```

3. When the shutdown appears to be complete, return to the MP main menu by pressing **`^b`**.

4. What command can you use to reset your system and reboot? Make it so! (Note: Some systems automatically power-off when halted via **`shutdown -h`**. If your system is powered-off, power it back on again with the **`pc`** command. This should automatically boot the system, too).

5. Now monitor the system's self-tests via the VFP.

© 2004 Hewlett-Packard Company

6. Watch the VFP for a few minutes, then access the console log to see some of the console messages generated by the HP-UX startup process.

7. When your system finishes booting, access the console interface again and login as root to verify the boot succeeded.

```
MP> co
Console Login: root
Password: **          Enter your root password here
```

8. Return to the MP main menu before proceeding.

```
# exit
^b
```


© 2004 Hewlett-Packard Company

Part 4: Configuring and Using MP User Accounts

The MP must be carefully secured to control access to your system console. In this part of the lab, you will learn how to create and manage MP user accounts to prevent unauthorized access to your system. This portion of the lab requires MP admin privileges.

1. Which MP command may be used to manage MP access accounts? Read the full description of this command via the MP **`he`** (help) utility.

2. Using the command you discovered in the previous question, create a new MP user account for yourself, then log out of the MP.

   - Use login name "student"
   - Also use "Student" as the user's real name
   - Use HPE as the user's organization name
   - Grant your new user "Operator" access to the MP
   - Configure the "Multiple Use" mode for the new account
   - Enable the account
   - Disable dialback access
   - Choose a password for the new user

3. In the previous question, you granted the “student” user “operator” level access to the MP. Which of the following MP commands should the new MP user be able to access? (Don’t try the commands yet. You will have a chance to try that in a moment.)

   a. **`co`**
   b. **`vfp`**
   c. **`rs`**
   d. **`ps`**
   e. **`ls`**
   f. **`lc`**
   g. **`uc/so`**

.

© 2004 Hewlett-Packard Company

4. Let's test your answers to the previous question. Without closing your original "admin" MP **`telnet`** session, open a second **`telnet`** window on your desktop, connect to the MP, and login as MP user "student" to verify that the new MP user account works.

5. Can the original admin user tell that there is a second user logged into the MP? Run the MP **`who`** command in the original **`telnet`** window to find out! (Hint: On some systems, you may need to go to the **`cm`** command menu to access the **`who`** command.)

6. What command can you use to warn other MP users that you are using the system? Make it so!

7. What happens if the "student" user attempts to access the console interface at the same time as the "admin" user? Try it! Type the **`co`** command in both telnet windows. Explain the results.

8. Remove the "student" MP user account. If you made any other changes to the MP configuration undo them as well before moving on.

© 2004 Hewlett-Packard Company

© 2004 Hewlett-Packard Company

Module 16 — Booting PA-RISC Systems

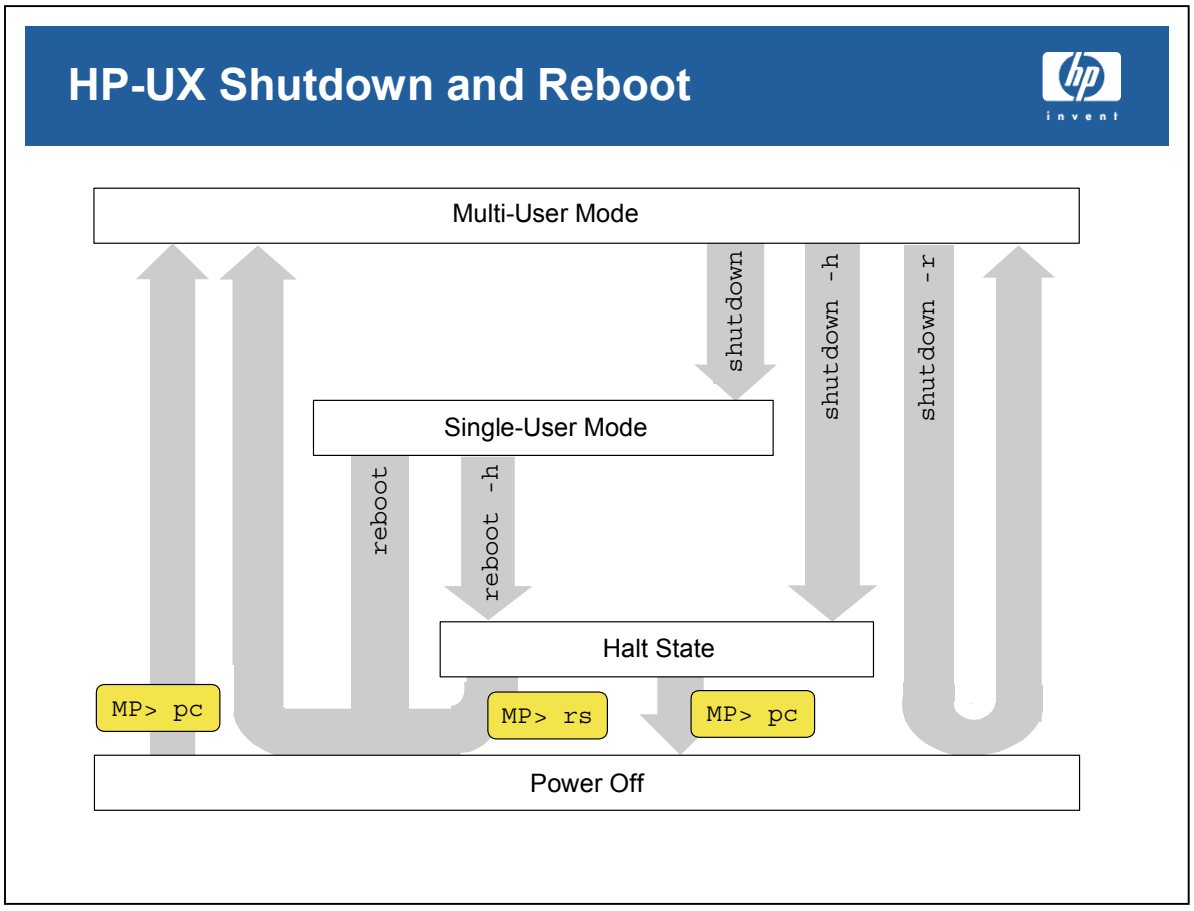
Objectives

Upon completion of this module, you will be able to do the following:

- Properly shut down an HP-UX system using **`shutdown`** and **`reboot`**.
- Describe how the PDC/BCH, ISL/IPL, and HPUX utility load the kernel in memory.
- Boot from the primary, alternate, or HA alternate boot device.
- Boot from a system recovery tape.
- Boot from an install CD or Ignite install server.
- Boot from an alternate kernel.
- Boot to single-user mode to perform system maintenance.

© 2004 Hewlett-Packard Development Company, L.P.

16–1. SLIDE: HP-UX Shutdown and Reboot



Student Notes

An HP-UX system will always be in one of several operation states. The system state determines which system services are available. Some of the system administrative tasks described in this course require you to change your current operation state. This chapter will present an overview of the functionality available in each state.

Most of the time, your system will be in *multi-user mode*. In multi-user mode, users can log in; file systems are mounted; and most system services and daemons are available.

There are occasions, however, when you may need to take your system down to *single-user* mode to perform administrative tasks. In single-user mode, users cannot log in, non-critical file systems are unmounted, and non-critical system daemons are shut down. Moving to single-user mode brings your system to a quiet state, from which you can do system backups, run **`fsck`**, or perform other system administrative tasks, without interference from user processes.

Bringing your system to a *halt state* unmounts all file systems, and kills all processes. Always halt HP-UX before powering-off your server or workstation.

© 2004 Hewlett-Packard Development Company, L.P.

Taking your system to the power-off state terminates power to all system hardware components, except the Management Processor card.

Several commands may be used to move between states on an HP-UX system:

- From *multi-user* mode, use the **`shutdown`** command to reboot, halt, or shutdown to single-user mode.
- From *single-user* mode, use the **`reboot`** command to reboot or halt the system.
- From the *halt state*, use the **`rs`** (reset) command from the Management Processor Command Menu to reset and boot the system. On workstations, push the reset button on the workstation chassis.
- From a *power-off state*, use the **`pc`** (power control) command from the Management Processor Command Menu to power on the system or simply press the power button.
- On a *hung system*, it may be necessary to use the **`rs`** or **`tc`** commands from the Management Processor Command Menu to reset and boot the system. On workstations, push the reset button on the workstation chassis.

Changing State from Multi-user Mode with `shutdown`

Executing the **`shutdown`** command terminates system activities in an orderly and consistent manner. The **`shutdown`** command performs the following tasks:

- Prompts the administrator for a broadcast message to send to all users.
- Broadcasts the warning message to all user terminal sessions.
- Grants a 60 second (by default) grace period for users to log out.
- Kills all user logins.
- Gracefully shuts down all non-critical processes using **`/sbin/rc*.d/K*`** kill scripts.
- Unmounts all non-critical file systems.

Depending on the option specified, **`shutdown`** will either leave the system in single-user mode (no options), the halt state (if **`-h`** was specified), or initiate a reboot (if **`-r`** was specified).

Common **`shutdown`** options:

`# shutdown -y 600`	*# shutdown to single-user mode in 600 seconds."-y" (yes) option prevents shutdown from requesting confirmation before proceeding.*
`# shutdown -hy 600`	*# shutdown to a halt state in 600 seconds."-y" (yes) option prevents shutdown from requesting confirmation before proceeding.*

© 2004 Hewlett-Packard Development Company, L.P.

```
# shutdown -ry 600        # reboot in 600 seconds without requesting confirmation.

# shutdown -ry 0          # reboot immediately without requesting confirmation.
```

Changing State from Single-user Mode with `reboot`

The **`reboot`** command uses the SIG_KILL signal to kill running processes. This takes the system down quickly, but can cause problems for applications and file systems. The **`shutdown`** command shuts down applications and processes more gracefully, and thus is the preferred method for halting or rebooting the system from multi-user mode. Reboot may be used if:

- The system is already in single-user mode.
- You need to bring the system down very quickly.

```
# reboot -h     # shutdown to a halt state (only use this from single-user mode).
# reboot        # reboot
```

Changing State from the Halt State with the `rs` MP Command

After halting a system or partition with either the **`reboot -h`** or **`shutdown -h`** command, you can initiate the boot process via the **`rs`** command on the MP Command Menu. Although the **`rs`** command may be executed on a running system, too, doing so may cause corruption.

```
MP> cm
MP:CM> rs

Execution of this command irrecoverably halts all system processing and
I/O activity and restarts the computer system.

Type Y to confirm your intention to restart the system: (Y/[N]) y
-> SPU hardware was successfully issued a reset.
```

Changing State via the `pc` MP Command

Once your system reaches the halt state, you can safely power-off the system hardware by either pressing the power switch on the system cabinet, or, on servers, by executing the **`pc`** command from the MP Command Menu (output on your system may be slightly different from the output shown below).

```
MP> cm
MP:CM> pc

Current System Power State: On      Power Switch State: On
   ON  - Turn the power On
   OFF - Turn the power Off

Enter your choice (ON/OFF) or [Q] to quit: off
Enter delay (in minutes  0-6000) to power Off (CR for no delay):
System will be powered Off in 0 minutes.
   Confirm? (Y/[N]): y
   -> System has been powered Off
```


© 2004 Hewlett-Packard Development Company, L.P.

If your system has been powered off for some reason, you can power it back on again by pressing the power button, or, on servers, by executing the **pc** command from the MP Command Menu.

```
MP> cm
MP:CM> pc

Current System Power State: Off      Power Switch State: On
   ON  - Turn the power On
   OFF - Turn the power Off

Enter your choice (ON/OFF) or [Q] to quit: on
Enter delay (in minutes  0-6000) to power On (CR for no delay):
System will be powered On in 0 minutes.
  Confirm? (Y/[N]): y
  -> System has been powered On
```

Changing State on a Hung System with the `tc` and `rs` MP Commands

On rare occasions, a bug in an application or the OS may leave your system in a "hung" state. In these situations, it may be necessary to use the MP **tc** or **rs** commands to reboot. Use these commands with care, as they may cause corruption.

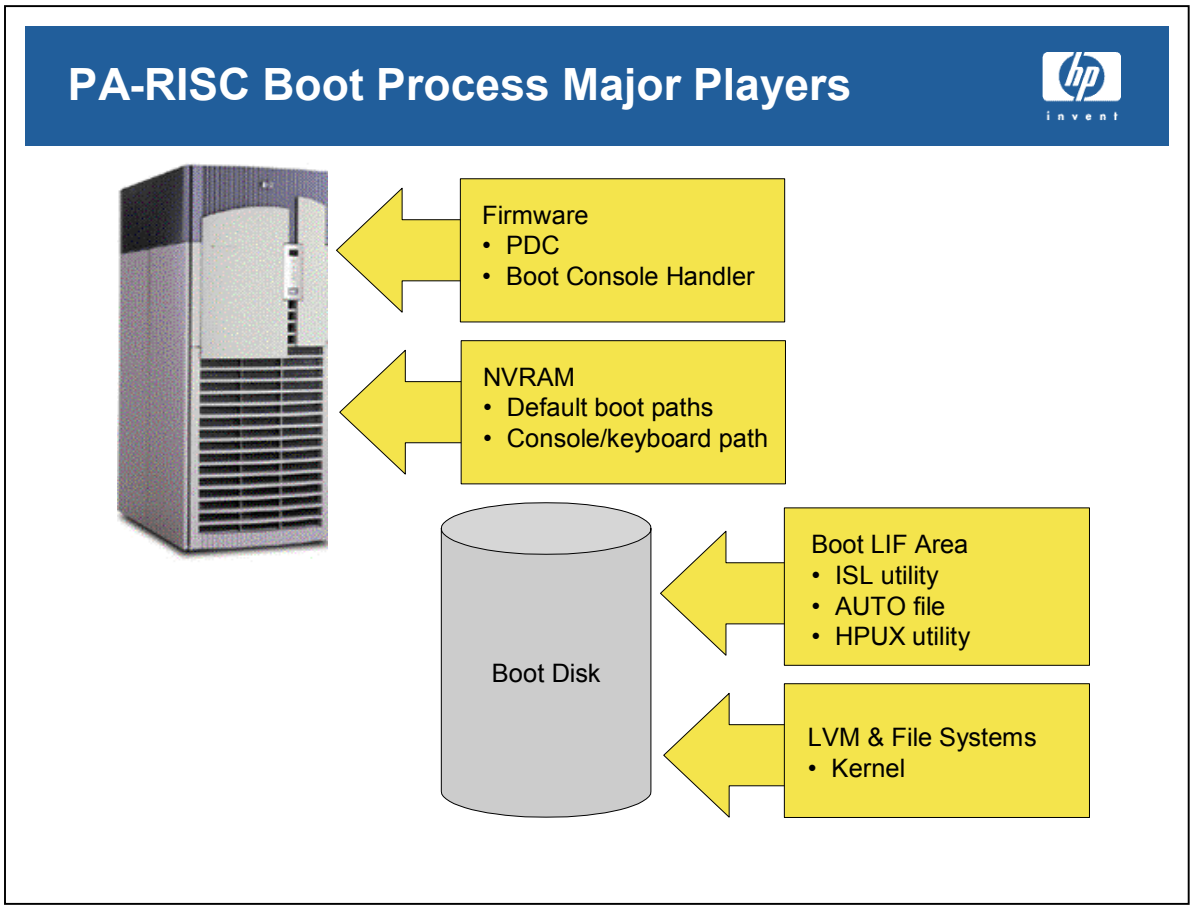
`shutdown` and `reboot` on Node and Virtual Partitions

When you execute the shutdown or reboot command on a node or virtual partition, the command only affects your current partition. Other partitions remain up and running.

Several additional **shutdown** and **reboot** options are required if you intend to shutdown a partition to reallocate partition resources. HP's Partition Management class (U5075S) discusses these options in detail.

© 2004 Hewlett-Packard Development Company, L.P.

16–2. SLIDE: PA-RISC Boot Process Major Players



Student Notes

After shutting down your system, you will eventually need to boot it back up again. The goal of the system boot process is to load the kernel executable from the boot disk into memory and start all the daemons and services necessary to bring the system back up to a fully functional state. There are several players involved in this process:

Firmware	Every PA-RISC SPU has a BootROM chip containing Power-on Self Test (POST), Processor Dependent Code (PDC), I/O Dependent Code (IODC), and Boot Console Handler" (PDC/BCH) firmware executables. These executables in memory in the early stages of the boot process to do a variety of hardware self-tests, and identify the system boot disk.
Non-Volatile RAM	The PDC/BCH consults Non-Volatile RAM (NVRAM) to determine the hardware paths of the primary boot disk, an alternate boot source, and the system console. The contents of NVRAM may be modified by the administrator via the BCH **`path`** command or the HP-UX **`setboot`** command..

© 2004 Hewlett-Packard Development Company, L.P.

Boot Area

Every system requires at least one boot disk. Each boot disk has a 2MB "Boot Area" containing the utilities needed to find and load the kernel. Files in the boot area are stored in a special "Logical Interchange Format" (LIF) that is not directly viewable with **`ls`**, **`cat`** or other regular UNIX commands. The primary files of interest in the LIF area are the Initial System /Program Loader (ISL/IPL), the AUTO file, and the HPUX utility. These will be discussed in more detail on the next slide.

LVM & File Systems:

The boot disk must also have a file system containing a kernel to boot. The kernel is typically stored in the **`/stand`** file system, with filename **`/stand/vmunix`**. Currently, the utilities used to load the kernel in memory are unable to read JFS file systems; thus the file system containing the kernel must be HFS.

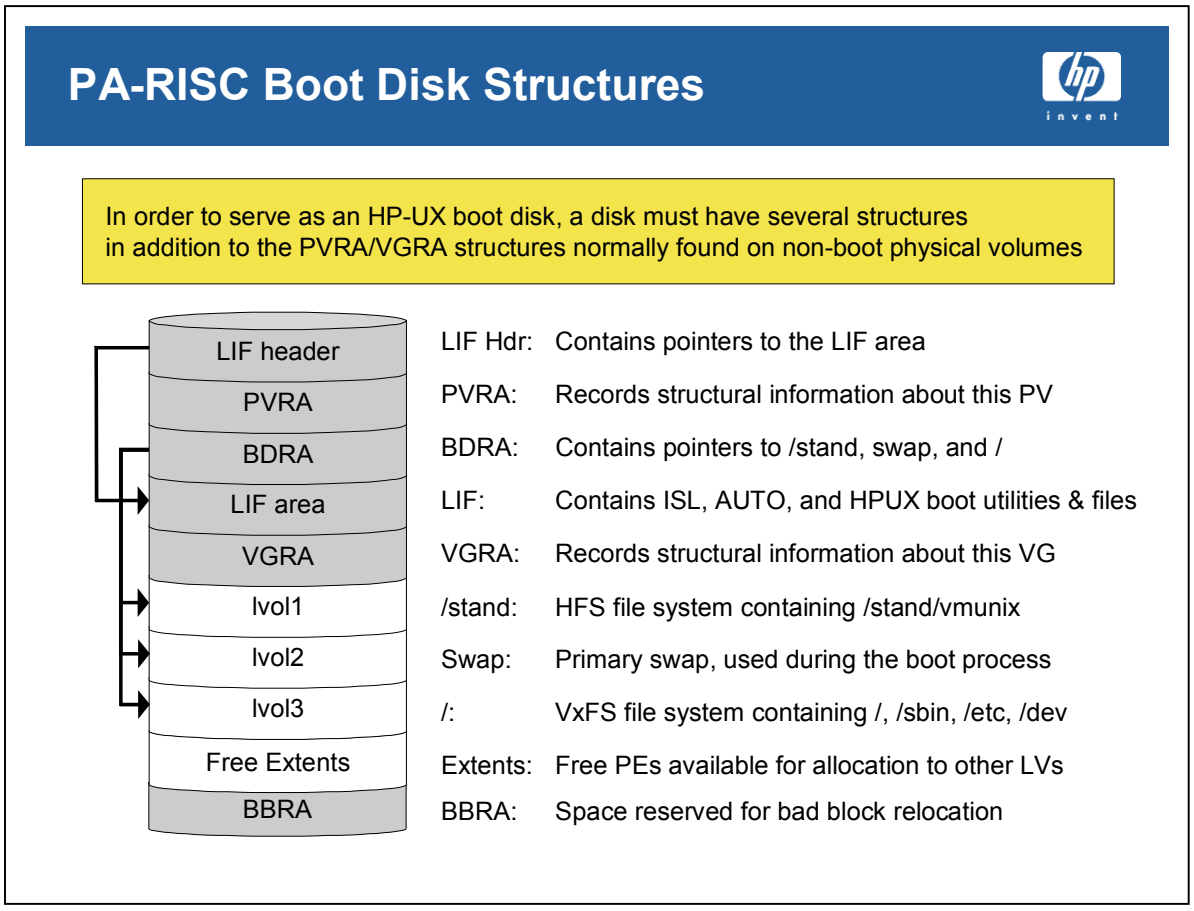
The boot disk also typically contains the "**`/`**" (root) file system. The "**`/`**" file system contains the**`/etc`**, **`/sbin`**, and **`/dev`** directories which are required in the early stages of the system startup process. The "**`/`**" file system may be either HFS or JFS.

Finally, the boot disk contains a primary swap area that is enabled and used early in the boot process.

These logical volumes will be discussed in more detail on the next slide.

© 2004 Hewlett-Packard Development Company, L.P.

16–3. SLIDE: PA-RISC Boot Disk Structure



Student Notes

LVM uses a variety of reserved areas on each physical volume to record structural information about the LVM environment. LVM boot disks require some additional special structures that are described in detail below.

LIF Header: At the top of every boot disk, there is a Logical Interchange Format (LIF) header structure. This structure contains information necessary to access and read the boot LIF area described below.

PVRA: The Physical Volume Reserved Area (PVRA) records the disk's PVID, and pointers to other important structures on the disk.

BDRA: The Boot Data Reserved Area (BDRA) contains pointers to the three key logical volumes used in the early stages of the boot process: **`/stand`**, **`/`**, and primary swap. The HPUX kernel loader is unable to read LVM extent maps; instead it determines the locations of these logical volumes using the pointers in the BDRA.

LIF Area: The Logical Interchange Format (LIF) area on a boot disk contains several boot utilities and files. The **`ISL`** executable is responsible for determining

© 2004 Hewlett-Packard Development Company, L.P.

which kernel to boot. It does this by reading the ASCII **AUTO** file, which is also stored in the LIF area. The **HPUX** utility is responsible for actually loading the kernel in memory. The LIF area files are stored in a special LIF file format that isn't readable using standard UNIX commands like **cp**, **ls**, and **cat**. The utilities required to manage the LIF area will be described later in the chapter.

VGRA: The Volume Group Reserved Area (VGRA) records structural information about the volume group that the disk belongs to. Recall that the volume group extent map is stored in this structure.

lvol1: **/dev/vg00/lvol1** contains the /stand HFS file system. This file system contains the **/stand/vmunix** kernel executable and other associated files.

lvol2: **/dev/vg00/lvol2** serves as primary swap, supplementing physical memory during the system boot process.

lvol3: **/dev/vg00/lvol3** contains the root VxFS file system. At a minimum, this file system contains the **/**, **/dev**, **/sbin**, and **/dev** directories.

Free Extents: Remaining space on the boot disk may be used to store other logical volumes, too. Though not required, many administrators choose to put **/usr**, **/var**, /tmp, and possibly even **/home** on the boot disk, too.

BBRA: The Bad Block Relocation Area (BBRA) is used to relocate data that would otherwise have been written to failed blocks in the extent allocation area on disk.

LVM Versus Non-LVM Boot Disks

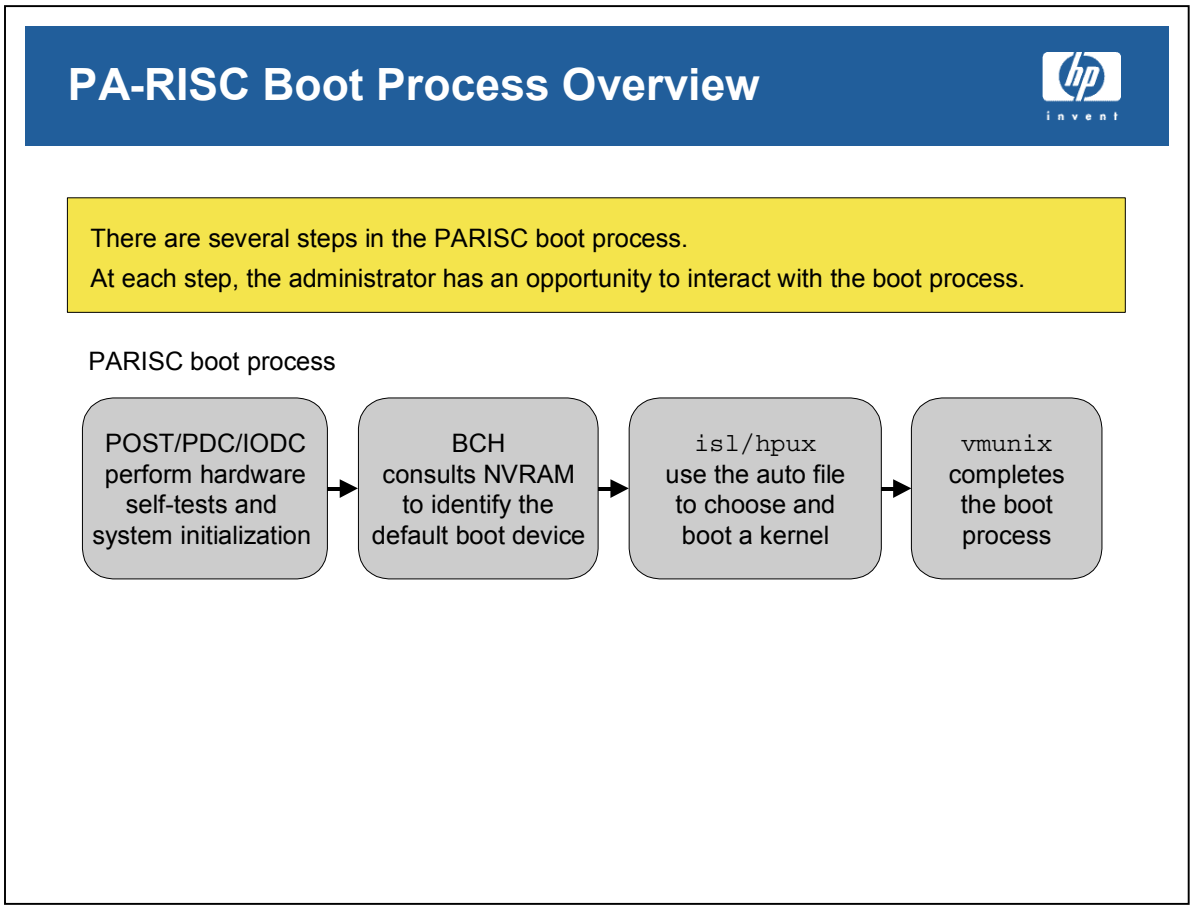
HP-UX boot disks may be configured using either LVM or the whole disk layout approach. The notes above describe the contents of an LVM boot disk.

Non-LVM boot disks have a 2MB boot LIF area; an HFS **/** file system that includes the **/**, **/stand**, **/sbin**, **/etc**, and **/dev** directories; and a primary swap section.

The layout of the boot disk is usually defined when you install HP-UX.

© 2004 Hewlett-Packard Development Company, L.P.

16–4. SLIDE: PA-RISC Boot Process Overview



Student Notes

Several steps are required to bring an HP-UX system to a fully functional state.

POST/PDC/IODC

Early in the boot process, the Power-On Self Test (POST), Processor Dependent Code (PDC), and I/O Dependent Code (IODC) perform a variety of hardware self-test and system initialization tasks.

BCH Chooses a Boot Disk

The Boot Console Handler (BCH) is responsible for determining the boot disk and system console hardware paths. These paths are configurable, and are recorded in Non-Volatile Random Access Memory (NVRAM). Next, the Initial System/Program Loader (ISL/IPL) is loaded in memory from the boot disk LIF area.

ISL/IPL and HPUX Choose and Boot a Kernel

The ISL/IPL consults the **`AUTO`** file in the boot disk LIF area to determine the default kernel boot string. After determining the kernel boot string, the ISL/IPL uses the HPUX utility to load the kernel in memory.

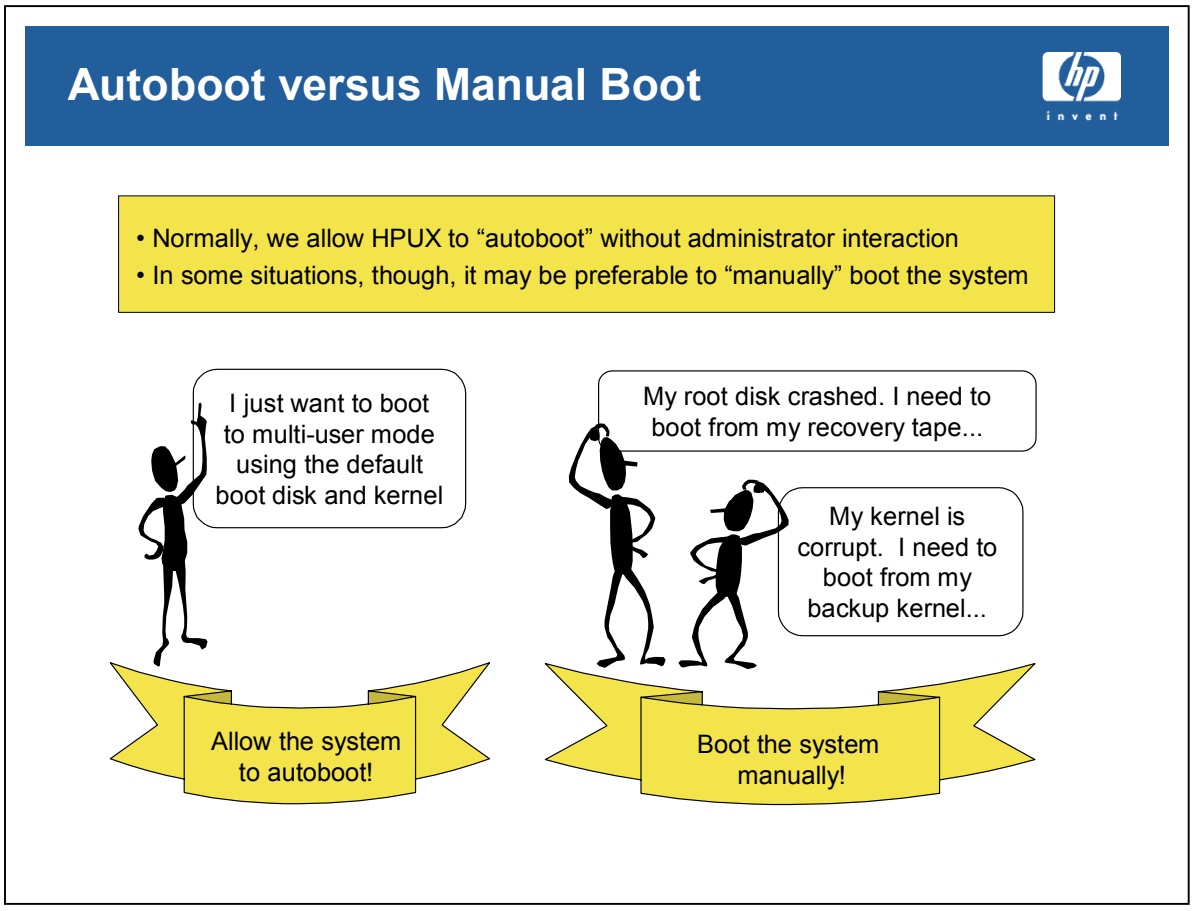
© 2004 Hewlett-Packard Development Company, L.P.

`vmunix` Brings the System to a Fully Functional State

The kernel then scans the hardware, mounts the root file system, and starts the **`init`** daemon. The **`init`** daemon starts the daemons and services necessary to bring the system up to multi-user mode.

© 2004 Hewlett-Packard Development Company, L.P.

16–5. SLIDE: Autoboot versus Manual Boot



Student Notes

Usually, the boot procedure described on the previous slide occurs without intervention from the System Administrator. The system boots from the primary boot disk to multi-user mode using the default kernel. This is known as an **autoboot**.

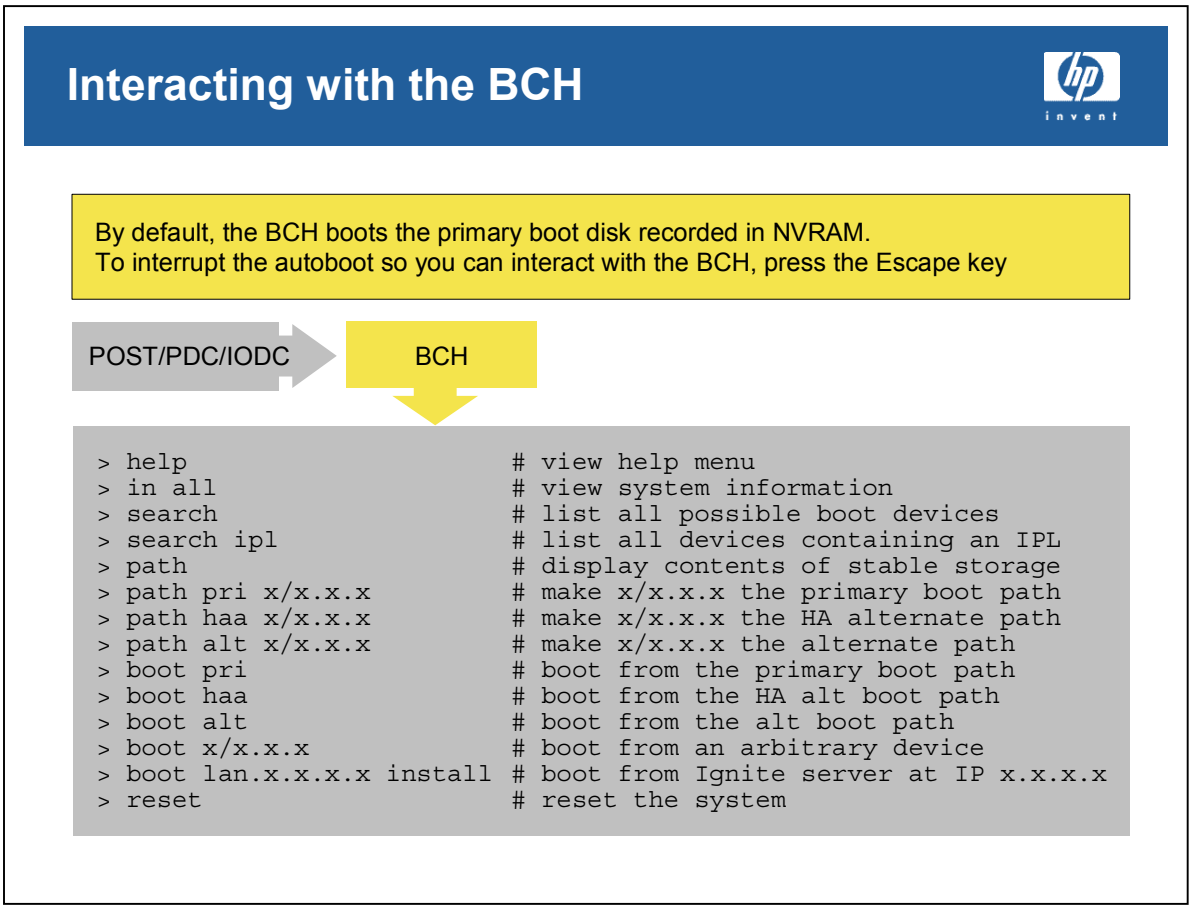
However, in some cases you may wish to override the autoboot process:

- To re-install the operating system from CD.
- To boot from a system recovery tape if the primary boot disk is corrupt.
- To boot from an alternate boot disk if the primary boot disk is corrupt.
- To boot from an alternate kernel if **`/stand/vmunix`** is corrupt.
- To boot to single-user mode to perform administrative tasks.
- To boot to single-user mode if the root password is lost.

The remainder of this chapter discusses the procedure used to manually boot your HP-UX system, and the functionality available at each step of the process.

© 2004 Hewlett-Packard Development Company, L.P.

16–6. SLIDE: Interacting with the BCH



Student Notes

Why Interact with the BCH?

After the boot process is initiated, the BCH checks stable storage to determine which disk to boot from, and continues the boot procedure without any further interaction with the administrator. However, interaction with the BCH may be required:

- To re-install the operating system from CD.
- To boot from a system recovery tape if the primary boot disk is corrupt.
- To boot from an alternate boot disk if the primary boot disk is corrupt.
- To boot from an alternate kernel if **`/stand/vmunix`** is corrupt.
- To boot to single-user mode to perform administrative tasks.
- To boot to single-user mode if the root password is lost.

© 2004 Hewlett-Packard Development Company, L.P.

Getting to the BCH

To interact with the BCH, press any key after initiating the boot process. The messages that appear on the screens that follow will vary somewhat from model to model. Newer models place you directly at a **`Main Menu>`** or **`>`** prompt, from which you can issue the BCH commands shown on the slide. Older models place you at an initial menu similar to the one shown below.

```
Device Selection          Device Path               Device Type
------------------------------------------------------------------

     P0                    scsi.6.0         HP      2213A
     P1                    scsi.5.0         HP      2213A
     P2                    scsi.3.0         HP      HP35480A
     P3                    scsi.2.0         HP      S6300.650A

     b)    Boot from specified device
     s)    Search for bootable devices
     a)    Enter Boot Administration mode
     x)    Exit and continue boot sequence
     ?)    Help

     Select from menu:b p0
```

At the **`Select from menu:`** prompt, you can boot from one of the devices listed by typing **`b p0`** as shown in the sample screen capture. Alternately, you can go to a **`Main Menu>`** prompt and issue any of the commands shown on the slide by typing **`a`** at the **`Select from menu:`** prompt.

Executing Commands at the BCH

The slide lists the commands most commonly used at the BCH. If you need more information about any of the BCH commands, you can always view a help screen:

```
Main Menu: Enter command or menu > help
```

The BCH **`in all`** (info all) command displays a detailed catalog of the system configuration. The sample output

```
Main Menu: Enter command or menu > in all
```

Recall that the primary purpose of the BCH is to choose a boot disk. You can search for potential boot devices on the system via the BCH **`search`** command.

```
Main Menu: Enter command or menu > search
```

The **`search`** command generates a list of all the disk and tape drives on your system. Most of the devices listed probably aren't really bootable. To list only devices that have boot areas containing an ISL/IPL, add an additional argument on the search command:

```
Main Menu: Enter command or menu > search ipl
```


© 2004 Hewlett-Packard Development Company, L.P.

Usually, the BCH chooses a boot disk by consulting NVRAM. Every system has both a primary (PRI) boot path, which the BCH boots from by default. Some (but not all) systems support a high availability alternate (HAA) disk, too, which may be configured to boot automatically if the primary boot disk fails. All systems support an alternate (ALT) boot path, too. If your system supports an HAA boot path, then the alternate boot path will likely reference the CDROM or tape drive. If your system doesn't support an HAA boot path, the alternate path may identify a backup boot disk. Use the BCH **`path`** command to see what primary and alternate boot paths are currently defined.

```
Main Menu: Enter command or menu >  path
```

You may wish to change the default boot paths if you configure a mirrored boot disk, or if you change the hardware configuration of your system. The primary boot path can be changed with the **`path`** command. Remember that you can view a list of available hardware paths via the **`search`** command.

```
Main Menu: Enter command or menu > path pri 0/0/1/1.6.0
```

The HAA and alternate boot paths may be changed, too:

```
Main Menu: Enter command or menu > path haa 0/0/1/2.4.0
Main Menu: Enter command or menu > path alt 0/0/1/2.3.0
```

To actually boot the system, use the **`boot`** command. You may boot using the primary (default), HAA, or alternate boot paths, or choose an entirely different path if, for instance, you wish to boot from the recovery tape.

```
Main Menu: Enter command or menu > boot pri
Main Menu: Enter command or menu > boot haa

Main Menu: Enter command or menu > boot alt
Main Menu: Enter command or menu > boot 0/0/1/2.0.0
```

To boot from an Ignite-UX network install server, use the **`boot lan install`** command. If you wish to boot from a specific server, specify the server's IP address following the **`lan`** keyword.

```
Main Menu: Enter command or menu > boot lan.10.1.1.1 install
```

At any time, you can restart the system by executing the BCH **`reset`** command.

```
Main Menu: Enter command or menu > reset
```

The slide only shows the most commonly used BCH commands. Others are shown in the table below. Note however, that not all commands are available on all models. See the BCH help screens for more information about any of these.

© 2004 Hewlett-Packard Development Company, L.P.

Command	Action
auto	Display state of autoboot/autosearch flags.
autosearch	Set state of autosearch flag.
autoboot	Set state of autoboot flag.
boot	Boot from primary/alternate path or specified device.
date	Read/Set the Real-Time Clock.
diagnostic	Show (**on**)/Conceal (**off**) diagnostic messages during boot.
exit	Return to previous menu.
help *item*	Display Help information for *item*.
info	Display boot/revision information.
information	Display boot/revision information.
lan_addr	Display LAN station address.
lanaddress	Display LAN station address.
monitor	Display/Set monitor type.
path	Display/Modify path information.
reset	Reset the system.
search	Search for boot device.
secure	Display/set secure boot mode.
show	Display the results of the previous search.

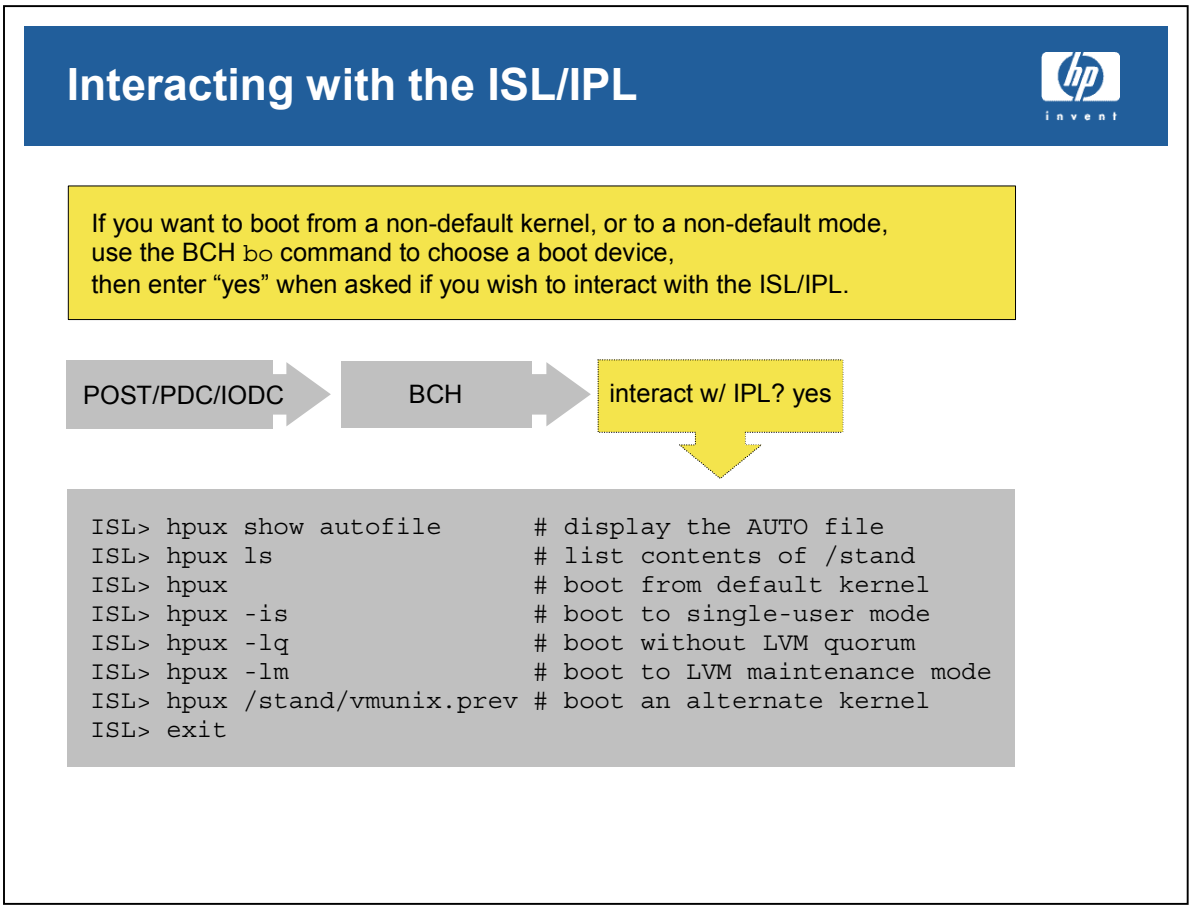
Viewing and Modifying NVRAM from Multi-User Mode

The slide above notes that you can view and change the contents of stable storage via the **path** command at the BCH prompt. However, you can also view or modify the contents of stable storage from multi-user mode using the **setboot** command:

```
# setboot                    Display contents of stable storage.
# setboot -a x/x.x.x         Make x/x.x.x the alternate boot device .
# setboot -p x/x.x.x         Make x/x.x.x the primary boot device.
```


© 2004 Hewlett-Packard Development Company, L.P.

16–6. SLIDE: Interacting with the ISL/IPL



Student Notes

Why Interact with the ISL/IPL?

After the BCH identifies a boot disk, the Initial System/Program Loader (ISL or IPL) is responsible for identifying a kernel to load, and for calling the HPUX kernel loader. Usually, the ISL reads the AUTO file and proceeds without any interaction with the administrator.

Interaction may be required, however,

- If the default kernel (**`/stand/vmunix`**) is unbootable.
- If you wish to boot to single-user mode rather than multi-user mode to perform maintenance tasks.
- If you forget the root password. (Booting to single-user mode logs you in as root at the console without prompting for a password.)

© 2004 Hewlett-Packard Development Company, L.P.

Getting to the ISL/IPL

In order to get to the **isl**, you will first need to initiate a manual boot as described on the previous slide. Be sure to include the **isl** argument on the end of the **boot** command:

```
Main Menu: Enter command or menu > boot pri isl  or
Main Menu: Enter command or menu > boot alt isl
```

Some models then ask if you wish to interact with the ISL/IPL. If asked, answer **yes**. Momentarily, this should place you at an **ISL>** prompt.

```
Interact with IPL (Y, N, or Cancel)?> y
```

Executing Commands with ISL

Although there are a number of diagnostic utilities available at the ISL, the most common command used at the ISL prompt is **hpux**. The slide shows some of the most commonly used options on **hpux**.

Left to its own devices, the ISL reads the **AUTO** file in your boot disk's boot LIF area to determine which kernel to boot. To view the default boot string in your **AUTO** file, use the **hpux show** command. If the **AUTO** file doesn't specify which kernel to boot, **hpux** boots using the default kernel, **/stand/vmunix**.

```
ISL> hpux show autofile
```

After determining which kernel you wish to boot from, you can load the chosen kernel into memory and continue the boot process by executing the **hpux** command. When executed without options or arguments, **hpux** simply loads the default **/stand/vmunix** in memory and boots the system to multi-user mode.

```
ISL> hpux
```

If your last system boot failed or hung, you may wish to boot the system to single-user mode to troubleshoot the problem. This proves especially useful when an autoboot fails because of errors in **/etc/passwd**, **/etc/fstab**, or other critical configuration files. This may also be useful if you have forgotten the root password; booting to single-user mode automatically logs root in on the system console without prompting for a password! Use the **-is** option to boot the system to single-user mode:

```
ISL> hpux -is
```

By default, an HP-UX system won't boot unless 51% of the disks in **vg00** are available. This is known as the LVM quorum restriction. You can, however, override this restriction via the **hpux -lq** command. This proves useful if you have a mirrored boot disk, and the primary boot disk is corrupted. You can still boot from the remaining mirror, but only if you execute **hpux -lq**. Be sure, however, to repair or replace the corrupted mirror as soon as possible!

```
ISL> hpux -lq
```


© 2004 Hewlett-Packard Development Company, L.P.

You can also boot into "LVM maintenance mode" by typing **hpux -lm**. You may need to boot to maintenance mode at some point to modify **vg00**'s configuration, extend the root file system, or modify primary swap. It is very easy to corrupt your LVM configuration in maintenance mode; use it with care!

```
ISL> hpux -lm
```

If you reconfigure your default kernel with an incompatible set of kernel parameters that leave your system unbootable, you may choose to boot from a backup kernel as shown at the bottom of the slide:

```
ISL> hpux /stand/vmunix.prev
```

The slide only discussed the most common commands executed at the ISL prompt. The tables below list some of the other commands and utilities available at the ISL. Additional information is available in the **isl(1M)**, **hpux(1M)**, and **boot(1M)** man pages.

help or **?**	List available ISL commands.
ls or **listf**	List available other utilities available in the LIF area.
autoboot on/off	Enable or disable the **autoboot** sequence parameter in stable storage.
autosearch on/off	Enable or disable the **autosearch** parameter in stable storage.
display	Display the primary boot, alternate boot, and console paths.
primpath 8/12.6.0	Set the primary boot path in stable storage.
altpath 8/12/5.0	Set the alternate boot path in stable storage.
conspath 8/24	Set the console hardware path in stable storage.
lsautofl	Display the contents of the **auto** file in the LIF area.

hpux	Boot to multi-user mode using **/stand/vmunix**.
hpux /stand/vmunix.prev	Boot to multi-user mode using a named kernel.
hpux ls	List the contents of **/stand**.
hpux show autofile	Display contents of **autofile**.
hpux set autofile "hpux"	Change contents of the **autofile**.
hpux -v	Display the version number of the **hpux** utility.
hpux -is	Boot to single-user mode.
hpux -lm	Boot to LVM maintenance mode (may be needed if boot disk is corrupted).

Viewing the AUTO File and Boot Area from Multi-User Mode

The slide above explains how to view the contents of the boot area and AUTO file from the ISL prompt. However, the boot area may also be viewed using the **lifls** and **lifcp** commands as shown below:

```
# lifls /dev/rdsk/c0t6d0          List the contents of a disk's boot area.
```


© 2004 Hewlett-Packard Development Company, L.P.

```
# lifcp /dev/rdsk/c0t6d0:AUTO -
```
View the contents of a disk's AUTO file.

The **lifcp** command's primary purpose is to copy files to and from the boot LIF area. However, putting a "**-** " in the destination field causes **lifcp** to copy the contents of a LIF area file to standard output.

© 2004 Hewlett-Packard Development Company, L.P.

16–7. LAB: Shutting Down and Rebooting Your System

Directions

Carefully follow the instructions below.

Part 1: Preliminary Steps

A portion of this lab requires you to interact with the ISL and boot menus, which can only be accomplished via a console login. If you are using remote lab equipment, access your system's console interface via the GSP/MP.

Part 2: System Shutdown

1. Currently, your system should be in multi-user mode. Note which file systems are mounted, and how many processes are running.

2. Shut down to single-user mode. Then check to see what processes and mounted file systems remain. What differences do you see between single- and multi-user modes?

3. From single-user mode, take your machine down to the halt state. What can you do in the halt state? Why might it be necessary to take your system down to the halt state?

   ```
   # reboot -h
   ```

4. Use the MP **`rs`** command or the system power switch to reset your server before proceeding to the next part of the lab. Immediately begin pressing the **`[Escape]`** key to interrupt the boot process.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Booting the System

1. If you didn't already do so in the previous part of the lab, reboot your system and interrupt the autoboot process. Proceed to the PDC/BCH main menu.

2. Search for possible boot devices on your system. How many SCSI devices does your system have? If a system isn't booting properly, this is one way of determining if all your SCSI devices are properly connected and powered on.

3. Now search for disks that contain an IPL that you can boot from. How many of your disks appear to be bootable?

4. Display your primary and alternate boot paths. Which disk is defined as your primary boot device?

5. Boot to the ISL from your primary boot disk. Be sure to specify that you want to interact with the ISL/IPL.

© 2004 Hewlett-Packard Development Company, L.P.

6. At the ISL prompt, get a list of valid ISL commands.

7. Interacting with the ISL may be useful if your primary kernel is corrupted and you have to boot from a backup kernel. Do you currently have a backup kernel in your /stand directory? (Kernel file names usually begin with vmunix.)

8. By default, the system boots to multi-user mode using **`/stand/vmunix`**. Boot to single-user mode from the default kernel.

9. Did the system prompt you for a password when you were brought to single-user mode? When might this be helpful?

10. Reboot your system again, but this time let it boot unattended using the default boot disk and kernel.

© 2004 Hewlett-Packard Development Company, L.P.

Part 4: (Optional) Mirroring the Boot Disk

In order to improve system stability, many administrators choose to mirror the system boot disk. Doing so guarantees that you system continues running, even if one of the boot disks or SCSI interface cards on your system is damaged. If time permits, follow the procedure below to mirror your student system's boot disk:

1. Use **ioscan** and **diskinfo** to find a disk large enough to mirror the existing boot disk.

   ```
   # ioscan -funC disk
   # diskinfo /dev/rdsk/boot
   # diskinfo /dev/rdsk/bootmirror
   ```

2. **pvcreate** the new disk and add it to the root volume group. The **-B** option on **pvcreate** reserves space for a "boot area" on the new boot disk. Files in the boot area are stored in a special format and may not be viewed or modified using the standard UNIX file system commands.

   ```
   # pvcreate -fB  /dev/rdsk/bootmirror
   # vgextend vg00 /dev/dsk/bootmirror
   ```

3. The new boot disk's boot area must have copies of the ISL and HP-UX utilities. These utilities may be created in the boot area using the **mkboot** command.

   ```
   # mkboot /dev/rdsk/bootmirror
   ```

4. Recall that the AUTO file is a special structure in the boot area at the top of each boot disk that determines how the kernel is loaded during the boot process. By default, an HP-UX system won't boot unless LVM "quorum" is met. Quorum is met when 51% of the disks in a volume group are present. If the boot disk is mirrored, we want the system to boot even if one of the mirrors in vg00 is unavailable. The two commands below modify the AUTO file on both boot disks to allow the system to boot even if vg00 doesn't meet the quorum restrictions during the boot process.

   ```
   # mkboot –a "hpux –lq" /dev/rdsk/boot
   # mkboot –a "hpux –lq" /dev/rdsk/bootmirror
   ```

5. Mirror all of the boot disk logical volumes to the new boot disk. Note that the names of the logical volumes on *your* boot disk may be different. Make sure you mirror all of the boot disk logical volumes! It may take several minutes to synchronize each mirror.

   ```
   # lvextend -m 1 /dev/vg00/lvol1 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol2 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol3 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol4 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol5 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol6 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol7 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol8 /dev/dsk/bootmirror
   ```


© 2004 Hewlett-Packard Development Company, L.P.

6. Update stable storage so your new mirror is listed as the alternate boot disk. Stable storage may be modified from the PDC menu, but it may also be modified from multi-user mode using the command shown below:

   ```
   # setboot -a 2/0/1.5.0    use your mirrored disk's hw path here
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 5: (Optional) Testing the New Boot Mirror

After configuring the new mirrored boot disk, test it!

1. Verify that the new PV was added to the volume group properly.

   ```
   # vgdisplay -v vg00
   ```

2. Check to see that the logical volumes are mirrored and current.

   ```
   # lvdisplay -v /dev/vg00/lvol1
   # lvdisplay -v /dev/vg00/lvol2
   # lvdisplay -v /dev/vg00/lvol3
   ```

3. List the files in the mirror's boot area, then display the AUTO files from both disks using the commands shown below.

   ```
   # lifls /dev/rdsk/bootmirror
   # lifcp /dev/rdsk/bootmirror:AUTO -
   # lifcp /dev/rdsk/primaryboot:AUTO -
   ```

4. Now for the real test: try to boot from your mirrored root disk! Do a shutdown, then interrupt the boot sequence. Use the PDC **`search`** command to see if your new boot disk is recognized. Assuming it is, go ahead and boot from it!

   ```
   # shutdown -ry 0
   [esc]
   Main Menu: Enter command or menu > search ipl
   Main Menu: Enter command or menu > boot alt
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Module 17 — Booting Integrity Systems

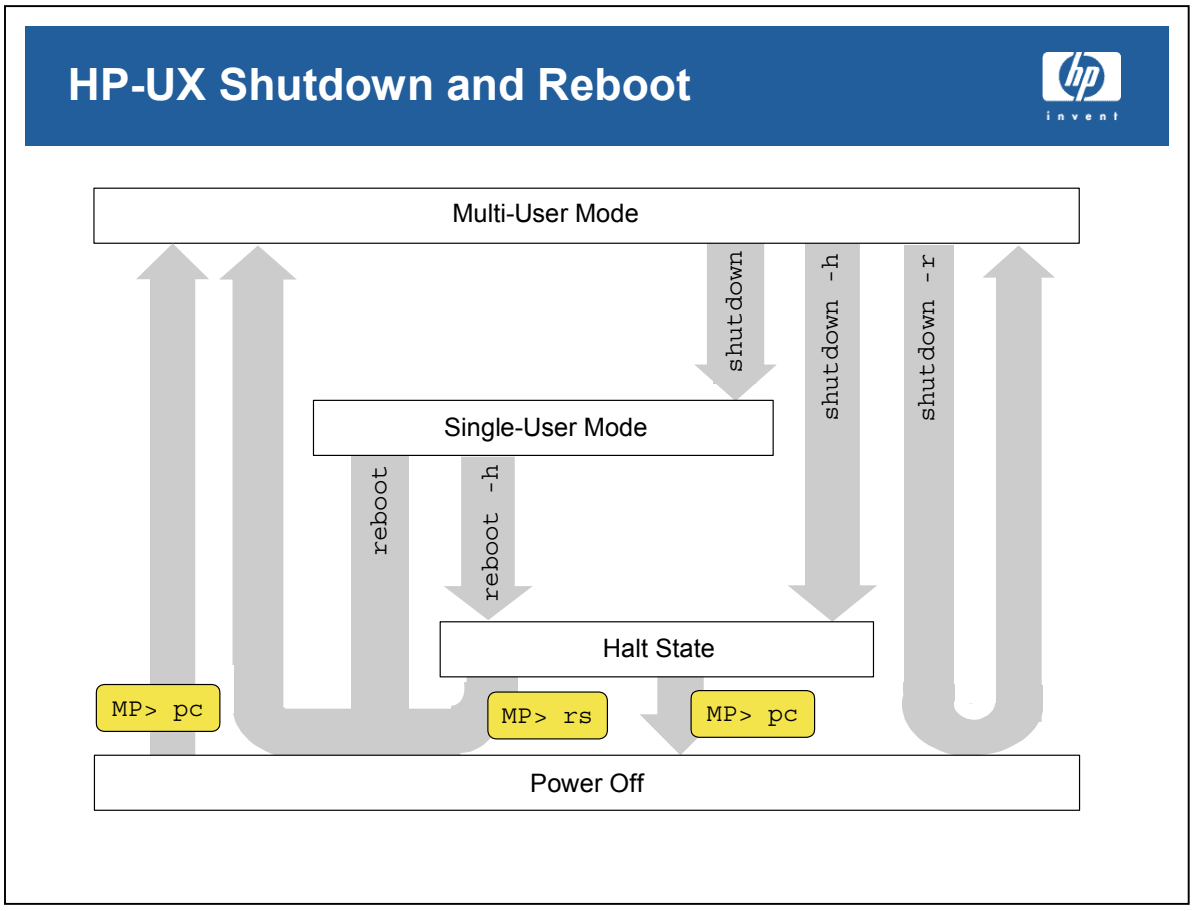
Objectives

Upon completion of this module, you will be able to do the following:

- Properly shut down an HP-UX system using **`shutdown`** and **`reboot`**.
- Describe how the EFI and **`hpux.efi`** utility load the kernel in memory.
- View system information from the BCH.
- Boot from the primary, alternate, or HA alternate boot device.
- Boot from a system recovery tape.
- Boot from an install CD or Ignite install server.
- Boot from an alternate kernel.
- Boot to single-user mode to perform system maintenance.

© 2004 Hewlett-Packard Development Company, L.P.

17–1. SLIDE: HP-UX Shutdown and Reboot



Student Notes

An HP-UX system will always be in one of several operation states. The system state determines which system services are available. Some of the system administrative tasks described in this course require you to change your current operation state. This chapter will present an overview of the functionality available in each state.

Most of the time, your system will be in *multi-user mode*. In multi-user mode, users can log in; file systems are mounted; and most system services and daemons are available.

There are occasions, however, when you may need to take your system down to *single-user* mode to perform administrative tasks. In single-user mode, users cannot login, non-critical file systems are unmounted, and non-critical system daemons are shut down. Moving to single-user mode brings your system to a quiet state, from which you can do system backups, run **`fsck`**, or perform other system administrative tasks, without interference from user processes.

Bringing your system to a *halt state* unmounts all file systems, and kills all processes. Always halt HP-UX before powering-off your server or workstation.

© 2004 Hewlett-Packard Development Company, L.P.

Taking your system to the power-off state terminates power to all system hardware components, except the Management Processor card.

Several commands may be used to move between states on an HP-UX system:

- From *multi-user* mode, use the **shutdown** command to reboot, halt, or shutdown to single-user mode.
- From *single-user* mode, use the **reboot** command to reboot or halt the system.
- From the *halt state*, use the **rs** (reset) command from the Management Processor Command Menu to reset and boot the system. On workstations, push the reset button on the workstation chassis. On cell-based systems, you can also use the **bo** (boot) command from the Management Processor to boot the system.
- From a *power-off state*, use the **pc** (power control) command from the Management Processor Command Menu to power on the system or simply press the power button.
- On a *hung system*, it may be necessary to use the **rs** or **tc** commands from the Management Processor Command Menu to reset and boot the system. On workstations, push the reset button on the workstation chassis.

Changing State from Multi-user Mode with `shutdown`

Executing the **shutdown** command terminates system activities in an orderly and consistent manner. The **shutdown** command performs the following tasks:

- Prompts the administrator for a broadcast message to send to all users.
- Broadcasts the warning message to all user terminal sessions.
- Grants a 60 second (by default) grace period for users to log out.
- Kills all user logins.
- Gracefully shuts down all non-critical processes using **/sbin/rc*.d/K*** kill scripts.
- Unmounts all non-critical file systems.

Depending on the option specified, **shutdown** will either leave the system in single-user mode (no options), the halt state (if **-h** was specified), or initiate a reboot (if **-r** was specified).

Common **shutdown** options:

`# shutdown -y 600`	*# shutdown to single-user mode in 600 seconds."-y" (yes) option prevents shutdown from requesting confirmation before proceeding.*
`# shutdown -hy 600`	*# shutdown to a halt state in 600 seconds."-y" (yes) option prevents shutdown from requesting confirmation before proceeding.*

© 2004 Hewlett-Packard Development Company, L.P.

```
# shutdown -ry 600        # reboot in 600 seconds without requesting confirmation.

# shutdown -ry 0          # reboot immediately without requesting confirmation.
```

Changing State from Single-user Mode with `reboot`

The **`reboot`** command uses the SIG_KILL signal to kill running processes. This takes the system down quickly, but can cause problems for applications and file systems. The **`shutdown`** command shuts down applications and processes more gracefully, and thus is the preferred method for halting or rebooting the system from multi-user mode. Reboot may be used if:

- The system is already in single-user mode.
- You need to bring the system down very quickly.

```
# reboot –h     # shutdown to a halt state (only use this from single-user mode).
# reboot        # reboot
```

Changing State from the Halt State with the `rs` MP Command

After halting a system or partition with either the **`reboot -h`** or **`shutdown -h`** command, you can initiate the boot process via the **`rs`** command on the MP Command Menu. Although the **`rs`** command may be executed on a running system, too, doing so may cause corruption.

```
MP> cm
MP:CM> rs

Execution of this command irrecoverably halts all system
processing and I/O activity and restarts the computer system.

Type Y to confirm your intention to restart the system: (Y/[N]) y
-> SPU hardware was successfully issued a reset.
```

On cell-based systems, you can also use the **`bo`** (boot) command from the Management Processor to boot the system.

Changing State via the `pc` MP Command

Once your system reaches the halt state, you can safely power-off the system hardware by either pressing the power switch on the system cabinet, or, on servers, by executing the **`pc`** command from the MP Command Menu. The output below was taken from an rx2600 server. The output on other models may be slightly different.

```
MP> cm
MP:CM> pc

Current System Power State: On      Power Switch State: On
   ON  - Turn the power On
   OFF - Turn the power Off
```


© 2004 Hewlett-Packard Development Company, L.P.

```
Enter your choice (ON/OFF) or [Q] to quit: off
Enter delay (in minutes  0-6000) to power Off (CR for no delay):
System will be powered Off in 0 minutes.
   Confirm? (Y/[N]): y
   -> System has been powered Off
```

If your system has been powered off for some reason, you can power it back on again by pressing the power button, or, on servers, by executing the **pc** command from the MP Command Menu (output on your system may be slightly different from the output shown below).

```
MP> cm
MP:CM> pc

Current System Power State: Off      Power Switch State: On
   ON  - Turn the power On
   OFF - Turn the power Off

Enter your choice (ON/OFF) or [Q] to quit: on
Enter delay (in minutes  0-6000) to power On (CR for no delay):
System will be powered On in 0 minutes.
  Confirm? (Y/[N]): y
  -> System has been powered On
```

Changing State on a Hung System with the `tc` and `rs` MP Commands

On rare occasions, a bug in an application or the OS may leave your system in a "hung" state. In these situations, it may be necessary to use the MP **tc** or **rs** commands to reboot. Use these commands with care, as they may cause corruption.

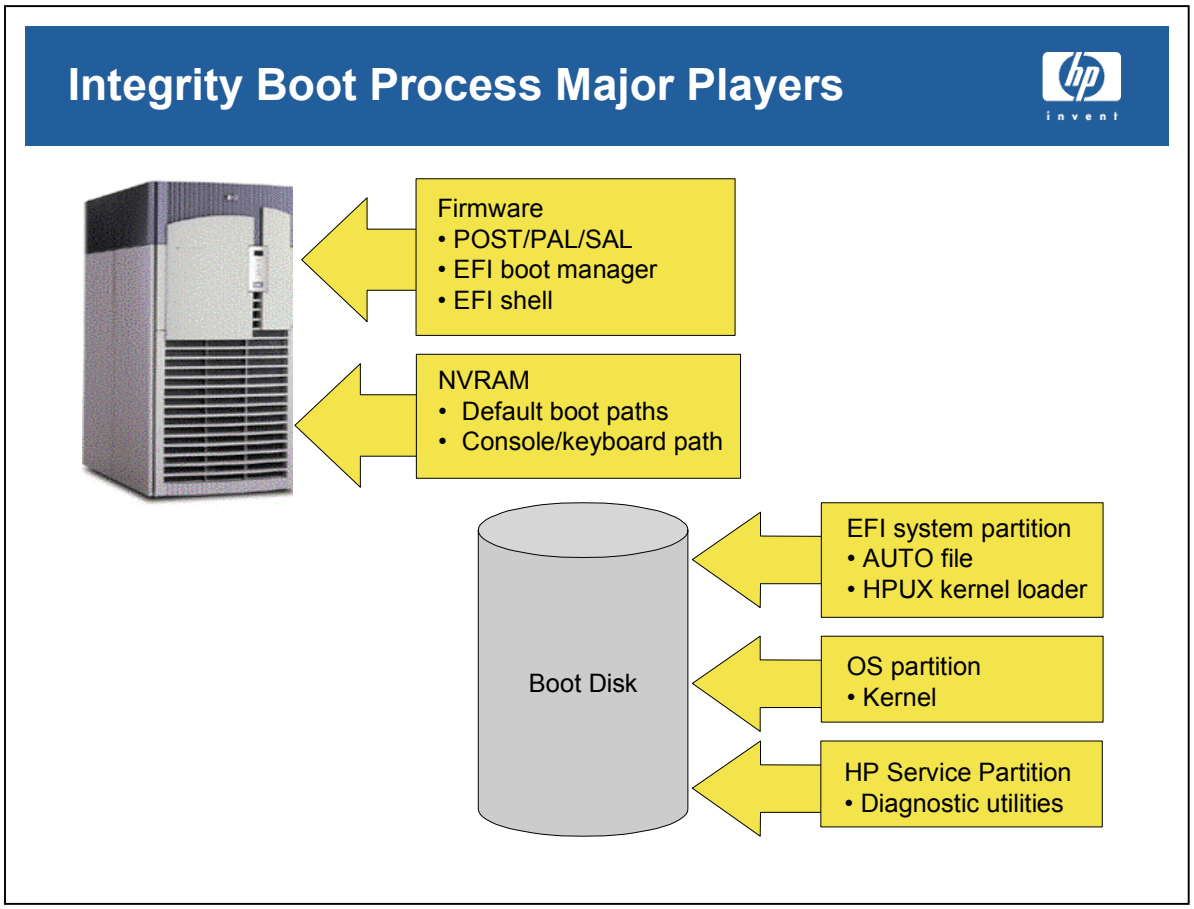
`shutdown` and `reboot` on Node and Virtual Partitions

When you execute the shutdown or reboot command on a node or virtual partition, the command only affects your current partition. Other partitions remain up and running.

Several additional **shutdown** and **reboot** options are required if you intend to shutdown a partition to reallocate partition resources. HP's Partition Management class (U5075S) discusses these options in detail.

© 2004 Hewlett-Packard Development Company, L.P.

17–2. SLIDE: Integrity Boot Process Major Players



Student Notes

After shutting down your system, you will eventually need to boot it back up again. The goal of the system boot process is to load the kernel executable from the boot disk into memory and start all the daemons and services necessary to bring the system back up to a fully functional state. Several major players are involved in this process:

Firmware: The Power-On Self Test (POST), Processor Abstraction Layer (PAL), and System Abstraction Layer (SAL) Firmware components are responsible for performing system and platform initialization and self-tests in the earliest stages of the boot process.

The EFI boot manager is the utility responsible for choosing a boot device. The boot manager includes a menu-based interface that allows the administrator to interact with the boot process. The EFI shell provides a more flexible command-line interface to the EFI firmware.

NVRAM: The EFI boot manager consults Non-Volatile RAM (NVRAM) to determine the hardware paths of the primary boot disk, alternate boot sources, and the system console. The contents of NVRAM may be

© 2004 Hewlett-Packard Development Company, L.P.

modified by the administrator via the EFI shell **`bcfg`** command or the HP-UX **`setboot`** command.

Boot Disk: The boot disk itself is subdivided into three EFI disk partitions.

The EFI system partition contains several utilities and files that are used to find and load an appropriate kernel in memory, including the **`hpux.efi`** utility and the **`auto`** configuration file.

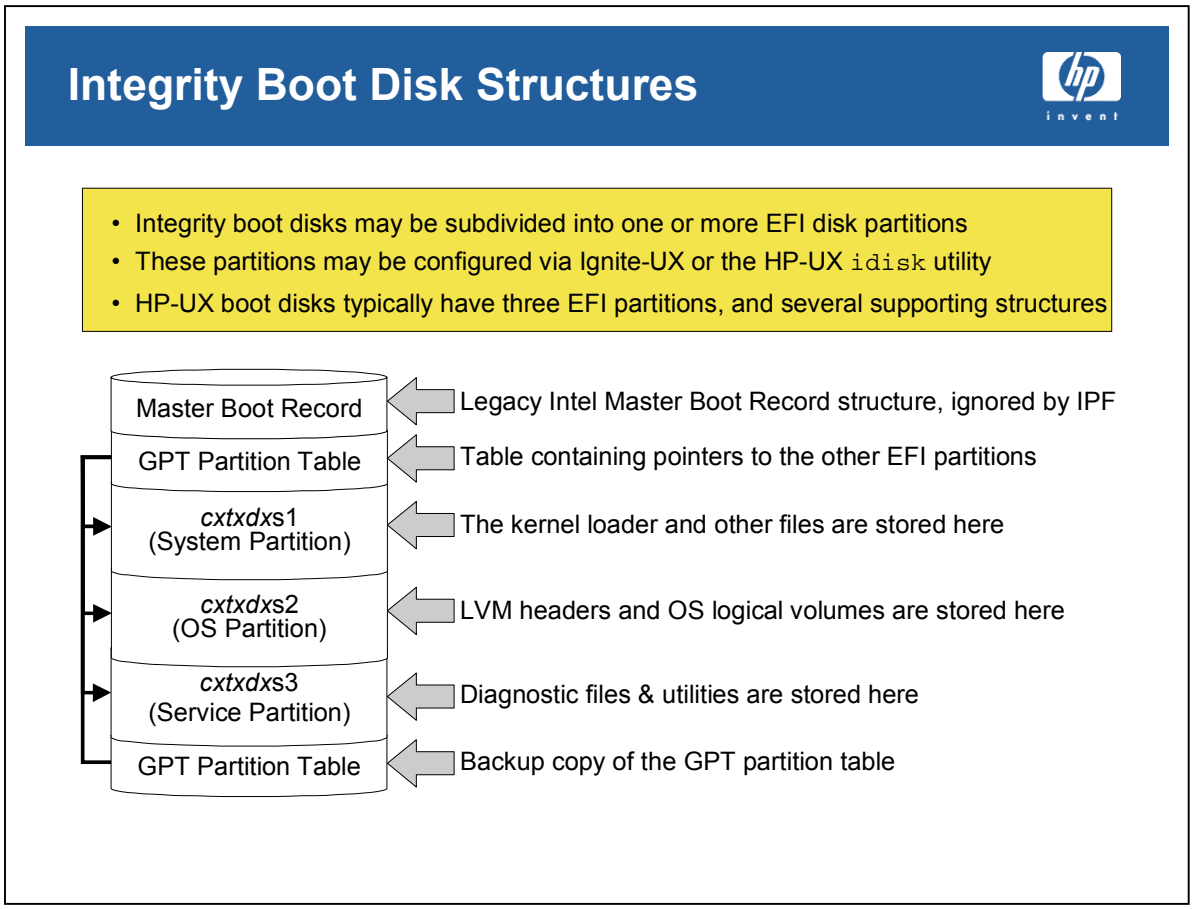
The OS partition on the boot disk contains LVM structures, the kernel and several file systems that play a critical role in the boot process.

The HP Service Partition (HPSP) contains diagnostic tools that may be used to diagnose hardware problems on an unbootable system.

Each of these partitions will be discussed in greater detail later over the next few pages.

© 2004 Hewlett-Packard Development Company, L.P.

17–3. SLIDE: Integrity Boot Disk Partitions



Student Notes

POST, PAL, SAL, and EFI firmware all reside in memory. However, other important players in the boot process such as the **`hpux.efi`** OS loader and the **`/stand/vmunix`** kernel executable, reside on the boot disk. The next few slides discuss the layout and structure of the boot disk in detail.

Integrity boot disks are typically subdivided into one or more EFI disk partitions. The EFI partitions on an HP-UX boot disk are initially created by Ignite-UX, but can also be manually viewed and managed via the new **`idisk`** utility.

NOTE:	EFI disk partitions are not related to HP-UX node partitions and virtual partitions. nPars and vPars subdivide a system's resources into independent OS instances. An EFI partition is simply a portion of disk space.

HP-UX boot disks typically have three EFI partitions, and several supporting structures:

MBR: The Master Boot Record (MBR) is located at the top of the disk. This is a legacy Intel structure that is ignored by the EFI firmware. Including an MBR at the top of the disk, however, makes it possible to

© 2004 Hewlett-Packard Development Company, L.P.

configure media that is readable on both Itanium and legacy Intel 32-bit systems.

GPT: Every EFI partition is assigned a Globally Unique IDentifier (GUID). The partition GUID's and locations are recorded in the EFI GUID Partition Table (GPT). This is a critical structure that is replicated at both the top and bottom of the disk.

EFI System Partition: At a minimum, the EFI system partition contains the OS loader that is responsible for loading the OS in memory during the boot process. On HP-UX boot disks, this OS loader is called **`\efi\hpux\hpux.efi`**. The **`\efi\hpux\auto`** file, which stores the system boot string, and several troubleshooting utilities are also stored in the system partition on HP-UX systems. Other OSes require different files in the system partition.

On HP-UX machines, the system partition is represented by device file **`/dev/rdsk/`*`cxtxdx`*`s1`**.

The EFI partition uses a file system structure which is similar to the FAT32 file system structure traditionally used in the Microsoft Windows Operating System. HP-UX includes several special commands such as **`efi_ls`**, **`efi_cp`**, and **`efi_rm`** that allow administrators to view and manage files in these EFI file systems. See the **`man`** pages for details.

OS Partition: The OS partition contains an encapsulated LVM boot physical volume, including all of the structures one would expect to see on an LVM boot disk: a PVRA, VGRA, BDRA, BBRA, and physical extents allocated to the logical volumes in vg00. In HP-UX, this partition is represented by device file **`/dev/rdsk/`*`cxtxdx`*`s2`**.

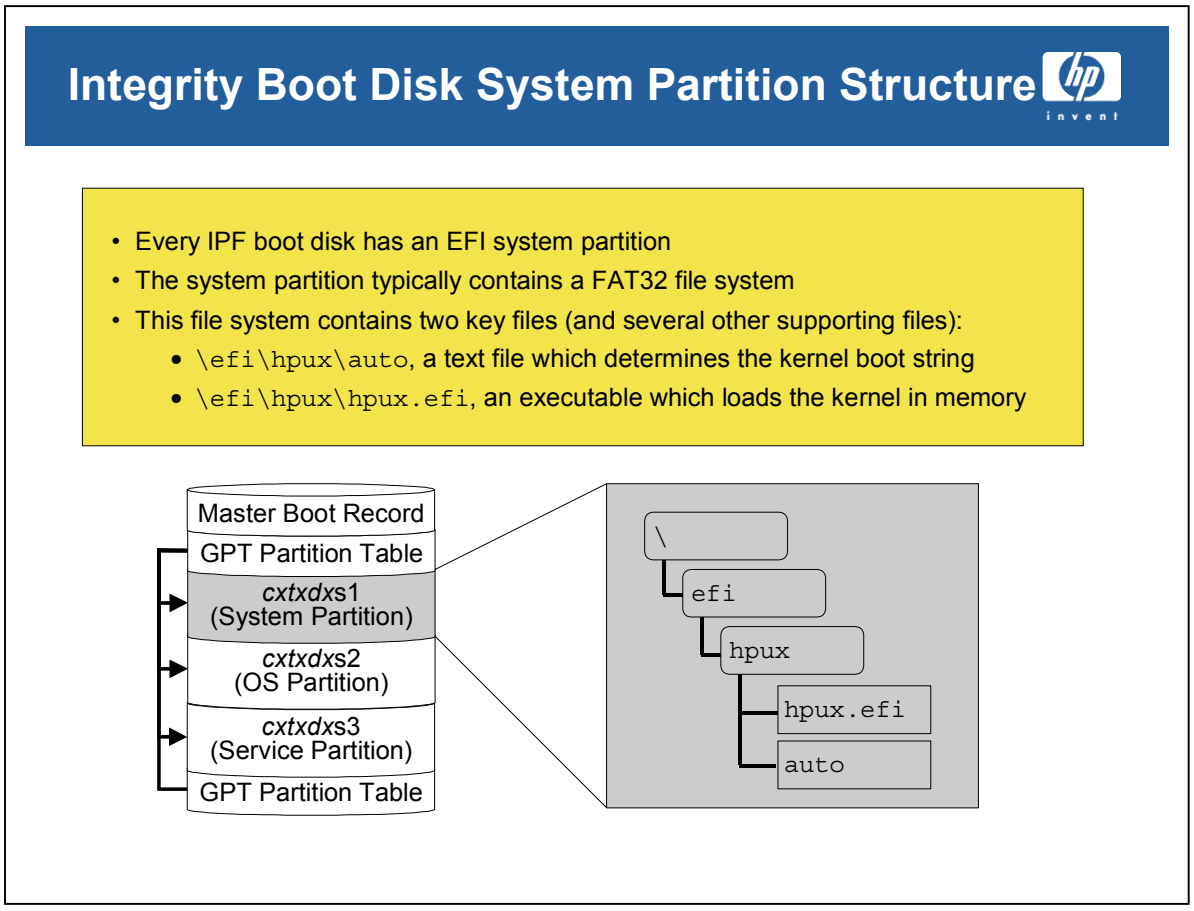
The contents of this partition may be accessed using traditional LVM commands such as **`pvcreate`** and **`pvdisplay`**.

Theoretically, a single disk may have multiple OS partitions, each containing a different operating system, but HP currently only supports one OS partition per disk.

HPSP: The HP Service Partition (HPSP) is FAT32 file system that may be used to store offline diagnostic utilities. In HP-UX, this partition is represented by device file **`/dev/rdsk/`*`cxtxdx`*`s3`**.

© 2004 Hewlett-Packard Development Company, L.P.

17–4. SLIDE: Integrity Boot Disk System Partition Structure



Student Notes

Every Integrity boot disk contains an EFI system partition. The system partition (represented by the **`/dev/rdsk/cxtxdxs1`**. device file in HP-UX) contains a variety of EFI executable files and utilities which are used to load the kernel, and facilitate troubleshooting and maintenance activities.

The EFI partition uses a FAT32 file system structure which is identical to the file system structure traditionally used in the Microsoft Windows Operating System. Several new utilities have been added to HP-UX to allow administrators to view and manage files in these EFI file systems.

Files and Tools in the HP-UX EFI System Partition

On an HP-UX boot disk, the most important files in the system partition are:

- **`\efi\hpux\hpux.efi`**, which is responsible for loading and running the kernel, and
- **`\efi\hpux\auto`**, which contains the kernel boot string that **`hpux.efi`** uses to determine which mode and kernel to boot.

© 2004 Hewlett-Packard Development Company, L.P.

The EFI is intended to be "extensible", so you may find other executables in the EFI system partition, too. Currently, HP includes a small suite of network troubleshooting tools in the **`\efi\hp\tools\network`** directory:

- **`ftp.efi`**, an FTP client
- **`ifconfig.efi`**, for enabling a LAN interface before booting HP-UX
- **`ping.efi`**, which is similar to the standard UNIX ping command
- **`route.efi`**, which is similar to the standard UNIX route command

For instructions on using these tools, run the **`\efi\hp\tools\network\inet.nsh`** EFI script from the EFI shell prompt.

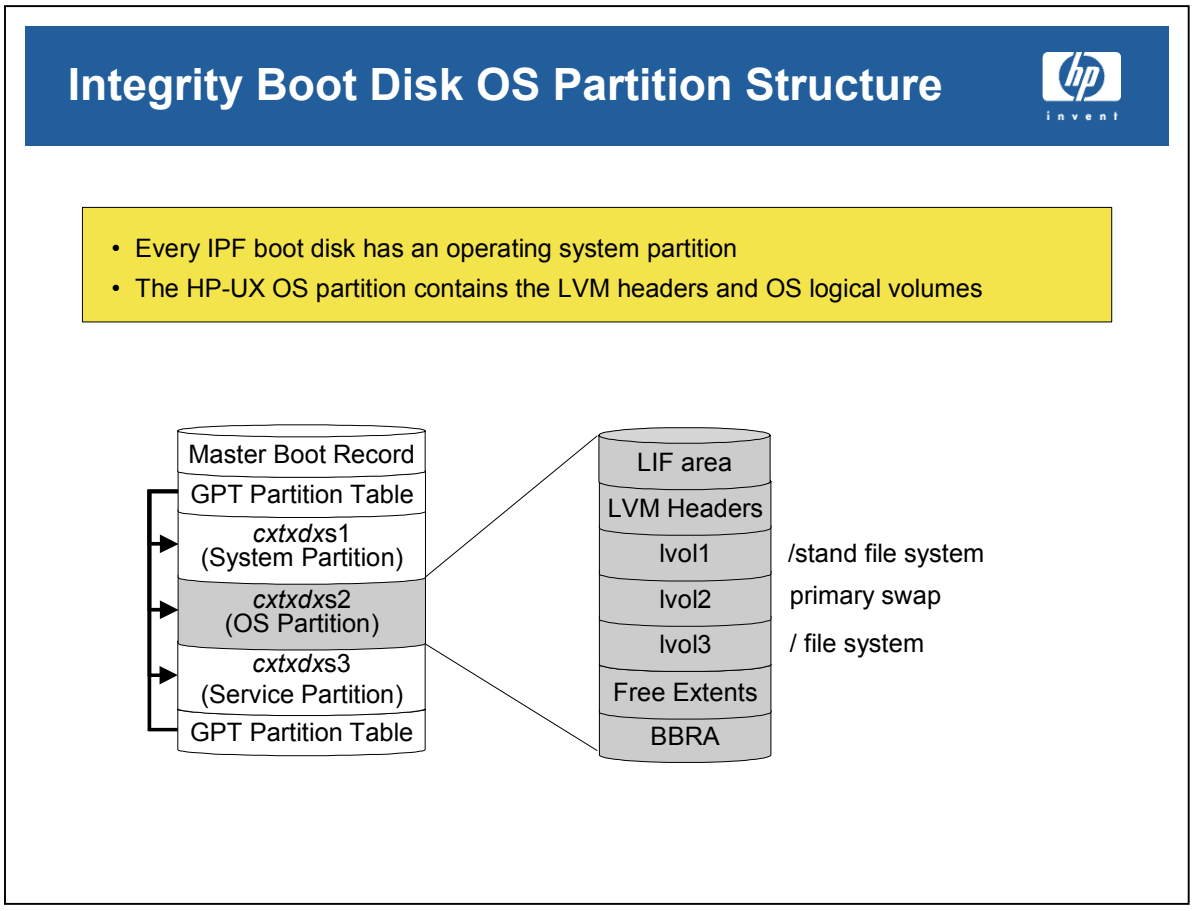
You can write and compile your own EFI executable tools and scripts, too, using an EFI compiler from Intel. See the **`tools`** link on **`http://www.intel.com/technology/efi/`** for more information.

Files and Tools in Other EFI System Partitions

The EFI is also intended to be OS-neutral. Thus, whether you are running HP-UX, Windows, or Linux on your Integrity server, your boot disk should contain an EFI system partition. Of course the EFI executables and files within the partition vary from OS to OS. For instance, the system partition on a Linux boot disk contains the **`lilo.efi`** (LInux LOader) kernel loader rather than **`hpux.efi`**.

© 2004 Hewlett-Packard Development Company, L.P.

17–5. SLIDE: Integrity Boot Disk OS Partition Structure



Student Notes

The OS partition contains an encapsulated LVM boot disk, including all of the structures one would expect to see on a traditional Whole Disk, LVM, or VxVM boot disk. The system partition may be accessed using the **`/dev/rdsk/cxtxdxs2`** device file.

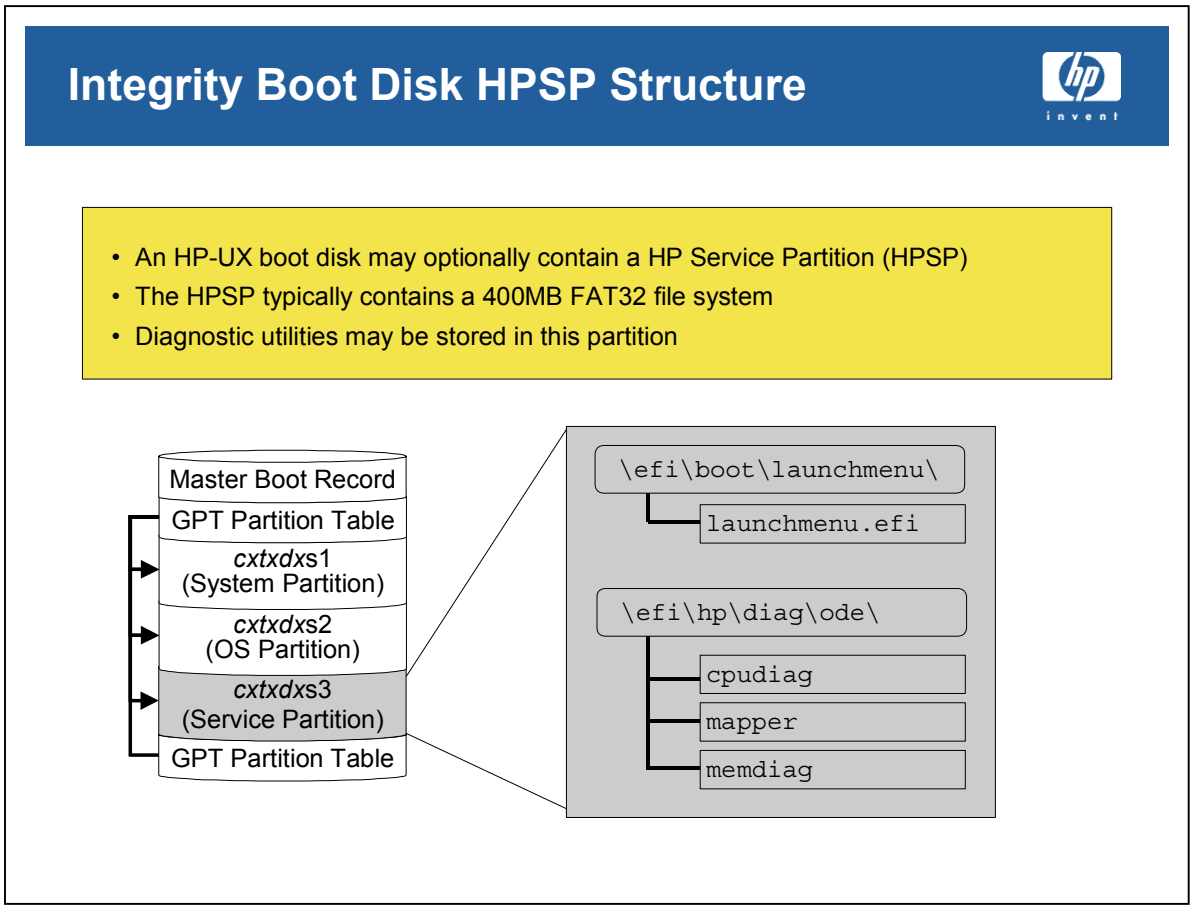
If you choose to configure your system using the Whole Disk option for the boot disk, the OS partition will contain a LIF area, a VxFS root file system, and primary swap.

If you choose to configure your system using LVM on the boot disk, the OS partition will contain a LIF area, PVRA, VGRA, BDRA, BBRA, and physical extents allocated to the **`/`** file system and primary swap logical volumes in vg00. You can view the contents of an LVM OS partition using the traditional LVM **`pvdisplay`** command.

If you choose to configure your system using VxVM on the boot disk, the OS partition will contain a VxVM root disk group. You can view the contents of the VxVM OS partition using the traditional VxVM **`vxdisk`** and **`vxprint`** commands.

© 2004 Hewlett-Packard Development Company, L.P.

17–6. SLIDE: Integrity Boot Disk HPSP Structure



Student Notes

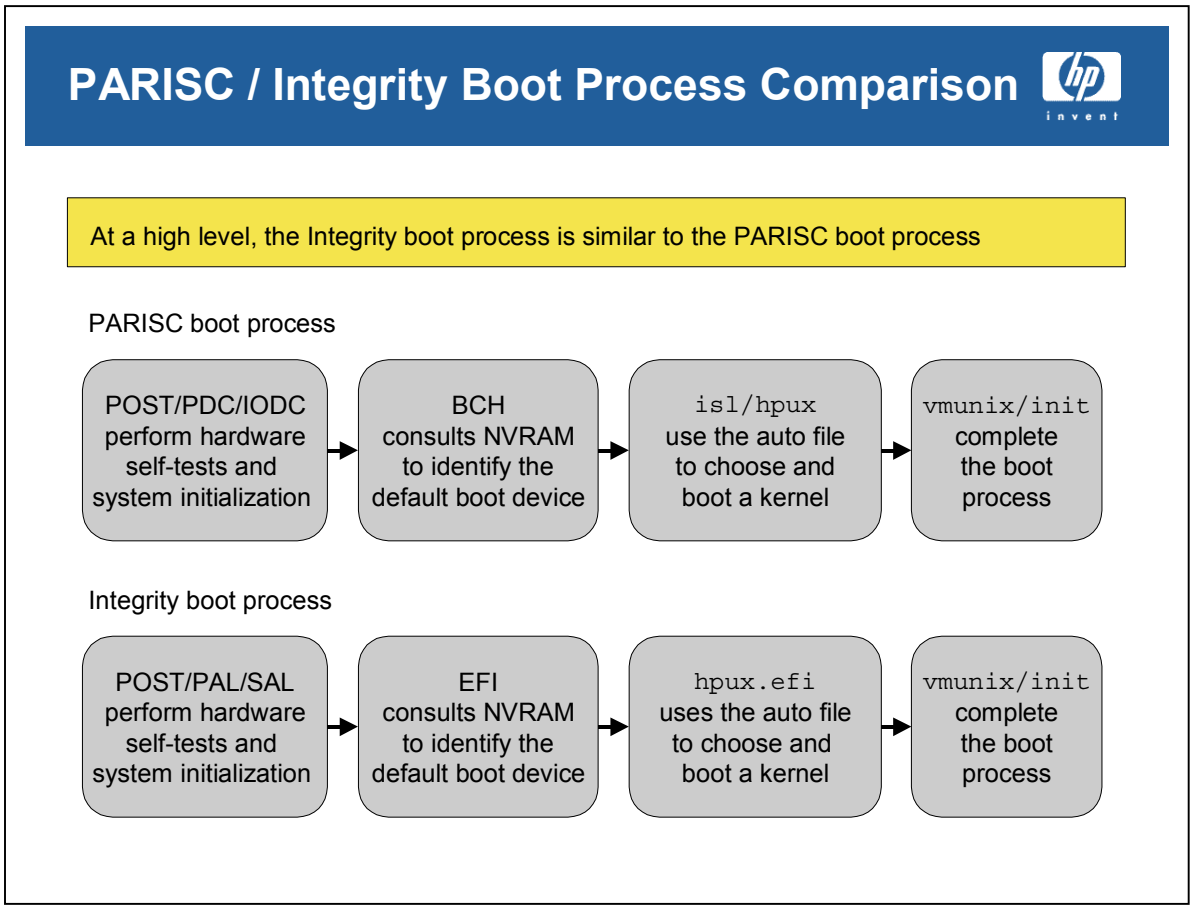
A new tool called e-Diagnostics ("e-Diag" for short) is now available for HP-UX 11i v2. The program allows you to troubleshoot hardware problems on a computer that cannot boot an operating system.

e-Diagnostics may be launched directly from the Offline Diagnostics and Utilities CDROM, which is included with every HP Integrity server, or from a 400MB, FAT32 Service Partition on the system boot disk. Running e-Diagnostics from the disk-based service partition provides much better performance than the CDROM-based solution.

The slide shows just a few of the many files that are typically found in the HPSP or on the Diagnostics and Utilities CDROM. The intuitive, menu-based interface for e-Diag may be executed by running **`\efi\boot\launchmenu\launchmenu.efi`** from either the Diagnostics and Utilities CD, or the HPSP.

© 2004 Hewlett-Packard Development Company, L.P.

17–7. SLIDE: PARISC/Integrity Boot Process Comparison



Student Notes

So far in this chapter we've discussed several firmware and disk-based components that play a part in the Integrity boot process. This slide shows how these players work together to boot an Integrity system to a fully functional state. For the sake of comparison, the slide also shows the corresponding steps in the PA-RISC boot process.

POST/PAL/SAL Firmware Perform Hardware Self-Tests and System Initialization

First, the POST/PAL/SAL firmware performs a variety of processor and hardware self-test and initialization tasks.

On the PA-RISC platform, the Power On Self Test (POST), Processor Dependent Code (PDC), and I/O Dependent Code (IODC) perform these tasks.

EFI Boot Manager Consults NVRAM to Identify the Default Boot Device

Next, the EFI boot manager consults NVRAM, displays a menu of potential boot devices, and proceeds to load a boot loader from the default boot device if autoboot is enabled. Alternatively, the administrator can use the EFI boot menu to select an alternate boot source

© 2004 Hewlett-Packard Development Company, L.P.

or use the EFI shell to interact with the system before the autoboot proceeds. On the PA-RISC platform, the Boot Console Handler (BCH) serves a similar role.

`hpux.efi` Uses the `auto` File to Choose and Load a Kernel

On HP-UX systems, the **`hpux.efi`** kernel loader consults the **`\efi\hpux\auto`** file on the EFI system partition to determine which kernel and mode to boot. Typically, the **`auto`** file suggests booting **`/stand/vmunix`** to multi-user mode.

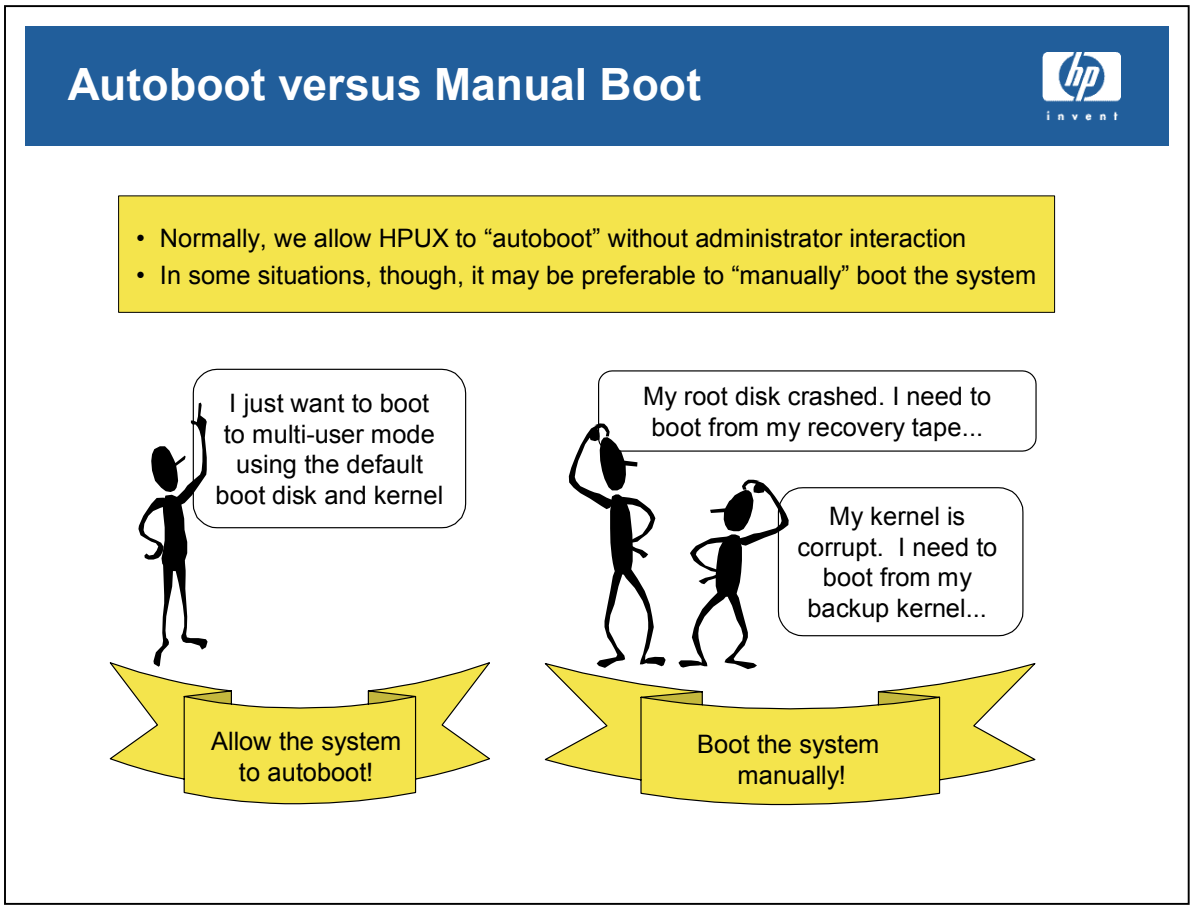
On PA-RISC systems, the Initial System Loader (**`ISL`**) uses the **`HPUX`** kernel loader and **`AUTO`** file to load the kernel in memory. The **`ISL`** and **`HPUX`** executables and the **`AUTO`** file are all stored in the boot area on the PA-RISC boot disk.

`vmunix` Brings the System to a Fully Functional State

The kernel (typically **`/stand/vmunix`**) then scans the hardware, mounts the root file system, and starts the **`init`** daemon. The **`init`** daemon starts the daemons and services necessary to bring the system up to multi-user mode. This portion of the process is identical on PA-RISC and Integrity systems.

© 2004 Hewlett-Packard Development Company, L.P.

17–8. SLIDE: Autoboot versus Manual Boot



Student Notes

Usually, the boot procedure described on the previous slide occurs without intervention from the System Administrator. The system boots from the primary boot disk to multi-user mode using the default kernel. This is known as an **autoboot**.

However, in some cases you may wish to override the autoboot process:

- To re-install the operating system from CD.
- To boot from a system recovery tape if the primary boot disk is corrupt.
- To boot from an alternate boot disk if the primary boot disk is corrupt.
- To boot from an alternate kernel if **`/stand/vmunix`** is corrupt.
- To boot to single-user mode to perform administrative tasks.
- To boot to single-user mode if the root password is lost.

The remainder of this chapter discusses the procedure used to manually boot your HP-UX system, and the functionality available at each step of the process.

© 2004 Hewlett-Packard Development Company, L.P.

Autoboot versus Manual Boot

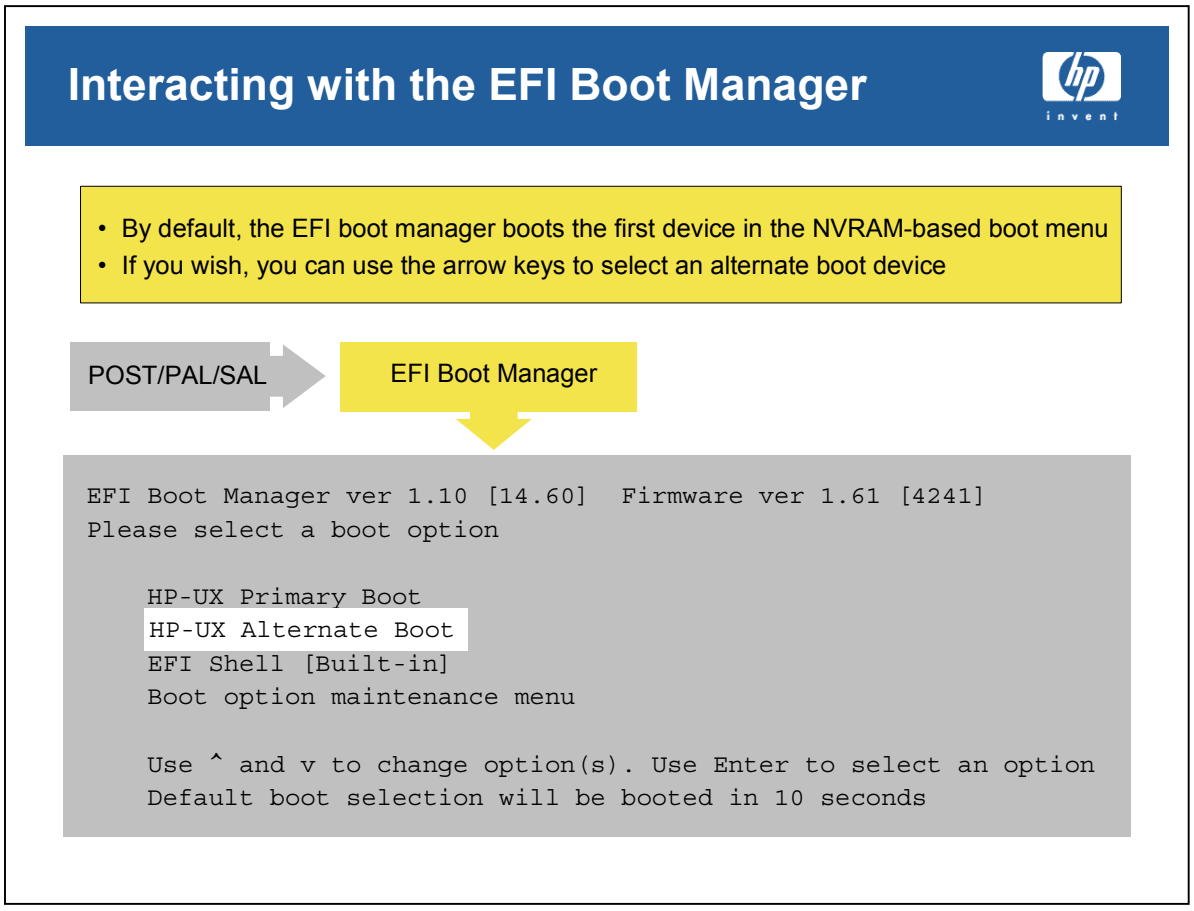
Usually, the boot procedure described on the above occurs without intervention from the System Administrator. The system boots from the primary boot disk to multi-user mode using the default kernel. This is known as an **autoboot**.

However, in some cases you may wish to override the autoboot process with a **manual boot**:

- To re-install the operating system from CD.
- To boot from an alternate boot disk if the primary boot disk is corrupt.
- To boot from a system recovery archive if no boot disk is available.
- To boot from an alternate kernel if **`/stand/vmunix`** is corrupt.
- To boot to single-user mode to perform administrative tasks.
- To boot to single-user mode if the root password is lost.

© 2004 Hewlett-Packard Development Company, L.P.

17–9. SLIDE: Interacting with the EFI Boot Manager



Student Notes

After you initiate the system boot process, the system will run some preliminary system initialization functions, and will eventually display an EFI boot manager menu screen similar to the menu on the slide above.

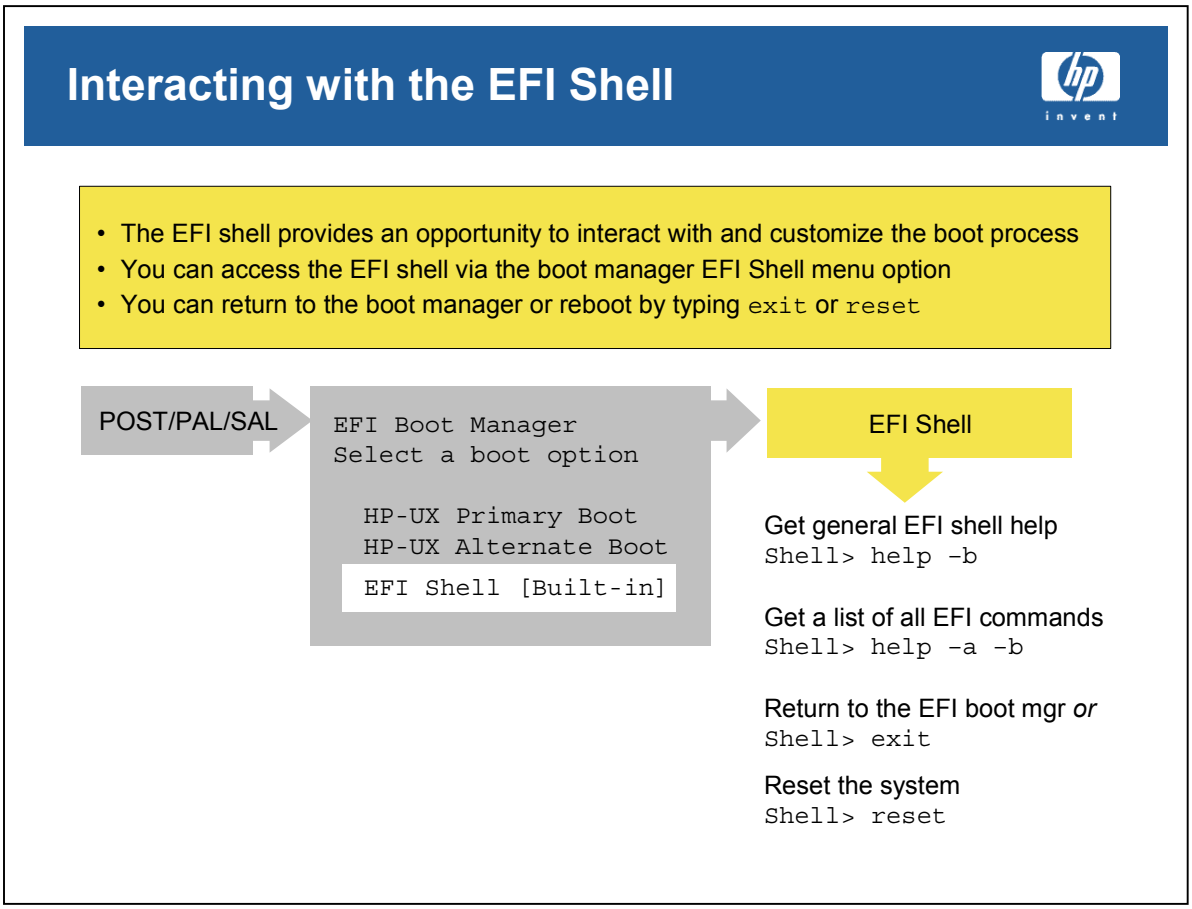
Left to its own devices, the EFI will automatically boot the first device on the menu. If the first boot device fails, the EFI proceeds to the next device on the list. If the second boot device fails, the EFI proceeds to the third.

If you want to explicitly choose an alternate device, simply use the up/down arrow keys to select the desired device and press **Return** before the ten second timeout expires.

Note that the boot manager menu is OS-neutral. If your system contains an HP-UX boot disk, a Linux boot disk, and a Windows boot disk, the boot manager can be configured to display all three boot sources.

© 2004 Hewlett-Packard Development Company, L.P.

17–10. SLIDE: Interacting with the EFI Shell



Student Notes

The last slide explained how to select a boot source via the menu-based EFI boot manager. The EFI shell provides a more flexible mechanism for interacting with the EFI firmware. From the shell you can:

- Boot from an arbitrary boot source, or even an alternate OS
- Customize the boot menu and autoboot settings
- Create customized batch scripts to execute at run time, and much more!

The EFI shell provides a rich command set for managing files and directories in EFI partitions, managing EFI and boot manager settings, storing and accessing variables. It even provides looping and branching functionality! For a complete list of EFI shell commands, see the EFI shell command reference guide at **`http://www.intel.com/software/products/college/efishell/images/efi_shell_cmnd_1_1.pdf`**.

The built-in shell commands may be supplemented by custom EFI shell scripts, and binary executables built using Intel's EFI byte code compiler.

© 2004 Hewlett-Packard Development Company, L.P.

Accessing and Leaving the EFI Shell

To access the EFI shell, interrupt the EFI autoboot process, and select the EFI shell option from the boot manager menu. The shell will automatically display a list of potential boot devices, run the **`startup.nsh`** shell startup script from the default boot disk, and display a **`Shell>`** prompt.

```
EFI Boot Manager ver 1.10 [14.60] Firmware ver 1.61 [4241]
Please select a boot option
    HP-UX Primary Boot
    HP-UX Alternate Boot
    EFI Shell [Built-in]
    Boot option maintenance menu
    Security/Password Menu

Loading.: Built-in EFI Shell
EFI Shell version 1.10 [14.60]
Device mapping table
fs0  : Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part1,SigB)
fs1  : Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part3,SigB)
blk0 : Acpi(HWP0002,0)/Pci(2|0)/Ata(Primary,Master)
blk1 : Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun0,Lun0)
blk2 : Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun1,Lun0)
blk3 : Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part1,SigB)
blk4 : Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part2,SigB)
blk5 : Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part3,SigB8)
startup.nsh> echo -off
  setting hpux path(\EFI\HPUX)...
  type 'fs[x]:' where x is your bootdisk (0, 1, 2...)
  type 'hpux' to start hpux bootloader
Shell>
```

To return to the boot manager menu, simply type **`exit`** at the EFI shell prompt.

```
Shell> exit
```

To reboot the system from the EFI shell, type **`reset`**.

```
Shell> reset
```

Executing EFI Shell Commands and Getting Help

The EFI shell provides a simple, yet powerful interface for executing commands.

- In order to execute an EFI command, just type the command name
  - File extensions may be omitted
  - Full pathnames aren't required if the EFI path variable is set properly
  - Commands are not case-sensitive
  - Most internal commands support a **`-b`** option to view one screen at a time

© 2004 Hewlett-Packard Development Company, L.P.

- The EFI supports several UNIX-like shell features
  - Use the arrow keys for editing and command line recall
  - Use **`*`**, **`?`**, and **`[]`** for filename generation
  - Use **`>`**, **`>>`**, **`2>`**, and **`2>>`** for output and error redirection
- Use the **`help`** command to learn more about EFI commands
  - Use **`help -a`** to view a list of all commands
  - Use **`help`** ***`command`*** to learn more about a specific command
  - Use **`help bch`** to view EFI equivalents for familiar BCH commands

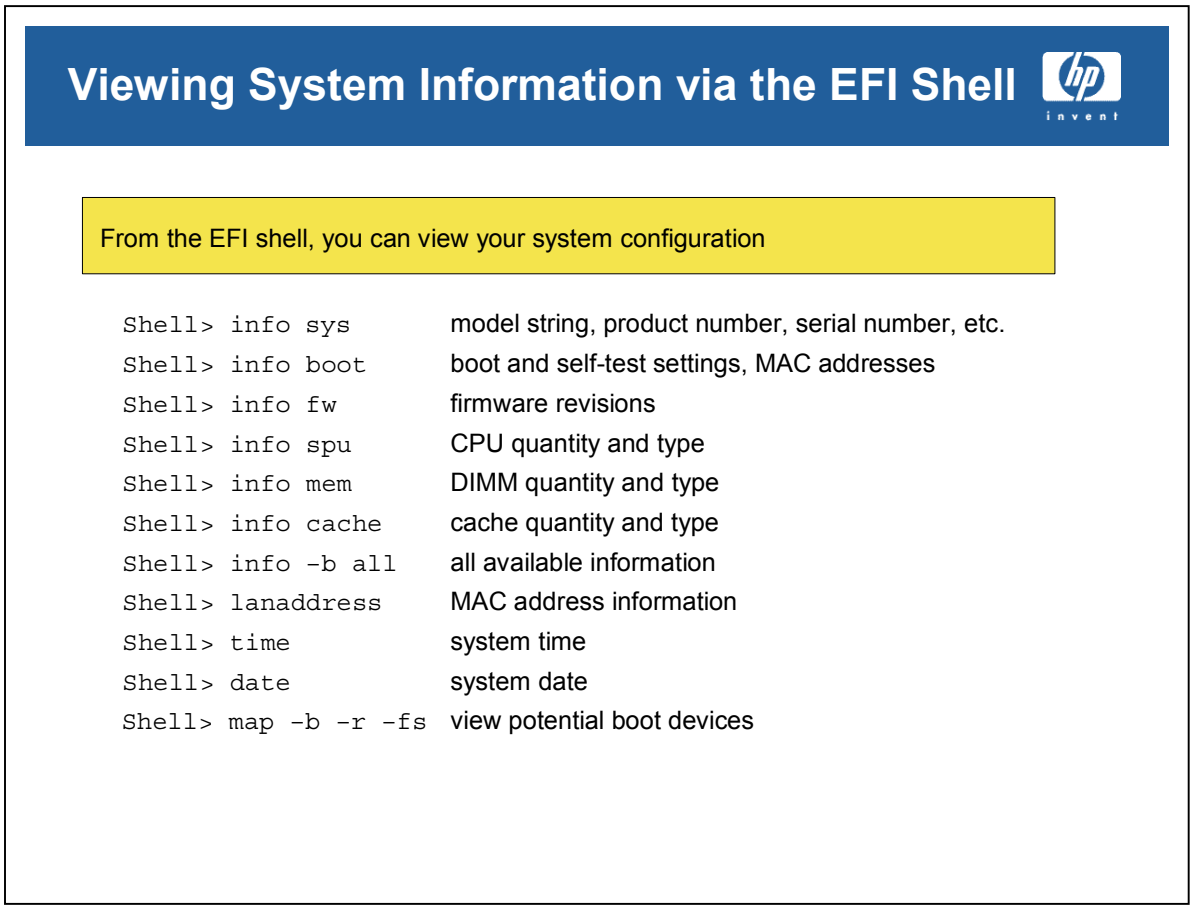
About the EFI Shell PATH Variable

Much like the UNIX shell, the EFI shell searches directories in a **`path`** variable to find executables. You can view or modify the contents of EFI shell variables (including **`path`**) via the EFI shell **`set`** command.

```
Shell> set path \efi\hpux
Shell> set
       path : .;\efi\hpux
```


© 2004 Hewlett-Packard Development Company, L.P.

17–11. SLIDE: Viewing System Information via the EFI Shell



Student Notes

The next few slides list some of the more frequently used EFI commands in a bit more detail. The commands shown on this slide may be used to view system configuration information.

Shell> info sys	*model string, product number, serial number, etc.*
Shell> info boot	*boot and self-test settings, MAC addresses*
Shell> info fw	*firmware revisions*
Shell> info spu	*CPU quantity and type*
Shell> info mem	*DIMM quantity and type*
Shell> info cache	*cache quantity and type*
Shell> info -b all	*all available information*
Shell> lanaddress	*MAC address information*
Shell> time	*system time*
Shell> date	*system date*
Shell> map -b -r -fs	*view potential boot devices (see next slide for details)*

To learn more about viewing a list of EFI devices with the **map** command, see the next slide.

© 2004 Hewlett-Packard Development Company, L.P.

Sample Output from the `info` Command

The **`info -b all`** command is perhaps the most useful of the commands listed above since it provides the most comprehensive output. Sample output is included below. The **`-b`** option displays the output one page at a time.

```
Shell> info -b all

SYSTEM INFORMATION

Date/Time:  Jan  6, 2004  08:56:31  (20:04:01:06:08:56:31)
Manufacturer:  hp
Product Name:  server rx2600
Product Number:  A6870A
Serial Number:  US21729721
UUID:  G0
System Bus Frequency:  200 MHz

PROCESSOR MODULE INFORMATION

      # of                L3      L4      Family/
CPU   Logical             Cache   Cache   Model          Processor
Slot  CPUs     Speed      Size    Size    (hex.)   Rev   State
----  -------  --------   ------  ------  -------  ---   ------------
0      1       900 MHz    3 MB    None    1F/00    B1    Active
1      1       900 MHz    3 MB    None    1F/00    B1    Active



MEMORY INFORMATION

     ---- DIMM A -----  ---- DIMM B -----
      DIMM   Current     DIMM   Current
---  ------ ----------  ------ ----------
 0    512MB     Active   512MB     Active
 1    512MB     Active   512MB     Active
 2    512MB     Active   512MB     Active
 3    512MB     Active   512MB     Active
 4    ----               ----
 5    ----               ----

   Active Memory    : 4096 MB
   Installed Memory : 4096 MB


I/O INFORMATION

BOOTABLE DEVICES

  Order  Media Type   Path
  -----  ----------   -------------------------------------
     1        CDROM   Acpi(HWP0002,0)/Pci(2|0)/Ata(Primary,Master)/CDROM(Entry0)
     2    HARDDRIVE   Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part1,SigB8)
     3    HARDDRIVE   Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun1,Lun0)/HD(Part3,SigB8)


 Seg  Bus  Dev  Fnc  Vendor  Device Slot
```


© 2004 Hewlett-Packard Development Company, L.P.

```
  #    #    #    #     ID      ID     #   Path
 ---  ---  ---  ---  ------  ------  ---  -----------
  00   00   01   00  0x1033  0x0035   XX  Acpi(HWP0002,0)/Pci(1|0)
  00   00   01   01  0x1033  0x0035   XX  Acpi(HWP0002,0)/Pci(1|1)
  00   00   01   02  0x1033  0x00E0   XX  Acpi(HWP0002,0)/Pci(1|2)
  00   00   02   00  0x1095  0x0649   XX  Acpi(HWP0002,0)/Pci(2|0)
  00   00   03   00  0x8086  0x1229   XX  Acpi(HWP0002,0)/Pci(3|0)
  00   20   01   00  0x1000  0x0030   XX  Acpi(HWP0002,100)/Pci(1|0)
  00   20   01   01  0x1000  0x0030   XX  Acpi(HWP0002,100)/Pci(1|1)
  00   20   02   00  0x14E4  0x1645   XX  Acpi(HWP0002,100)/Pci(2|0)
  00   40   01   00  0x103C  0x1029   03  Acpi(HWP0002,200)/Pci(1|0)
  00   60   01   00  0x1000  0x0021   02  Acpi(HWP0002,300)/Pci(1|0)
  00   80   01   00  0x1011  0x0019   01  Acpi(HWP0002,400)/Pci(1|0)
  00   E0   01   00  0x103C  0x1290   XX  Acpi(HWP0002,700)/Pci(1|0)
  00   E0   01   01  0x103C  0x1048   XX  Acpi(HWP0002,700)/Pci(1|1)
  00   E0   02   00  0x1002  0x5159   XX  Acpi(HWP0002,700)/Pci(2|0)



BOOT INFORMATION

   Monarch CPU:

   Current  Preferred
   Monarch   Monarch   Possible Warnings
   -------  ---------  -----------------
         0          0

   AutoBoot: ON - Timeout is : 7 sec

   Boottest:

   BOOTTEST Settings Default Variable

   OS is not speedy boot aware.

   Selftest     Setting
   ---------    --------------
   early_cpu    Run this test
   late_cpu     Run this test
   platform     Run this test
   chipset      Run this test
   io_hw        Run this test
   mem_init     Run this test
   mem_test     Run this test

   LAN Address Information:

   LAN Address        Path
   -----------------  ----------------------------------------
   Mac(00306E1E7EE0)  Acpi(HWP0002,0)/Pci(3|0)/Mac(00306E1E7EE0))
  *Mac(00306E1E9EA9)  Acpi(HWP0002,100)/Pci(2|0)/Mac(00306E1E9EA9))

FIRMWARE INFORMATION

   Firmware Revision: 2.21 [4334]
```


© 2004 Hewlett-Packard Development Company, L.P.

```
   PAL_A Revision: 7.31/7.31
   PAL_B Revision: 7.59

   SAL Spec Revision: 3.01
   SAL_A Revision: 2.00
   SAL_B Revision: 2.21

   EFI Spec Revision: 1.10
   EFI Intel Drop Revision: 14.61
   EFI Build Revision: 1.22

   POSSE Revision: 0.10

   ACPI Revision: 7.00

   BMC Revision 1.50
   IPMI Revision: 1.00
   SMBIOS Revision: 2.3.2a
   Management Processor Revision: kd.05.07


WARNING AND STOP BOOT INFORMATION


CHIP REVISION INFORMATION
```

Chip Type	Logical ID	Device ID	Chip Revision
Memory Controller	0	122b	0022
Root Bridge	0	1229	0022
Host Bridge	0000	122e	0022
Host Bridge	0001	122e	0022
Host Bridge	0002	122e	0022
Host Bridge	0003	122e	0022
Host Bridge	0004	122e	0022
Host Bridge	0006	122e	0022
Host Bridge	0007	122e	0022
Other Bridge	0	0	0002
Other Bridge	0	0	0004
Baseboard MC	0	0	0150

© 2004 Hewlett-Packard Development Company, L.P.

17–12. SLIDE: Managing EFI File Systems via the EFI

Managing EFI File Systems via the EFI

The EFI provides a rich set of commands for managing files on the EFI system partition.

You can use these commands to edit the auto file, or to find an OS loader on the EFI partition.

EFI file system commands are based on traditional UNIX and DOS file commands.

Note that EFI file systems use DOS backslashes rather than UNIX forward slashes!

Select an EFI device containing a FAT32 file system
```
Shell> map -r
Shell> fs1:
```

Use `cd` change to a different directory
```
fs1:\> cd \efi\hpux
```

Use `cd` without arguments to determine the current directory
```
fs1:\> cd
```

Use `ls` or `dir` to list the contents of a directory
```
fs1:\> ls -b
```

Use `type` or `cat` to view the contents of a file
```
fs1:\> cat -b \efi\hpux\auto
```

Use `mkdir` or `md` to create directories
```
fs1:\> mkdir \efi\hpux\backup
```

Use `cp` or `copy` to copy a file
```
fs1:\> cp \efi\hpux\auto \efi\hpux\backup\
```

Use `edit` to edit a file
```
fs1:\> edit \efi\hpux\auto
```

To execute an EFI program, just type its full pathname
```
fs1:\> fs1:\efi\hpux\hpux.efi
```

Student Notes

You can also use the EFI shell to manage files on the EFI system partition, or any other EFI file system. This might be useful if you need to make changes to the **`auto`** file which specifies your kernel boot string, or if you were searching for an OS loader. Some administrators use these commands, too, to create EFI batch scripts.

The EFI system and service partitions typically contain EFI file systems, which are similar to the FAT32 format that is popular on Windows systems. Note that FAT32 file systems use backslashes ("**`\`**") rather than forward-slashes ("**`/`**") in directory paths.

The EFI commands used to access and manage EFI file systems are an odd amalgamation of traditional UNIX and DOS commands. A few examples are included below:

First, select a device containing a EFI file system. Simply type the partition name followed by a "**`:`**"

```
Shell> map -r
Shell> fs1:
```


© 2004 Hewlett-Packard Development Company, L.P.

You can change directories within the file system via the **cd** command. Note that EFI path names use back-slashes rather than UNIX forward slashes. You can also use **..** to represent the parent directory.

```
fs1:> cd \efi\hpux
```

Use the **cd** command without any arguments to determine your present working directory. The EFI shell doesn't support a **pwd** command.

```
fs1:\> cd
```

Use the **ls** or **dir** command to list the contents of an EFI directory. If you don't specify a directory as an argument, you will see the contents of the current directory. The **-b** option displays one page at a time.

```
fs1:\> ls –b
```

Use the **cat** or **type** command to view the contents of an ASCII file in an EFI file system. The **-b** option displays one page at a time.

```
fs1:\> cat -b \efi\hpux\auto
```

Use the UNIX-like **mkdir** command to create directories. Multiple arguments may be specified on a single command line.

```
fs1:\> mkdir \efi\hpux\backup
```

Use the UNIX-like **cp** command to copy files. Use **-r** (recursive) when copying directories.

```
fs1:\> cp \efi\hpux\auto \efi\hpux\backup\
```

Use the **edit** command to edit files. Use the arrow keys to navigate the file contents, and backspace to delete text. Use **[Escape][s]** to save, and **[Escape][q]** to quit.

```
fs1:\> edit \efi\hpux\auto
```

In order to run EFI executables in an EFI file system, simply type the full pathname for the desired EFI program. If the program is in the currently selected partition, you may omit the drive name prefix.

```
fs1:\> fs1:\efi\hpux\hpux.efi
```

Though not shown on the slide, you can also use the UNIX-like **rm** command to remove files from FAT32 file systems.

```
fs1:\> rm \efi\hpux\auto\backup\
```


© 2004 Hewlett-Packard Development Company, L.P.

Aliases

The EFI shell provides an **alias** command very similar to the POSIX **alias** command. To view a list of currently defined aliases, run the **alias** command without any arguments. The aliases shown below are predefined in HP's EFI implementation.

```
fs1:\> alias
    dir  : ls
    md   : mkdir
    rd   : rm
    del  : rm
    copy : cp
    find : ls -r
    ll   : ls
    cat  : type
```

You can also define additional aliases. The example below creates a **pwd** alias which displays the present working directory.

```
fs1:\> alias pwd cd
```

By default aliases persist across reboots. To define a temporary alias, add the **-v** (volatile) option.

```
fs:\> alias –v pwd cd
```

To delete an alias, use the **-d** option.

```
fs1:\> alias –d pwd
```

Managing EFI File Systems Via HP-UX

EFI file systems aren't accessible via traditional UNIX file system commands. However, several EFI-specific utilities are now available to manage EFI partitions on a booted HP-UX system:

- **idisk** for creating EFI partitions
- **efi_fsinit** for creating an EFI file system on an EFI partition
- **efi_ls** for listing the contents of EFI file systems
- **efi_cp** for copying files to and from EFI file systems
- **efi_mkdir** for creating directories in a EFI file system
- **efi_rmdir** for removing directories from a EFI file system
- **efi_rm** for removing files from an EFI file system

© 2004 Hewlett-Packard Development Company, L.P.

The examples described below show some common uses for these commands.

- The **/tmp/idf** file describes the desired disk layout. Based on the sample file below, the EFI system partition will be 100MB. The HP Service Partition will be 400MB. All remaining space will be allocated to the HP-UX OS partition.

- The **idisk** command creates partitions based on the parameters in the file following the **-f** option (**/tmp/idf** in this case). The **-w** option is required if you wish to write changes to the disk. **idisk** also automatically creates EFI file systems in the system and HPSP partitions, so there is no need to explicitly run the **efi_fsinit** command.

- The **mkboot -e** command copies **hpux.efi** and other EFI executables from **/usr/lib/efi** to the new system partition. The **-l** option is required if the disk will be configured using a volume manager such as LVM or VxVM. Use the **-h** option instead if you plan to use the "whole disk" approach.

```
# vi /tmp/idf
  3
  EFI 100MB
  HPUX 100%
  HPSP 400MB
# idisk -wf /tmp/idf /dev/rdsk/c0t1d0
# mkboot -e -l /dev/rdsk/c0t1d0
```

You can use the **efi_ls** command to list the contents of an EFI file system. The **-d** option specifies the device filename of the partition you wish to access. The example below displays the contents of the **/efi/hpux/** directory on the **/dev/rdsk/c0t0d0s1** EFI partition. Note that the HP-UX **efi*** commands require forward slashes in EFI pathnames, while the EFI shell utilities require backslashes.

```
# efi_ls -d /dev/rdsk/c0t0d0s1 /efi/hpux
```

You can edit files stored in an EFI file system, too. Use the **efi_cp** command to copy the desired file to a file in **/tmp**, edit the temporary file with your favorite editor, and return it to the EFI system partition with **efi_cp**.

```
# efi_cp -d /dev/rdsk/c0t0d0s1 /efi/hpux/auto /tmp/auto
# vi /tmp/auto
# efi_cp -d /dev/rdsk/c0t0d0s1 -u /tmp/auto /efi/hpux/auto
```

For more detailed information, see the **efi(4)** man page, or the man pages for any of these commands.

© 2004 Hewlett-Packard Development Company, L.P.

17–13. SLIDE: Managing the EFI Boot Menu via the EFI

Managing the EFI Boot Menu via the EFI Shell

From the EFI shell, you can customize the EFI boot menu via the `bcfg` command

View available devices

```
Shell> map -b -r -fs
```

Define the primary, high availability alternate, and alternate boot devices

```
Shell> fs1:
fs1:\> bcfg boot add 1 \efi\hpux\hpux.efi "HP-UX Primary Boot"
fs1:\> fs2:
fs2:\> bcfg boot add 2 \efi\hpux\hpux.efi "HP-UX HA Alternate Boot"
fs2:\> fs3:
fs3:\> bcfg boot add 3 \install.efi "HP-UX Alternate"
```

Remove the third item from the boot menu

```
Shell> bcfg boot rm 3
```

View the resulting boot menu

```
Shell> bcfg boot dump
```

Enable or disable the EFI autoboot feature

```
Shell> autoboot on|off|10
```

Student Notes

Managing the EFI boot menu items is one of the most common tasks performed from the EFI shell. During the boot process, the EFI uses the boot manager menu to identify the default boot source. If the first device on the boot menu fails to boot, the EFI attempts to boot each successive device until it finds a viable boot source. If you configure a mirrored boot disk you will probably want to add the new mirror to the boot menu. You might also choose to add your DVD drive to the boot menu to simplify installation and recovery processes.

The steps below may be used to add three boot sources to the top of the EFI boot manager menu.

First, use the **`map`** command to view a list of available devices. **`-b`** displays one screen at a time. **`-r`** refreshes the list to ensure that we are viewing the most current data. The **`-fs`** option only lists partitions containing EFI file systems. EFI system partitions, which are typically included in the boot menu, always contain a EFI file system. EFI hardware paths are different than HP-UX hardware paths. The **`ioscan -efC disk`** command will show you which EFI hardware path is associated with each disk.

Next, define the first (primary), second (high availability alternate), and third (alternate) boot sources. For each source, select a device and add it to the menu via the **`bcfg`** command.

© 2004 Hewlett-Packard Development Company, L.P.

- The first **boot** argument indicates that you wish to modify the boot menu. The **bcfg** command may also be used to add and remove drivers.
- The second **add** argument indicates that you wish to add an item to the boot menu. You can also remove (**rm**) a menu item, or rearrange/move menu items (**mv**).
- The third argument determines where each device appears in the boot menu. Existing menu items are demoted down the menu to make space for each new boot source.
- The fourth argument specifies the default boot executable on each device. On HP-UX boot disks, the kernel loader is typically called **\efi\hpux\hpux.efi**. The install executable on the HPUX install DVDs is typically called **\install.efi**.
- The fifth argument specifies the text string to display in the boot menu.

```
Shell> fs1:
fs1:\> bcfg boot add 1 \efi\hpux\hpux.efi "HP-UX Primary Boot"
fs1:\> fs2:
fs2:\> bcfg boot add 2 \efi\hpux\hpux.efi "HP-UX HA Alternate
Boot"
fs2:\> fs0:
fs0:\> bcfg boot add 3 \install "HP-UX Alternate"
```

You can also remove items from the boot menu via the **bcfg** command.

```
Shell> bcfg boot rm 3
```

To view the boot menu contents at any time, type **bcfg boot dump**

```
Shell> bcfg boot dump
```

During the boot process, the EFI firmware automatically displays a menu of boot devices recorded in NVRAM. By default, the boot manager waits ten seconds, then automatically boots the first device in the menu. As noted earlier, you can press **Escape** during the 10-second grace period if you wish to choose an alternate boot device. The EFI **autoboot** command makes it possible to disable the autoboot sequence entirely, or change the timeout period.

```
Shell> autoboot off    # Turn off the autoboot feature
Shell> autoboot on     # Turn on the autoboot feature
Shell> autoboot 10     # Change the default autoboot grace period
```


© 2004 Hewlett-Packard Development Company, L.P.

17–14. SLIDE: Booting an Arbitrary Device via the EFI Shell

Booting an Arbitrary Device via the EFI Shell

From the EFI shell, you can boot from a variety of boot sources

View available devices, and identify your desired boot device

```
Shell> map -b –r -fs
```

Run the `hpux.efi` kernel loader from an alternate boot disk

```
Shell> fsn:\efi\hpux\hpux.efi
```

Run the diagnostics utility from the disk-based HPSP partition, or from the diagnostics CD

```
Shell> fsn:\efi\boot\launchmenu\launchmenu.efi
```

Run the Ignite-UX install utility from the install DVD

```
Shell> fsn:\install.efi
```

Run the Ignite-UX install utility from an Ignite-UX server:

```
Shell> lanboot select
01   Acpi(HWP0002,0)/Pci(3|0)/Mac(00306E1E7EE0)
02   Acpi(HWP0002,100)/Pci(2|0)/Mac(00306E1E9EA9)
Select Desired LAN: 1
```

Student Notes

View available devices, and identify the **`fsn:`** label for your desired boot device/partition.

```
Shell> map -b –r –fs
```

Run the **`hpux.efi`** kernel loader from an alternate boot disk. Replace **`fsn:`** with the file system label associated with the alternate boot disk.

```
Shell> fsn:\efi\hpux\hpux.efi
```

Run the diagnostics utility from the disk-based HPSP partition, or from the diagnostics CD. Replace **`fsn:`** with the file system label associated with either the HPSP or the CDROM drive.

```
Shell> fsn:\efi\boot\launchmenu\launchmenu.efi
```

Run the Ignite-UX install utility from the install DVD. Replace **`fsn:`** with the file system label associated with the CDROM drive.

```
Shell> fsn:\install.efi
```

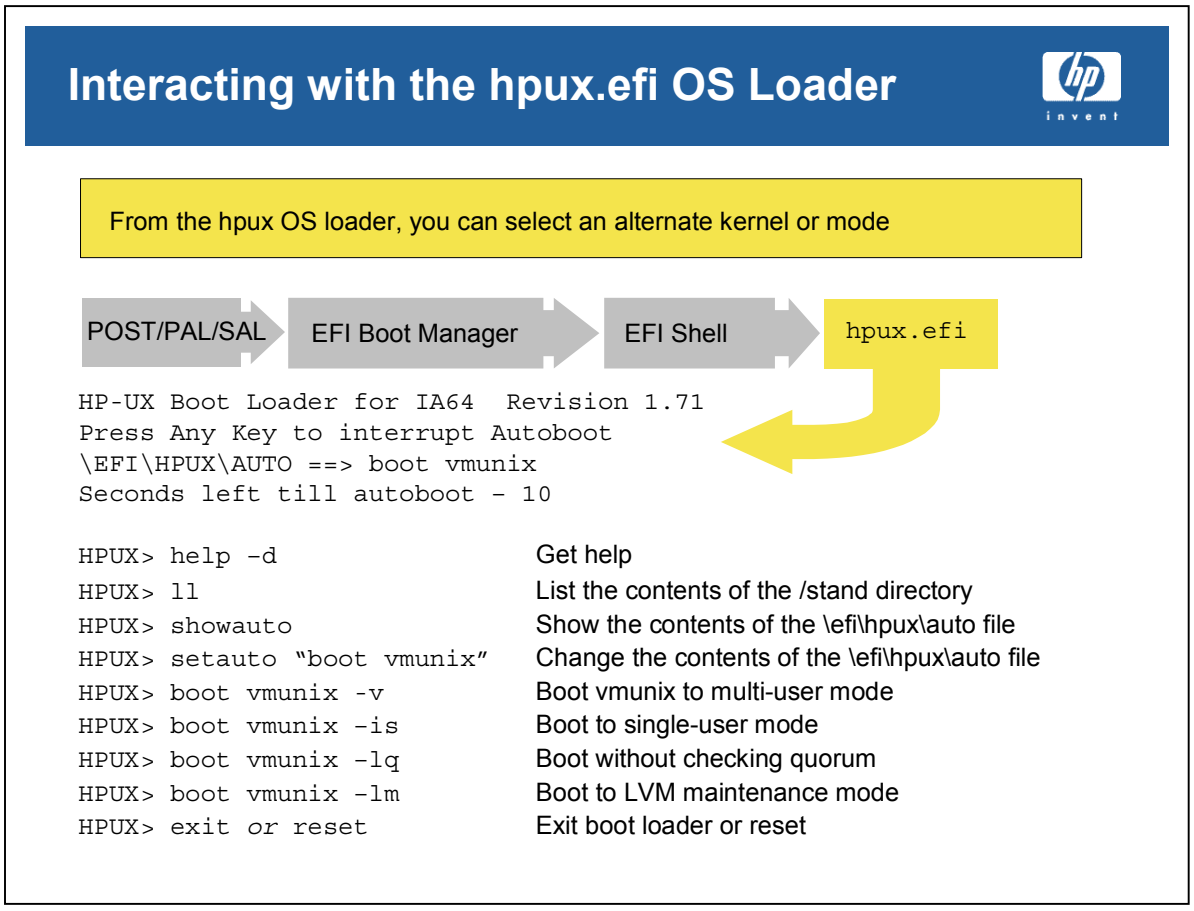

© 2004 Hewlett-Packard Development Company, L.P.

Run the Ignite-UX install utility from an Ignite-UX server. The EFI uses a BOOTP broadcast mechanism to automatically identify an appropriate Ignite server, so no IP address is required. If you have multiple LAN interfaces, though, you will have to specify which card should be used to contact the server.

```
Shell> lanboot select
01  Acpi(HWP0002,0)/Pci(3|0)/Mac(00306E1E7EE0)
02  Acpi(HWP0002,100)/Pci(2|0)/Mac(00306E1E9EA9)
Select Desired LAN: 1
```


© 2004 Hewlett-Packard Development Company, L.P.

17–15. SLIDE: Interacting with the hpux.efi OS Loader



Student Notes

After you select an HP-UX boot device to boot using either the EFI boot manager or the EFI shell, the EFI loads and runs the **\efi\hpux\hpux.efi** OS loader. If the selected boot device contains a **\efi\hpux\auto** file, **hpux.efi** reads this file to determine which kernel to load, and which mode to boot. You should see a message similar to the following:

```
HP-UX Boot Loader for IPF  --  Revision 1.73
Press Any Key to interrupt Autoboot
\EFI\HPUX\AUTO ==> boot vmunix
Seconds left till autoboot - 10
   Type 'help' for help
```

After 10 seconds, **hpux.efi** automatically boots the system based on the contents of the **auto** file.

If there is no **auto** file, or if you interrupt the autoboot sequence by pressing **Escape**, you will find yourself at an **HPUX>** prompt. At this point, you can manually interact with the kernel loader. The commands available at the HPUX OS loader prompt are similar to the

© 2004 Hewlett-Packard Development Company, L.P.

options on the PARISC **hpux** command. A sampling of available commands are listed below. For more information, type **help** at the **HPUX>** prompt.

`HPUX> ll`	*List contents of /stand*
`HPUX> showauto`	*View contents of the \efi\hpux\auto file*
`HPUX> boot backup`	*Boot the backup kernel*
`HPUX> boot vmunix -v`	*Boot vmunix to multi-user mode*
`HPUX> boot vmunix –is`	*Boot to single-user mode*
`HPUX> boot vmunix –lq`	*Boot without checking LVM quorum*
`HPUX> boot vmunix –lm`	*Boot to LVM maintenance mode*
`HPUX> boot vmunix –tm`	*Boot to tunable maintenance mode*
`HPUX> exit`	*Return to the EFI boot manager*

NOTE:	Tunable maintenance mode boots the kernel using system default kernel parameters. This will be described in greater detail in the kernel tuning module later in the course.

© 2004 Hewlett-Packard Development Company, L.P.

17–16. SLIDE: For Further Study

For Further Study

Intel offers a number of free and for-fee EFI related courses.
Look for the Extensible Firmware Interface course section on
`https://shale.intel.com/SoftwareCollege/CourseCatalog.asp`

Read the EFI shell documentation at
`http://www.intel.com/software/products/college/efishell/images/efi_shell_cmnd_1_1.pdf`

HP's *HP System Partitions Guide* on `http://docs.hp.com` includes a chapter on booting and resetting partitions on both HP-UX 11iv1 and HP-UX 11iv2

For HP-UX command syntax descriptions and options, see the HP-UX online man pages.

For EFI command syntax descriptions and options, access the EFI help screens by typing `help` at the `Shell>` prompt

To learn more about the HP Service Partition visit:
`http://docs.hp.com/hpux/onlinedocs/diag/ode/ode_ediag.htm`

Student Notes

© 2004 Hewlett-Packard Development Company, L.P.

17–17. LAB: Booting Integrity Servers

Directions

Carefully follow the instructions below.

Part 1: Preliminary Steps

A portion of this lab requires you to interact with the EFI and HPUX interfaces, which can only be accomplished via a console login. If you are using remote lab equipment, access your system's console interface via the MP.

Part 2: Shutting Down the System

1. Currently, your system should be in multi-user mode. Note which file systems are mounted, and how many processes are running.

2. Shut down to single-user mode. Then check to see what processes and mounted file systems remain. What differences do you see between single- and multi-user modes?

3. From single-user mode, take your machine down to the halt state. What can you do in the halt state? Why might it be necessary to take your system down to the halt state?

   ```
   # reboot -h
   ```

4. Use the MP **`rs`** command or the system power switch to reset your server before proceeding to the next part of the lab. Return to the console interface and immediately begin pressing the **`[Escape]`** key to interrupt the boot process.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Interacting with the EFI Boot Manager and Shell

1. If you didn't already do so in the previous part of the lab, reboot your system and interrupt the EFI autoboot process. At this point, you should see the EFI boot manager menu:

```
EFI Boot Manager ver 1.10 [14.60] Firmware ver 1.61 [4241]

Please select a boot option

    HP-UX Primary Boot
    EFI Shell [Built-in]
    Boot option maintenance menu
    Security/Password Menu
```

2. Select the EFI shell option from the boot manager menu.

3. Use the EFI **`info -b all`** command to view your system configuration.

   a. What is your system product name / model string?

   b. How many CPUs do you have?

   c. How much memory do you have?

   d. How many bootable devices do you have?

   e. How many LAN interfaces do you have?

   f. What are their MAC addresses?

© 2004 Hewlett-Packard Development Company, L.P.

4. Use the **`map -r -fs`** command to view a list of potential boot devices. Record the **`fsn:`** names below, and see if you can determine which hardware path each **`fsn:`** name represents. You may need to refer back to the EFI hardware addressing slide earlier in the book to interpret the hardware paths.

5. Type **`fsn:`** to select the system partition on your primary boot disk, where **`fsn:`** is the drive name that you identified in the previous step.

6. Use the EFI **`cd`** and **`ls`** commands to explore the file system and find the **`hpux.efi`** OS loader and the **`auto`** file. What are the full pathnames for these files?

7. Which EFI shell command can be used to view the contents of the **`auto`** file? Try it!

8. Recall that the **`bcfg`** command may be used to view and manage the EFI boot menu. Use the appropriate **`bcfg`** arguments to dump the current boot menu to the screen.

9. Remove your primary boot disk from the EFI boot manager menu.

© 2004 Hewlett-Packard Development Company, L.P.

10. Oops! We didn't really want to remove the primary boot disk from the boot manager. Add the primary boot disk back to the top of the boot menu. Label the menu item "HP-UX Primary Boot".

11. Dump the EFI boot menu to the screen again to verify your work.

12. What command can you execute from the EFI shell to boot the system? Make it so! When given an opportunity to interrupt the HPUX kernel loader autoboot, press **Escape**.

© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Interacting with the HPUX kernel loader

In this part of the lab, you will have a chance to explore the HPUX kernel loader.

1. If you didn't interrupt the autoboot process in the previous part of the lab, reboot again, and press **Escape** when the HPUX autoboot message appears.

```
HP-UX Boot Loader for IA64  Revision 1.71

Press Any Key to interrupt Autoboot
\EFI\HPUX\AUTO ==> boot vmunix
Seconds left till autoboot - 10  Escape
  Type 'help' for help
```

2. Type the **help** command to view a list of available commands.

3. Which command can you use to list the kernels available in **/stand**? Try it!

4. Which command can you use to view the contents of the **auto** file? What is the purpose of the **auto** file?

5. Do whatever is necessary to boot **vmunix** to single-user mode.

© 2004 Hewlett-Packard Development Company, L.P.

6. Did the system prompt you for a password when you were brought to single-user mode? When might this be helpful?

7. Reboot your system again, but this time let it boot unattended using the default boot disk and kernel.

© 2004 Hewlett-Packard Development Company, L.P.

Part 5: (Optional) Configuring a Partition Table for a Mirrored Boot Disk

Many administrators choose to mirror the boot disk to ensure that the failure of a single interface or controller card doesn't jeopardize a system's availability. The last few parts of the lab walk you through the process required to mirror an Integrity boot disk. First, you need to identify create three EFI partitions on the new boot disk mirror.

1. Use **ioscan** and **diskinfo** to identify a disk large enough to mirror the existing boot disk.

   ```
   # ioscan -funC disk
   # diskinfo /dev/rdsk/c0t1d0
   ```

2. Create the system, OS, and service partitions with the HP-UX **idisk** command. Allocate 500MB for the EFI system partition, 400MB for the HPSP partition, and all the remaining space for the OS partition. Replace **c0t1d0** in the command below with your new mirror's block device file name.

   ```
   # vi /tmp/idf
     3
     EFI 500MB
     HPUX 100%
     HPSP 400MB
   # idisk -wf /tmp/idf /dev/rdsk/c0t1d0
   ```

3. Create device files for the new partitions.

   ```
   # insf -eC disk
   ```

4. Verify the partition table. Near the end of the **idisk** output, you should see three partitions. Replace **c0t1d0** with your new mirror's raw device file name.

   ```
   # idisk /dev/rdsk/c0t1d0
   ```

5. Verify that the device files were created properly. There should be block and raw ***cxtxdx*s1**, ***cxtxdx*s2**, and ***cxtxdx*s3** device files for the new boot mirror.

   ```
   # ioscan -funC disk
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 6: (Optional) Configure the Mirrored Boot Disk's EFI System Partition

After you create the partition table, you can initialize and populate the EFI system partition.

1. Populate the **/efi/hpux/** directory in the new EFI system partition with the **mkboot** command. This command copies the **hpux.efi** boot loader and other files to the system partition.

   ```
   # mkboot –e –l /dev/rdsk/c0t1d0
   ```

2. By default , the LVM "quorum" rules require at least 51% of the disks in vg00 to be available at boot time. However, we want our system when just 50% of the mirrors in vg00 are available. Thus, we must disable the default quorum check by changing the auto file boot string to **boot vmunix –lq** on both disks. You could do this from the EFI shell, but we'll use the HP-UX **efi_cp** command to avoid rebooting.

   ```
   # vi /tmp/auto
     boot vmunix –lq
   # efi_cp –d /dev/rdsk/c0t0d0s1 /tmp/auto /efi/hpux/auto
   # efi_cp –d /dev/rdsk/c0t1d0s1 /tmp/auto /efi/hpux/auto
   ```

3. Verify the contents of the system partition.

   ```
   # efi_ls –d /dev/rdsk/c0t1d0s1
   ```

4. Verify the contents of the **auto** file in the system partition.

   ```
   # efi_cp –d /dev/rdsk/c0t1d0s1 –u /efi/hpux/auto /tmp/efi.auto
   # cat /tmp/efi.auto
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 7: (Optional) Configure the Mirrored Boot Disk's OS Partition

Next, configure the OS partition. Recall that the OS partition is essentially an encapsulated LVM boot disk.

1. Initialize and add the new OS partition to vg00. Be sure to specify s2, the OS partition!

```
# pvcreate -fB /dev/rdsk/c0t1d0s2
# vgextend vg00 /dev/dsk/c0t1d0s2
```

2. Mirror *all* vg00 logical volumes. If your vg00 has non-default logical volume names, you may need to manually **`lvextend`** each logical volume.

```
# for lv in /dev/vg00/lvol*
  do
     lvextend -m 1 $lv
  done
```

3. Add a line to **`/stand/bootconf`** so SDUX knows which disks are boot disks. When patches are **`swinstall`**'ed for LIF area utilities, SDUX consults **`/stand/bootconf`** to determine which disks have LIF areas that might need to be patched. Be sure to specify s2, the OS partition!

```
# vi /stand/bootconf
    l /dev/dsk/c0t0d0s2
    l /dev/dsk/c0t1d0s2
```

4. Verify that the disk was added to vg00, and that the logical volumes are sync'ed. The **`LV Status`** field for each logical volume should report **`syncd`**.

```
# vgdisplay -v vg00
```

5. Verify that the BDRA was updated properly. You should see references to both disks in the boot, root, and swap portions of the output.

```
# lvlnboot -v
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 8: (Optional) Add the Mirrored Boot Disk to the EFI Boot Manager Menu

Finally, add the new mirrored disk to the EFI boot manager menu. This ensures that the EFI will automatically attempt to boot the mirrored boot disk if the primary boot disk fails.

1. You can do this via the EFI **`bcfg`** command, but we'll use the HP-UX **`setboot`** command instead to avoid rebooting. Update the EFI boot manager primary, high availability, and alternate boot paths.

```
# setboot -p 0/1/1/0.0.0
# setboot -h 0/1/1/0.1.0
# setboot -a 0/0/2/0.0.0.0
```

2. Verify that the boot paths are configured properly in NVRAM.

```
# setboot
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 9: (Optional) Booting from the Mirrored Boot Disk

After you complete the mirroring process, it is a good idea to verify your work by booting from the new mirror.

1. Initiate the boot sequence.

```
# shutdown –ry 0
```

2. Interrupt the EFI boot manager autoboot.

```
EFI Boot Manager ver 1.10 [14.60] Firmware ver 1.61 [4241]
[Escape]
```

3. Select your mirrored disk from the boot manager selection menu.

```
EFI Boot Manager ver 1.10 [14.60] Firmware ver 1.61 [4241]
Please select a boot option
   HP-UX Primary Boot
   HP-UX Alternate Boot
   EFI Shell [Built-in]
```

4. Verify which disk you booted from when the boot process completes.

```
# grep “Boot device’s HP-UX HW path” /var/adm/syslog/syslog.log
  vmunix: Boot device's HP-UX HW path is: 0.0.0.0.1.0
```


© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 18 — Configuring the HP-UX 11i v1 Kernel

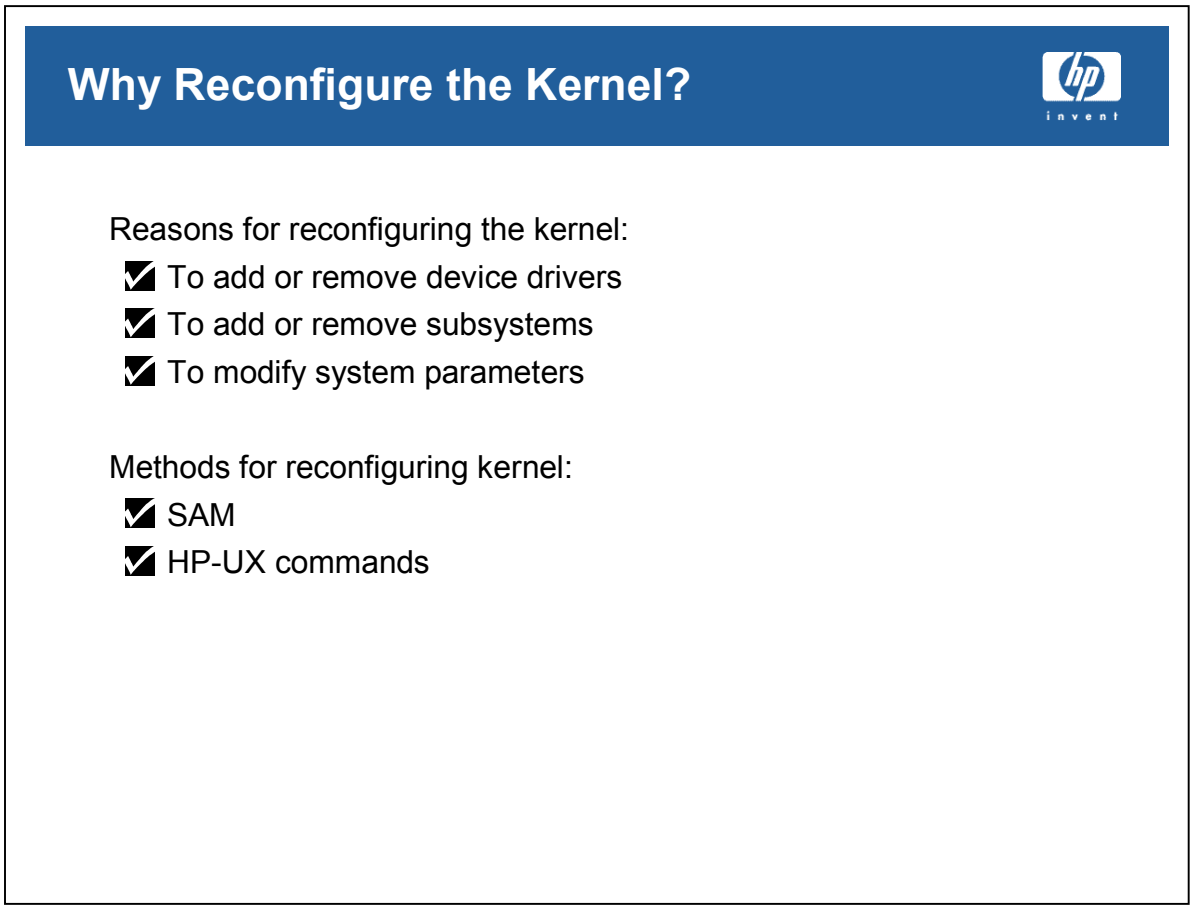
Objectives

Upon completion of this module, you will be able to do the following:

- List three reasons for reconfiguring the kernel.
- Compare and contrast static versus dynamic kernel modules.
- Describe the structure of the **`/stand`** directory.
- Add and remove device drivers with SAM.
- Add and remove kernel subsystems with SAM.
- Modify tunable kernel parameters with SAM.
- Boot from a backup kernel.

© 2004 Hewlett-Packard Development Company, L.P.

18–1. SLIDE: Why Reconfigure the Kernel?



Student Notes

There are several situations in which kernel reconfiguration is necessary. The kernel must be reconfigured to change the following:

Drivers	Kernel drivers contain the code that the kernel requires to communicate with interface cards, devices, and other hardware components. When you add a new device or interface card to your system, you may need to add a driver to the kernel. Conversely, if you remove a device or interface card, it may be beneficial to remove the associated driver from the kernel to save space in RAM.
Subsystems	HP-UX includes several subsystems that may be managed by the system administrator to add or remove system functionality. The "LVM" subsystem allows the kernel to configure and use logical volumes. The "CD-ROM/9000" subsystem allows the kernel to access CDFS file systems. The "LAN/9000 subsystem" allows the kernel to communicate with a local area network. These are a few of the most commonly used kernel subsystems that you may choose to configure.

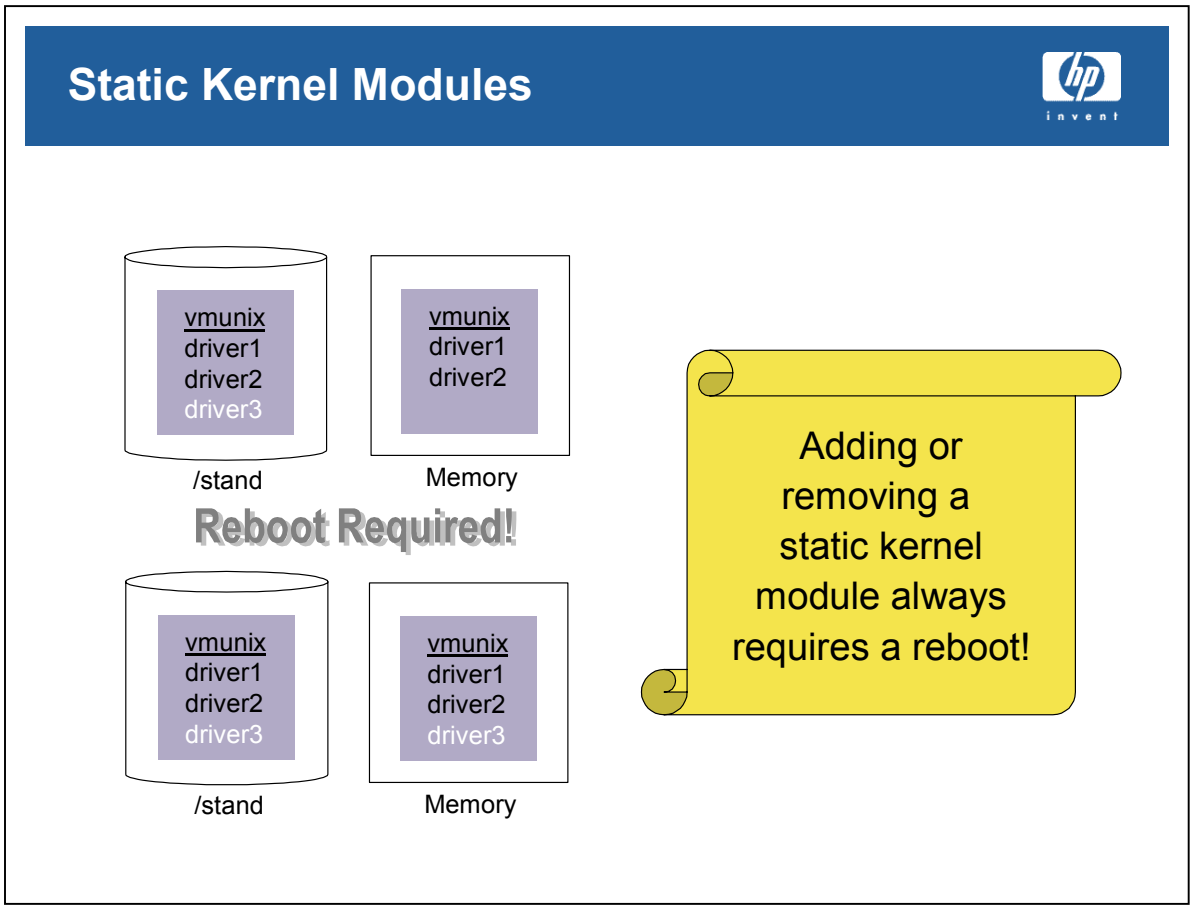
© 2004 Hewlett-Packard Development Company, L.P.

Tunable Kernel Parameters	Tunable kernel parameters affect the behavior of the system as well as the size of the kernel. Many kernel table sizes are determined by system parameters. For example, the maximum number of concurrent processes is determined by the size of the process table. The system parameter **`nproc`** determines the size of the process table. Extreme care should be taken when modifying system parameters. Often times, when installing software, such as databases, the software vendor will advise you to modify system parameters in order for their product to function properly.

You can modify the kernel either by using SAM, the System Administration Manager utility, or by using HP-UX commands. In most cases, the system must be rebooted to load the new kernel and put the modifications in place.

© 2004 Hewlett-Packard Development Company, L.P.

18–2. SLIDE: Static Kernel Modules



Student Notes

An HP-UX kernel is constructed from multiple kernel modules. Adding additional modules provides additional functionality to your kernel, and removing a module removes functionality.

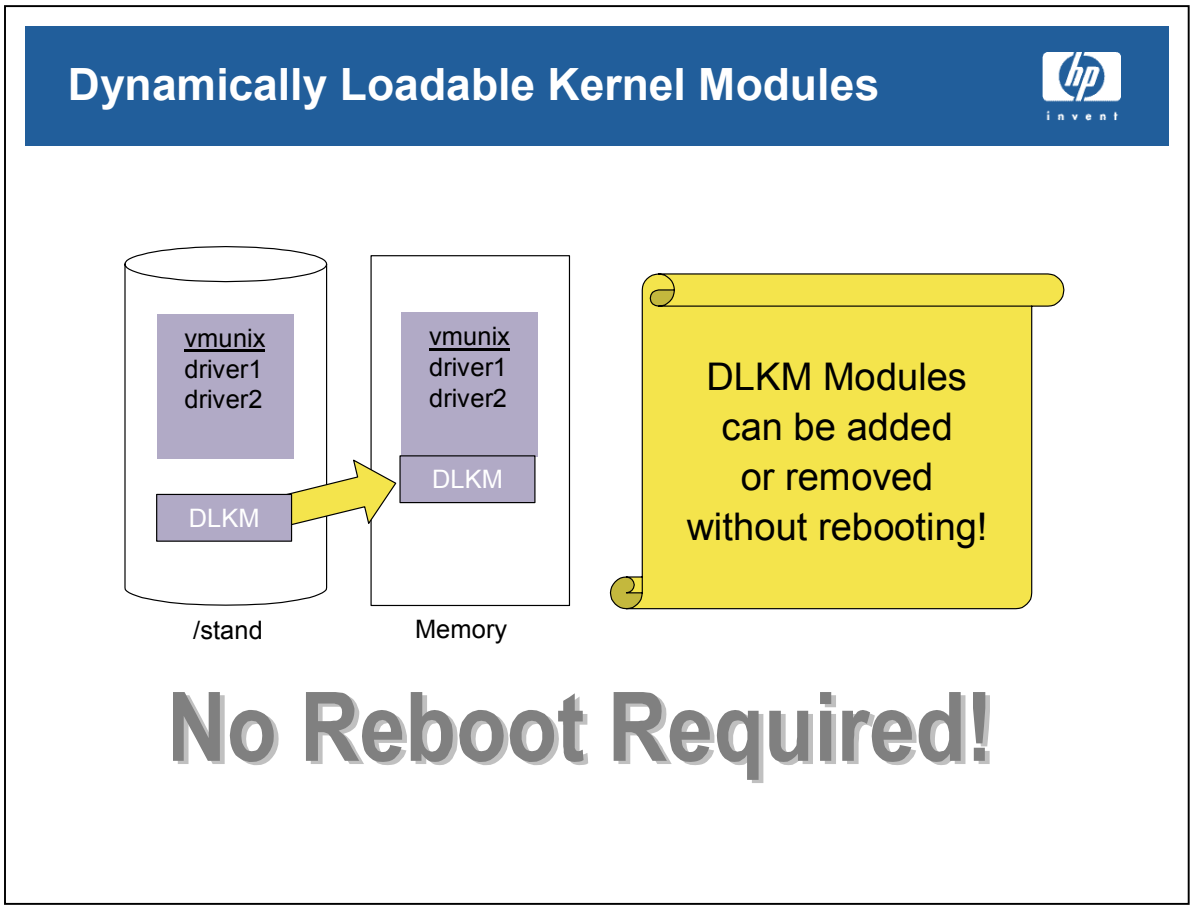
Prior to version 11.x of HP-UX, adding or removing a driver or subsystem module required the administrator to

- Rebuild the entire kernel on disk.
- Then reboot the system to move the new kernel into place.

Thus, the kernel was essentially **static**. Changes could not be made to the running kernel without a system reboot.

© 2004 Hewlett-Packard Development Company, L.P.

18–3. SLIDE: Dynamically Loadable Kernel Modules



Student Notes

Version 11.x of HP-UX introduced the infrastructure necessary to support dynamically loadable kernel modules (DLKMs). DLKM drivers can be dynamically loaded into the running kernel as needed, then unloaded when no longer needed. This offers a number of advantages:

High availability: DLKMs don't require a system reboot to add needed drivers and subsystems.

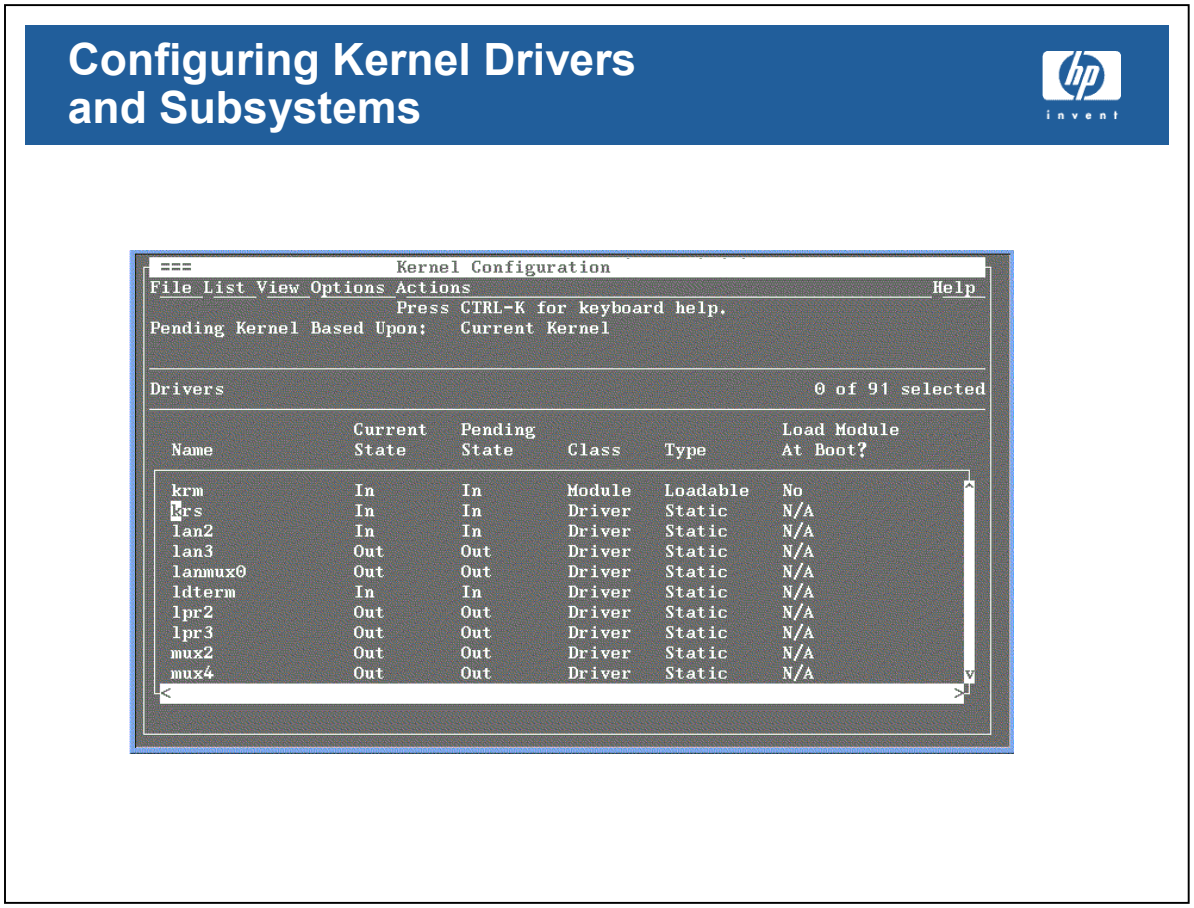
Efficiency: DLKMs that the kernel no longer needs can be unloaded.

DLKM is being implemented in phases. Though the DLKM architecture is in place at version 11.x of HP-UX, most HP-UX drivers are still static at this point. Subsequent releases will offer DLKM drivers and support online replacement of DLKM modules.

HP-UX version 11i added the ability to dynamically tune some kernel parameters, too!

© 2004 Hewlett-Packard Development Company, L.P.

18–4. SLIDE: Configuring Kernel Drivers and Subsystems



Student Notes

SAM provides the safest, most intuitive mechanism for managing the HP-UX kernel. Under the SAM "Kernel Configuration" functional area, there are several sub-areas:

- drivers
- subsystems
- dump devices
- configurable parameters

This slide concentrates on driver and subsystem configuration, while the following slide discusses tunable kernel parameters.

© 2004 Hewlett-Packard Development Company, L.P.

Kernel Subsystems

HP-UX includes several subsystems that may be managed by the system administrator to add or remove system functionality. The "LVM" subsystem allows the kernel to configure and use logical volumes. The "CD-ROM/9000" subsystem allows the kernel to access CDFS file systems. The "LAN/9000 subsystem" allows the kernel to communicate with a local area network. These are a few of the most commonly used kernel subsystems that you may choose to configure.

In order to add a subsystem, go to the SAM -> Kernel Configuration -> Subsystems functional area. The resulting SAM screen will list all of the available subsystem names and descriptions. The "Current State" column indicates which subsystems are currently included in the kernel. The "Pending State" column indicates which subsystems have been marked for inclusion the next time the kernel is recompiled. Search the list of subsystems, select the subsystem you wish to add, and choose the "Add" item from the "Actions" menu. Similarly, you can choose the "Remove" item from the "Actions" menu any time you wish to remove a subsystem from the kernel.

Changes made to the "Pending State" column don't take effect immediately; the selected subsystems won't be added or removed until you recompile and reboot.

Kernel Drivers

Kernel drivers contain the code that the kernel requires to communicate with interface cards, devices, and other hardware components. When you add a new device or interface card to your system, you may need to add a driver to the kernel. Conversely, if you remove a device or interface card, it may be beneficial to remove the associated driver from the kernel to save space in RAM.

In order to add a driver, go to the SAM -> Kernel Configuration -> Drivers functional area. The "Drivers" screen, much like the "Subsystems" screen, lists driver names, descriptions, and current and pending state columns. There are three other columns, however, that don't appear on the "Subsystems" screen. The "Class" column indicates which drivers are part of DLKM "Modules", and which are standard static "Drivers". The "Type" column indicates which drivers are "Static" modules, and which are dynamically "Loadable" kernel modules. Note that drivers that are included in DLKM modules can be configured directly into the kernel as "Static" modules or can be dynamically loaded as "Loadable" modules. The "Load Module At Boot" column only applies to dynamically loadable kernel modules; DLKM modules can be loaded immediately at boot time, or dynamically on an as-needed basis.

To change the state of driver, select the driver from the object list, then select the "Add" or "Remove" item from the "Actions" menu. If the selected driver is "Static", then the "Pending State" column will change to reflect the desired state, but the driver's "Current State" won't change until you recompile the kernel and reboot.

If the selected driver is a DLKM driver, SAM presents a dialog box where you will have an opportunity to define the module's attributes. You can specify whether the module should be statically compiled into the kernel, or dynamically loaded. If you choose to load the module dynamically, you can configure the module to load immediately at boot time, or on an as-needed basis. Set these parameters as you wish, then click [Modify], and finally [OK]. If you chose to load the module dynamically, the "Current State" column should change immediately!

© 2004 Hewlett-Packard Development Company, L.P.

Templates

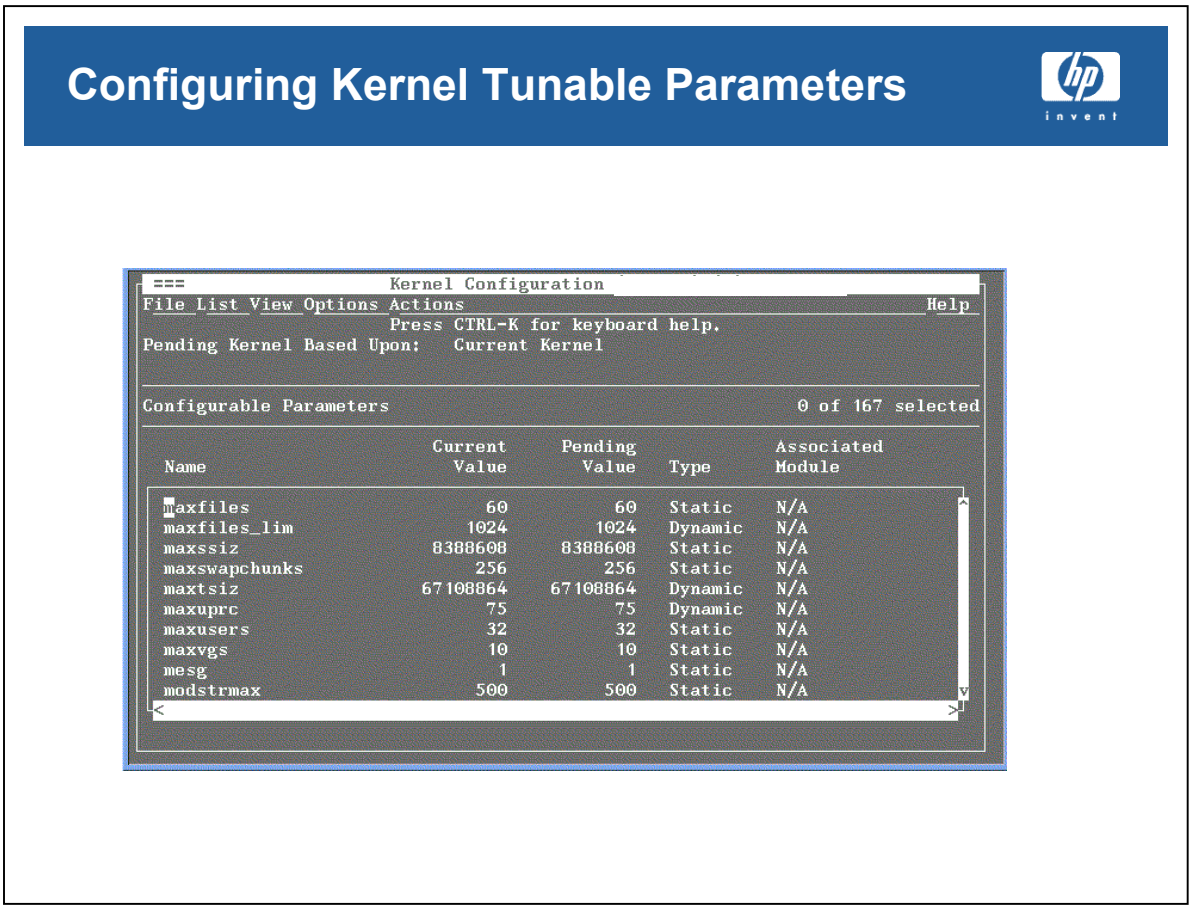
If you manage multiple identical systems, you may wish to configure one system, then use that system's kernel configuration as a "template" for the other hosts. To do this, copy the template system's **`/stand/system`** file to each of the systems you wish to configure. Then launch SAM on the target systems, and go to the "Drivers" or "Subsystems" screen. Select **`Templates`** from the **`Actions`** menu. Select **`Load a Template`**. Enter the full path name of the **`system`** file that you copied from the template system. To view the template values, use the **`View->Columns`** menu selection to assign a column position to the template values. Template values will then automatically be displayed in any view or modify dialog boxes. The template values are not automatically applied to the kernel you are building. To apply template values choose **`Templates`** from the **`Actions`** menu. **`Select Apply Template Values`**. SAM will allow you to choose to apply template values to configurable parameters and drivers or subsystems.

Compiling the New Kernel

When you have completed all the kernel changes you wish to make using SAM, choose the **`Process New Kernel`** action to build a new static kernel based on the "Pending Values" that you selected.

© 2004 Hewlett-Packard Development Company, L.P.

18–5. SLIDE: Configuring Kernel Tunable Parameters



Student Notes

Certain kernel operating parameters can be configured to fit specific system needs, resulting in better performance or more effective allocation of resources. The size of the "process table," for instance, is determined by the **`nproc`** kernel parameter. The number of processes that a single user can have running at any given time is defined by the **`maxuprc`** parameter.

The ideal value for each parameter is often determined by the system's particular hardware configuration, the specific mix of applications the system runs, and the trustworthiness of system users, factors that vary widely from system to system.

Kernel tunable parameters are most easily changed via SAM. In order to change a kernel parameter, launch SAM, and go to the SAM -> Kernel Configuration -> Configurable Parameters screen. Each line in the "Configurable Parameters" object list has several fields:

Name:	Lists the names of the configurable parameters.
Current Value:	Lists the parameter values in the currently running kernel.
Pending Value:	Lists the parameter values that will take effect after the next kernel rebuild and reboot.

© 2004 Hewlett-Packard Development Company, L.P.

Type:	Beginning with HP-UX 11i, some configurable parameters can be modified without a system reboot. These dynamically tunable parameters are identified by a "Dynamic" label in the "Type" field. Changing a "Static" parameter requires a kernel recompile and reboot. Currently, only ten tunables are dynamic.
Associated Module:	Parameters associated with DLKM modules will identify the associated module's name in this field.
Description:	Description of the parameter's purpose.

In order to change a configurable parameter, select a parameter, then click the "Modify Configurable Parameter" menu item from the "Actions" menu. This displays a dialog box in which you can modify the chosen parameter. Most parameters are defined as explicit values, but some are defined by formulas based on other kernel parameter values.

Change the value or formula as you wish, then click [OK]. If the parameter is dynamically tunable, the "Current State" value should change immediately. Other changes will take effect after the next kernel recompile and reboot.

Warning!

Changing kernel parameters to improper or inappropriate values or combinations of values can cause data loss, system panics, or other (possibly very obscure and/or difficult to diagnose) operating anomalies, depending on which parameters are set to what values.

- Never set any system parameter to a value outside the allowable range for that parameter. (SAM refuses to store values outside of the allowable range.)

Many parameters interact, and their values must be selected in a balanced way. Your application vendors should be able to provide some guidance. Also, the SAM Help menus include a tutorial that explains the purpose of many of the configurable kernel parameters. Also take a look at the "HP-UX Kernel Tuning and Performance Guide" on **`http://docs.hp.com`**.

Tuned Parameter Sets

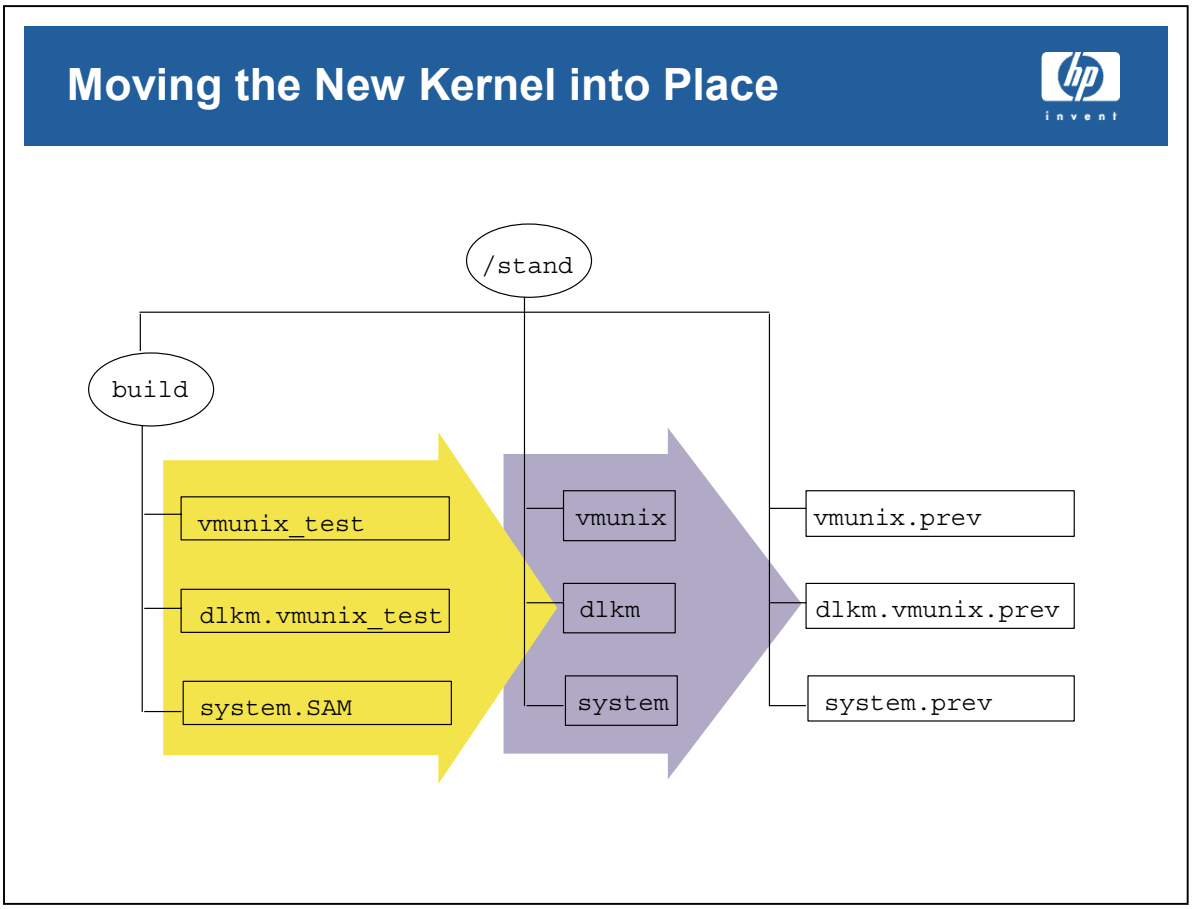
If you wish, you can choose to set your system parameters using predefined Kernel Parameter Sets. Choose the "Actions -> Apply Tuned Parameter Set" menu item to view the list of tuned parameter sets that is included in your version of HP-UX.

Compiling the New Kernel

When you have completed all the kernel changes you wish to make using SAM, choose the **`Process New Kernel`** action to build a new static kernel based on the "Pending Values" that you selected.

© 2004 Hewlett-Packard Development Company, L.P.

18–6. SLIDE: Moving the New Kernel into Place



Student Notes

After you have made your desired changes to the pending kernel configuration, and have selected **`Process New Kernel`** from the SAM **`Actions`** menu, SAM builds a new kernel for you.

The newly-built kernel has three components:

`vmunix_test`	The static kernel executable.
`dlkm.vmunix_test`	Compiled DLKM modules associated with the new kernel.
`system.SAM`	Text file listing the static drivers, subsystems, and parameters incorporated in the new kernel.

SAM initially creates the new kernel in the **`/stand/build`** directory. At next reboot, however, the system will boot from the kernel in **`/stand/vmunix`**. Thus, you must move the new kernel and its associated files into place under **`/stand`** and reboot before your changes take effect.

© 2004 Hewlett-Packard Development Company, L.P.

You should also retain a copy of your existing kernel, in case your new kernel proves to be unbootable. Backup copies of the existing kernel files are typically kept in the **/stand** directory with **.prev** extensions.

SAM will make a backup copy of your existing kernel and immediately move your new kernel into place for you if you select **Move Kernel into Place and Shutdown/Reboot System Now**. If users are currently logged into your system, you may prefer to move the new kernel into place and reboot later. This requires several steps at the command line:

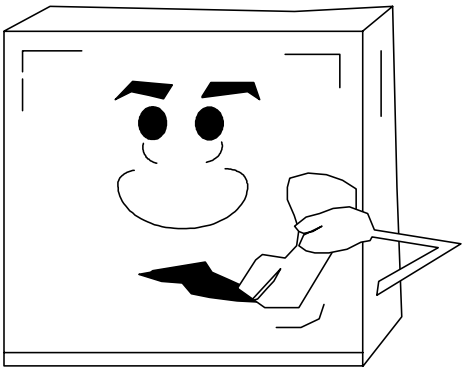
```
# cd /stand
# cp /stand/system /stand/system.prev
# cp /stand/build/system.SAM /stand/system
# kmupdate /stand/build/vmunix_test
# shutdown -ry 0
```

NOTE: At HP-UX 11.x you should not use **cp** or **mv** to replace the current **vmunix** file or **dlkm** directory. Use the **kmupdate** command instead.

© 2004 Hewlett-Packard Development Company, L.P.

18–7. SLIDE: What If the New Kernel Won't Boot?

What If the New Kernel Won't Boot?



If the new kernel fails to boot, boot from the backup kernel!

```
[POWER]
[ESC]
Main Menu> boot pri
Interact with the IPL? y
ISL> hpux -is /stand/vmunix.prev
```

Student Notes

You may have made changes to the kernel using either HP-UX commands or SAM. Although SAM tries to check for invalid and incompatible settings, it is not foolproof. Booting from the new kernel is the only way to verify that your changes were truly successful.

If the system won't boot from your new kernel, or if it boots but does not perform acceptably, you may need to return to your prior kernel configuration. Fortunately, the system maintains a backup copy of your previous kernel.

Booting from the Backup Kernel

To boot the backup kernel, **`/stand/vmunix.prev`**, boot the system manually as described below:

`[POWER]`	*Power-on the system.*
`[ESC]`	*Hit the escape key to interrupt the boot process.*
`Main Menu> boot pri`	*Boot from the primary boot disk.*
`Interact with the IPL? y`	*Do interact with the IPL/ISL.*
`ISL> hpux -is /stand/vmunix.prev`	*Boot from the original kernel.*

© 2004 Hewlett-Packard Development Company, L.P.

Moving the Original Kernel Back into Place

After booting the system, you may wish to make the original kernel the default so future reboots can proceed without any intervention from the administrator:

```
# cd /stand                       Move to the /stand directory.
# mv system.prev system           Move the original system file back into place.
# kmupdate /stand/vmunix.prev     Move the original kernel back into place.
# shutdown -ry 0                  Reboot the system with the original kernel.
```


© 2004 Hewlett-Packard Development Company, L.P.

18–8. LAB: Kernel Configuration

Directions

Follow the directions below carefully.

Part 1: Preliminary Steps

A portion of this lab requires you to interact with the ISL and boot menus, which can only be accomplished via a console login. If you are using remote lab equipment, access your system's console interface via the GSP/MP.

Part 2: Managing Kernel Drivers and Subsystems with SAM

The kernel requires device drivers to communicate with devices on your system. Without the proper driver, the kernel is unable to access a device. As you add new devices and interface cards to your system, you may need to install new drivers. Removing unused drivers may save some space in memory.

1. SAM is the easiest tool for managing kernel drivers. Go to the SAM kernel driver screen by selecting: **`SAM --> Kernel Configuration --> Drivers`** This should give you a list of all the drivers currently installed on your system. The **`Current State`** column indicates which of the drivers are actually configured in your kernel. The **`Pending State`** column indicates which drivers will be included in the next kernel rebuild. Based on this list, can your kernel successfully communicate with a tape drive that requires the stape driver? Can you dynamically load or unload the stape driver without rebooting, or is it a **static** kernel driver that requires a reboot?

2. Select a "static" driver that isn't yet configured in your kernel, and use the Actions menu to add the driver to the kernel.

3. What changed on the SAM screen to indicate that the static driver you selected will be included in the next kernel rebuild?

© 2004 Hewlett-Packard Development Company, L.P.

4. Dynamically load the **krm** DLKM driver:

   - Select the **krm** driver from SAM's list of available drivers.
   - Select **Actions --> Add driver to kernel**.

   You should be sitting in the **Set Kernel Module Attributes** window at this point. DLKM drivers may be statically or dynamically loaded in the kernel. If you set the module type to Static, the new module will become available only at the next kernel build and reboot.

   If you choose **Loadable**, the driver becomes available immediately. If you choose **Load automatically at boot**, the kernel will load the DLKM automatically during the boot process. Otherwise, the kernel will load only the DLKM as needed.

   - Ensure that the Module Type is **Loadable**.
   - Ensure that **Load Automatically at Boot** is set to **No**.
   - If you changed either of the settings above, click **Modify**.
   - Click **OK** to make the changes take effect.

5. What changed on the SAM screen to indicate that the module is immediately available for use by devices on your system?

6. You can also use SAM to manage kernel subsystems. Choose **List --> Subsystems**. Is the diagnostic subsystem configured? Is the fibre channel subsystem configured?

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Tuning a Kernel Parameter

1. Choose **`List --> Configurable Parameters`**. What is the maximum number of processes your kernel can support? How many files can be open simultaneously?

2. Double the value of the **`maxuprc`** kernel parameter, which defines the maximum number of concurrent processes a user can run. Which fields in the object list changed as a result of your actions? **`Current State`**? **`Pending State`**? Or both? Why?

3. Look a bit further down the list of parameters until you find **`maxusers`**. Double the value of this kernel parameter. Does this change take effect immediately? Why?

4. After changing the value of **`maxusers`**, note that the pending value for **`nproc`** changed as well. Select the **`nproc`** parameter and select **`Actions --> Modify Configurable Parameter`**. How does the definition of this parameter differ from the **`maxusers`** parameter? Why do you think **`nproc`** is defined in this manner?

© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Rebuilding the Kernel

1. Now that you have marked several pending changes to the kernel, choose **`Actions --> Process New Kernel`**. Before moving the kernel into place, look at the next question.

2. Shortly, SAM will ask if you want to move the new kernel into place and reboot. SAM built your new kernel in the **`/stand/build`** directory. Which three files or subdirectories in **`/stand/build`** must be moved into place in **`/stand`** before rebooting with the new kernel?

3. Allow SAM to move the new kernel into place and reboot.

4. When your system returns, try the following:

   ```
   # lsdev              # Lists currently configured static kernel drivers.
   # kmtune | more      # Lists your currently running kernel's tunable parameters.
   ```

   Was your new static kernel driver included in the new kernel? Did the value of **`nproc`** increase?

© 2004 Hewlett-Packard Development Company, L.P.

Part 5: Booting from the Backup Kernel

Hopefully, your new kernel will boot successfully and bring you up to multiuser mode. In some cases, however, the parameters in your new kernel may conflict and cause the boot to fail. Fortunately, SAM keeps a backup of your previous kernel in **/stand** called **/stand/vmunix.prev.**

1. Reboot your system to single user mode using the backup kernel.

2. Use **lsdev** and **kmtune** to check your parameter values and driver list again. Which kernel appears to be running? (Note: The **lsdev** and **kmtune** executables are in **/usr/sbin**, so you will need to do a **mount -a** first.)

   ```
   # mount -a
   # kmtune | more
   # lsdev
   ```

   Hopefully, the driver and parameter lists reflect the original kernel, not the kernel you tuned in the earlier part of this lab.

3. Move your backup kernel and associated files back into place as the default kernel and reboot again.

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 19 — Configuring the HP-UX 11i v2 Kernel

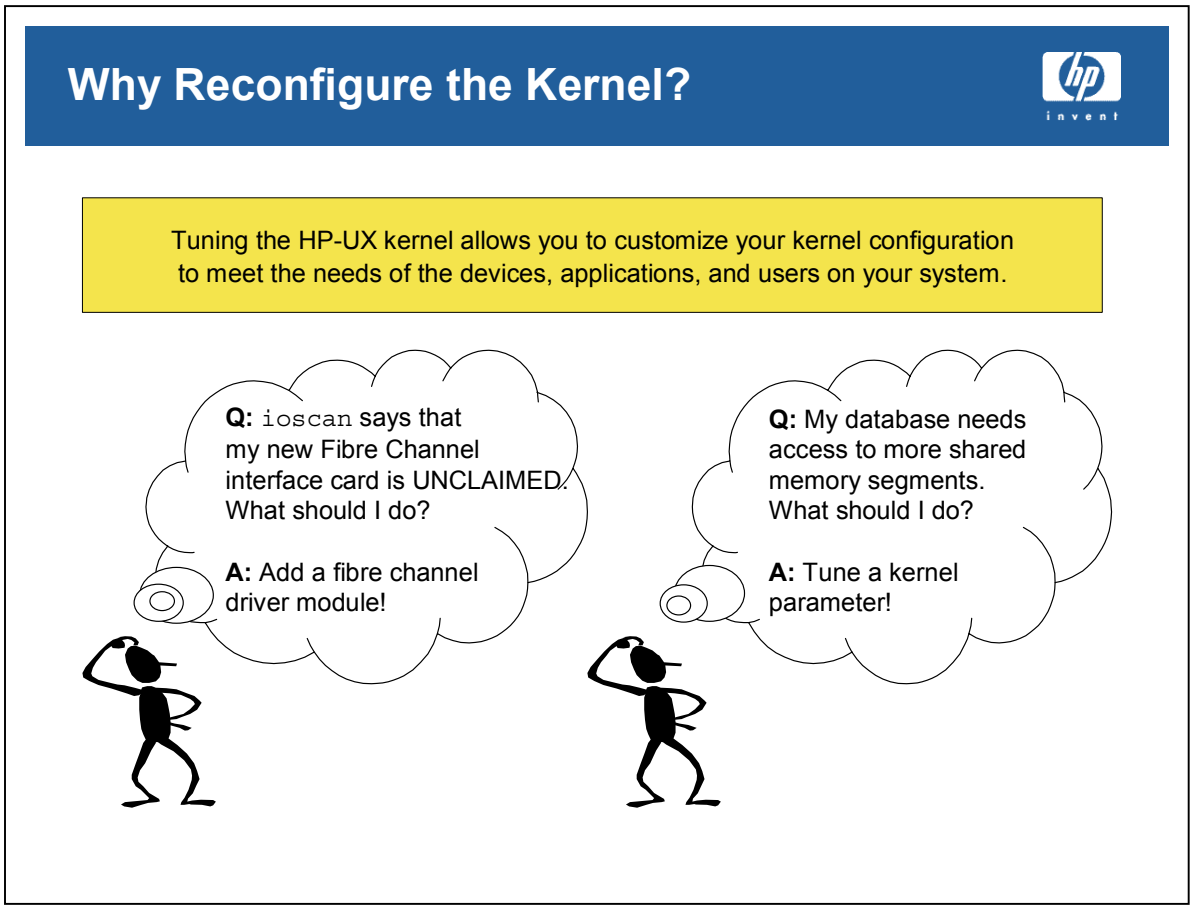
Objectives

Upon completion of this module, you will be able to do the following:

- Describe the purpose of kernel configurations.
- Describe the purpose of kernel modules.
- Describe the purpose of kernel module tunable parameters.
- Describe the purpose of the kernel registry services.
- View, import, export, load, and save kernel configurations with **`kconfig`**.
- Tune the kernel using the **`kcweb`** interface.
- Boot from an alternate kernel.
- Boot to tunable maintenance mode.

© 2004 Hewlett-Packard Development Company, L.P.

19–1. SLIDE: Why Reconfigure the Kernel?



Student Notes

HP-UX systems are used for many purposes. Low-end desktop workstations are often used to run single-user desktop applications, while high-end servers may be used to run enterprise database applications with thousands of concurrent user connections. In order to ensure that your system provides maximum performance, it may be necessary to "tune" the kernel to match your users' and applications needs.

Driver Modules	Kernel drivers contain the code that the kernel requires to communicate with interface cards, devices, and other hardware components. When you add a new device or interface card to your system, you may need to add a driver to the kernel. Conversely, if you remove a device or interface card, it may be beneficial to remove the associated driver from the kernel to save space in RAM.

© 2004 Hewlett-Packard Development Company, L.P.

Subsystem Modules

HP-UX includes several subsystems that may be managed by the system administrator to add or remove system functionality. The **`lvm`** module allows the kernel to configure and use logical volumes. The **`cdfs`** module allows the kernel to access CDFS file systems. These are a couple of the most commonly used kernel subsystems that you may choose to configure.

Tunable Kernel Parameters

Tunable kernel parameters affect the behavior of the system as well as the size of the kernel. Many kernel table sizes are determined by system parameters. For example, the maximum number of concurrent processes is determined by the size of the process table. The system parameter **`nproc`** determines the size of the process table. Extreme care should be taken when modifying system parameters. Oftentimes, when installing software, such as databases, the software vendor will advise you to modify system parameters in order for their product to function properly.

© 2004 Hewlett-Packard Development Company, L.P.

19–2. SLIDE: Kernel Configuration Command Overview

Kernel Configuration Command Overview

HP-UX provides several commands for configuring & managing the kernel.
This course will focus on the web-based configuration utility, `kcweb`.

Command	Purpose
`kcweb`	web-based configuration tool
`kconfig`	command-line tool for managing kernel configurations
`kcpath`	command-line tool for determining the current kernel path
`kcmodule`	command-line tool for managing kernel modules
`kctune`	command-line tool for managing tunable parameters
`kcusage`	command-line tool for monitoring resource usage
`kcalarm`	command-line tool for setting kernel tunable parameter alarms
`kclog`	command-line tool for viewing the kernel configuration log

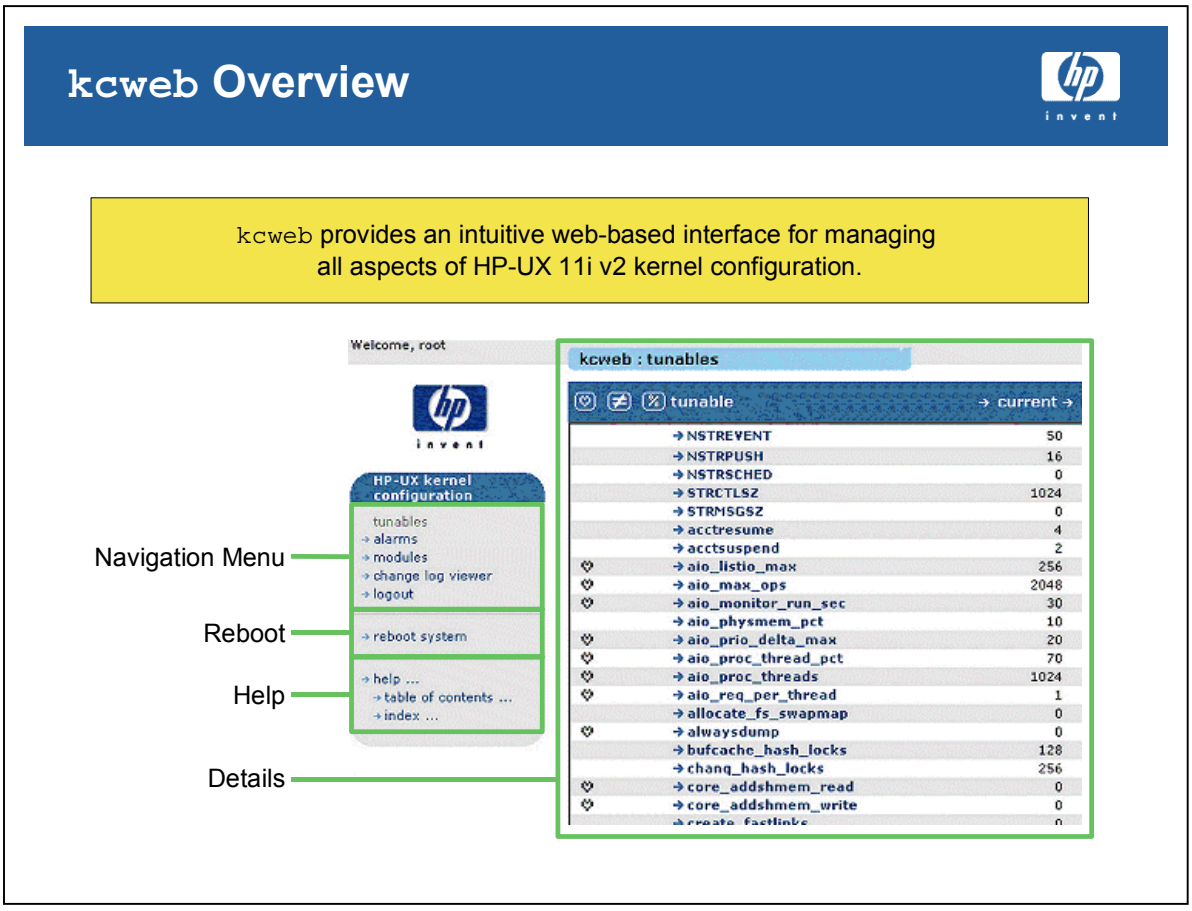
Student Notes

In HP-UX 11i v2, you can tune the kernel using a web-based interface called **`kcweb`** or a collection of command-line utilities. The table below provides a brief explanation of each command. Though we will briefly discuss a few of the command-line utilities, this course will focus primarily on tuning the kernel via the web-based interface.

Command	Purpose
`kcweb`	web-based configuration tool
`kconfig`	command-line tool for managing kernel configurations and **`system`** files
`kcpath`	command-line tool for determining the current kernel path
`kcmodule`	command-line tool for managing kernel modules
`kctune`	command-line tool for managing tunable parameters
`kcusage`	command-line tool for monitoring resource usage
`kcalarm`	command-line tool for setting kernel tunable parameter alarms
`kclog`	command-line tool for viewing the kernel configuration log

© 2004 Hewlett-Packard Development Company, L.P.

19–3. SLIDE: `kcweb` Overview



Student Notes

`kcweb` provides an intuitive, web-based interface for managing all aspects of HP-UX 11i v2 kernel configuration.

The navigation menu at top left provides links to the functional areas available in **`kcweb`**. From this menu, you can choose to manage tunable parameters, kernel resource alarms, kernel modules, or the kernel log,

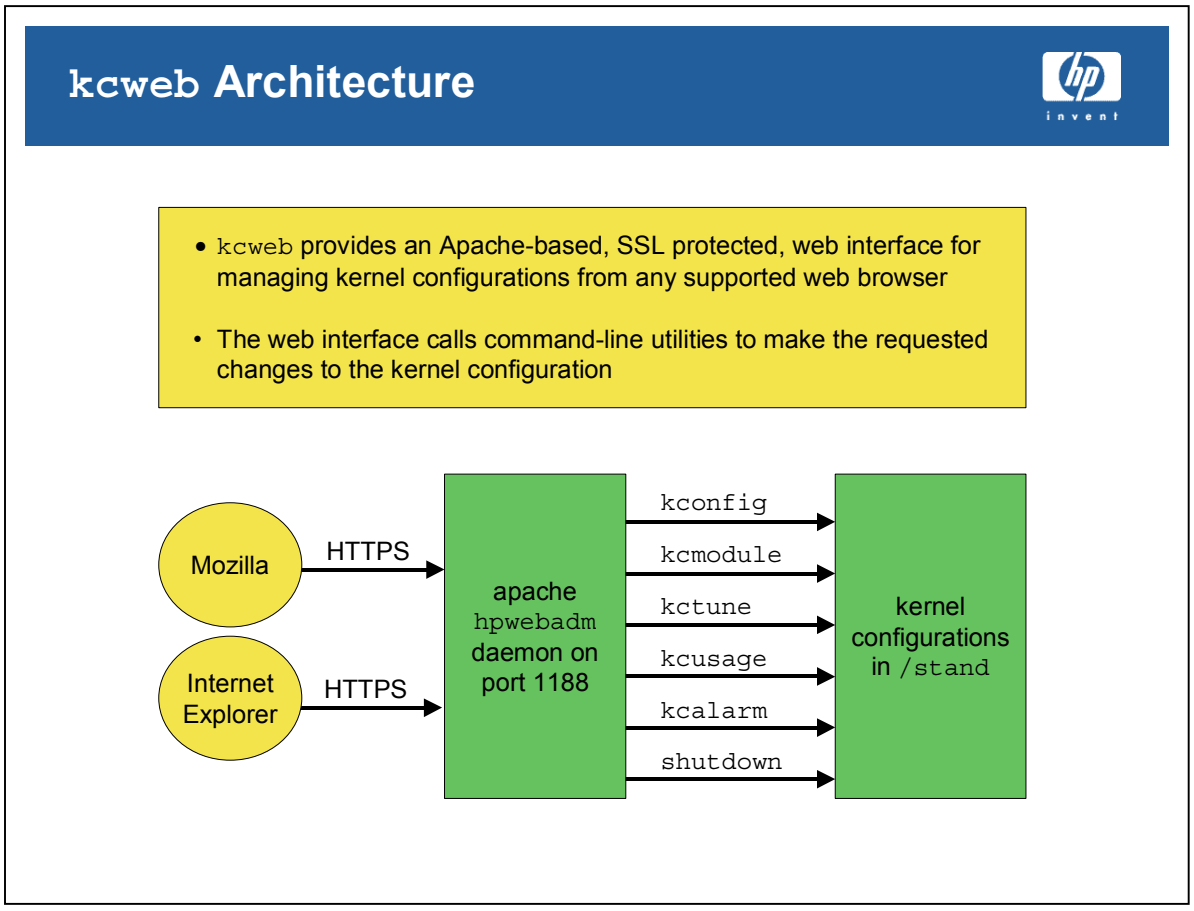
The **`reboot`** button at middle left allows you to reboot the system without leaving the web interface. This is particularly convenient when you are tuning the kernel remotely via your web browser.

The help area provides a searchable index of help topics, including detailed information about each tunable kernel parameter.

When you select a functional area from the navigation menu, **`kcweb`** displays a detail screen in a pane on the right. Later in the chapter you will have an opportunity to explore the detail panes in much greater detail.

© 2004 Hewlett-Packard Development Company, L.P.

19–4. SLIDE: `kcweb` Architecture



Student Notes

`kcweb` service is provided by a dedicated Apache/hpwebadm server daemon, which runs on port#1188. Administrators can access the **`kcweb`** interface via a Mozilla 1.2 or Netscape 4.0 or 6.0 web browser running locally on the HP-UX server, or via Internet Explorer 5.0 or 6.0 running on a PC (Note that Netscape 7.0 is not supported). You must have cookies enabled on your browser to use **`kcweb`**. Cookies are used to track user sessions.

`kcweb` uses Secure Socket Layer (SSL) technology to secure traffic passing between the Apache/hpwebadm daemon and the administrator's browser. Apache/hpwebadm is the same daemon that provides the **`pdweb`** web interface for managing peripheral devices and interface cards.

When **`kcweb`** receives a request to modify the kernel configuration, it uses standard command line utilities such as **`kctune`** and **`kcmodule`** to make the changes. These commands are logged in **`/var/sam/log/samlog`**, which may be viewed by typing **`/usr/sam/bin/samlog_viewer`**.

© 2004 Hewlett-Packard Development Company, L.P.

19–5. SLIDE: Using `kcweb` via a Local Browser

Using `kcweb` via a Local Browser

To manage kernel configurations on your localhost, simply run `kcweb`. `kcweb` will:

- Create SSL server certificates if needed
- Launch the Apache/administration server if it isn't already running
- Launch a web browser
- Shutdown the server daemon 5 minutes after you close your browser

Verify that your DISPLAY variable is set

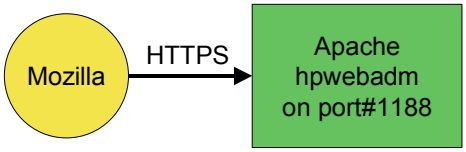
```
# echo $DISPLAY
```

Verify that your BROWSER variable is set

```
# export BROWSER=/opt/mozilla/mozilla
```

Run kcweb and enter the root username and password when prompted

```
# kcweb [-F] &
```



Student Notes

Administrators can access the **`kcweb`** interface from a browser running locally on the target server, or from a browser running remotely on another UNIX system or PC.

Several steps are required if you wish to access the **`kcweb`** interface on the localhost:

1. Verify that your **`DISPLAY`** variable is set. Running **`kcweb`** via a remote X-windows session or X-windows emulator such as ReflectionX is risky since X-windows traffic isn't secure. In fact, **`kcweb`** automatically verifies that the **`kcweb`** server's hostname matches the hostname in the **`DISPLAY`** variable. If the hostnames don't match, **`kcweb`** won't start unless you add the **`-F`** (force) option. If you need to configure a kernel on a remote target server, consider using the more secure technique described on the next slide.

   ```
   # echo $DISPLAY
   ```

2. Verify that your **`BROWSER`** variable is set. If the **`BROWSER`** variable isn't defined, **`kcweb`** launches **`/opt/mozilla/mozilla`**, the default browser in 11i v2.

   ```
   # export BROWSER=/opt/mozilla/mozilla
   ```


© 2004 Hewlett-Packard Development Company, L.P.

3. Run **kcweb** and enter the root username and password when prompted. If **kcweb** is executed without any options, the server daemon will stop automatically after a period of inactivity. If you want the server daemon to continue running until the next reboot, add the **-s startssl** option. If you are accessing the target system via a remote X-windows session, add the **-F** (force) option, too.

   ```
   # kcweb &
   ```

   kcweb will:

   - Create SSL certificates for the server if they haven't already been created.
   - Launch the Apache/hpwebadm server daemon if it isn't already running.
   - Launch the web browser defined in your **BROWSER** environment variable.

© 2004 Hewlett-Packard Development Company, L.P.

19–6. SLIDE: Using `kcweb` via a Remote Browser

Using `kcweb` via a Remote Browser

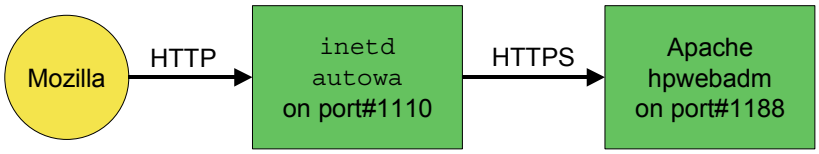
If you prefer to access the `kcweb` functionality from a remote browser, configure the inetd/`autowa` daemons to start the server daemon when necessary.

Enable/disable inetd autostart functionality for Apache/hpwebadm daemon

```
server# waconf -a on|off
server# netstat -an | grep 1110
        tcp 0 0 *.1110 *.* LISTEN
```

Launch a web browser and enter the root username/password when prompted

```
client# mozilla http://server:1110/casey/top.cgi
```



Student Notes

It is also possible to access the Apache/hpwebadm daemon from a remote web browser on a PC or another UNIX system. In order for this to work, though, you must first start the Apache/hpwebadm daemon on the system you wish to manage. There are two ways to accomplish this: you can either start and stop the Apache/hpwebadm daemon manually, or you can configure the **`autowa`** daemon to start the Apache/hpwebadm daemon for you on an as-needed basis.

Manually starting the Apache/hpwebadm daemon for remote access

If you prefer to manually start the Apache/hpwebadm daemon each time you need to modify the kernel, use this procedure.

1. Remotely login on the target server via **`telnet`**, **`rlogin`**, or SSH, and start the Apache/hpwebadm daemon

   ```
   server# kcweb -s startssl
   server# kcweb -s status
   ```

2. Back on your local client system, point your browser to the server's URL and enter the root password when prompted. Be sure to type **`https`** in the URL rather than **`http`** since

© 2004 Hewlett-Packard Development Company, L.P.

kcweb uses Secure Socket Layer (SSL) technology to encrypt and authenticate your session. If this doesn't work, verify that your firewall allows access from your local desktop to port 1188 on the target server.

```
client# mozilla https://server:1188/casey/top.cgi
```

3. When you are done configuring the kernel the Apache/hpwebadm daemon will run for a few more minutes, the die automatically after a period of inactivity. If you prefer to stop the daemon explicitly, remotely login on the target server and use the **kcweb -s stop** command to kill the server process.

```
server# kcweb -s stop
server# kcweb -s status
```

Automatically Starting the Apache/hpwebadm Daemon for Remote Access

If you typically access the **kcweb** functionality from a remote browser, save yourself the hassle of manually starting the server daemon each time by enabling **kcweb** autostart functionality in **/etc/inetd.conf**.

The autostart functionality causes the **inetd** daemon to listen for incoming **kcweb** requests on port#1110. When a browser contacts port#1110, **inetd** launches the **/opt/hpwebadmin/lbin/autowa** (auto web admin) server process. **autowa**, in turn, launches Apache/hpwebadm on port#1188. Shortly after the session terminates, the Apache/hpwebadm daemon dies. When **inetd** receives another request for service, it launches **autowa** again.

Two simple steps are required to configure the autostart functionality.

1. Enable/disable **inetd** autostart functionality for the Apache/hpwebadm service in **/etc/inetd.conf** and **/etc/services**. You can modify the appropriate lines automatically by running the **waconf** command, or you can use the **vi** editor to manually remove the **#** sign preceding the **autowa** lines in **/etc/inetd.conf** and **/etc/services**. Verify that **inetd** is **LISTEN**ing for incoming requests on port 1110.

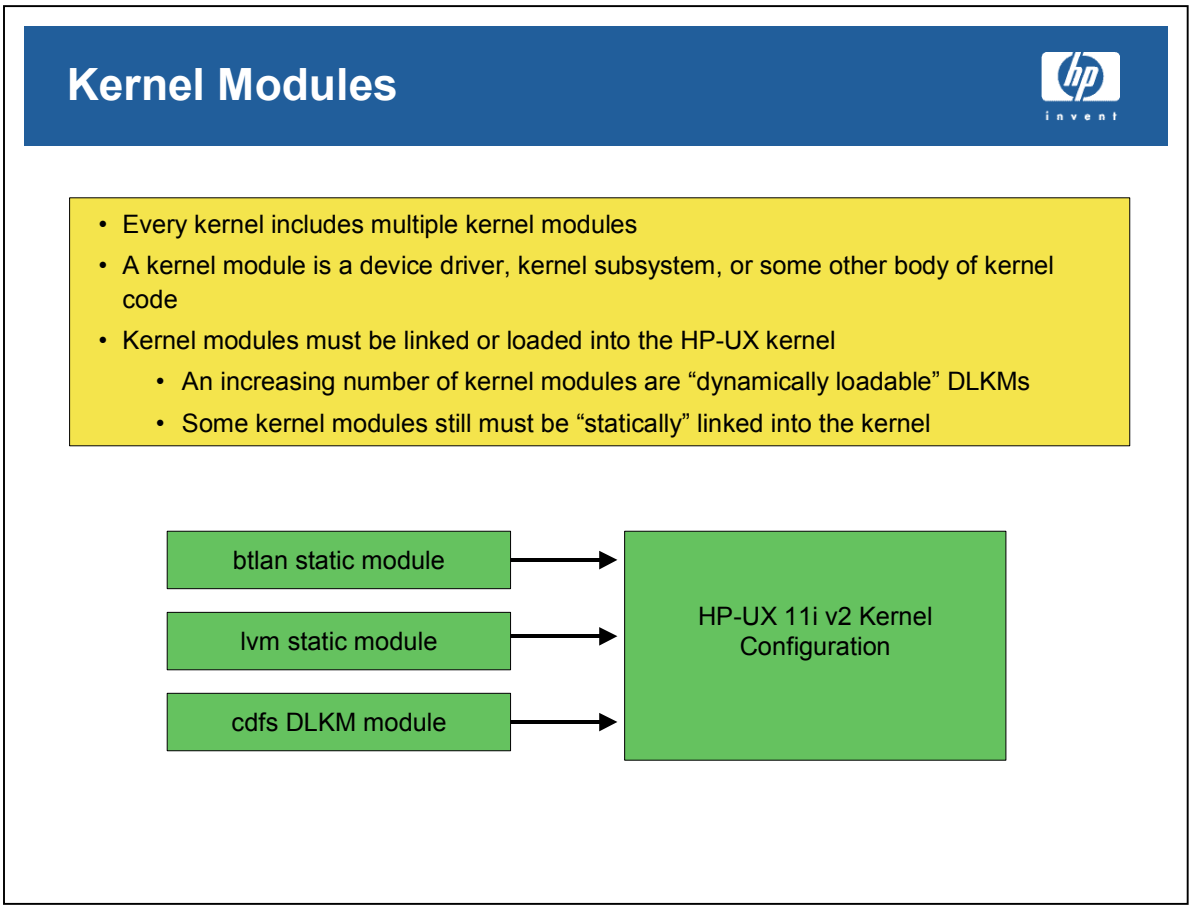
```
server# waconf -a on|off
server# netstat -an | grep 1110
        tcp 0 0 *.1110 *.* LISTEN
```

2. Launch a web browser and enter the root username/password when prompted. If this doesn't work, make sure your firewall permits traffic from your browser client to port#1110 and port#1188 on the target server.

```
client# mozilla https://server:1110/casey/top.cgi
```


© 2004 Hewlett-Packard Development Company, L.P.

19–7. SLIDE: Kernel Modules



Student Notes

The HP-UX kernel contains an administrator-defined list of kernel modules, each of which may contain a driver, a subsystem, or some other body of kernel code. In order to access a peripheral device, the kernel must include the kernel driver module associated with that device. There are also kernel modules that provide major features of the operating system such as CDFS file systems, VxFS file systems, LVM, and NFS. Kernel modules can be added and removed as system needs change.

Using Kernel Modules

There are two basic ways to use a kernel module. Traditionally, kernel modules have been statically bound into the main kernel executable, **`/stand/vmunix`**. This method is simple, but it has some significant disadvantages. In particular, it requires system administrators to re-link the kernel executable and reboot the system in order to add, remove, or patch a kernel module.

Since HP-UX 11.00, HP-UX has supported Dynamically Loadable Kernel Modules (DLKMs). A DLKM can be loaded into the running kernel without rebuilding **`/stand/vmunix`** or rebooting. Most dynamically loadable modules are also dynamically unloadable, so they can be removed from the running kernel without rebooting. System administrators prefer kernel

© 2004 Hewlett-Packard Development Company, L.P.

modules that support dynamic loading because they can reconfigure their kernel more quickly, and without requiring costly downtime. Module developers prefer dynamically loadable modules because they can test changes without having to wait for systems to reboot.

However, there are some disadvantages to dynamically loadable modules, too. There is a slight performance penalty when a module is dynamically loaded. It usually isn't noticeable, but may be an issue in some performance-critical environments. For this reason, some system administrators may prefer to dynamically load and unload modules while determining the desired system configuration, but may choose to configure a static kernel containing those modules for production use.

As of this writing, HP does not yet support dynamic unloading of certain types of modules. This support will be added in a future release of HP-UX.

© 2004 Hewlett-Packard Development Company, L.P.

19–8. SLIDE: Kernel Module States

Kernel Module States

- A module's state determines if/when the module will be accessible on the system
- HP-UX 11i v2 recognizes five different module states

State	Description
unused	The module isn't included in the kernel configuration
static	The module is statically bound into the `vmunix` kernel executable
auto	The module will be dynamically loaded into the kernel when needed
loaded	The module has been dynamically loaded into the kernel
best	The module is in the "best" state, as defined by the module developer

Student Notes

System administrators specify how a module should be used by putting it into one of five states: unused, static, loaded, best, or auto. These states are described in the table below.

State	Description
`unused`	The module isn't included in the kernel configuration.
`static`	The module is statically bound into the **`vmunix`** executable.
`auto`	The module will be dynamically loaded into the kernel when something tries to use it.
`loaded`	The module is dynamically loaded into the kernel.
`best`	The module will be put into the state identified by the kernel module developer as its "best" state. Typically this will be **`auto`** (if supported by the module). If auto isn't supported, it will likely be loaded (if the module is dynamically loadable). If the module supports neither the **`auto`** nor the **`loaded`** states, the best available state will likely be **`static`**.

Some modules do not support all states. To see which states a module supports, run

© 2004 Hewlett-Packard Development Company, L.P.

`kcmodule -v` ***`modulename`***. You can change a module's state via the **`kcmodule`** command or **`kcweb`**.

Changing a module's state may or may not require a kernel re-link and reboot. The table below describes exactly what happens when changing between various module states.

© 2004 Hewlett-Packard Development Company, L.P.

From state…	To state… `unused`	To state… `static`	To state… `loaded`	To state… `auto`
`unused`	N/A	kernel re-linked with the module; will be used at next boot	module loaded dynamically; no reboot needed**	module marked for load at next use; no reboot needed**
`Static`	static kernel re-linked without the module; will be unused at next boot	N/A	kernel re-linked without the module; will be loaded dynamically during next boot	kernel re-linked without the module; will be marked for load at first use after next boot
`Loaded`	loaded module unloaded dynamically; no reboot*,**	kernel re-linked with module; will be used at next boot	N/A	module remains loaded; will be marked for load at first use after next boot
`Auto`	module will not be loaded at next use, or at first use after next boot	kernel re-linked with the module; will be used at next boot	module loaded dynamically; no reboot**	N/A

* If the module is busy, the change will be held for next boot.
** To force the change to be held for next boot, add the **`kcmodule -h`** option or select the appropriate checkbox in **`kcweb`**.

© 2004 Hewlett-Packard Development Company, L.P.

19–9. SLIDE: Kernel Module State Changes

Kernel Module State Changes

- The system administrator can force a module state change via `kcmodule` or `kcweb`
- Sometimes, HP-UX may change a module's state automatically

Cause	Description
explicit	The system administrator explicitly chose the current state
best	The system administrator chose to use the module, but didn't choose a specific state, so the module is placed in its "best" state, as determined by the module developer
auto	The module was in the auto state, and was automatically loaded when something tried to use it (DLKMs only)
required	The module is required by the kernel
depend	The module is in use because some other module in the configuration depends on it

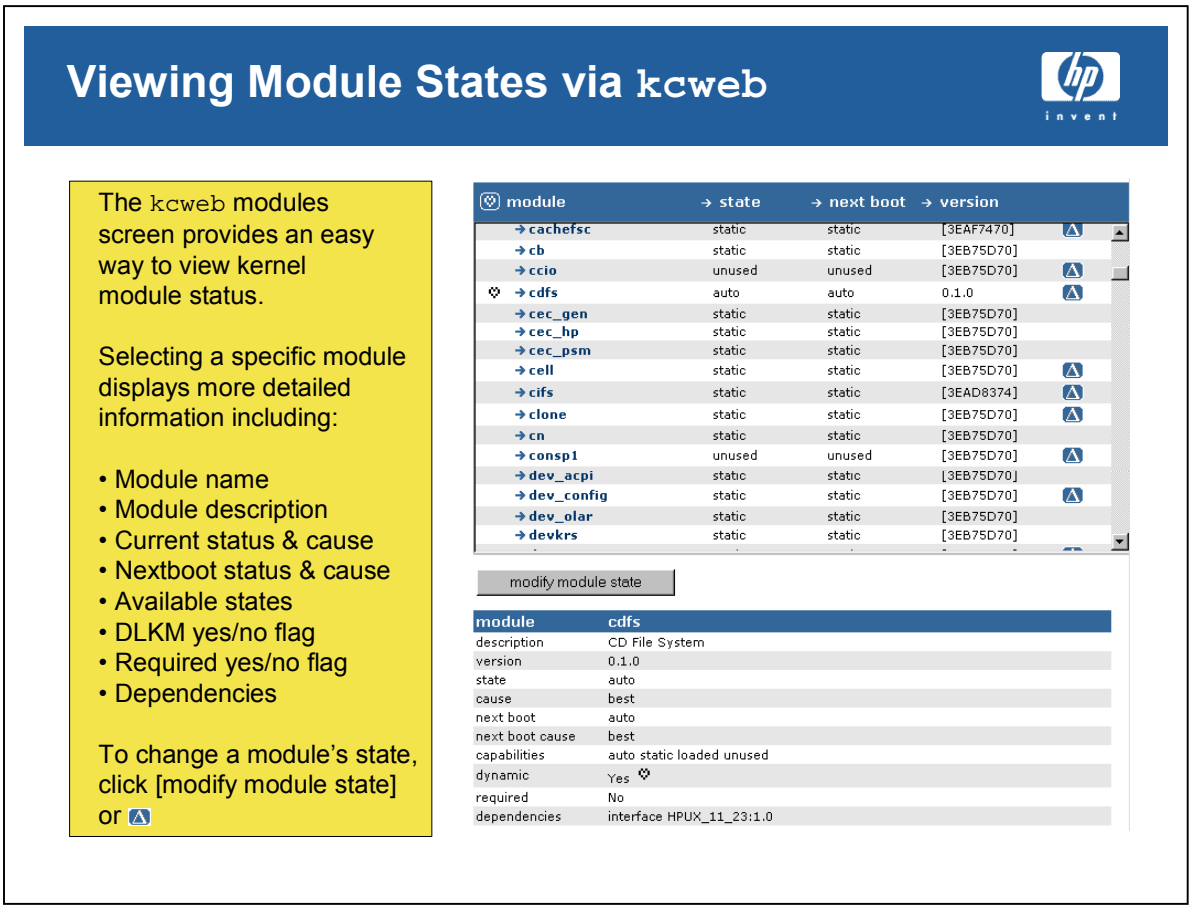
Student Notes

Occasionally when you add a new interface card or peripheral device, it may be necessary to add a new kernel module to your kernel configuration. The system administrator can force a module state change via the **`kcmodule`** or **`kcweb`** command. HP-UX may change a module's state automatically, too. The **`kcmodule`** command reports what caused each module to enter its current state. The table below lists the reasons a module might change state.

Cause	Description
`explicit`	The system administrator explicitly chose the current state.
`best`	The system administrator chose to use the module, but didn't choose a specific state, so the module is placed in its "best" state, as determined by the module developer.
`auto`	The module was in the auto state, and was automatically loaded when something tried to use it (DLKMs only).
`required`	The module is required by the kernel.
`depend`	The module is in use because some other module in the configuration depends on it.

© 2004 Hewlett-Packard Development Company, L.P.

19–10. SLIDE: Viewing Module States via `kcweb`



Student Notes

The **`kcweb modules`** screen provides an easy way to both view and manage kernel modules. After launching **`kcweb`**, select the **`modules`** link on the **`kcweb`** navigation pane. This should generate a summary list of the modules in your current configuration in a pane at top right. The columns in the report are described below.

Column	Description
	Is the module dynamically loadable?
`module`	What is the module's name?
`State`	What is the module's current state?
`next boot`	What will be the module's state after nextboot?
`version`	What is the module's version number?

To sort the list of modules based on any column's contents, click the appropriate column header.

To view module details, click the → arrow to the left of the desired module name. The selected module's details should appear in a detail pane at bottom right. The module detail pane reports:

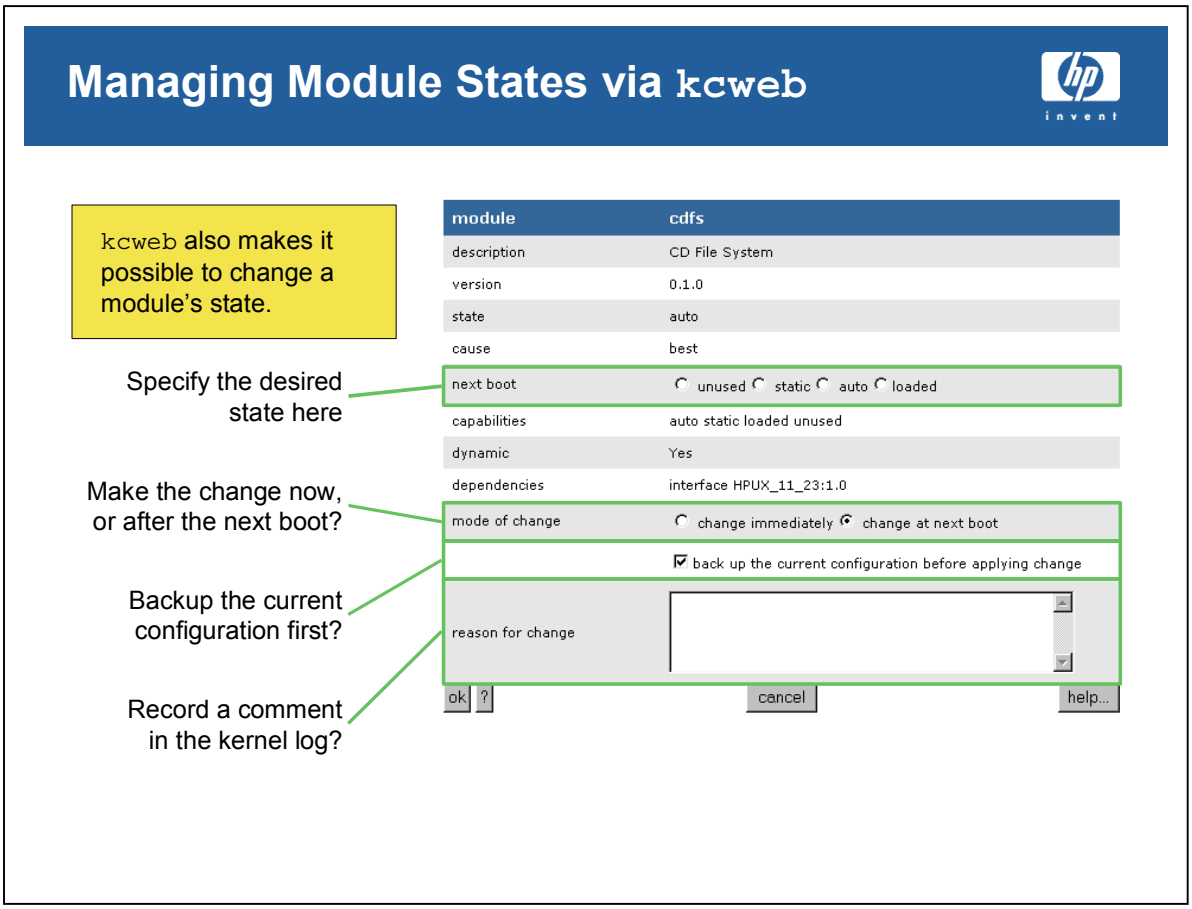
© 2004 Hewlett-Packard Development Company, L.P.

Field	Description
module	What is the module's name?
description	What does the module do?
version	What is the module version?
state	What is the module state?
cause	What caused the module to enter it's current state?
next boot	What will be the module's state after nextboot?
next boot cause	What will determine the module's state after nextboot?
capabilities	What states does the module support?
dynamic	Is the module dynamically loadable?
required	Is the module a required module?
dependencies	Is the module dependent on any other modules?

To change a module's state, click the icon to the right of the parameter you wish to modify, or click the **[modify module state]** button in the middle of the screen.

© 2004 Hewlett-Packard Development Company, L.P.

19–11. SLIDE: Managing Module States via `kcweb`



Student Notes

You can also change a module's state from the **`kcweb modules`** screen. To change a module's state, click the icon to the right of the parameter you wish to modify, or click the **`[modify module state]`** button in the middle of the screen. Either action should open a dialog box similar to the one shown on the slide above. The dialog box contains several fields.

Field	Description
`module`	What is the module's name?
`description`	What does the module do?
`version`	What is the module version?
`state`	What is the current module state?
`cause`	What caused the module to enter it's current state?
`next boot`	What will be the module's state after nextboot?
`capabilities`	What states does the module support?
`dynamic`	Is the module dynamically loadable?
`dependencies`	Is the module dependent on any other modules?
`mode of change`	Change immediately if possible, or hold for next boot?
`backup?`	Do you wish to backup the current configuration?

© 2004 Hewlett-Packard Development Company, L.P.

reason for change	Why are you making the change?

Modify the fields as desired, then click **[OK]** to make the changes. If the tunable is static, or if you selected the **change at next boot** option, the change won't take effect until you reboot.

Changing Module States from the Command Line

If you prefer to manage module states from the command line, you can use the **kcmodule** command-line utility. Some common options are described in the table below.

Command	Description
kcmodule	Shows all optional modules and their current states.
kcmodule –a	View all modules, including required modules which aren't normally modified by the system administrator.
kcmodule -v *module*	Shows verbose information about a module.
kcmodule *module=state*	Sets the module state. Sometimes it is not possible to make the change immediately; for example the **cdfs** module can't be unloaded while a CD file system is mounted. A change that moves a module in or out of the **static** state can never be applied immediately; the change will always be held for next boot. You can specify multiple changes on the **kcmodule** command line. However, if any change on the **kcmodule** command line has to be held for next boot, they all will be.
kcmodule -h *module=state*	Sets module state, but holds until next boot. Sometimes you may want to force a change to be held for next boot, instead of applying it immediately. In this case, use the **kcmodule –h** (hold) command to force this behavior. HP recommends that this flag be used only when the next boot is expected to be soon. Before rebooting a system, the administrator should execute **kconfig -D** to review changes being held for next boot.
kcmodule -K *module=state*	Don't backup the kernel before changing state. When changing module states, **kcmodule** supports the **–B** and **–K** flags to specify backup behavior. The default **-B** options specifies that a

© 2004 Hewlett-Packard Development Company, L.P.

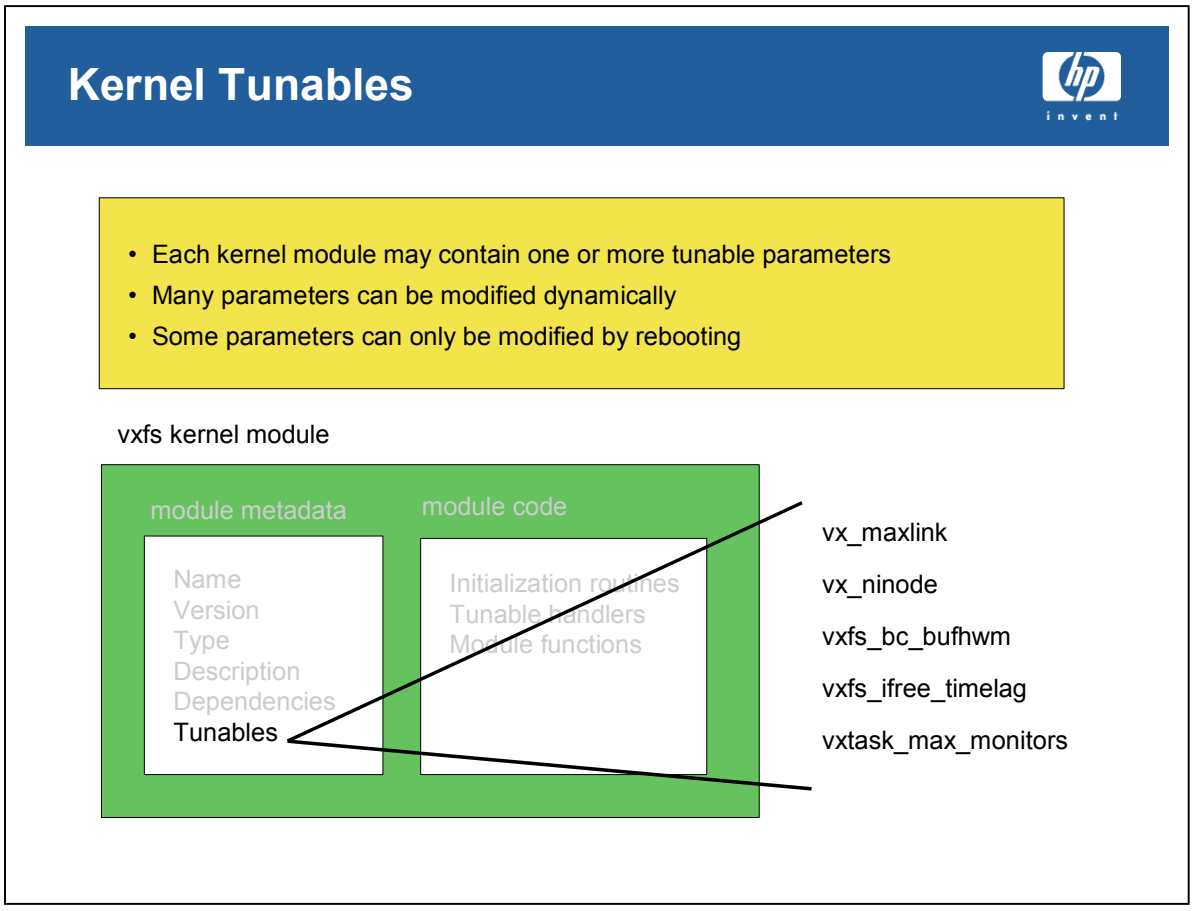
	backup of the current configuration should be made before applying changes. The **-K** option explicitly prevents the writing of the backup configuration.
kcmodule -c *config mod=state*	Apply changes to a named configuration rather than the **current** configuration.
kcmodule -D	Show changes held for next boot. Always review pending changes before rebooting!

In all of these examples you can add the **-C "*comment*"** option to record a comment in **/var/adm/kc.log**.

When modules are moved in or out of the **static** state, the **kcmodule** command will run for quite a while. This is because such changes require that the kernel executable be re-linked. If you have multiple changes to make, it is best that you list them all on the same **kcmodule** command line or make the changes in a system file and import it (using **kconfig -i**). Either of these techniques will ensure that the kernel executable is re-linked only once.

© 2004 Hewlett-Packard Development Company, L.P.

19–12. SLIDE: Kernel Tunables



Student Notes

Every kernel module may contain one or more tunable parameters. The system administrator can modify these tunable parameters based on the needs of the applications and users on the system. For instance, if you install a new application on the system, it may be necessary to increase the **`nproc`** kernel parameter, which determines the maximum number of processes in the process table, or **`nfile`**, which determines the maximum number of files that may be open at any given time.

Some kernel parameters are dynamically tunable, while others are static. Changes made to dynamically tunable kernel parameters immediately impact the running kernel. Changes made to static kernel parameters don't take effect until the system reboots.

© 2004 Hewlett-Packard Development Company, L.P.

19–13. SLIDE: Types of Tunables

Types of Tunables

HP-UX 11i v2 supports five different types of tunable parameters

Type	Description
Static	Changes require re-linking, and rebooting
Dynamic	Changes may be applied immediately
Private	Tunables are hidden from listings, but may be tuned
Automatic	Default values are set automatically at boot time, but can be overridden by the administrator
User-Defined	May be used as variables in other tunables' formulas

Student Notes

HP-UX 11i v2 now supports five different types of tunable parameters

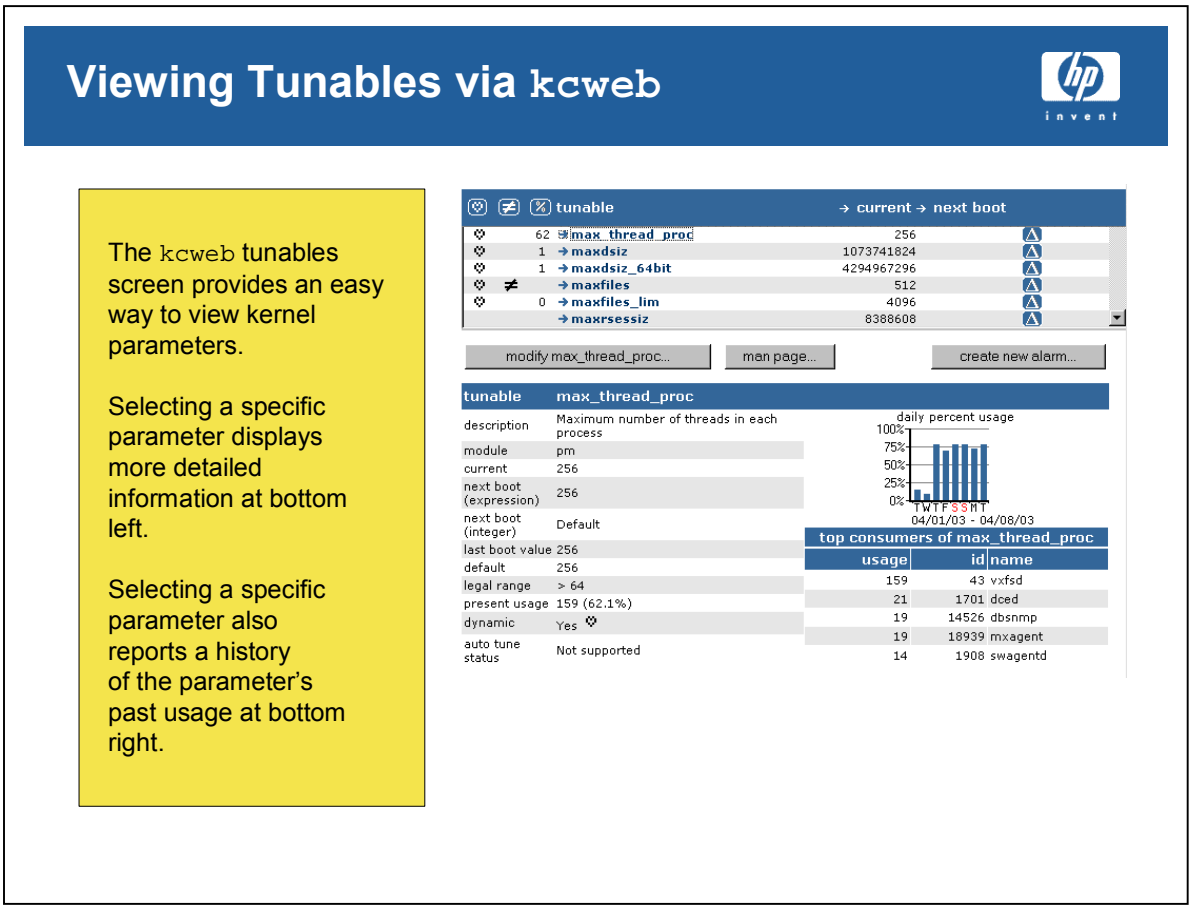
Type	Description
`static`	Static tunable changes require re-linking the kernel, and rebooting.
`dynamic`	Dynamic tunable changes may be applied immediately to the running kernel, without rebooting. 11i v2 introduces many new dynamic tunables.
`private`	Private tunables are only intended for use by developers and HP support personnel. Private tunables can be set and queried like any other tunable, except that they do not appear in **`kctune`**'s or **`kcweb`**'s default tunable lists: they must be explicitly provided as arguments to the **`kctune`** command.

© 2004 Hewlett-Packard Development Company, L.P.

automatic	Every tunable has a default value that is used if the administrator doesn't specify otherwise. Many kernel parameters have fixed default values that remain the same across systems and across reboots. Automatic tunables, however, don't have a fixed default value; rather, a default value is calculated at boot time. For example, the default value of **nflocks** is computed as either 1200 or 4096, depending on the amount of system memory. For dynamic tunables, the default value is calculated whenever the administrator sets the tunable to state **default**; if the tunable is static, it's calculated at boot time (assuming the tunable is set to **default** when the system boots).
user-defined	In HP-UX 11i v2, system administrators can create their own user-defined tunables if they choose. These tunables don't affect the operation of the system directly, but they can be used to compute the values of other tunables. For example, an administrator might choose to create a **num_databases** tunable, and then set several kernel tunables based on the **num_databases** value. A subsequent change to the value of **num_databases** would cause all of the related kernel tunable values to be changed as well. No user-defined tunables are shipped with HP-UX 11i v2. These are left to administrators to define. While **kcweb** supports the use of user-defined tunables, you can only create them via **kctune**. Earlier versions of HP-UX included a pseudo-parameter called **maxusers** that worked much like the new user-defined kernel parameter functionality. Kernel parameters that were previously calculated based on the value of **maxusers** have been converted to automatic kernel parameters. However, you can manually recreate **maxusers** or similar user-defined kernel parameters if you wish.

© 2004 Hewlett-Packard Development Company, L.P.

19–14. SLIDE: Viewing Tunables via `kcweb`



Student Notes

The **`kcweb tunables`** screen provides an easy way to both view and manage kernel tunables. After launching **`kcweb`**, select the **`tunables`** link on the **`kcweb`** navigation pane. This should generate a summary list of the tunables in your current configuration in a pane at top right.

Select a tunable name from the tunable summary list to view details at the bottom of the screen. The tunable details pane has several columns:

Column	Description
	Is the parameter dynamically tunable?
	Has the parameter been modified from the default?
	What percentage of the resource is being used?
`tunable`	What is the tunable's name?
`current`	What is the current value for this tunable?
`nextboot`	What is the nextboot value for this tunable?

To sort the tunables based on any column, just click the appropriate column header.

© 2004 Hewlett-Packard Development Company, L.P.

To view a particular tunable's details, click the → icon to the left of the tunable name. This displays a tunable detail pane at the bottom right of the screen. The tunable detail screen reports several fields:

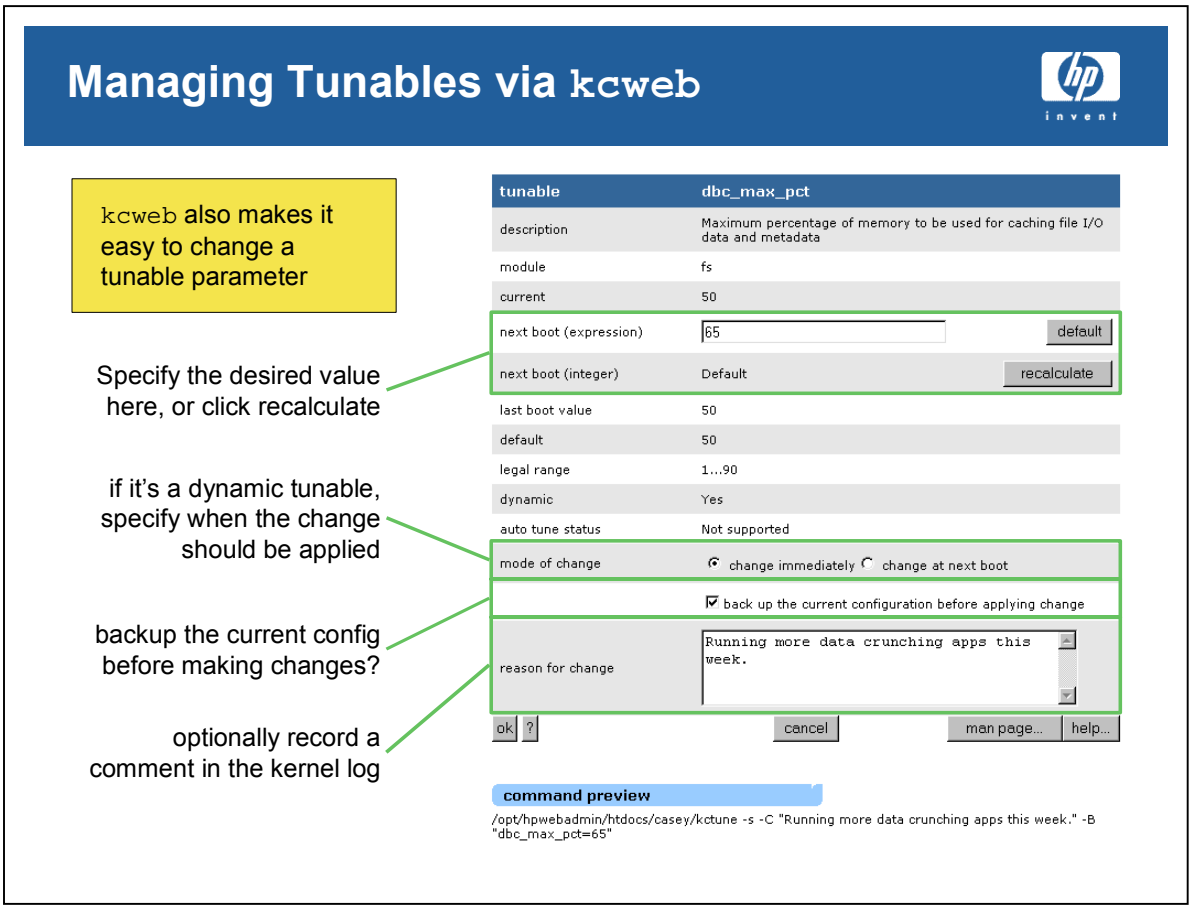
Field	Description
`Tunable`	What is the tunable's name?
`description`	What does the tunable control?
`module`	What module does the tunable belong to?
`current`	What is the current tunable value?
`next boot (integer)`	What will the tunable's value be at next reboot?
`next boot (expression)`	What formula (if any) determines the tunable's value?
`last boot value`	What was the tunable's value at last boot?
`default`	What is the tunable's default value?
`legal range`	What is the tunable's legal range?
`present usage`	What % of the tunable's resource is currently in-use?
`dynamic`	Is the parameter dynamically tunable?
`auto tune status`	Is the parameter automatically tunable?
`constraints`	Are there any dependencies on other tunables?

For some tunables, you may see a usage graph and other usage information in a pane at bottom right. Resource usage reporting and alarms will be discussed later in this chapter.

To change a tunable's value, click the icon to the right of the tunable, or click `[modify]`.

© 2004 Hewlett-Packard Development Company, L.P.

19–15. SLIDE: Managing Tunables via `kcweb`



Student Notes

`kcweb` provides a convenient, intuitive mechanism for modifying tunable parameter values.

Simply click the icon to the right of the desired tunable, and click through the resulting dialog box. The dialog box contains several fields:

© 2004 Hewlett-Packard Development Company, L.P.

Field	Description
tunable	What is the tunable's name?
description	What does the tunable control?
module	What module does the tunable belong to?
current	What is the current tunable value?
next boot (expression)	What will be the tunable value at next reboot?
next boot (integer)	What formula (if any) determines the tunable's value?
last boot value	What was the tunable's value at last boot?
default	What is the tunable's default value?
legal range	What is the tunable's legal range?
dynamic	Is the parameter dynamically tunable?
auto tune status	Is the parameter auto-tunable at boot time?
constraints	Are there any constraints or dependencies?
mode of change	Change immediately if possible, or hold for next boot?
backup?	Do you wish to backup the current configuration?
reason for change	Why are you making the change?

Modify the fields as desired, then click **[OK]** to make the change. If the tunable is static, or if you selected the **change at next boot** option, the change won't take effect until you reboot.

Tunable Range Checking

Whenever you attempt to change the value of a tunable, the kernel compares the new value to the legal range field in the module's metadata. **kctune** reports any out-of-range errors, which **kcweb**, in turn, displays to the user. Tunable handler functions (not accessible by the customer) also validate proposed values for the tunables they control.

Some tunables have constraints on their values, which are shown in the verbose output. Sometimes these are minimum and/or maximum values. Other times there are fixed relationships between tunables (for example, **acctresume** must be greater than **acctsuspend**) or restrictions on the allowed values (for example, **dnlc_hash_locks** must be a power of two). kcweb and kctune enforce these constraints whenever you chage a tunable parameter.

There are other constraints, not shown by **kcweb** or **kctune**, that are based on the current state of the system and can change over time (for example, **nproc** cannot be set to less than the number of processes currently running). These constraints are enforced only when changing the currently running system, and not when making changes held for use at next boot or changes to a saved configuration.

A common kind of validation check enforces a required relationship between two or more tunables. For example, the tunable **nkthread** (number of kernel threads) must always be set to a greater value than the tunable **nproc** (number of processes).

Every tunable should have a man page which documents allowed ranges and dependencies.

© 2004 Hewlett-Packard Development Company, L.P.

`kcweb` Limitations

Some tunable management tasks are only available via **`kctune`**.

- **`kcweb`** doesn't provide a mechanism for creating user-defined parameters.
- **`kcweb`** may only be used to modify the current configuration, not named configurations.
- **`kcweb`** doesn't provide a mechanism for modifying private tunables.

Modifying Tunables via `kctune`

If you prefer to manage kernel tunables from the command line, you can use the **`kctune`** command-line utility.

Command	Description
`kctune`	List all tunable values
`kctune -D`	List values held for next boot
`kctune 'maxuprc=512'`	Set a value immediately or [-h]old for nextboot
`kctune 'maxuprc=nproc/30'`	Define a formula
`kctune 'maxuprc=default'`	Allow system to use the default value
`kctune 'maxuprc+=32'`	Increment a tunable
`kctune 'maxuprc>=1024'`	Set to n if not already greater than n
`kctune -u mytunable=256`	Create a user-defined tunable
`kctune -c myconfig maxuprc=512`	Set tunable in a saved configuration

In all of these examples you can add the **`-C "comment"`** option to record a comment in **`/var/adm/kc.log`** kernel change log.

`kctune` automatically verifies that all parameters are within allowed boundaries.

Changes to static tunables and tunables modified via **`kctune -h`** will be held until next boot. Other changes should take effect immediately.

More About User-Defined Parameters

User-defined parameters can only be configured via the command-line, and follow C-style identifier syntax. Here are some of the guidelines for choosing the tunable name:

- A name may not exceed 32 characters. Most names are 8-16 characters. Names must start with a letter, and may contain only letters, digits, and underscores.
- Tunable names typically begin with a module name, module vendor name, or some other mnemonic that ensures the tunable name will be unique. In keeping with this, consider using a similar mnemonic scheme for user-defined tunables. (This rule also ensures that related tunables sort together.)
- Subsequent words in a tunable name should be separated by underscores.
- Use all lower case unless the meaning cannot be understood without capitalization.

© 2004 Hewlett-Packard Development Company, L.P.

- Do not start or end a name with an underscore.

- Omit "num" and "max" from a tunable name unless they are needed to contrast with another tunable name (e.g. a "min" of the same thing). If an abbreviation is necessary, only use common abbreviations such as "min" and "max".

Learning More about Tunable Parameters

In past versions of HP-UX, tunable parameters were often poorly documented. HP-UX 11i v2 includes an online man page in manual section 5 for every non-private tunable parameter. You can view these man pages as you would any other man page.

```
# man 5 maxuprc
```

The kernel tunable man pages are also available on **http://docs.hp.com**, and via the **[help]** button in **kcweb**. A sample tunable parameter man page is included below.

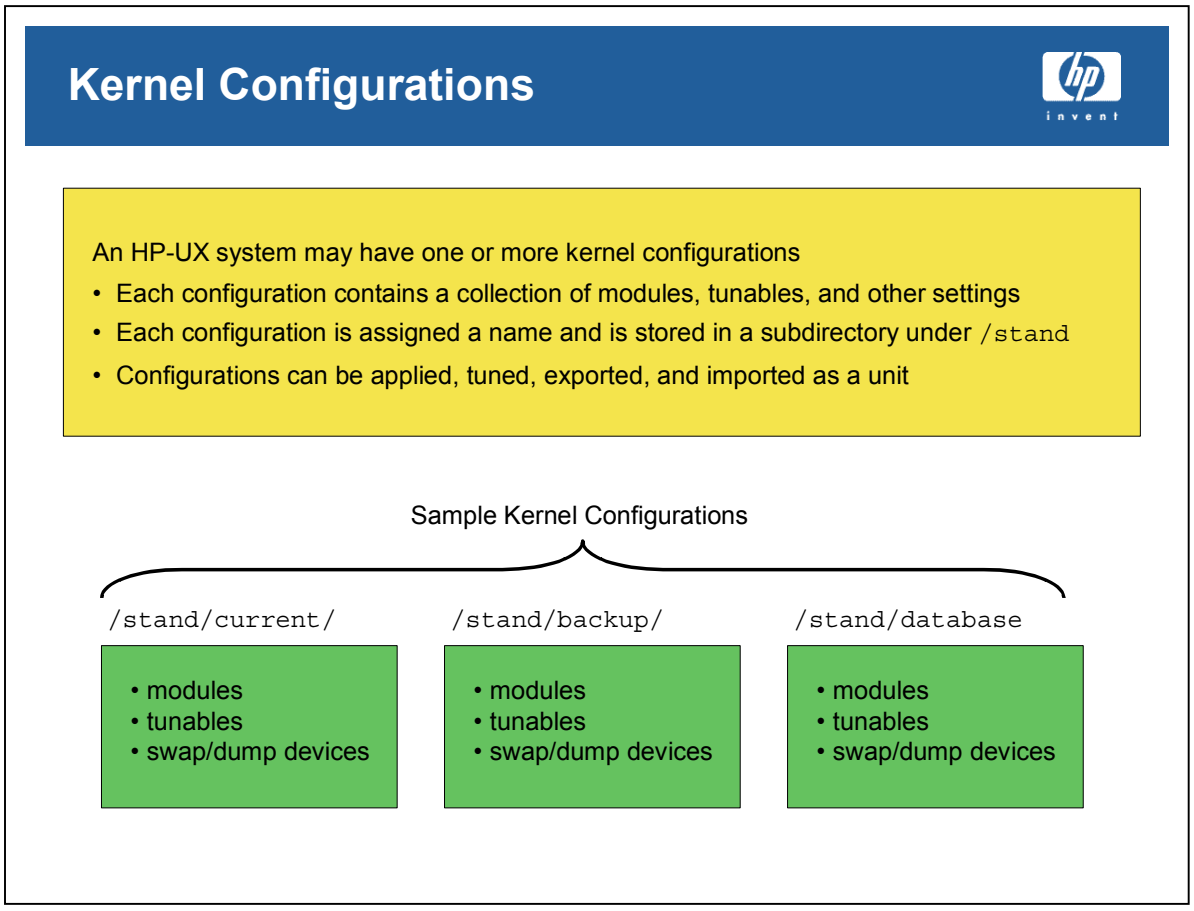
```
maxuprc(5)
                          Tunable Kernel Parameters
NAME

  maxuprc - limits the maximum number of concurrent user processes
  per user

VALUES

  Failsafe
  256

  Default
  256

  Allowed values
  Between 3 and (nproc -5).

  Recommended value
  256

DESCRIPTION

  maxuprc is a dynamic tunable that limits the maximum number of
  processes per user.  Only root can have more than the number of
  processes limited by maxuprc.

  Who is Expected to Change This Tunable?

  System administrators can change the value of maxuprc depending on
  the usage of the system.

  Restrictions on Changing
  None.  This tunable is dynamic.
```


© 2004 Hewlett-Packard Development Company, L.P.

```
  When Should the Value of This Tunable Be Raised?
  The value of maxuprc should be changed if users require more
  processes than what they are currently allowed by maxuprc.  If
  fork() fails with an error value of [EAGAIN], it could be an
  indication that maxuprc was reached by that particular user.
  However, that is not the only reason that could cause fork() to
  fail with an errno value of [EAGAIN].  [EAGAIN] may be returned if
  nproc system tunable has been reached or memory on the system has
  been exhausted.

  What Are the Side Effects of Raising the Value?
  Raising the value of maxuprc allows a single user to consume a
  greater percentage of the system resources.

  When Should the Value of This Tunable Be Lowered?
  The value of maxuprc should be lowered when individual users are
  monopolizing system resources by running too many concurrent
  processes.

  What Are the Side Effects of Lowering the Value?
  Applications that depend on a large number of processes may behave
  differently or fail.  Existing processes will continue to run but
  new process creations which would result in the user exceeding
  maxuprc will fail.

  What Other Tunable Values Should Be Changed at the Same Time?
  When tuning maxuprc you should keep in mind the value of nproc.

WARNINGS

  All HP-UX kernel tunable parameters are release specific. This
  parameter may be removed or have its meaning changed in future
  releases of HP-UX.

AUTHOR

  maxuprc was developed by HP.

SEE ALSO

  nproc(5)
```


© 2004 Hewlett-Packard Development Company, L.P.

19–16. SLIDE: Kernel Configurations



Student Notes

In HP-UX 11i v2, the kernel modules and tunable parameters specified by a system administrator via the **`kcweb`** or the **`kc*`** commands are recorded in a structure known as a "kernel configuration".

- Each configuration contains a collection of modules, parameters, and other settings
- Each configuration is assigned a name and is stored in a subdirectory under **`/stand.`** The subdirectory name serves as the kernel configuration's name. The system depicted on the slide above has three kernel configurations.
- Configurations can be applied, tuned, exported, and imported as a unit.

The next few slides describe several common kernel configuration management tasks.

© 2004 Hewlett-Packard Development Company, L.P.

19–17. SLIDE: Special Kernel Configurations

Special Kernel Configurations

HP-UX automatically maintains several special kernel configurations

Configuration	Description
`/stand/current/`	The configuration for the currently running kernel
`/stand/backup/`	Copy of the current configuration prior to changes
`/stand/nextboot/`	Link to the configuration to be used at the next boot
`/stand/last_install/`	Kernel configuration originally installed with OS
`/stand/crashconfig/`	Automatically created after any crash dump

There are also two special files in /stand:
- `/stand/vmunix` is a link to `/stand/current/vmunix`
- `/stand/system` is a link to `/stand/nextboot/system`

Student Notes

Several kernel configurations are automatically created and maintained by the system.

Kernel Configuration	Purpose
`/stand/current/`	This directory defines the configuration of the currently running kernel. By default, all kernel changes that can be made dynamically are applied to this configuration.
`/stand/backup/`	Before the administrator makes any changes to the current kernel configuration, HP-UX copies the existing kernel configuration to this directory. If you modify the kernel and discover that the new configuration doesn't work, you can boot using this **`backup`** configuration.
`/stand/nextboot/`	If the administrator makes changes to static kernel modules or parameters, or if the administrator specifically requests that changes be held until next boot, those changes are recorded in the **`nextboot`** configuration. During the next system reboot, the **`nextboot`** configuration becomes the current configuration.

© 2004 Hewlett-Packard Development Company, L.P.

/stand/last_install/	This is a backup of the kernel configuration that was created during the last OS install.
/stand/crashconfig/	At boot time, the system determines if there is a viable crash dump. If there is, a copy of the kernel and configuration are saved in the directory **/stand/crashconfig**. When the **/sbin/rc** program runs the **/sbin/init.d/savecrash** script during the boot process, **savecrash** copies the kernel and associated files to the directory specified in the **/etc/rc.config.d/savecrash** configuration file.

There are also two special files in the **/stand** directory.

- **/stand/vmunix** is a link to **/stand/current/vmunix**, the static executable image for the current kernel. This file should never be manually renamed or moved. Always use the **kc*** commands to modify the running kernel!

- **/stand/system** is a link to **/stand/nextboot/system**, which contains an ASCII list of the modules and kernel parameters included in the kernel configuration.

Kernel configuration directory contents

Each kernel configuration has a dedicated directory under **/stand**. The files in the kernel configuration directory determine the kernel configuration's features and functionality.

Configuration	Description
/stand/*configname*/README	A README file reminding system administrators that they shouldn't modify any file in this directory.
/stand/*configname*/.config	A flag file, marking this as a directory containing a kernel configuration. Also used as a lock file for the configuration.
/stand/*configname*/bootfs/	Contains a **stand/current/** directory, under which are symbolic links to the **.config** file, **krs** files, and those module object files that correspond to modules capable of loading during kernel boot. The **hpux.efi** OS loader uses this directory to populate the RAM file system used during the boot process.
/stand/*configname*/mod/	Contains the module object files and preparation scripts for each kernel module used by the configuration (i.e., in a state other than unused). The module object files are named with the module name (no extension). The **kc*** commands invoke the preparation scripts before and after loading and unloading a module. The scripts are

© 2004 Hewlett-Packard Development Company, L.P.

	named ***modulename*.prep**.
/stand/*configname*/krs/	Contains the kernel registry files for the kernel configuration. The kernel registry is used to record module states and tunable parameter values.
/stand/*configname*/config.krs.lkg/	Contains a "last known good" backup copy of the kernel registry files as they existed following the last successful boot.
/stand/*configname*/system	Contains an ASCII list of the modules and tunable parameters included in the kernel. As in earlier releases of HP-UX, **system** files may be manually edited. However, it is much safer to modify the kernel via the **kc*** commands.
/stand/*configname*/vmunix	The kernel executable used with this kernel configuration.

vmunix, module object files, and preparation scripts are often shared between configuration directories using hard links.

CAUTION:	With the exception of the **system** file, the files in these directories could only be modified by the **kc*** kernel configuration commands. Administrators should *never* manipulate or edit these files directly.

© 2004 Hewlett-Packard Development Company, L.P.

19–18. SLIDE: Kernel Configuration Command Overview

Kernel Configuration Command Overview

Kernel configurations may be managed via the `kconfig` command

Command/Option	Description
`kconfig -v`	List available configurations and their contents
`kconfig -a conf`	Display the contents of a configuration
`kconfig -r conf newconf`	Rename a configuration
`kconfig -c conf newconf`	Copy a configuration
`kconfig -d conf`	Delete a configuration
`kconfig -s conf`	Save a copy of the running configuration
`kconfig -l conf`	Load a configuration into the running kernel
`kconfig -n conf`	Select a configuration for use after next boot
`kconfig -e conf /tmp/system`	Create a system file from a configuration
`kconfig -i conf /tmp/system`	Create a configuration from a system file
`kconfig -t conf "for new myapp"`	Assign a title to a configuration
`kconfig -D`	View a list of pending changes
`kcpath -d`	Determine the currently booted config name

Student Notes

`kconfig` is the preferred command for configuring and managing HP-UX 11i v2 kernel configurations. The command supports a host of options for copying, deleting, saving, and loading kernel configurations. The table below describes the most common options.

Command	Description
`kconfig -v`	List available configurations.
`kconfig -a conf`	Display the contents of a configuration. Lists all of the modules and tunables in the named configuration. If no configuration is specified, the command displays the contents of the running kernel.
`kconfig -r conf newconf`	Rename a configuration.
`kconfig -c conf newconf`	Copy a configuration. Particularly useful if you want to make a backup copy of a configuration before making changes.
`kconfig -d conf`	Delete a configuration.

© 2004 Hewlett-Packard Development Company, L.P.

kconfig -s *conf*	Save a copy of the running configuration.
kconfig -l *conf*	Load a configuration into the running kernel, overwriting any changes that were being held for next boot. If all of the changes required by the new configuration may be applied immediately, the configuration is applied immediately. However, if the new configuration impacts any static tunables or modules, the configuration won't be applied until the next reboot.
kconfig -n *conf*	Select a configuration for use after next boot. The configuration specified here will be used as the current kernel the next time the system reboots.
kconfig -e *conf /tmp/system*	Create (export) a **system** file from a configuration. The resulting **system** file contains an ASCII list of the modules and tunables in the named configuration. This file can be copied, imported, and loaded on another host to propagate a kernel configuration to multiple machines. If no configuration is specified, the currently running kernel configuration will be exported, including any changes to it that are being held for next boot.
kconfig -i *conf /tmp/system*	Create (import) a configuration from a **system** file. In order to apply/load the newly imported configuration, run **kconfig -l**.
kconfig -t conf "*for new myapp*"	Assign a title to a configuration. The title appears in the **kconfig -v** output.
kconfig -D	Display any changes that are pending the next system reboot.
kcpath -d	Determine the currently booted configuration's directory path.

You can also include the **-C "*comment*"** option on any **kconfig** command if you want to document your actions in the **/var/adm/kc.log** file.

© 2004 Hewlett-Packard Development Company, L.P.

19–19. SLIDE: Example: Creating a New Configuration

Example: Creating a New Configuration

- Rather than modify the current kernel, you may wish to create a named configuration
- Multiple named configurations may be stored in `/stand`

Create a new named configuration based on the currently running configuration

```
# kcmodule -s MyConfiguration
```

Modify modules and tunables in the new configuration

```
# kcmodule -c MyConfiguration LCentIf=static
# kctune -c MyConfiguration nproc=6000
```

Review the new named configuration

```
# kconfig -v
# kconfig -av MyConfiguration
```

Load the new named configuration

```
# kconfig -l MyConfiguration
# shutdown -ry 0
```

Student Notes

Rather than modify the current kernel, you may wish to create a new configuration. Multiple configurations may be stored in **`/stand`**, each with a distinct name. When benchmarking a new application, you may wish to test several different combinations of tunable parameters; simply store each test kernel as a separate kernel configuration. The procedure below explains how to do this.

1. Create a new configuration based on the currently running configuration.

   ```
   # kcmodule -s MyConfiguration
   ```

2. Modify modules and tunables in the new configuration. Be sure to use the **`-c`** option to apply the changes to the named configuration rather than the current configuration. The syntax of the **`kcmodule`** and **`kctune`** commands will be described in detail later in the module.

   ```
   # kcmodule -c MyConfiguration …
   # kctune -c MyConfiguration …
   ```


© 2004 Hewlett-Packard Development Company, L.P.

3. Review the new configuration. The **-v** option lists the available kernel configurations. The **-av** options display the contents of a specific configuration.

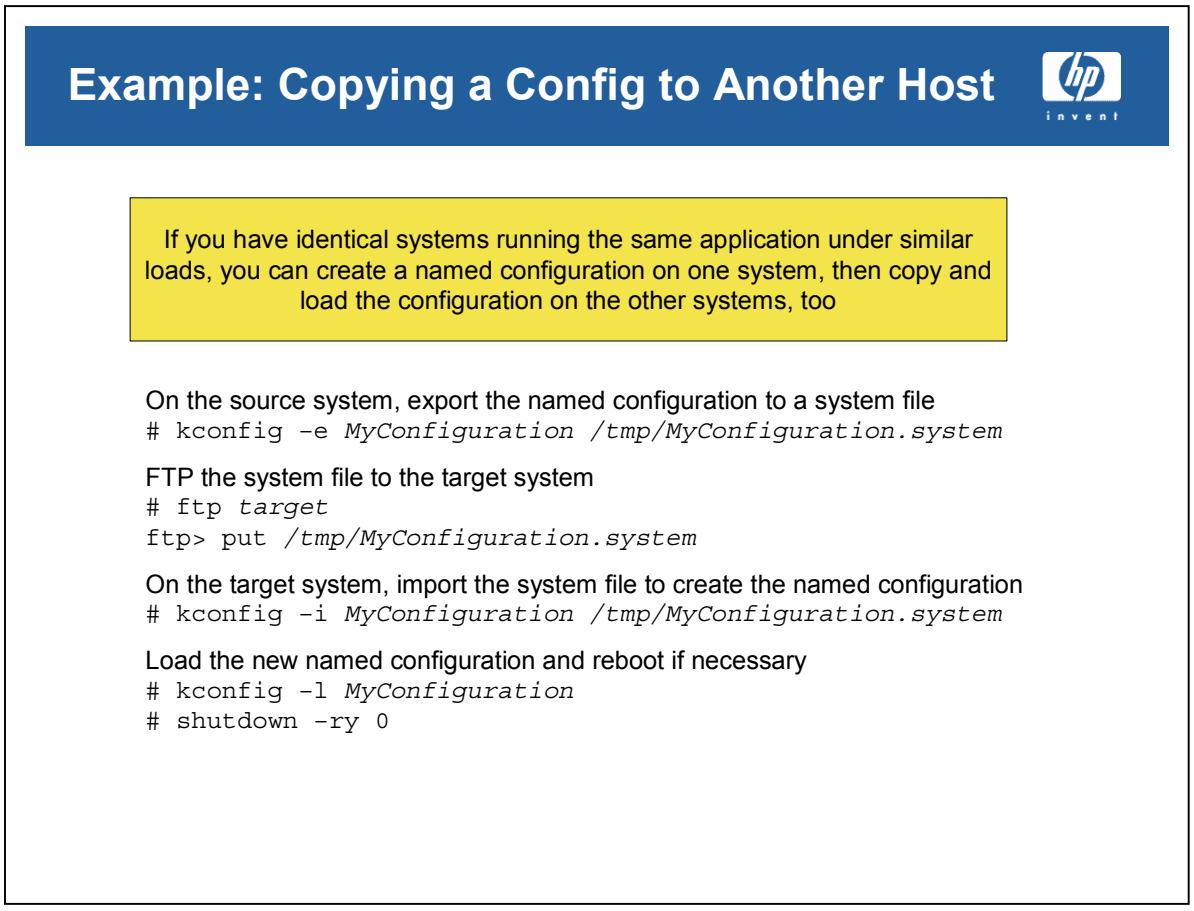
```
# kconfig -v
# kconfig -av MyConfiguration
```

4. Load the new configuration. If possible, HP-UX will apply the new configuration immediately. If applying the new configuration requires changes to static tunable parameters, though, neither the static nor dynamic parameters will be changed until the next reboot.

```
# kconfig -l MyConfiguration
# shutdown -ry 0
```


© 2004 Hewlett-Packard Development Company, L.P.

19–20. SLIDE: Example: Copying a Config to Another Host



Student Notes

If you have identical systems running the same application under similar loads, you can create a configuration on one system, then copy and load the configuration on the other systems, too.

1. On the source system, export the named configuration to a **`system`** file.

   ```
   # kconfig -e MyConfiguration /tmp/MyConfiguration.system
   ```

2. FTP the **`system`** file to the target system.

   ```
   # ftp target
   ftp> put /tmp/MyConfiguration.system
   ```

3. On the target system, import the **`system`** file to create the named configuration. The configuration name on the new system need not match the configuration name on the original system.

   ```
   # kconfig -i MyConfiguration /tmp/MyConfiguration.system
   ```


© 2004 Hewlett-Packard Development Company, L.P.

4. Load the new configuration. If possible, HP-UX will apply the new configuration immediately. If applying the new configuration requires changes to static tunable parameters, though, neither the static nor dynamic parameters will be changed until the next reboot.

```
# kconfig -l MyConfiguration
# shutdown -ry 0
```


© 2004 Hewlett-Packard Development Company, L.P.

19–21. SLIDE: Example: Loading a Configuration

Example: Loading a Configuration

- You can load any configuration at any time
- Many module and tunable changes apply immediately
- Changes to static modules and tunables won't apply until the next boot

List available configurations

```
# kconfig -v
# kconfig -av MyConfiguration
```

Load a named configuration

```
# kconfig -l MyConfiguration
```

Reboot if the named configuration modified any static modules or tunables

```
# shutdown -ry 0
```

- You can also specify a configuration at boot time via the `hpux.efi` kernel loader

After interrupting the HP-UX autoboot process, list and load the desired configuration

```
HPUX> 11
HPUX> boot MyConfiguration
```

Student Notes

You can load any named configuration on your system at any time. If possible, HP-UX applies the new configuration immediately. If applying the new configuration requires changes to static tunable parameters, though, neither the static nor dynamic parameters will be changed until the next reboot.

1. List available configurations and their contents.

```
# kconfig -v
# kconfig -av MyConfiguration
```

2. Load a named configuration.

```
# kconfig -l MyConfiguration
```

3. Reboot if the named configuration modified any static modules or tunables.

```
# shutdown -ry 0
```

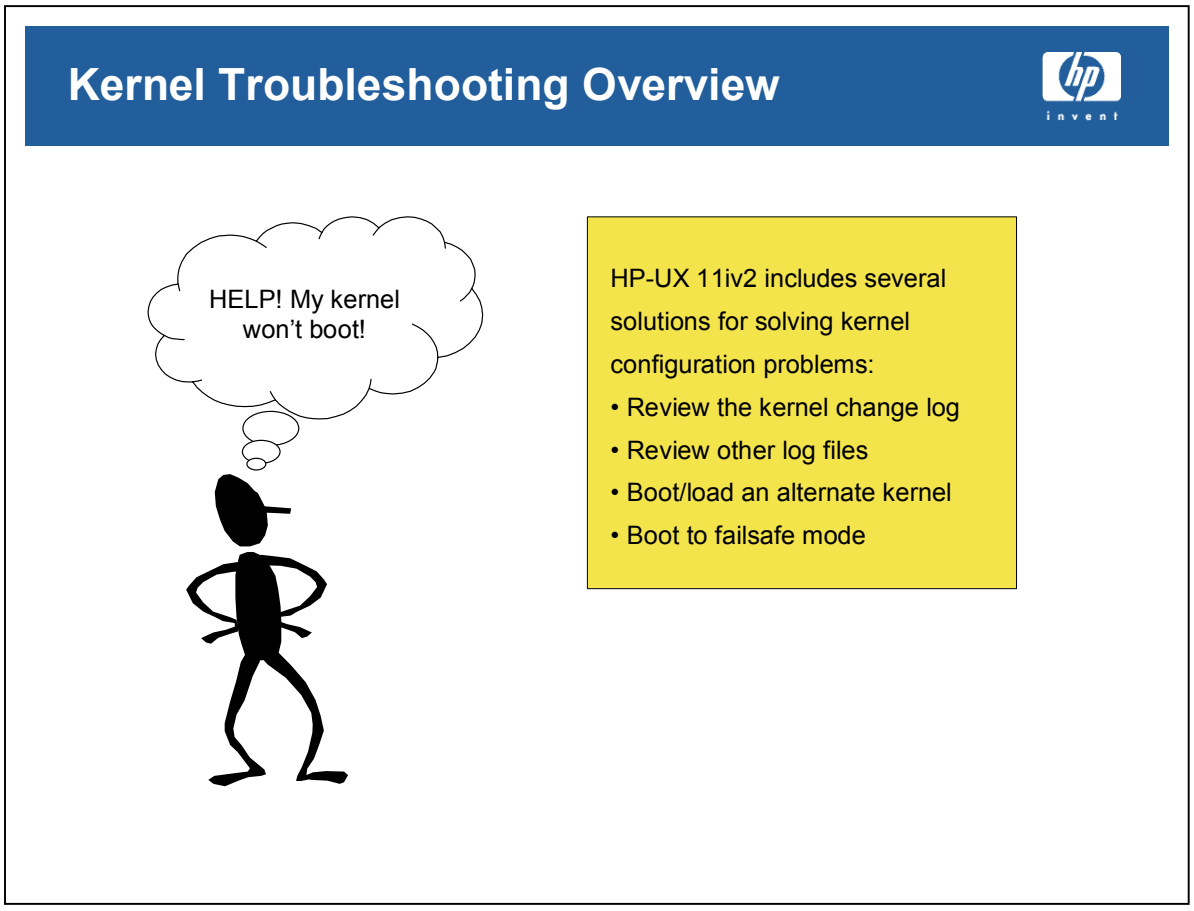

© 2004 Hewlett-Packard Development Company, L.P.

You can also specify a named configuration at boot time via the **hpux.efi** kernel loader. Allow the EFI boot manager to autoselect a boot disk, but interrupt the **hpux.efi** kernel loader. Use the **ll** command to view the contents of **/stand**, then boot the desired configuration.

```
HP-UX Boot Loader for IA64  Revision 1.71
 Press Any Key to interrupt Autoboot
 \EFI\HPUX\AUTO ==> boot vmunix
 Seconds left till autoboot - 10  Escape
   Type 'help' for help

HPUX> ll
HPUX> boot MyConfiguration
```


© 2004 Hewlett-Packard Development Company, L.P.

19–22. SLIDE: Kernel Troubleshooting Overview



Student Notes

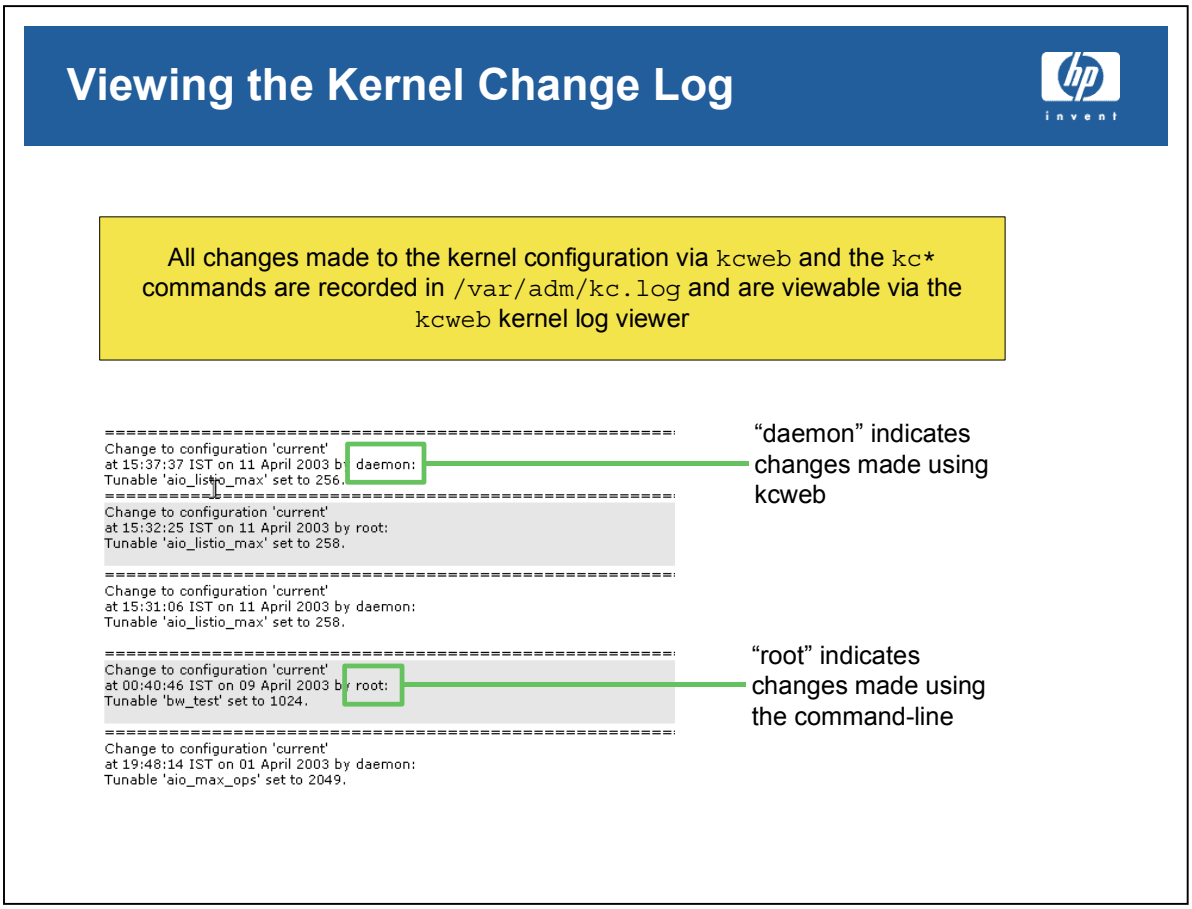
Although the HP-UX 11i v2 **`kc*`** tools automatically verify that tunable parameters fall within allowed boundaries and attempt to verify dependencies, poorly chosen kernel parameters may still cause an application – or the entire system – to crash. Fortunately, HP-UX 11i v2 also provides a number of tools to help troubleshoot these situations.

- The kernel change log records all kernel configuration changes.
- Other log files record actions and errors logged by **`kcweb`**.
- 11i v2 makes it very easy to boot or load a backup or alternate kernel if the current kernel configuration becomes unstable.
- If all else fails, every HP-UX 11i v2 tunable parameter has a hard-coded "failsafe" value. Booting to tunable maintenance mode boots the system to single-user mode using these "failsafe" parameters.

The next few slides discuss each of these tools in detail.

© 2004 Hewlett-Packard Development Company, L.P.

19–23. SLIDE: Viewing the Kernel Change Log



Student Notes

It is often useful to know what changes have been made on your system. In HP-UX 11i v2, all kernel configuration changes made via **kcweb** and the **kc*** commands are automatically recorded in **/var/adm/kc.log**. The commands automatically log what is changed, by whom, and when. Administrators can optionally add additional comments to the log file via the **-C** option on **kcmodule**, **kconfig**, and **kctune**, or the **comment** field on the **kcweb modules** and **tunables** screens.

kc.log is an ASCII text file that can be viewed via the **more** command, **kcweb**, or a dedicated log viewer called **kclog**. To view the log file from **kcweb**, select the **change log viewer** link from the navigation menu.

When you encounter a problem with your kernel configuration, review the kernel change log for recent changes, and consider using the **kcmodule** and **kctune** commands to reverse the changes until you find a stable configuration.

Viewing the Kernel Change Log with `kclog`

You can also view the kernel change log from the command line via the **more** command.

© 2004 Hewlett-Packard Development Company, L.P.

```
# more /var/adm/kc.log
```

Using the **kclog** command, you can view the contents of the kernel log filter log entries by search key, module, or tunable. Consider the examples below.

View the most recent log entry for the current kernel.

```
# kclog
```

View the last 5 entries for a specific named kernel configuration.

```
# kclog –c weekendConfig 5
```

Search for entries containing a specified string.

```
# kclog –f "Oracle"
```

View entries for a specific tunable or module.

```
# kclog –n "maxdsiz"
```

View entries for all tunables.

```
# kclog –t tunable
```

Add a general comment in the kernel log.

```
# kclog –C "Add this comment to kernel config log"
```

Include a comment when configuring a kernel module/tunable.

```
# kcmodule –C "comment" …
# kctune –C "comment" …
```


© 2004 Hewlett-Packard Development Company, L.P.

19–24. SLIDE: Viewing the `kcweb` Logs

Viewing the `kcweb` Logs

`kcweb` log messages are recorded in a number log files

Log File	Purpose
`/var/sam/log/samlog`	hpwebadm log messages
`/var/opt/hpwebadmin/logs/error_log`	Apache error messages
`/var/opt/hpwebadmin/logs/access_log`	Apache access log
`/var/adm/kc.log`	`kc*` command log messages
`/var/adm/syslog/syslog.log`	System log messages

Student Notes

Several other files log actions and errors generated by **`kcweb`**.

Log File	Purpose
`/var/sam/log/samlog`	hpwebadm log messages; records **`kc*`** commands executed by the Apache/hpwebadm daemon. This file must be viewed using the **`samlog_viewer`** utility.
`/var/opt/hpwebadmin/logs/error_log`	Apache error messages; records failed attempts to access hpwebadm via Apache.
`/var/opt/hpwebadmin/logs/access_log`	Apache access log; records successful attempts to access hpwebadm via Apache.
`/var/adm/kc.log`	**`kc*`** command log messages; records all **`kc*`** commands executed from either kcweb or the command line.
`/var/adm/syslog/syslog.log`	System log messages

© 2004 Hewlett-Packard Development Company, L.P.

19–25. SLIDE: Booting from an Alternate Kernel

Booting from an Alternate Kernel

If the default `vmunix` kernel configuration won't boot,
boot from the `backup` kernel configuration, or any other named kernel configuration.

Initiate the boot process , then interrupt the HPUX kernel loader autoboot when prompted

```
HP-UX Boot Loader for IA64  Revision 1.71
Press Any Key to interrupt Autoboot
\EFI\HPUX\AUTO ==> boot vmunix
Seconds left till autoboot - 10  Escape
  Type 'help' for help
```

Choose a kernel and boot

```
HPUX> ll                    List available kernel configurations
HPUX> boot backup           Boot the backup kernel, or …
HPUX> boot mykernel         Boot any other named kernel configuration
```

Student Notes

If your system fails to boot as a result of changes made to the default kernel configuration, it may be necessary to boot from the **`/stand/backup/`** kernel configuration, or any other named kernel configuration.

First, initiate the boot process and interrupt the HPUX kernel loader autoboot when prompted.

```
HP-UX Boot Loader for IA64  Revision 1.71
Press Any Key to interrupt Autoboot
\EFI\HPUX\AUTO ==> boot vmunix
Seconds left till autoboot - 10 Escape
Type 'help' for help
```

Choose a kernel and boot.

```
HPUX> ll              # List available kernel configurations
HPUX> boot backup     # Boot the backup kernel, or …
HPUX> boot mykernel   # Boot any other named kernel configuration
```


© 2004 Hewlett-Packard Development Company, L.P.

Once the system finishes booting, review the kernel change log to review what changes may have caused the default kernel to fail.

© 2004 Hewlett-Packard Development Company, L.P.

19–26. SLIDE: Booting to Tunable Maintenance Mode

Booting to Tunable Maintenance Mode

Booting to tunable maintenance mode boots to single-user mode using failsafe tunable parameters hard-coded in the kernel modules. DLKMs are not loaded. After booting to tunable maintenance mode, review the kernel log, modify the kernel as necessary, and reboot.

Initiate the boot process , then interrupt the HPUX kernel loader when prompted

```
HP-UX Boot Loader for IA64  Revision 1.71
Press Any Key to interrupt Autoboot
\EFI\HPUX\AUTO ==> boot vmunix
Seconds left till autoboot - 10  Escape
  Type 'help' for help
```

Boot to tunable maintenance mode

```
HPUX> boot vmunix -tm          Boot to tunable maintenance mode
```

Student Notes

If the default kernel configuration won't boot and you don't have a backup kernel configuration, you can boot to tunable maintenance mode. Booting to tunable maintenance mode boots to single-user mode using failsafe tunable parameters hard-coded in the kernel modules. DLKMs are not loaded. After booting to tunable maintenance mode, review the kernel log, modify the kernel as necessary, and reboot.

Initiate the boot process, then interrupt the HPUX kernel loader when prompted

```
HP-UX Boot Loader for IA64  Revision 1.71
Press Any Key to interrupt Autoboot
\EFI\HPUX\AUTO ==> boot vmunix
Seconds left till autoboot - 10  Escape
Type 'help' for help
```

Boot to tunable maintenance mode.

```
HPUX> boot vmunix -tm        # Boot to tunable maintenance mode
```


© 2004 Hewlett-Packard Development Company, L.P.

19–27. SLIDE: For Further Study

For Further Study

- See the following manuals on `http://docs.hp.com`:
  HP-UX 11i Version 2 Release Notes
  Managing Systems and Workgroups: A Guide for HP-UX System Administrators
  Tunable Kernel Parameters
  Dynamically Tunable Parameters in HP-UX 11i
- Also read the man pages for the kernel configuration commands:
  `kconfig(5), kclog(1m), kcmodule(1m), kcpath(1m), kctune(1m)`
- All tunable parameters now have man pages, too.
  For a complete list of kernel tunables, type `kctune`.

Student Notes

If you want to learn more about HP-UX 11i v2 kernel configuration, have a look at the following documents.

- *Managing Systems and Workgroups: A Guide for HP-UX System Administrators* on **`http://docs.hp.com`** has a chapter on HP-UX 11i v2 kernel configuration
- Also read the online man pages for the kernel configuration commands:
  `kconfig(5)`, **`kclog(1m)`**, **`kcmodule(1m)`**, **`kcpath(1m)`**, **`kctune(1m)`**
- All tunable parameters now have man pages, too.
  For a complete list of kernel tunables, type **`kctune`**.

© 2004 Hewlett-Packard Development Company, L.P.

19–28. LAB: Configuring the HP-UX 11i v2 Kernel

Directions

Carefully follow the instructions below.

Preliminary Steps

1. Portions of this lab require console access. Login as root via your server's MP console interface.

2. Portions of this lab require access to a web browser, too. If you are accessing your server directly via a PC desktop, simply launch Internet Explorer from your Start menu. If you are accessing your server from another UNIX system, you can use your client's Mozilla browser.

3. If your instructor tells you to disable your browser's proxy settings, do so now. Ask your instructor for details.

Part 1: Configuring and using kcweb

Our goal in this portion of the lab is to configure, launch, and navigate the **`kcweb`** interface.

1. In your console window, enable the **`kcweb`** autostart functionality.

2. Point your desktop browser to the **`kcweb`** autostart URL, **`http://`*`server`*`:1110/casey/top.cgi`**.

3. You may see a dialog box noting that the server is using a certificate issued by "a company you haven't chosen to trust". Click "Accept this certificate permanently" when asked if you wish to proceed.

© 2004 Hewlett-Packard Development Company, L.P.

4. When prompted for your username and password, enter the root username and password.

5. Logout out of the **`kcweb`** interface.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Managing Kernel Modules

When you add a new interface card or choose to implement a previously unused HP-UX feature, it sometimes becomes necessary to add a kernel module to your kernel configuration. This portion of the lab provides an opportunity to add and remove both static and dynamic kernel modules via **`kcweb`**.

1. Before you make any changes to the kernel configuration, it's a good idea to save a copy of your current configuration. Call the configuration copy "**`OldConfig`**".

2. If you didn't launch **`kcweb`** in the previous part of the lab, do so now.

3. View the list of kernel modules that are on your system.

   a. Does your kernel contain the **`sdisk`** module required to support SCSI disks?
   b. Does your kernel contain the **`stape`** module required to support SCSI tapes?
   c. Does your kernel contain the **`nfs_server`** module required to support NFS servers?

4. The **`LCentIf`** module supports parallel interfaces. You probably don't have a parallel interface on your Integrity server, so change its state to **`unused`** to remove it from the kernel.

5. Was the module removed from the running kernel immediately? Explain.

© 2004 Hewlett-Packard Development Company, L.P.

6. The **cdfs** module, which supports the CDFS file system, is the first major module to fully implement HP-UX 11i v2's DLKM functionality. View the detailed description of the **cdfs** module by selecting it from the module list in **kcweb**.

   a. What is the current state of the module?
   b. What other states are supported?

7. The initial state of the module should be **auto**, which indicates that the module will automatically be loaded when needed. However, you can force the module to be loaded immediately by changing the state to **loaded**. Try it!

8. Check the state again. What is the current state now? Did you have to reboot? What is the current state cause?

9. Can you unload a module that's already loaded? Try it! Change the **cdfs** module's state to **unused**. Then review the verbose **kcmodule** output again. Can you think of a situation that might prevent the administrator from marking the **cdfs** module **unused**?

10. In the last two questions, you should have discovered that the **cdfs** module can be dynamically loaded and unloaded from the running kernel, without rebooting the system. Thus, if one had to install a CDFS patch on the system, one could temporarily unload the module from the running kernel, patch the module, then reload it – no reboot required!

    Return the **cdfs** module to the best/auto state.

© 2004 Hewlett-Packard Development Company, L.P.

NOTE:	The next couple questions require a CD/DVD drive with a loaded CD/DVD. Only proceed to questions #10 & #11 if you have a CDROM/DVD drive with a CD/DVD. However, everyone should complete #12 before moving on to the next part of the lab.

11. (Optional) Currently, the **`cdfs`** module state should be **`auto`**. Mount the CDROM, then click the **`Reload`** or **`Refresh`** button in your web browser to see what happened! Note the current **`state`** and the **`cause`** field.

12. (Optional) Can you change the **`cdfs`** module's state to **`unused`** while a CDROM is mounted? Try it!

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Managing Tunable Parameters

This part of the lab gives you an opportunity to experiment with 11i v2 tunable parameters. You may complete this portion of the lab using **`kcweb`**.

1. Before you make any changes to the kernel configuration, it's a good idea to save a copy of your current configuration. Call the configuration copy "**`OldConfig`**". If you created an **`OldConfig`** backup earlier in the lab, you can skip this step.

2. First let's consider the **`nswapdev`** kernel parameter. Have a look at the man page for this parameter. What does this parameter do? When might it be necessary to change it?

3. Increase the **`nswapdev`** parameter by 10. Does this change take effect immediately?

4. Now review the man page for **`maxuprc`**. What does this parameter do? When might it be necessary to change it?

5. Increase the **`maxuprc`** parameter by 50. Does the change take effect immediately?

© 2004 Hewlett-Packard Development Company, L.P.

6. What happens if you accidentally add a few extra zeros on the end of the **`maxuprc`** value you entered in the previous question? Try it!

7. Since we don't have any applications running on the system, the current value of **`maxuprc`** is probably too high. Drop **`maxuprc`** back to its default value.

8. Reboot your system to make sure the new kernel configuration boots successfully.

© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Managing Kernel Configurations

1. Assuming your updated kernel configuration booted properly, save a copy of the updated configuration. Call it **`NewConfig`**. Assign a title to the new configuration, too: "**`Approved new kernel configuration`**".

2. After testing the configuration thoroughly, you may choose to apply the configuration to your production server. Export a system file based on your current configuration. Name the file **`system.NewConfig`**.

3. View the resulting system file with the **`more`** command.

4. What would be required to apply this system on the production server? (Record the commands below, but don't actually execute them.)

5. As it turns out, management decided not to install the new application after all. Load the **`OldConfig`** configuration that you created at the beginning of the exercise, then reboot.

© 2004 Hewlett-Packard Development Company, L.P.

Part 5: Logging and Troubleshooting

1. Let's undo all of the parameter changes you made in the previous parts of the lab. Reboot your system. When you get to the HPUX OS loader stage, terminate the autoboot sequence.

2. Boot to tunable maintenance mode. What distinguishes tunable maintenance mode from a normal boot?

3. Once you reach tunable maintenance mode, mount all of your file systems. Then use the **more** command to review recent kernel changes that were recorded in the kernel change log.

4. Reboot your system again. This time, interrupt the autoboot process, list the contents of `/stand`, and boot from the `OldConfig` configuration that you created back in Part 2.

© 2004 Hewlett-Packard Development Company, L.P.

Module 20 — Managing Software with SD-UX

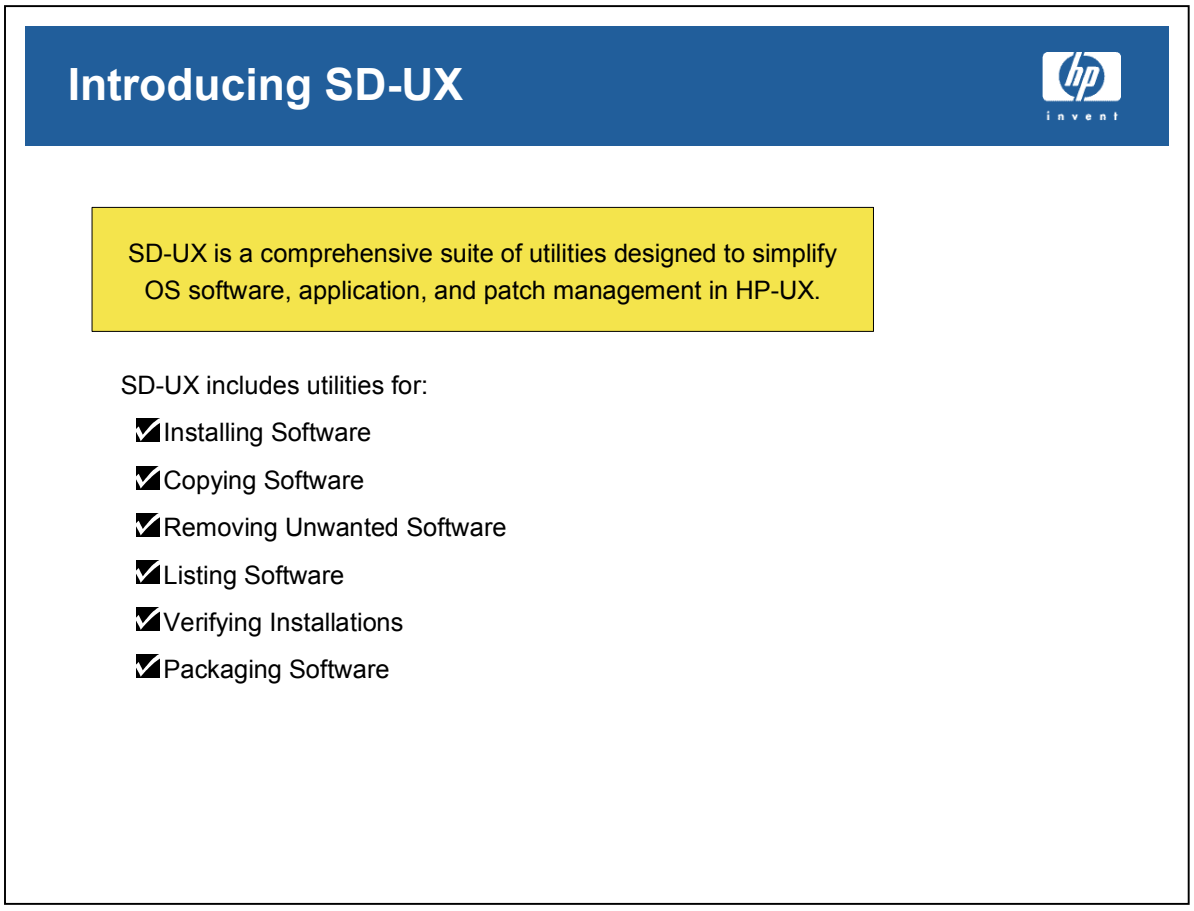
Objectives

Upon completion of this module, you will be able to do the following:

- Relate SD-UX terms and definitions.
- Install software using SD-UX.
- Verify software using SD-UX.
- List software using SD-UX.
- Remove software using SD-UX.

© 2004 Hewlett-Packard Development Company, L.P.

20–1. SLIDE: Introducing SD-UX



Student Notes

What Is SD-UX?

SD-UX is a comprehensive suite of utilities designed to simplify OS software, application, and patch management in HP-UX. These commands are provided as part of the HP-UX operating system and offer many options for flexible software management tasks, such as:

- Installing or updating software on local systems (**`swinstall`**).
- Building and configuring a network software server, or copying software from a distribution source or media onto a system (**`swcopy`**).
- Removing software from your system (**`swremove`**).
- Listing software that is installed on your system or on various media, CD-ROM, tape, etc. (**`swlist`**).

© 2004 Hewlett-Packard Development Company, L.P.

- Verifying that software products are compatible with your system before actually installing them (**`swverify`**), and reporting if modifications have been made to the installed software (**`swverify`**).

- Creating software "packages" that make later software installations quicker and easier (**`swpackage`** and Software Package Builder).

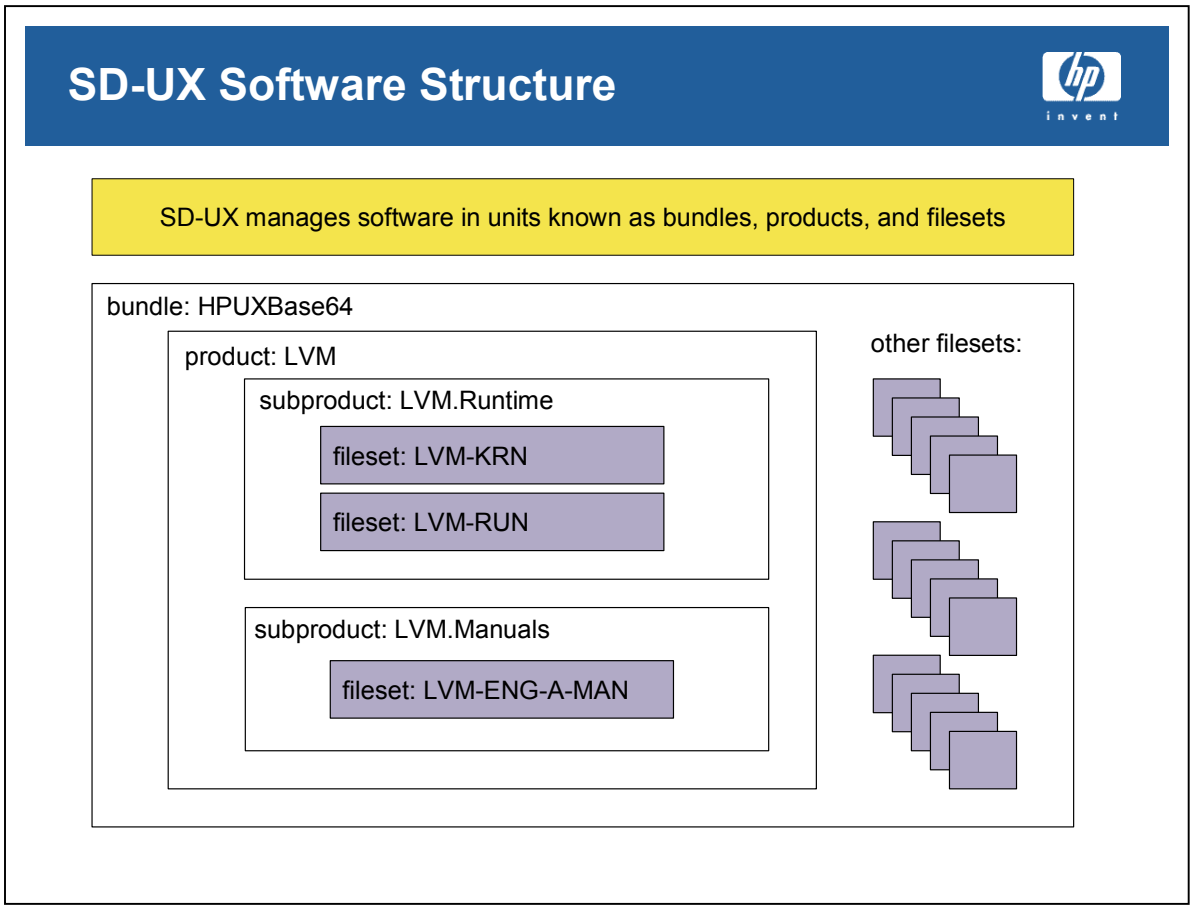
- Configuring, unconfiguring, or reconfiguring installed software (**`swconfig`**).

SD-UX Provides GUI, TUI and Command Line Interfaces

All SD-UX commands can be executed from either shell scripts or the command line with a series of options and arguments. The **`swinstall`**, **`swcopy`**, **`swremove`**, and **`swlist`** utilities also provide intuitive menu-based graphical and terminal interfaces.

© 2004 Hewlett-Packard Development Company, L.P.

20–2. SLIDE: SD-UX Software Structure



Student Notes

Software in SD-UX is treated as a hierarchy of objects including bundles, products, subproducts and filesets.

Filesets	Filesets include all the files and control scripts that make up a product. They are the smallest manageable (selectable) SD-UX software objects.
Subproducts	Subproducts are used to group logically related filesets within a product if the product contains several filesets. The same fileset can be part of more than one subproduct.
Products	A product is a collections of subproducts and filesets. The SD-UX commands maintain a product focus but still allow you to specify subproducts and filesets. Different versions of software can be defined for different platforms and operating systems, as well as different revisions of the product itself. Several different versions could be included on one distribution media or depot.
Bundles	Collections of filesets, possibly from several different products, that are "encapsulated" by HP for a specific purpose. Bundles can be stored in

© 2004 Hewlett-Packard Development Company, L.P.

software depots and copied, installed, removed, listed, configured and verified as single entities. Most administrators install and manage software at the bundle level.

SD-UX commands refer to this product structure in the form: **`bundle[.]`** or product **`[.[subproduct.]fileset]`** with periods separating each level.

Examples

An example of a bundle is:

```
HPUXBaseOS            B.11.23  HP-UX Base OS
```

An example of a product is:

```
LVM                   B.11.23  LVM
```

Examples of subproducts are:

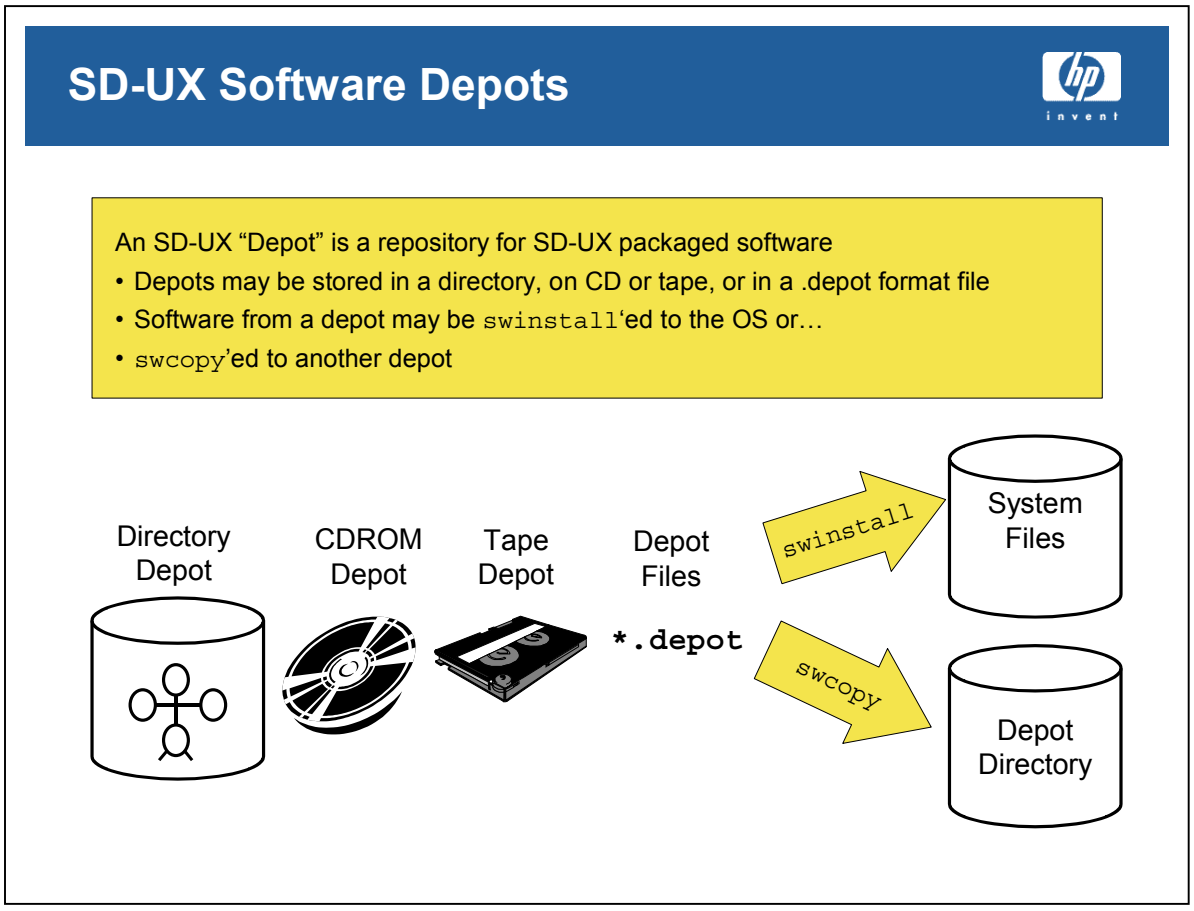
```
LVM.Manuals           B.11.23  LVM Manuals
LVM.MinimumRuntime    B.11.23  LVM MinimumRuntime
LVM.Runtime           B.11.23  LVM Runtime
```

The Runtime subproduct contains all the filesets in the Minimum Runtime, as well as some additional filesets. Examples of filesets are:

```
LVM.LVM-ENG-A-MAN   B.11.23        LVM Manuals
LVM.LVM-JPN-E-MAN   B.11.23        LVM Manuals
LVM.LVM-JPN-S-MAN   B.11.23        LVM Manuals
LVM.LVM-KRN         B.11.23        LVM Kernel components
LVM.LVM-RUN         B.11.23        LVM Runtime commands
```


© 2004 Hewlett-Packard Development Company, L.P.

20–3. SLIDE: SD-UX Software Depots



Student Notes

An SD-UX depot is a repository for software packaged using the SD-UX utilities. Software can be copied from depot to depot using the **swcopy** command, or it can be installed to the local files and directories using the **swinstall** command. There are several types of software depots:

Directory Depot — Software in a directory depot is stored under a normal directory on your file system (by default **/var/spool/sw**). The files and directories in a depot directory must adhere to the SD-UX naming convention and standard. When using the SD-UX commands, you refer to a directory depot via its top-most directory. A host can contain several depots. For example, a designated software distribution server on your network might contain a depot of word processing software, a depot of CAD software and a spreadsheet software depot, all on the same server.

CDROM Depot — When a directory depot is burned to a CDROM, the depot directory on the CDROM may be treated as an SD-UX depot source. Most software from HP is distributed on CDROMs containing SD-UX depots.

© 2004 Hewlett-Packard Development Company, L.P.

Tape Depot	Occasionally, a colleague or HP support person may send you an SD-UX format tape depot.
`*.depot` files	Patches and software that you download from the **`http://itrc.hp.com`** and **`http://software.hp.com`** websites can be **`swinstall`**'ed from a .depot file.

Network Source Depots

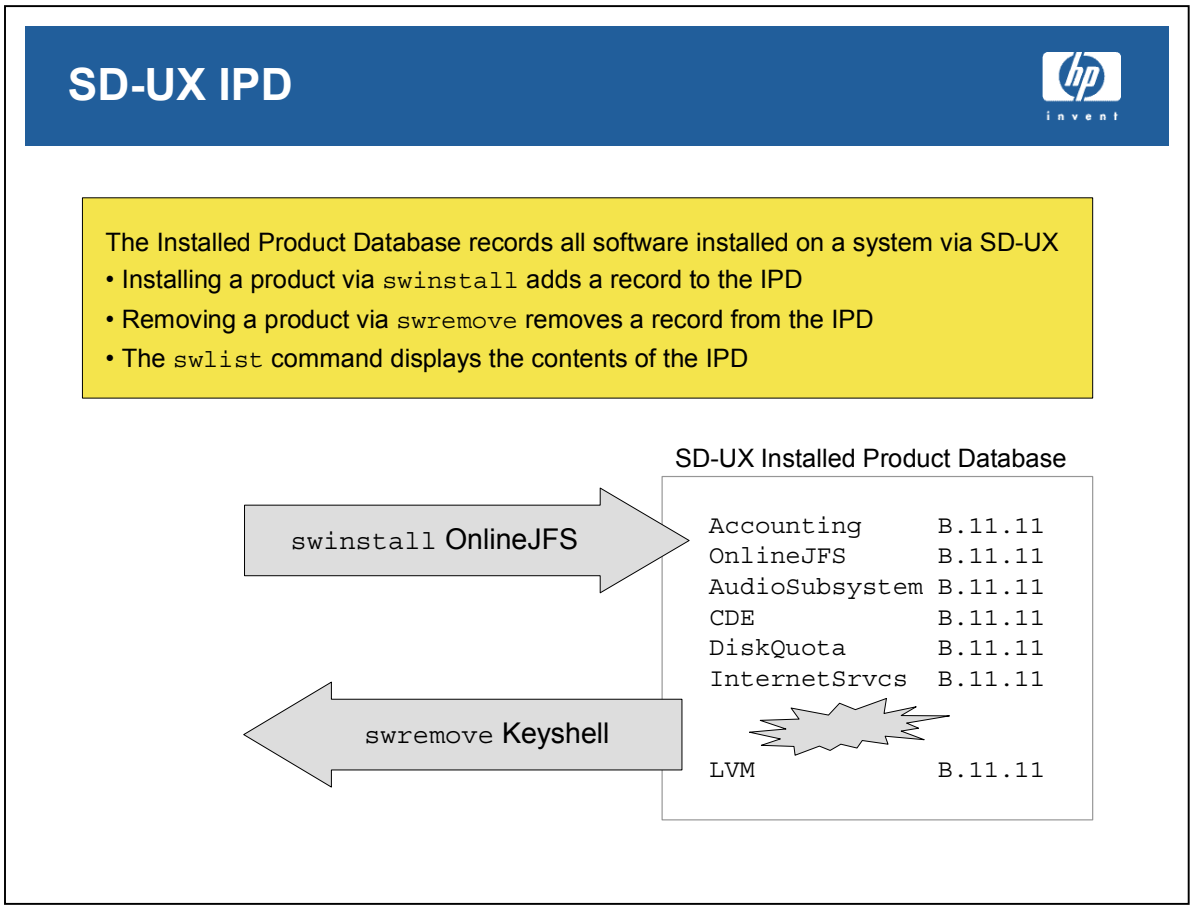
If a depot resides on a system that is connected to a network, then that system can be a network source for software. Other systems on the network can install software products from that server instead of installing them each time from a tape or CD-ROM.

A network source offers these advantages over installing directly from tape or CD-ROM:

- Several users can "pull" software down to their systems (over the network) without having to transport the tapes or disks to each user.
- Installation from a network server is faster than from tape or CD-ROM.
- Many different software products from multiple tapes, CD-ROMs and network servers can be combined into a single depot serving all others on the network.

© 2004 Hewlett-Packard Development Company, L.P.

20–4. SLIDE: SD-UX IPD



Student Notes

In order to manage software intelligently on a host, SD-UX must know what software is currently installed. SD-UX records this information in the Installed Product Database.

- When installing a new software bundle, product, or fileset, the new software must be added to the host's IPD.
- When removing software, the software must be removed from the host's IPD.
- Listing software is simply a matter of querying the IPD.

The IPD is stored in a directory structure under **`/var/adm/sw/products`** and is managed by the SD-UX utilities.

NOTE: Never manually edit the IPD, or manually remove, rename, or modify binaries and libraries that have been installed via **`swinstall`**; doing so may leave the IPD in an inconsistent state. If you install/remove software with the **`swinstall`** and **`swremove`** commands, SD-UX automatically updates the IPD for you.

© 2004 Hewlett-Packard Development Company, L.P.

20–5. SLIDE: SD-UX Daemons and Agents

SD-UX Daemons and Agents

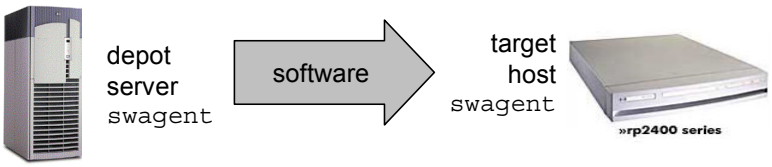
The SD-UX `swagentd` daemon starts automatically at run level 2, and must be running in order to install or remove SD-UX software

`swagentd`

- Starts at run level 2, and runs continuously thereafter.
- Listens for SD-UX requests.
- Schedules `swagent` agents for each SD-UX request.

`swagent`

- Performs software management tasks.
- Started as needed by `swagentd`.



Student Notes

SD-UX uses software agent (**`swagent`**) processes to accomplish software management tasks. When installing software, two **`swagents`** are required. One **`swagent`** must run on the host containing the depot from which the software is being pulled, and one **`swagent`** process must run on the host on which the software is to be installed. Other SD-UX utilities require **`swagent`** processes as well.

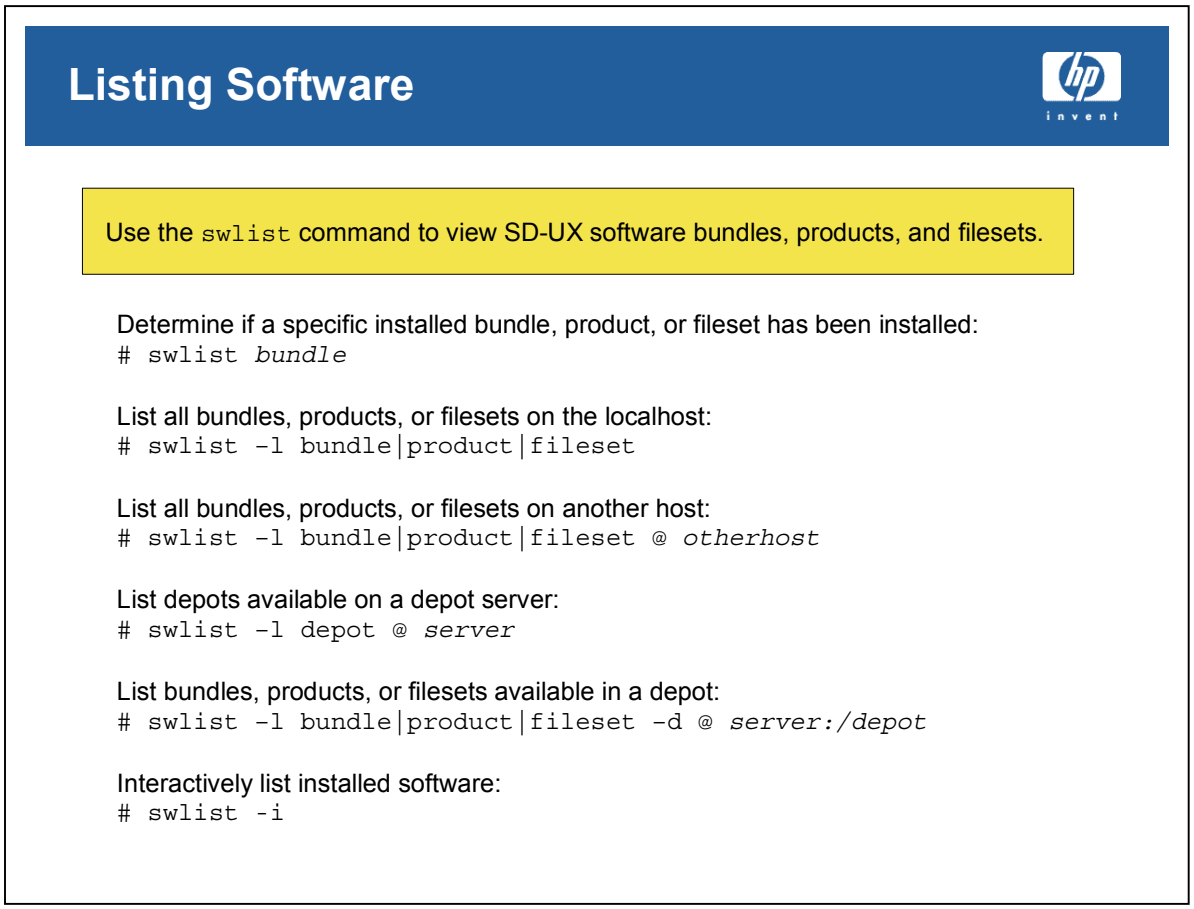
`swagent` processes are started on an as-needed basis by the **`swagentd`** daemon. The **`swagentd`** daemon must be running in order to perform any SD-UX software management tasks. The **`swagentd`** daemon starts automatically at run level two, and should run continuously on the system until shutdown.

If **`swagentd`** dies, you can restart it with the commands below.

```
# /sbin/init.d/swagentd start       # start it
# ps -ef | grep swagentd            # check it
```


© 2004 Hewlett-Packard Development Company, L.P.

20–6. SLIDE: Listing Software



Student Notes

The remaining slides in the chapter discuss some of the basic SD-UX commands required to manage software on your system. You can use the swlist command to list software in a depot, or on your system. Have a look at the following examples:

Determine if a specific installed bundle, product, or fileset has been installed:

```
# swlist bundle
```

List all bundles, products, or filesets on the localhost. If you leave off the **`-l bundle|product|fileset`** option, **`swlist`** lists bundles and products not included in other bundles.

```
# swlist -l bundle|product|fileset
```

List all bundles, products, or filesets on another host by querying the target system's **`swagentd`** daemon.

```
# swlist -l bundle|product|fileset @ otherhost
```


© 2004 Hewlett-Packard Development Company, L.P.

List depots available on a depot server. Recall that a server may have multiple depots if it serves clients running different versions of the OS, or if the administrator chooses to keep OS and application software in different depots.

```
# swlist -l depot @ server
```

List bundles, products, or filesets available in a depot. After determining which depots are available from a depot server, you will probably want to view the software in those depots.

```
# swlist -l bundle|product|fileset -d @ server:/depot
```

Interactively list installed software. If you prefer to use a GUI/TUI interface, add the **-i** option.

```
# swlist -i
```

swlist supports several other less common options. See the man page for details.

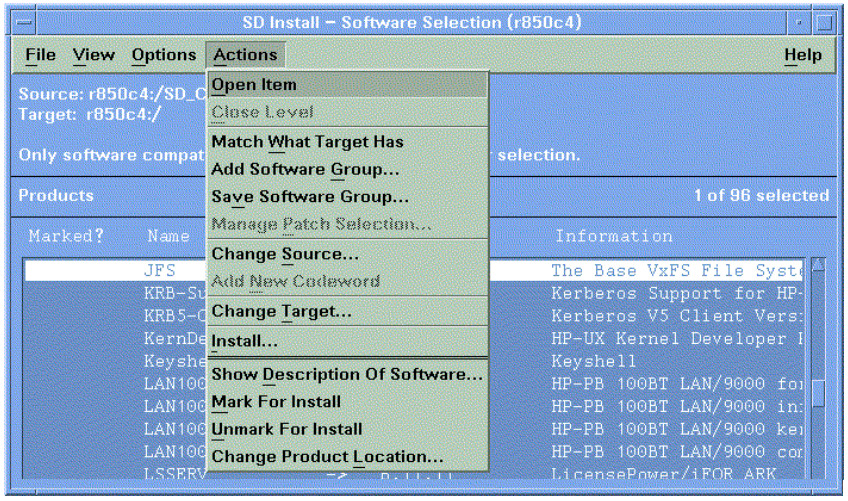
© 2004 Hewlett-Packard Development Company, L.P.

20–7. SLIDE: Installing and Updating Software

Installing and Updating Software

Use any of the following three commands to install/update software:

```
# swinstall -s svr:/depot -x autoreboot=true bundle
# swinstall -s svr:/depot -x autoreboot=true -x match_target=true bundle
# swinstall
```



Student Notes

Software may be installed from an SD-UX depot using the **`swinstall`** command.

Running `swinstall` From the Command Line

Several options are required if you want to non-interactively install software.

The **`-s`** option identifies the session's source depot. If the source is a directory or CDROM depot, specify the depot directory's pathname. If the source is a tape, use the tape drive device file. If the source is a .depot file, specify the file's full pathname.

```
# swinstall -s /dev/rmt/0m              # from a local tape depot
# swinstall -s /var/spool/sw            # from a local directory depot
# swinstall -s /cdrom                   # from a CD mounted on /cdrom
# swinstall -s depothost:/depotpath     # from a network depot
# swinstall -s /tmp/myapp.depot         # from a depot file
```

The **`-x autoreboot=true`** option indicates that **`swinstall`** can reboot the system if necessary. This option is generally required when installing products or patches containing kernel filesets.

© 2004 Hewlett-Packard Development Company, L.P.

Normally when executing the swinstall commandline, you must specify which product or bundle you wish to install or update. The **match_target** option, if set to true, selects software by locating filesets on the source that match the target system's installed filesets.

If executed without any options, swinstall launches a menu-based TUI or GUI interface, depending on the value of the **DISPLAY** variable.

```
# swinstall –s svr:/depot –x autoreboot=true bundle
# swinstall –s svr:/depot –x autoreboot=true -x match_target=true
bundle
# swinstall
```

The remaining notes below discuss the swinstall GUI interface in detail.

Choosing Source Depot

When you invoke the **swinstall** GUI/TUI, you will initially be prompted for your desired depot type, and the hostname and path of the depot you wish to install software from. You can skip the source depot selection dialog box by providing the source depot path on the command line:

```
# swinstall -s /dev/rmt/0m           # from a local tape depot
# swinstall -s /var/spool/sw         # from a local directory depot
# swinstall -s /cdrom                # from a CD mounted on /cdrom
# swinstall -s depothost:/depotpath  # from a network depot
```

Choosing the Software View

Once you have specified a software source, **swinstall** presents a list of software available from the selected depot. By default, **swinstall** lists all bundles on the depot, as well as products that aren't included in the listed bundles. If you prefer to simply view the products on the depot select:

```
View -> Change Software View -> Start with Products
```

You can return to the mixed bundle/product view by selecting:

```
View -> Change Software View -> Start with Top
```

Selecting Software to Install

Once the list of available software appears, you may select which software you wish to install. You may select:

- Entire bundles
- Selected products
- Selected filesets

To drill down to subproducts or individual filesets, double-click on any of the listed bundles or products. Once you find the software you wish to install:

© 2004 Hewlett-Packard Development Company, L.P.

Select a product/bundle/fileset with the space bar (TUI) or mouse (GUI).

Select `Actions --> Show Description of Software` to view a software description.

Select `Actions --> Mark for Install`.

To meet dependencies, `swinstall` may automatically select additional filesets.

Repeat steps 1-4 to select additional software.

Select `Actions --> Install (Analysis)` to install the selected software.

Selecting Software to Update

The procedure described above can also be used to update already installed software to a newer version. Simply select the products to update using **`Actions --> Mark`** for **`Install`**, then choose **`Actions --> Install Analysis`**.

If you wish to update all of the software on your machine, you can use **`Actions --> Match what Target Has`** to automatically select all of the depot software that matches products and bundles already installed on your machine. Then choose **`Actions --> Install Analysis`** to start the update. By default, **`swinstall`** does not reinstall filesets if the same version already exists on your machine. If you want to reinstall the same version, you can change the install options by selecting **`Options --> Change Options`**.

Starting the Install or Update

After marking software to install or update and selecting **`Actions --> Install`** (analysis), **`swinstall`** checks available disk space and software dependencies to ensure that the install will be successful. A dialog box appears so you can monitor the analysis process.

After the analysis completes, click on the **`Logfile`** and **`Disk Space`** buttons to see the results of the analysis. Assuming the analysis is successful, click the [OK] button to begin the install or update.

Viewing the Install or Update Log

`swinstall` writes all of its actions to the file **`/var/adm/sw/swinstall.log`**.
You should check this file for possible errors, and follow any instructions that are given.

The **`/var/adm/sw/swinstall.log`** file contains a description of the events and any errors that occurred during the update process. The following items are message labels and their meanings. Search the file for **`ERROR`**, **`WARNING or NOTE`**. These labels record anything important to know about the update process.

`=======`	Indicates that a task within **`swinstall`** is beginning or has completed.
`ERROR`	Indicates that the program cannot proceed, or that it needs corrective action. In some cases this impacts **`swinstall`** so much that it cannot continue.

© 2004 Hewlett-Packard Development Company, L.P.

WARNING Usually indicates that the program can continue. However, it does say something went wrong or requires attention (now or later). Read the information attached to the WARNING and perform the tasks noted.

NOTE Indicates that something out of the ordinary or worth special attention has happened. The message may require no action on your part. In other cases, the **NOTE:** message will require action. In some cases you must infer the action that is necessary after the update.

A Note about Protected Software

Most HP software products are shipped to you on CD-ROM as "protected" products. That is, they cannot be installed or copied unless you provide a "codeword" and customer ID. Software that is unlocked by a codeword can only be used on computers for which you have a valid license to use that software. *It is your responsibility to ensure that the codeword and software are used in this manner.* The codeword for a particular software product is found on the CD-ROM certificate, which you receive from HP. It shows the codeword and the customer ID for which the codeword is valid. One codeword usually unlocks all the products on the CD-ROM you have purchased. When an additional HP software product is purchased, an additional codeword will be provided by HP. Just enter the new codeword and customer ID, and they will be merged with any previously entered codewords. A codeword for a particular customer ID and CD-ROM only needs to be entered once per target system. The codeword and customer ID are stored for future reference in **/var/adm/sw/.codewords**. SD-UX will prompt you for these codewords or numbers prior to the installation of protected software. You can enter or change the numbers via the Graphical User Interface (using **Add New Codeword** from the **Actions** menu) or by using the appropriate default (**-x codeword=** ***xxxx*** **and -x customer_id=** ***xxx***) on the command line. A sample codeword certificate is shown below:

```
HP Sales Order Number:12345678-90123C
Date:14Feb96
DISC PART#:B3108-31083
CUSTOMER ID: 12345678-90123C
CODEWORD: 1234 5678 9012 3456 7890 1234 5678


PRODUCT NUMBER      PRODUCT DESCRIPTION
---------------      -------------------
B2491A                MirrorDisk/UX
B3701AA               GlancePlus Pak
```

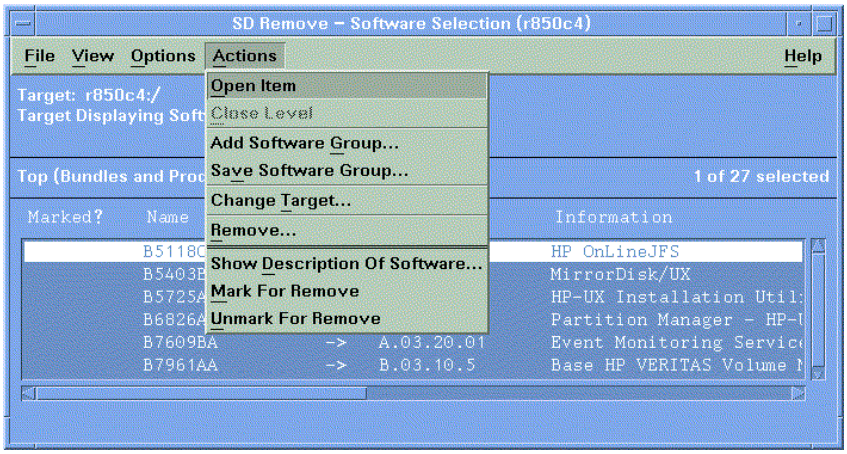

© 2004 Hewlett-Packard Development Company, L.P.

20–8. SLIDE: Removing Software

Removing Software

Use either of these commands to remove software:

```
# swremove -x autoreboot=true mybundle
# swremove -i
```



Student Notes

SD-UX also provides a mechanism for safely removing software from your system. This may be useful if:

- The software is no longer needed on the system.
- The disk space occupied by the software is needed for some other purpose.

The **`/usr/sbin/swremove`** command can be used to remove software. Like swinstall, **`swremove`** supports both a command-line and a GUI/TUI interface.

From the command line, simply run the following. If the software you wish to remove doesn't contain any kernel filesets, **`swremove`** won't actually reboot the system.

```
# swremove -x autoreboot=true mybundle
```

When executed without any options or arguments, **`swremove`** launches a **`swinstall`**-like GUI/TUI interface. The steps below explain how to remove software via the GUI/TUI interface:

1. Run **`/usr/sbin/swremove`**.

© 2004 Hewlett-Packard Development Company, L.P.

2. Select the product/bundle/fileset to remove with the mouse or space bar.

3. Select A**ctions --> Mark for Remove**.

4. Select A**ctions --> Remove (analysis)**.

After checking dependencies, **swremove** removes the appropriate files and directories, and removes the product's entry from the IPD. After the remove is complete, you should check the log to ensure that the remove was successful.

© 2004 Hewlett-Packard Development Company, L.P.

20–9. LAB: Hands-On, Using the Software Distributor

Directions

Your instructor will tell you the address of a depot that you should use for this lab exercise, and a product that you should install. Record this information below:

Depot hostname: _____

Depot pathname: _____

Perform all the tasks suggested below. Record the commands you use, and answer all the questions.

1. Remove the "perl" product from your student machine. Any errors or problems?

2. Do an **`swlist`** to see if the perl product was successfully removed.

3. Now reinstall the perl product from the depot server suggested by your instructor. Will the install require a reboot? Must any dependencies be met?

4. Verify that perl is correctly installed on your machine.

© 2004 Hewlett-Packard Development Company, L.P.

5. Create your own local perl depot by **`swcopy`**ing the perl product from the depot suggested by your instructor to your own local **`/var/spool/sw/perldepot`** depot.

6. Did **`swcopy`** automatically register the new depot? **`swlist`** the depots on your machine to find out!

7. What products are included in your perl depot? How can you find out?

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.

Module 21 — Managing Patches with SD-UX

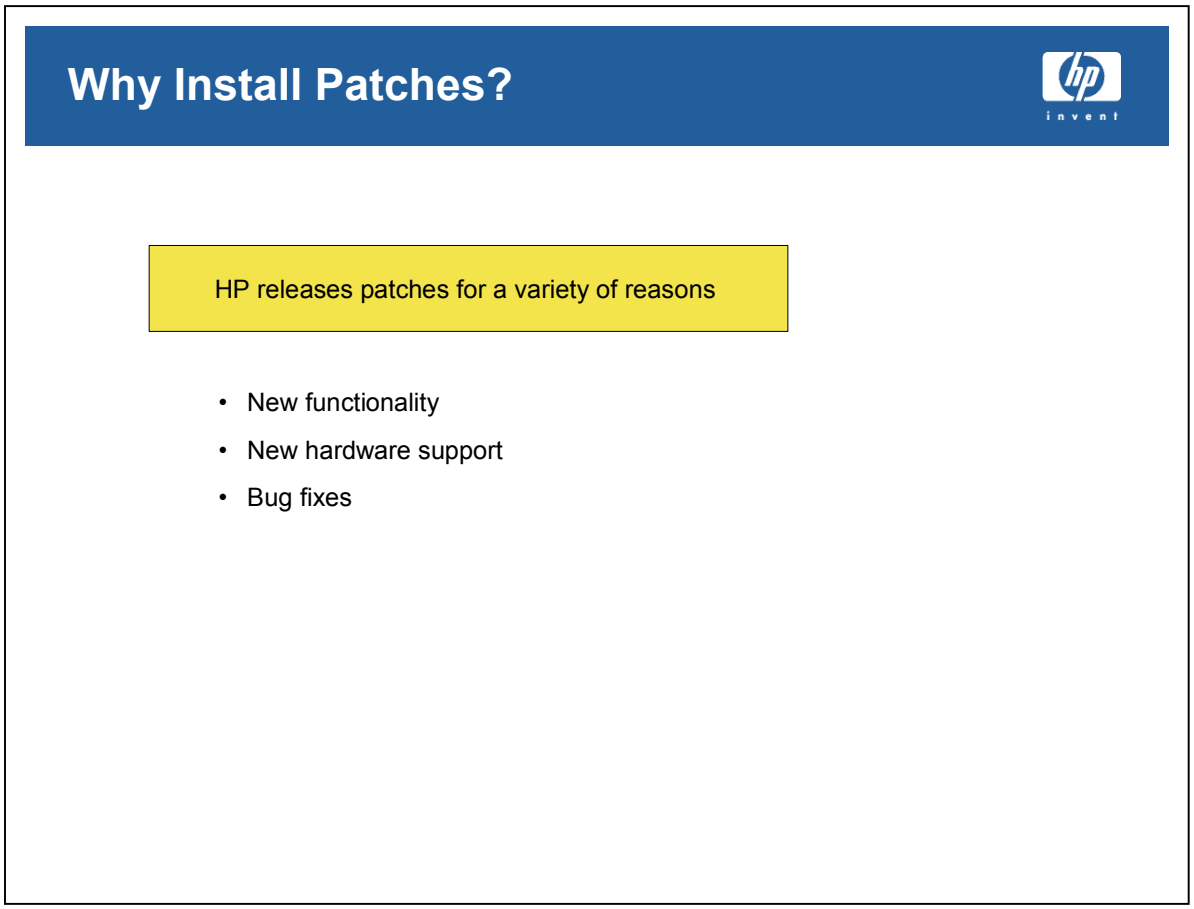
Objectives

Upon completion of this module, you will be able to do the following:

- Describe the purpose of HP-UX patches.
- Describe HP's patch rating scheme.
- Describe HP's patch naming convention.
- Describe the concept of a "superseding" or "cumulative" patch.
- List five sources for obtaining HP-UX patches.
- Download and install patches from the ITRC web patch database.
- Install patches from a Response Center tape.
- Install patches from the HP Support Plus and Software Pack patch bundles.
- List currently installed patches.
- Remove unnecessary patches.

© 2004 Hewlett-Packard Development Company, L.P.

21–1. SLIDE: Why Install Patches?



Student Notes

As a system administrator you will likely be responsible for installing patches for the HP-UX operating system and other applications and utilities. Patches may be installed for several different reasons:

- New functionality: HP is constantly incorporating new functionality in the HP-UX operating system. In order to make this functionality available to the installed customer base without requiring a full operating system update, HP sometimes provides these new features in the form of a patch.

  Example: PHCO_27035 was one of several HP-UX 11.11 enhancement patches that introduced support for the **`/etc/shadow`** file.

- New hardware support: HP-UX may require a patch to support new types of processors, interface cards, and devices.

  Example: PHKL_27154 was one of several HP-UX 11.11 hardware enablement patches required to support the PA8800 processor, which was introduced after 11.11 was first released.

© 2004 Hewlett-Packard Development Company, L.P.

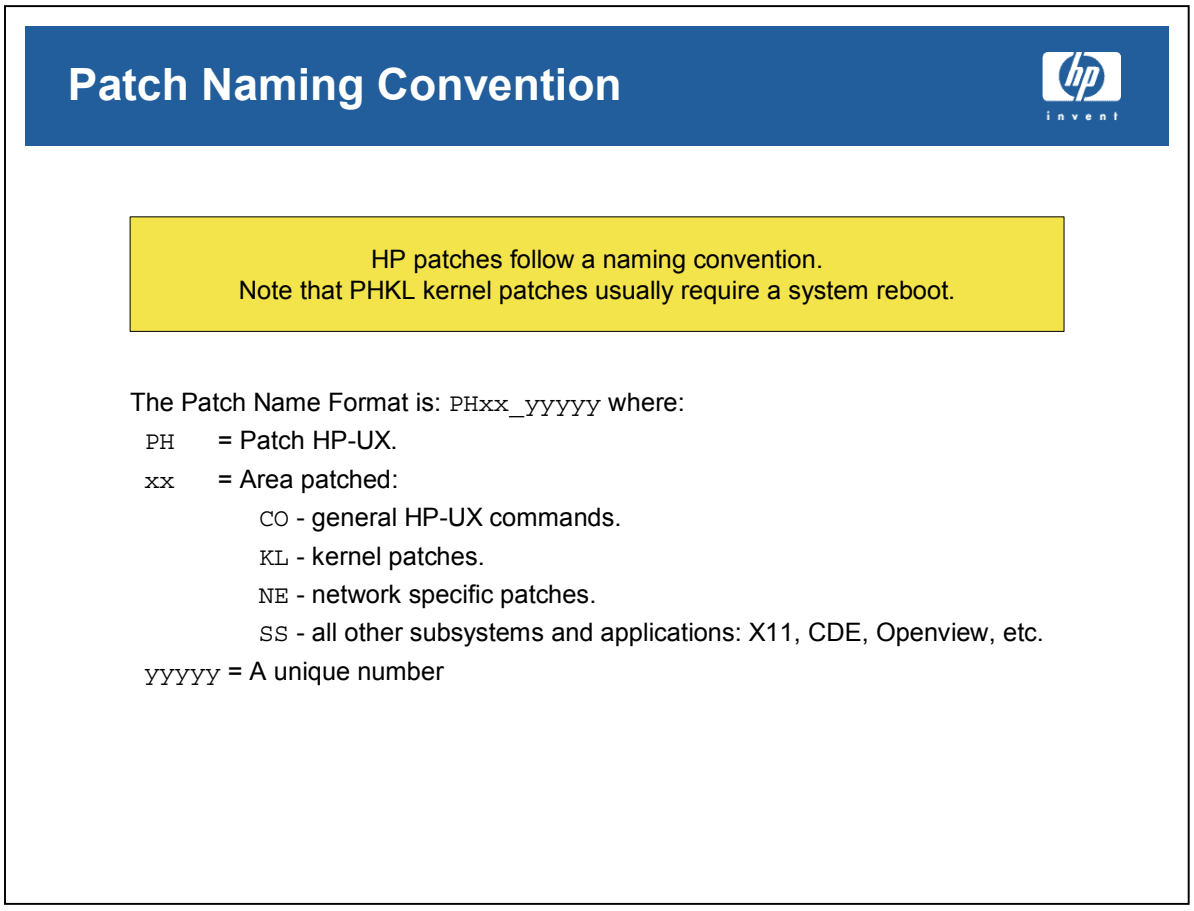
- Bug fixes: HP regularly provides patches to fix bugs identified in the operating system and other applications. It is especially important to install “Critical Fix” patches that patch critical issues that may result in downtime.

  Example: PHKL_24026 was a critical HP-UX 11.11 VxFS patch that addressed a number of problems that occasionally led to VxFS file system corruption.

Some patches provide both enhancements and bug fixes. For instance, PHKL_26743 was an HP-UX 11.11 LVM cumulative patch which fixed a variety of LVM issues that were known to cause system panics, and introduced some performance enhancements.

© 2004 Hewlett-Packard Development Company, L.P.

21–2. SLIDE: Patch Naming Convention



Student Notes

Browsing a list of patches can be overwhelming — there are literally thousands of patches available for various versions of HP-UX. The patch naming convention ensures that the administrator can identify a patch's general functionality by simply reading the patch name.

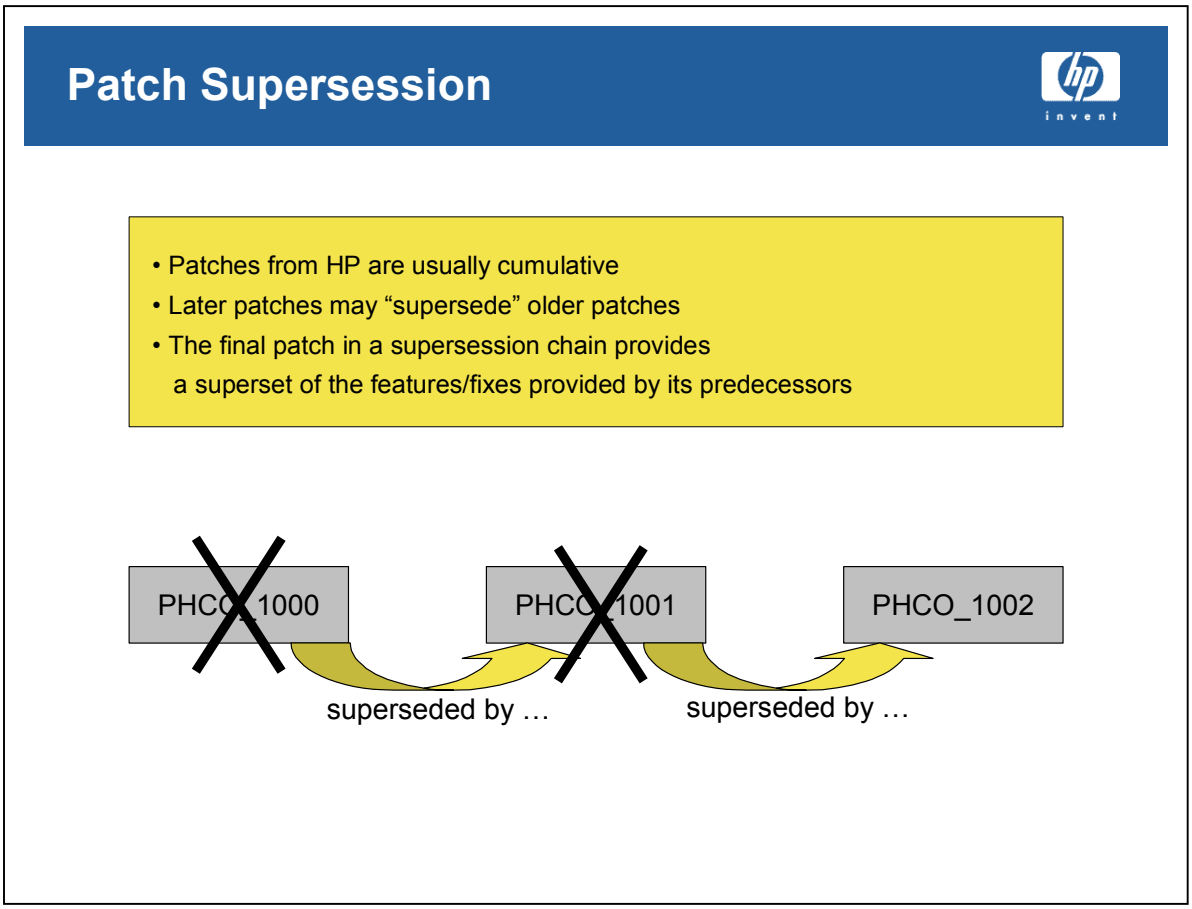
- PHCO patches patch commands, libraries, and man pages.
- PHKL patches patch kernel filesets, and often require a system reboot.
- PHNE patches patch network services and related commands.
- PHSS patches patch other OS subsystems and applications

Examples:

`PHCO_6349`	*SAM patch*
`PHKL_6681`	*Patch for process management*
`PHNE_6372`	*LAN products patch*
`PHSS_6433`	*MC/ServiceGuard patch*

© 2004 Hewlett-Packard Development Company, L.P.

21–3. SLIDE: Patch Supersession



Student Notes

Over time, multiple patches may be released for a single product or fileset. When this occurs, each patch must include all of the functionality introduced by its predecessor patches. The newer patch is said to supersede all earlier patches. A series of patches, each replacing the previous patch, forms a supersession chain. In general, the patch numbers increase along a patch supersession chain.

In the example on the slide, PHCO_1000 is superseded by PHCO_1001, which is superseded by PHCO_1002. PHCO_1002 includes all of the functionality introduced by PHCO_1000 and PHCO_1001.

When a patch is superseded, it remains on the system, but is not active. Only the final patch in the chain is in the "active" applied state. Since patches are designed to be cumulative, it isn't necessary to install *all* patches in a supersession chain; if you install a superseding patch, it isn't necessary to install the patch's predecessors. In fact, the presence of a superseding patch will *prevent* the installation of any preceding patch.

© 2004 Hewlett-Packard Development Company, L.P.

21–4. SLIDE: Patch Ratings

Patch Ratings

Every patch is assigned a star rating that indicates how thoroughly the patch has been tested.
Many customers only install 2- and 3-star patches.

Rating	Explanation
★	HP has done functional testing to verify that the patch fixes the problem that it purports to fix. Unwanted side effects were not discovered.
★★	Patch has been installed in a reasonable number of customer environments with no problems reported.
★★★	Patch has been stress- and performance-tested by HP in simulated customer mission-critical environments using common application stacks.

Student Notes

If a product has been patched by several superseding patches, should you install the most recent patch in the supersession chain? Not necessarily. HP assigns a rating to each patch to help customers identify appropriate patches for their systems. The higher the rating, the lower the risk of side-effects, and the more suitable the patch is for mission critical environments.

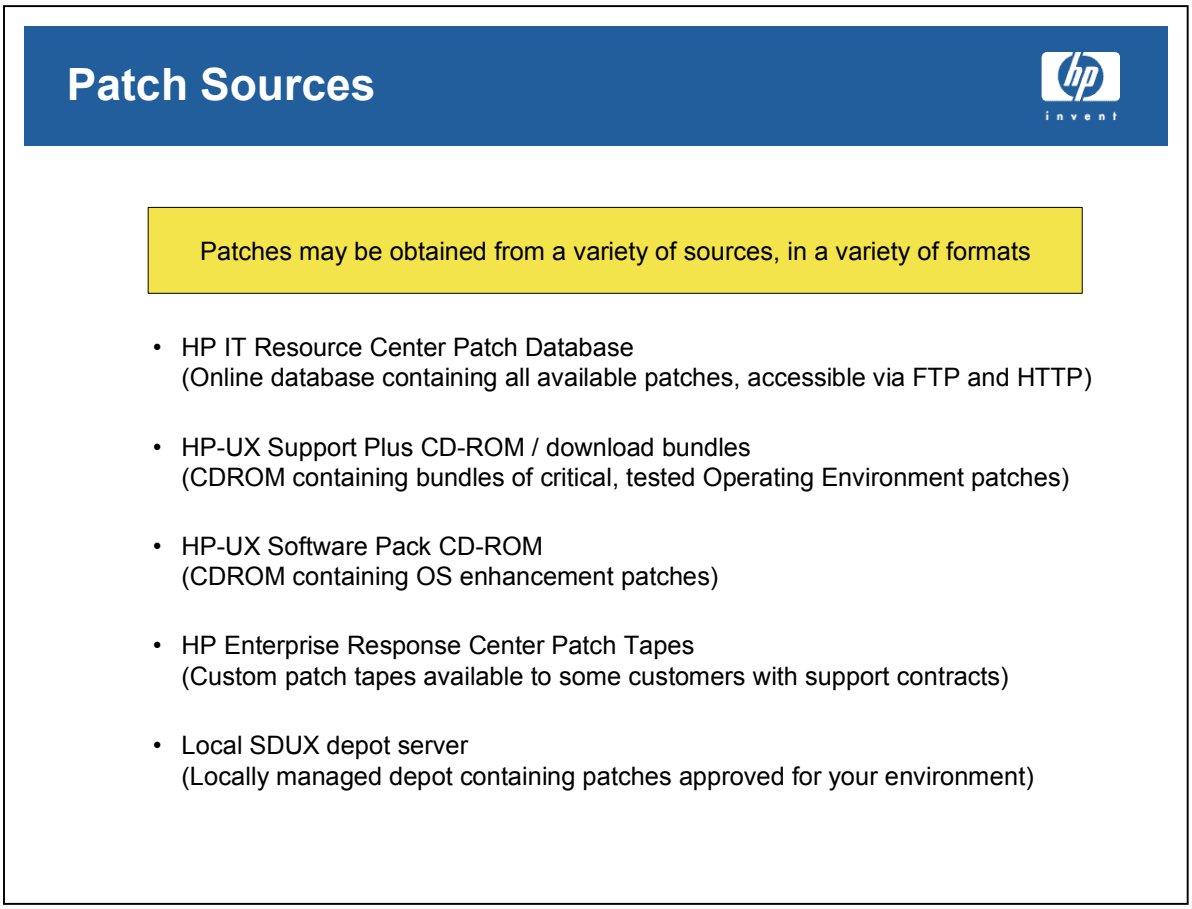
In the interest of timeliness, HP releases a patch after it meets HP's minimum patch quality standards. Patches are assigned a rating of 1-star upon initial release. If a patch has a 1-star rating and you haven't encountered the symptoms that the patch addresses, you may wish to defer installing the patch until further testing verifies that the patch is safe.

After initial release, patches undergo testing both in customer environments and within HP. The patch's rating is then updated accordingly as HP's confidence in the patch increases. Patches may be upgraded from 1-star to 2-stars at any time.

Patches undergo testing for promotion to a 3-star rating on a quarterly basis. 3-star patches are stress- and performance-tested by HP in simulated customer mission critical environments using common application stacks. Not all patches undergo this testing.

© 2004 Hewlett-Packard Development Company, L.P.

21–5. SLIDE: Patch Sources



Student Notes

Patches can be obtained from several sources.

- HP IT Resource Center Patch Database
  (Online database containing all available patches, accessible via FTP and HTTP)
- HP-UX Support Plus CD-ROM / download bundles
  (CDROM containing bundles of critical, tested Operating Environment patches)
- HP-UX Software Pack CD-ROM
  (CDROM containing OS enhancement software)
- HP Enterprise Response Center Patch Tapes
  (Custom patch tapes available to some customers with support contracts)
- Local SDUX depot server
  (Locally managed depot containing patches approved for your environment)

© 2004 Hewlett-Packard Development Company, L.P.

HP IT Resource Center Patch Database

The ITRC web site has a searchable database containing engineering notes, software status and release bulletins, security bulletins, and patches. You can search for and download individual patches from the ITRC via any web browser:

```
http://www.itrc.hp.com
```

The ITRC patch database is also available via anonymous FTP, though the FTP interface doesn't provide the search and analysis tools that are available on the web. If you want to download a specific patch via FTP, go to:

```
ftp://ftp.itrc.hp.com
```

Each patch is packaged as a **shar** archive containing a text file description of the patch, and an SD-UX depot format file from which you can install the patch itself.

Obtaining Patches from the HP Support Plus CD-ROM / Download Bundles

If you have an HP support contract, you should receive an HP-UX Support Plus CDROM each quarter, which contains tested bundles of HP-UX Operating Environment patches. The CD does not contain HP Openview or other application patches. The administrator can select and install appropriate patches via the **swinstall** command. Note that each updated Support Plus release is cumulative and completely replaces any previous Support Plus release.

- Each Support Plus CD is HP-UX version specific and may include the following bundles: A Quality Pack depot containing 2- and 3-star critical fix patches.
- A Hardware Enablement (HWE) bundle that provides patches required for new systems and for add-on hardware supported on HP-UX 11.0 and HP-UX 11i v1 OEs. The HWE bundles also provide patches that support the latest boot devices and I/O adapters with the Ignite-UX tools.
- HP-UX 11i v1 Required Patch Bundle (BUNDLE11i) delivers patches for several kernel header files that were incorrectly built and shipped on the initial HP-UX 11i v1 Operating Environment (OE) media as well as patches required during a cold install or update to an HP-UX 11i v1 OE release. BUNDLE11i is included on all HP-UX 11i v1 Support Plus and OE CD1 media beginning in February 2001.
- Diagnostics, including Support Tool Manager (STM) for online diagnostics, ODE (off-line diagnostics), EMS hardware monitors, Predictive Support, EMS Kernel Resource Monitor, and the Instant Capacity on Demand (iCOD) client products.
- HP Instant Support Enterprise Edition (ISEE) bundle that is a support solution that enables delivery of HP remote support services over the Internet. HP ISEE provides continuous hardware event monitoring and automated notification to identify and prevent potential critical problems during contract coverage hours.

Support contract customers receive Support Plus CDROMs via quarterly media kit updates. Customers without support contracts can download Support Plus bundles from **http://www.software.hp.com** free of charge.

© 2004 Hewlett-Packard Development Company, L.P.

Obtaining Patches from the HP-UX Software Pack CD-ROM

The HP-UX Software Pack (or Additional Core Enhancements in 11.00 and 10.20) is a CDROM containing new core enhancement patches that can optionally be installed on your system. The current 11.11 software pack includes:

- HP-UX Compressed Dump Version A.01.01
- Base HP-UX VERITAS File System 3.5 (HP OnlineJFS/JFS 3.5)
- HP-UX Interrupt Migration
- HP-UX Processor Sets
- HP-UX 11i Internet Protocol Version 6 (IPv6)

The Software Pack CDROM is included with the CDROM media kits that are distributed to support customers on a quarterly basis.

Obtaining Patches from HP Enterprise Response Center Patch Tapes

Some customers with support contracts can request custom patch tapes from the HP Enterprise Response Center. Response Center tapes use the standard SD-UX depot format.

Obtaining Patches from a Local SDUX depot server

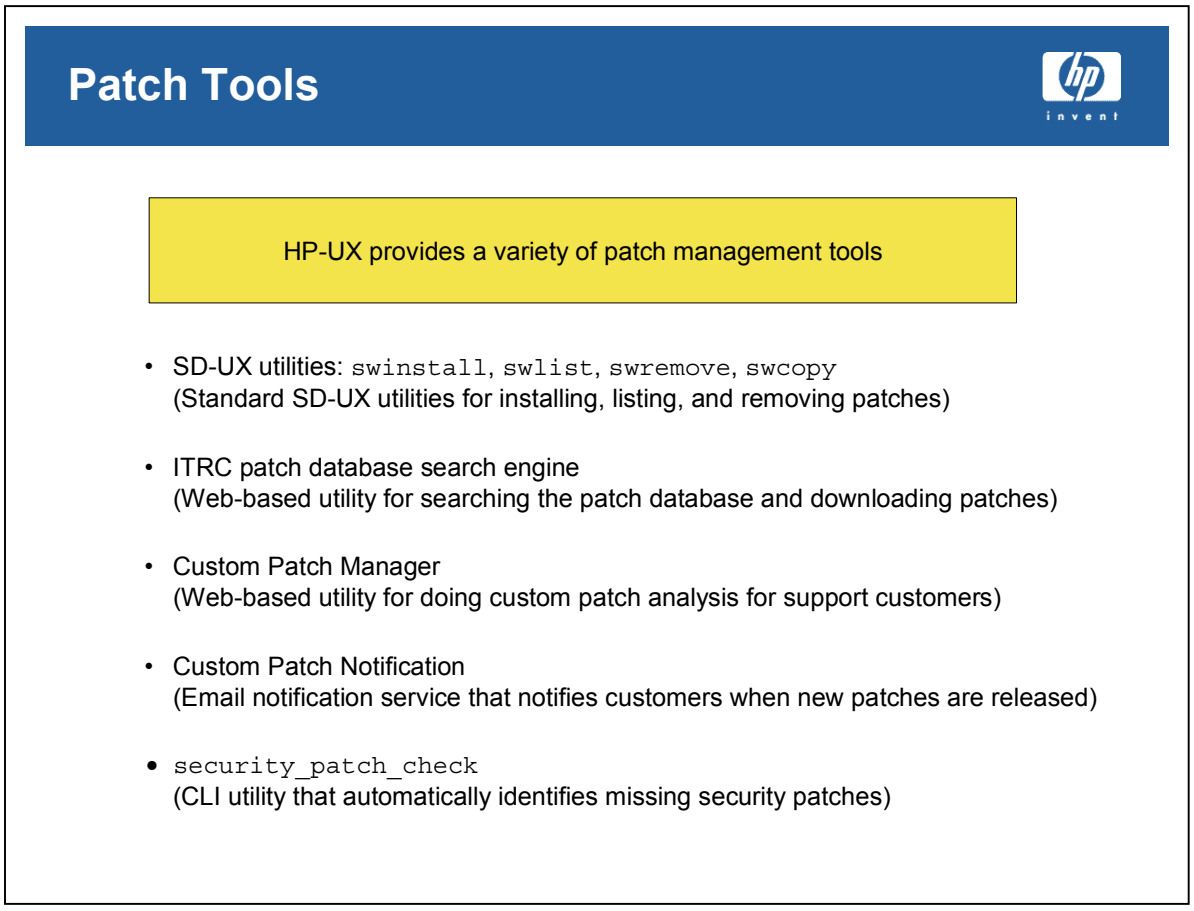
Organizations with multiple HP-UX systems may choose to configure an SD-UX depot server to ensure consistent, reliable software installs.

- Instead of managing stacks of CDROMs and tapes, you can **`swinstall`** software and patches from your SD-UX depot server. This is especially helpful when installing systems that don't have a CDROM or tape drive available.
- A depot server provides a single point of administration for your software and patch updates.
- Installing all of your hosts from a central depot server ensures that all hosts have a similar software/patch image.
- Configuring a depot server makes it possible to remotely install and manage software. Individual hosts on your network can "pull" software from the depot server. With HP-UX 11i, it is now possible to "push" software installs and updates from a depot server to one or more remote targets.
- After you select a patch, product, or bundle in a depot, **`swinstall`** auto-selects other products from the depot that your selected product requires.
- When **`swinstall`**ing software from an SD-UX depot, if the depot contains patches for the user-selected product(s), **`swinstall`** will automatically select and install those patches at the same time that it installs the selected product itself. This can significantly decrease the amount of downtime required to update software and patches on a system.

Attend HP's Patch Management course (H8339S) to learn more about SD-UX depot creation and management.

© 2004 Hewlett-Packard Development Company, L.P.

21–6. SLIDE: Patch Tools



Student Notes

HP offers a number of tools and utilities for installing and managing HP-UX patches.

- SD-UX utilities: You can install, remove, list, and copy patches using the standard SD-UX software management utilities. Several special options are available specifically for managing patches. These will be discussed in detail later in this chapter.

- ITRC patch database search engine: The ITRC maintains a database of every patch ever released for HP-UX. The website also includes a powerful search engine that makes it relatively easy to search the database for patches pertaining to symptoms you see on your systems. The core features of the ITRC patch search engine will be covered later in this chapter. For more information, visit `http://www1.itrc.hp.com/service/patch/mainPage.do`.

© 2004 Hewlett-Packard Development Company, L.P.

- Custom Patch Manager (CPM): CPM is a patch analysis tool on the ITRC website that analyzes the products and patches currently on your system and recommends current patches that appear to be missing. CPM can even download and install the patches for you. CPM is only available to customers who have support contracts. Follow the CPM link on **`http://us-support2.external.hp.com/common/bin/doc.pl`** for more information.

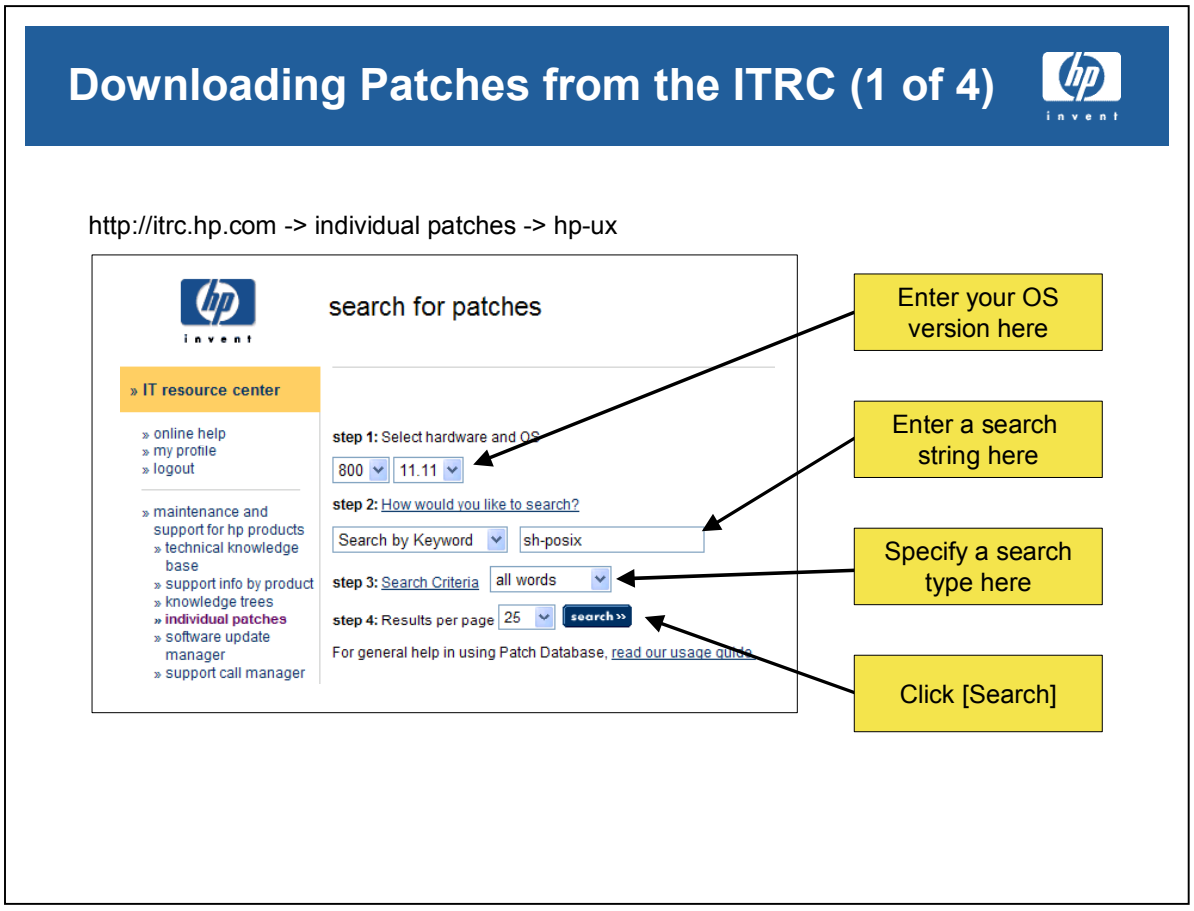
- Custom Patch Notification: Customers that use CPM can register to receive email notification messages when new patches are released that apply to products installed on their systems. This service is only available to customers who have support contracts. Follow the Custom Patch Notification link on **`http://us-support2.external.hp.com/common/bin/doc.pl`** for more information.

- **`security_patch_check`**: This useful tool compares the list of patches installed on your system to a catalog of critical security patches on the ITRC FTP site, then generates a list of security patches that are missing on your system. The tool is very easy to configure, and ensures that your system always has the most current security patches available. You can download and install the tool and its manual page from **`http://software.hp.com`**.

All of these tools are described in detail in HP Education's Patch Management course (H8339S).

© 2004 Hewlett-Packard Development Company, L.P.

21–7. SLIDE: Downloading Patches from the ITRC (1 of 4)



Student Notes

The remaining slides in the chapter describe the process required to install patches from a variety of sources. We'll start with the ITRC web patch database, since that's the most commonly used patch source.

1. Access the ITRC web patch database.

   a. Point your browser to **`http://itrc.hp.com.`**

   b. Click the **`login`** link at top left to login. If you don't have an ITRC login, create one on the resulting login screen.

   c. After logging in, click the **`individual patches`** link under **`maintenance and support`**.

   d. On the **`patch database main`** screen, you can either enter a patch ID directly in the text box at the top of the page, or click on **`hp-ux`** to search for patches by keyword.

© 2004 Hewlett-Packard Development Company, L.P.

2. On the **`search for patches`** screen, enter your hardware type and OS version. You can determine your hardware type and OS version via the **`uname -a`** command.

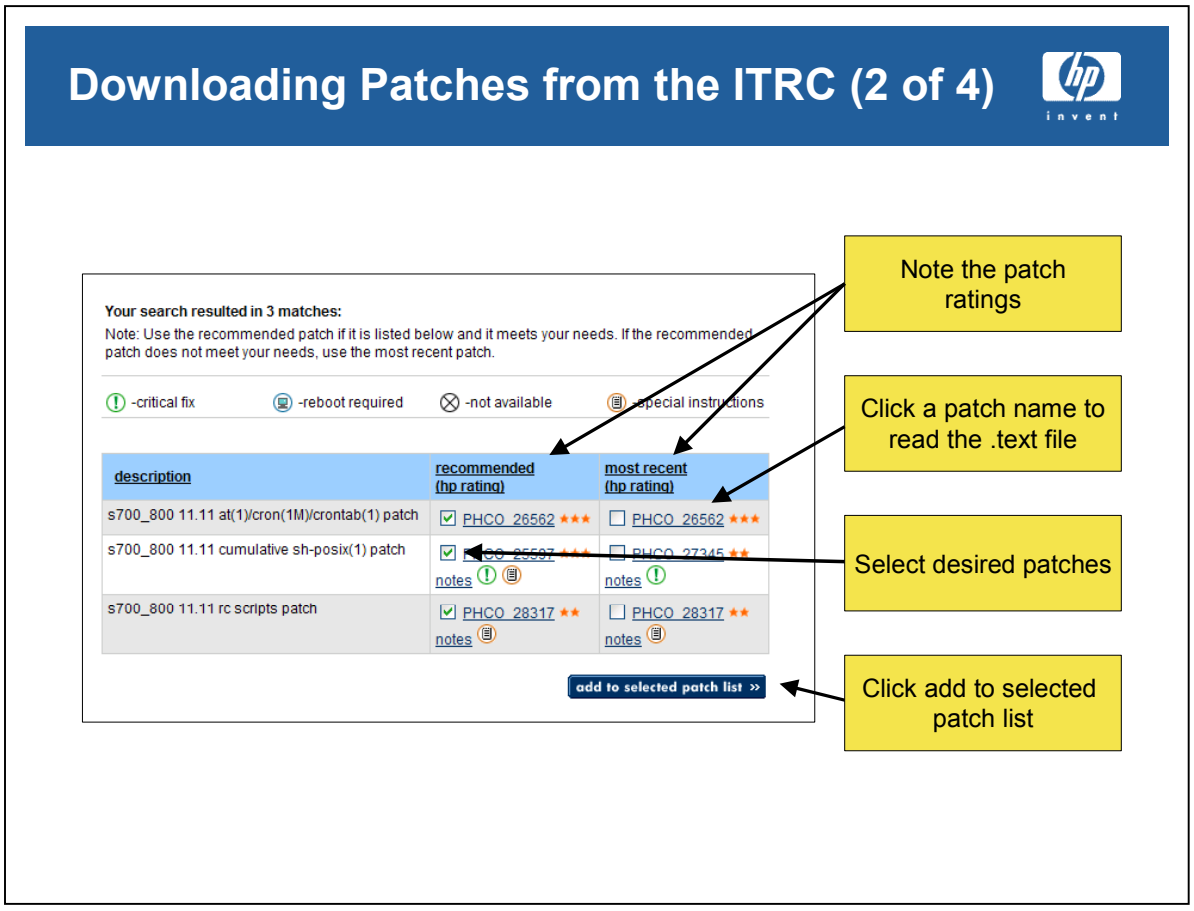
3. Next, use the **`Search by Keyword`** pull-down list to specify how you wish to search. You can search by keyword, by patch ID, or you can browse the entire patch catalog. If you choose to search by keyword or patch ID, enter the keyword or patch ID you wish to search for. The web analysis tool can perform a Boolean search, and a variety of other search types, as described in the table below.

Search criteria selected	keywords entered	search will return
Any word	HP-UX 11.0	Any document that includes *any one* of the individual words you enter, in this case *HP-UX* or *11.0* . Use spaces or commas to separate words.
All Words	HP-UX 11.0	Any document which includes *all* of the individual words you enter, in this case *HP-UX* and *11.0* . Use spaces or commas to separate words.
Exact Phrase	HP-UX 11.0	Any document containing the *entire phrase* , including such content as *HP-UX 11.0 release* or *Upgrade to HP-UX 11.0*. If you prefer to search on individual keywords, select the *All Words* search criteria.
Boolean OR	HP-UX OR MPE	Any document containing either *HP-UX* or *MPE*
Boolean AND	HP-UX AND MPE	Any document containing both *HP-UX* and *MPE*
Boolean NOT	HP-UX AND NOT 10.20	Any document containing *HP-UX* but not containing *10.20*
Boolean NEAR	HP-UX NEAR shutdown	Any document containing both words or phrases *HP-UX* and *shutdown* within 10 words of each other
Boolean ()	HP-UX and (11.0 or 11.11)	Any document which includes *HP-UX* and either *11.0* or *11.11* or both

4. Specify how many results you wish to display per page, then click **`Search`**.

© 2004 Hewlett-Packard Development Company, L.P.

21–8. SLIDE: Downloading Patches from the ITRC (2 of 4)



Student Notes

Clicking **`Search`** should cause the ITRC to display a list of patches that match the keyword(s) you specified. Read the list carefully.

1. Each row in the table should display a "Recommended Patch" as well as a "Most Recent Patch". If the most recent patch for a given product is a 2- or 3-star patch, the "Recommended" and "Most Recent" patch IDs may be identical. If the most recent patch is only a 1-star patch, however, the analysis tool may recommend using the more stable 2- or 3-star predecessor patch instead.

2. Before you install a patch, it is always a good idea to review the patch readme file. The readme files are available online. Click the patch name and read the resulting text description carefully. Determine if the patch fixes the symptoms that you've seen on your system, determine if a reboot is required, and read the installation instructions carefully. Read the instructions especially carefully if the ITRC patch grid displays a "special instructions" icon below the patch ID.

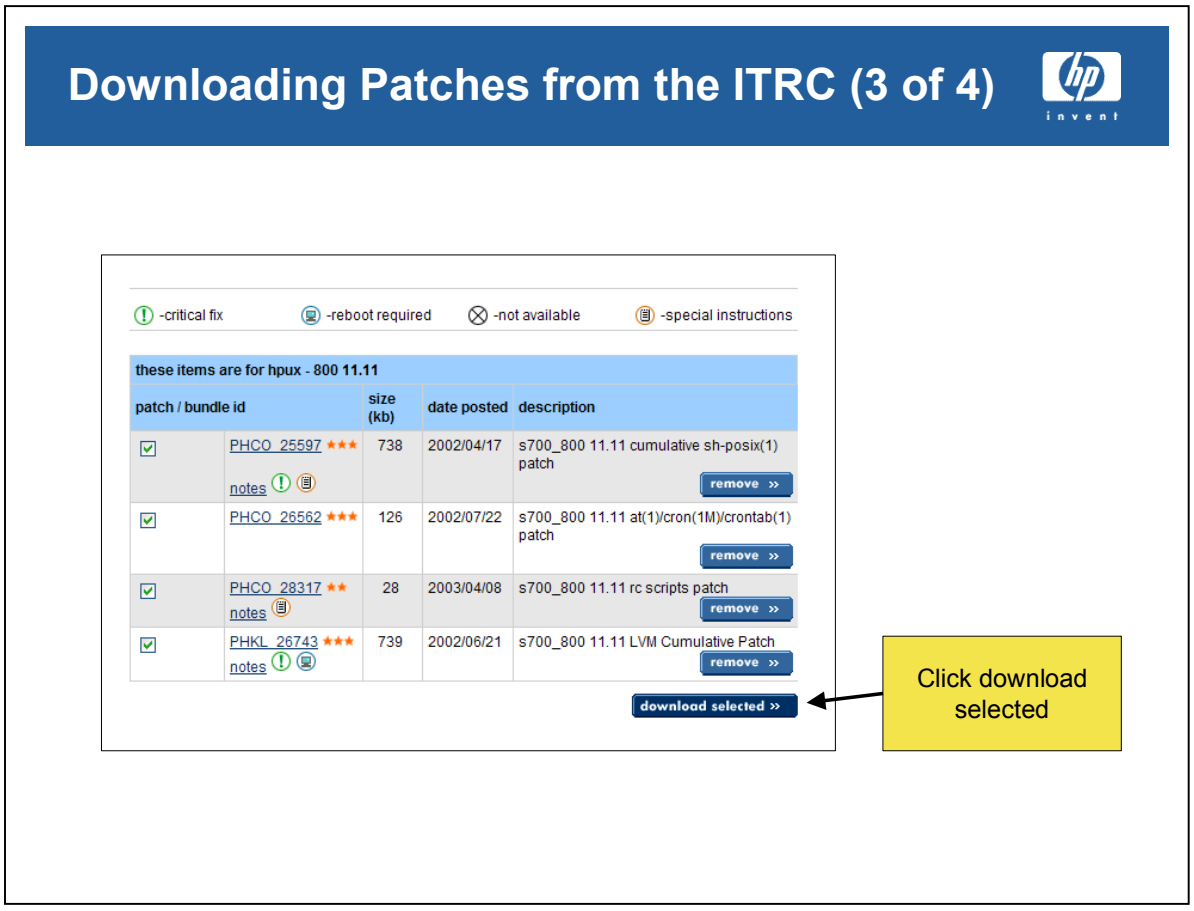
© 2004 Hewlett-Packard Development Company, L.P.

3. If you want to proceed to download the patch, click the checkbox to the left of the patch ID.

4. After selecting all of the desired patches, click the **`add to selected patch list`** button.

© 2004 Hewlett-Packard Development Company, L.P.

21–9. SLIDE: Downloading Patches from the ITRC (3 of 4)

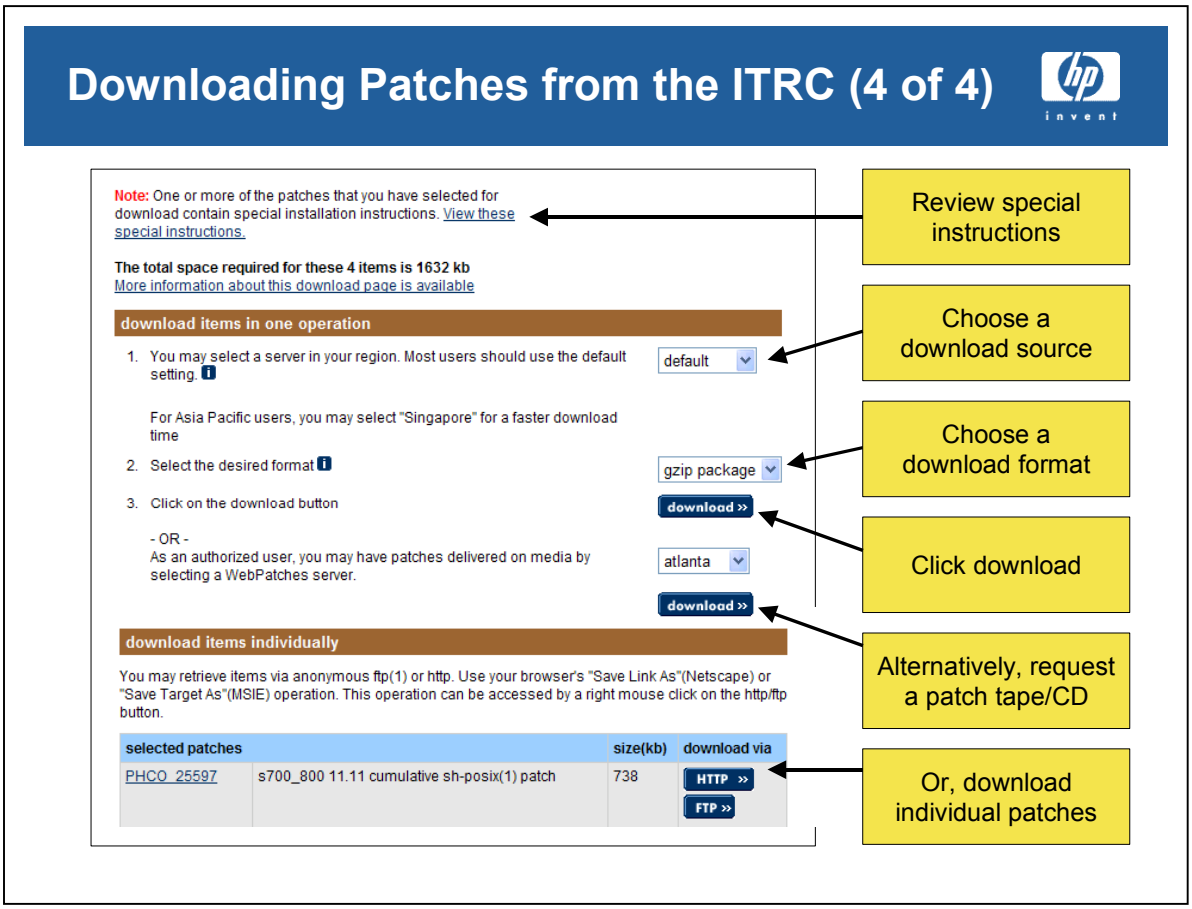


Student Notes

After clicking the **`add to selected patch list`** button, the ITRC database automatically adds any additional patches that may be required to meet dependencies. Review the resulting list, then click **`download selected`** to proceed to the patch download screen.

© 2004 Hewlett-Packard Development Company, L.P.

21–10. SLIDE: Downloading Patches from the ITRC (4 of 4)



Student Notes

Follow the instructions on the patch download page to download your selected patches.

1. At the top of the download patches screen, click the **`View special instructions`** link, and print the resulting page. You will need these special instructions when you install the patches later.

2. Choose a download source. For the Americas, Europe, Africa, and the Middle East, the default source is recommended. Customers in Asia and the Pacific should consider using the pull-down menu to select Singapore instead.

3. Choose a download format for the patch download. Options include: zip, gzip, and tar. If you intend to extract the achive on a Windows system, it might make sense to choose the zip format. If you intend to extract the patches on an HP-UX box, though, it is better to use the gzip format since gzip is now included with HP-UX.

4. Click the download button and select a destination directory in the resulting pop-up dialog box. Many administrators download patches to **`/tmp`**.

5. Support contract customers may be able to request a patch tape from the response center. Ask your HP support representative for more information.

© 2004 Hewlett-Packard Development Company, L.P.

6. At the bottom of the screen, you may alternatively download the patches one at a time via either FTP or HTTP. The gzip mechanism described above is much more convenient if you need to download multiple patches.

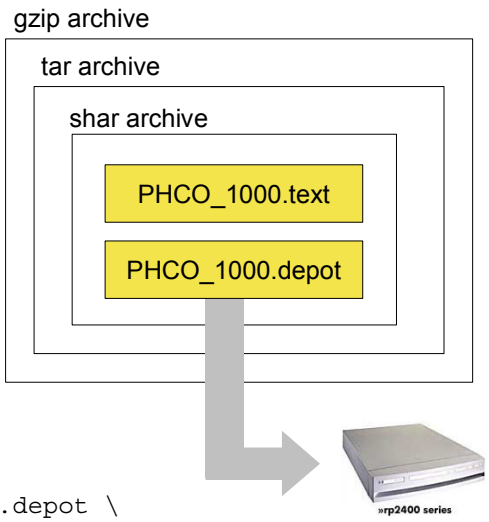
© 2004 Hewlett-Packard Development Company, L.P.

21–11. SLIDE: Installing a Single Patch from the ITRC

Installing a Single Patch from the ITRC

1. Do a full backup
2. Unzip the archive:
   `# gzip -d /tmp/patches.tgz`
3. Untar the archive:
   `# tar -xvf /tmp/patches.tar`
4. Unshar each patch:
   `# sh /tmp/PHCO_1000`
5. Read the resulting .text file carefully:
   `# more /tmp/PHCO_1000.text`
6. Install each patch:

```
# swinstall -s /tmp/PHCO_1000.depot \
            -x autoreboot=true \
            -x patch_match_target=true
```



Student Notes

Several steps are required to install a single patch from the ITRC.

1. First, do a full system backup.

2. Unzip the patch archive that you downloaded from the ITRC. This should create a **`patches.tgz`** file. If you only downloaded a single patch from the ITRC, you can skip this step.

   ```
   # gzip -d /tmp/patches.tgz
   ```

3. Next, untar the unzipped archive. This should create a separate shar archive file for each patch you selected to be included in the download. Again, if you only downloaded a single patch from the ITRC, you can skip this step.

   ```
   # tar -xvf /tmp/patches.tar
   ```


© 2004 Hewlett-Packard Development Company, L.P.

4. Unshar the patch you wish to install. This should yield a **.text** file and a **.depot** file for the patch.

```
# sh /tmp/PHCO_1000
```

5. Read the **.text** file that describes the patch. Are there any dependencies or conflicts? Will installing this patch require a reboot? What is the purpose of the patch? Are there any special installation instructions? Note that the **.text** file even includes instructions for installing the patch. A complete description of the **.text** file fields follows this procedure.

```
# more /tmp/PHCO_1000.text
```

6. **swinstall** the patch. The **-s** option shown on the slide identifies the patch depot as the source for the **swinstall** session. The **autoreboot=true** option allows **swinstall** to automatically reboot the system if necessary during the install. If the patch doesn't include any new kernel filesets, **swinstall** won't reboot the system. The **patch_match_target=true** option automatically selects patches that apply to the filesets already installed on your system. If you want to manually select patches to install via the **swinstall** GUI, execute **swinstall** without any command line options.

```
# swinstall -s /tmp/PHCO_1000.depot \
            -x autoreboot=true \
            -x patch_match_target=true
```

The .text File

The text file will include the following information:

Patch name	The name of the patch.
Patch description	A one-line description that describes the cumulative patch.
Creation date	The date that the patch was created.
Post date	The date that the patch was released for general distribution.
Hardware platforms – OS releases	The hardware platforms and HP-UX OS releases on which this patch can be installed.
Products	This field lists the product name and all product revisions to which this patch applies if it is a patch for an optional (that is, non-core operating system) product. If the patch is for the core operating system, the value in this field is N/A.
File sets	This is a list of all the file sets with their respective revision, architecture, and vendor attribute that will be updated by this patch.
Automatic reboot?	Either Yes or No depending on whether or not this patch will cause a system reboot after the installation succeeded.

© 2004 Hewlett-Packard Development Company, L.P.

Status — Either General Release or Special Release depending on the support status of the patch.

- With General Release, the patch should be installed on all systems meeting the OS, product, and dependency requirements.
- With Special Release, the site-specific patches for installation at one specific customer or set of customers, and other non-General Release patches.

Critical — Yes | No followed by text. A patch is considered "critical" if it fixes a critical problem or supersedes a patch fixing a critical problem. A problem is considered critical if at least one of the following conditions is true:

- Causes the system (OS/kernel) to fail, hang, crash or panic.
- Causes a major application to fail and thus impact severely the system's operation.
- Causes data loss or corruption.

Category tags — Identifies the type of patch. Possible values can be one or combinations of the following:

critical	Fixes one or more of the critical conditions depicted above under Critical".
panic	Fixes a system panic.
halts_system	Fixes a problem that leads to a system halt or hang.
corruption	Fixes a data corruption problem.
memory_leak	Fixes a memory loss problem that may lead to severe performance degradation and/or system halt.
defect_repair	Fixes a software defect.
hardware_enablement	Provides support for new hardware.
Enhancement	Provides new functionality.
general_release	Should be installed on any system with the appropriate software installed.
special_release	Should only be installed under specific conditions.
trial_patch	Preliminary patch. Should be replaced by the **general_release** or **special_release** version as it becomes available.

Path name — The path name is the patch's storage location on the ITRC systems. The only valid location for 11.x patches is

© 2004 Hewlett-Packard Development Company, L.P.

	/hp-ux_patches/s700_800/11.X/PHxx_yyyyy. See also under "Hardware Platforms - OS Releases" above.
Symptoms	The external symptoms of the problem.
Defect description	A detailed description of the defect that specifically addresses the explicit conditions which caused the problem (if known), and how to reproduce the problem (if known). Also can include methods to verify if the patch needs to be installed.
SR	All Service Request (SR) numbers addressed by this patch and all its predecessors. An SR is a formal request from a customer to have a defect resolved or a feature added to HP software.
Patch files	The full, installed path names of all files in this patch. If the patch replaces an object module in a library, the full path of the library is listed, with the object module following in parentheses. For example, if a patch replaces the object module **vers.o** in the library **/usr/conf/lib/libhp-ux.a**, the path listed would be **/usr/conf/lib/libhp-ux.a**(**vers.o**). Each entry is listed. A new copy of the **/sbin/mkboot** command is available in the patch. This new version applies to both 32-bit and 64-bit platforms, so it will be applied to both platforms. The **/usr/share/man/man1m.Z/mkboot.1m** man page will be installed on both 32-bit and 64-bit platforms, if the file **set OS-Core KERN-ENG-A-MAN** is installed on the system.

what(1) output	The **what(1)** output for each file or library object file listed in the Patch Files field. The **what** string is a means of identifying the software version, and thereby verifying that the patch is installed. Usually the **what** string references the name of the patch that introduced it. For cumulative patches, this may be one of the superseded patch names. As with the Patch Files information the **what(1)** output is listed with the corresponding base file set. An sample **what(1)** output is:
	
	`OS-Core.CORE-SHLIBS,fr=B.11.00,fa=HP-UX_B.11.00_32/64,v=HP:`
	`/usr/lib/libc.1:`
	`PATCH-11.00:PHCO_13284 libc.1 Nov 19 1997 16:35:42`

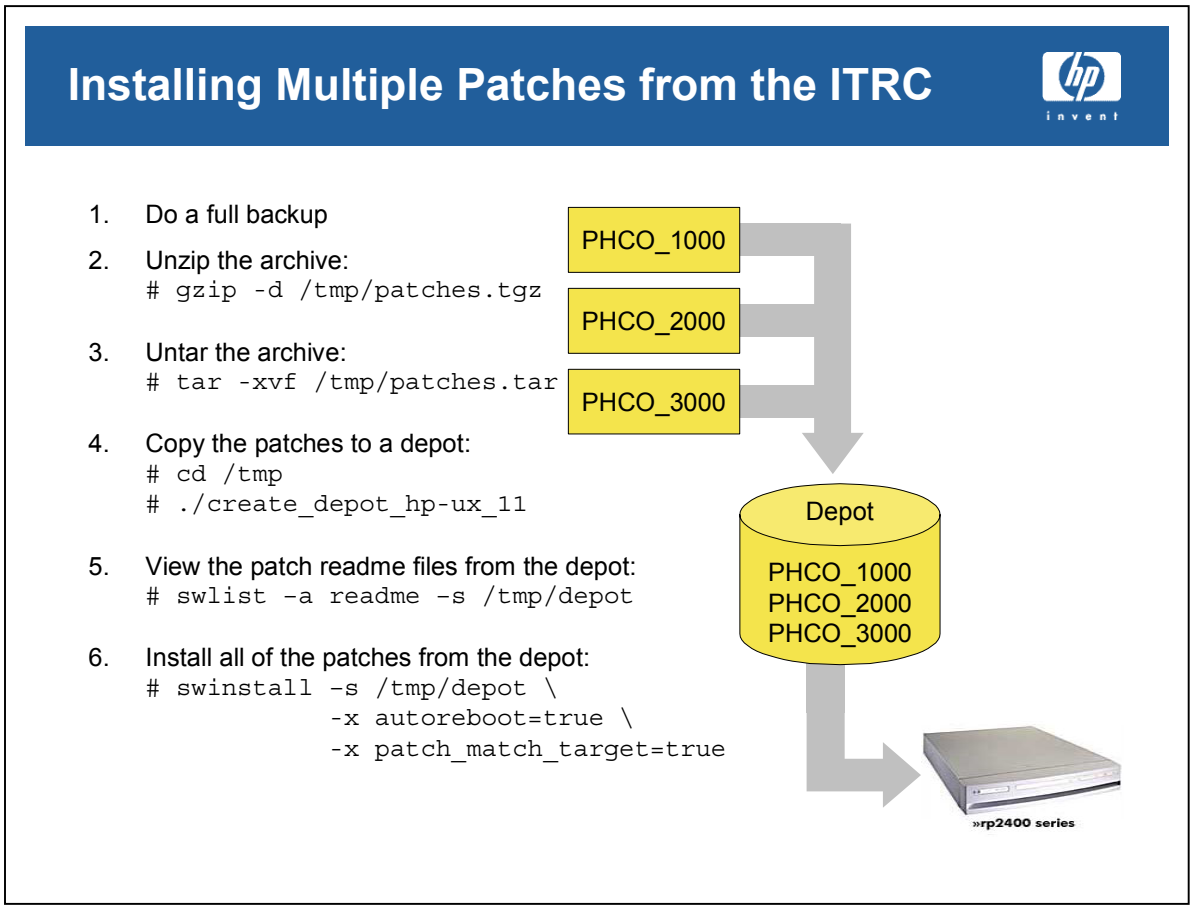
Patch conflicts	All known patch conflicts, both on a file basis as well as on a behavioral basis.
Patch dependencies	All patches that must be installed to ensure proper operation of this patch.
Hardware dependencies	Specific system models to which this patch is limited.
Other dependencies	Any non-patch and non-hardware requirements that may exist.

© 2004 Hewlett-Packard Development Company, L.P.

Supersedes	The list of all patches superseded by this patch.
Equivalent patches	All known comparable patches for other hardware platforms and OS releases, not including this patch.
Patch package size	The SD-UX depot size in Kbytes.
Installation instructions	The standard installation instructions common to all patches.
Special installation Instructions	Any special instructions not included in those mentioned above.

© 2004 Hewlett-Packard Development Company, L.P.

21–12. SLIDE: Installing Multiple Patches from the ITRC



Student Notes

If you downloaded multiple patches from the ITRC, and wish to install all of them at once, the procedure is a bit different.

1. First, do a full system backup in case the patch installation fails.

2. Unzip the patch archive that you downloaded from the ITRC. This should create a **`patches.tgz`** file.

   ```
   # gzip -d /tmp/patches.tgz
   ```

3. Next, untar the unzipped archive. This should create a separate shar archive file for each patch you selected to be included in the download, as well as a **`create_depot_hp-ux_11`** script that we will use in the next step.

   ```
   # tar -xvf /tmp/patches.tar
   ```


© 2004 Hewlett-Packard Development Company, L.P.

4. Run the **create_depot_hp-ux_11** script. The script unshars the patch shar archives that were included in your **patches.tgz** file, creates an SD-UX directory depot in **./depot**, and **swcopy**s the patches from the **.depot** files to the new directory depot.

```
# cd /tmp
# ./create_depot_hp-ux_11
```

5. View the patch **readme** files from the depot. Watch for special instructions and patches that require a reboot.

```
# swlist –a readme –s /tmp/depot
```

6. Install the patches from the directory depot. The **-s** option identifies the patch depot as the source for the **swinstall** session. The **autoreboot=true** option allows **swinstall** to automatically reboot the system if necessary during the install. The **patch_match_target=true** option automatically selects patches that apply to the filesets already installed on your system. If you want to manually select patches to install via the **swinstall** GUI, execute **swinstall** without any command line options.

```
# swinstall –s /tmp/depot \
            -x autoreboot=true \
            -x patch_match_target=true
```

create_depot_hp-ux_11 Syntax

A customized **create_depot_hp-ux_11** script is automatically included in each **patches.tgz** archive that you download from the ITRC. By default, the script unshars the patch shar archives that were included in your **patches.tgz** file, creates an SD-UX directory depot in **./depot**, and **swcopy**s the patches from the **.depot** files to the new directory depot.

The command recognizes several options and arguments that may be used to alter the default behavior:

```
Usage: create_depot [-h] [-i] [-u] [-f]
                       [-d depot_name]
                    [-b bundle_tag]
                    [-t bundle_title]
```


© 2004 Hewlett-Packard Development Company, L.P.

A detailed description of the options follows below:

-h	Displays help and usage information.
-i	Runs interactively, prompting the user for each piece of information described below.
-u	Attempts to use existing patch **.depot** files. If the patch **.depot** files haven't been unshar'ed yet, **-u** causes **create_depot_hp-ux_11** to fail.
-f	If the **./depot** directory already exists, the **-f** option is required to force the script to add the new patches to the existing depot.
-d ***depot_name***	Specifies a target depot. By default, patches are copied to **./depot**. You may wish to collect all of your patches from multiple patch downloads in a single patch depot, so you have all of your patches in a single location. If the target depot already exists, you must specify the **-f** option.
-b ***bundle_tag***	Specifies a bundle wrapper tag for the patches. When you **swlist** the **./depot** patch depot, the specified patch bundle will be listed rather than the individual patch names. If you consolidate patches from multiple patch download sessions in a single depot, bundle tags make it easier to determine which patches were downloaded when. The default bundle tag is "**BUNDLE**".
-t ***bundle_title***	Associates a more descriptive bundle title with your patch bundle. The default title is **Patch Bundle**.

Note that the list of patches to be copied to the target depot directory is hard-coded in the **create_depot_hp-ux_11** script. If you want to copy other patches to a depot, you should use the **swcopy** command.

© 2004 Hewlett-Packard Development Company, L.P.

21–13. SLIDE: Installing Patches from CD

Installing Patches from CD

1. Do a full backup.
2. `# mkdir /patchCD`
3. `# mount /dev/dsk/cxtxdx /patchCD`
4. `# ls /patchCD`
5. `# more /patchCD/GOLDBASE11i.readme`
6.
```
# swinstall    -s /patchCD/GOLDBASE11i \
               -x autoreboot=true \
               -x patch_match_target=true
```

Student Notes

Many administrators choose to proactively install patches on their systems from the HP Support Plus CD. Any customer with a support contract can subscribe to this service, and receive a tested bundle of the latest HP-UX patches on a quarterly basis. Customers who don't have a support contract can download the Support Plus patch bundles from **`http://software.hp.com`**.

© 2004 Hewlett-Packard Development Company, L.P.

Each Support Plus CD contains diagnostic software, several README files, and several patch bundles. There are three general types of patch depots and bundles on the Support Plus CDROMs:

Depot Name	Bundle Name	OS	Purpose
/cd/QPK1100	**QPK1100**	11.00	HP-UX 11.00 "Quality Pack" bundle containing recommended Core OS patches.
/cd/HWE1100	**HWE1100**	11.00	HP-UX 11.00 "Hardware Enablement" bundle containing patches required to support new hardware.
/cd/BUNDLE11i	**BUNDLE11i**	11i	HP-UX 11i v1 depot/bundle containing critical kernel patches. This bundle is required.
/cd/GOLDQPK11i	**GOLDBASE11i**	11i	HP-UX 11i "Quality Pack" bundle containing recommended Core OS patches.
/cd/GOLDQPK11i	**GOLDAPPS11i**	11i	HP-UX 11i "Quality Pack" bundle containing recommended patches for applications such as Netscape, Apache, and others that are included in the 11i Operating Environments.
/cd/HWEnable11i	**HWEnable11i**	11i	HP-UX 11.00 "Hardware Enablement" bundle containing patches required to support new hardware.
/cd/DIAGNOSTICS	**OnlineDiag**	11.00 and 11i	Diagnostic and Event Monitoring Services (EMS) software for identifying system hardware problems. Though this isn't a patch bundle, the OnlineDiag product changes so frequently that it is included on the Support Plus CD and should be updated each time you receive a new media kit.

Installing Patches from the Support Plus CD

Several steps are required to install patches from the Support Plus CD:

1. Anytime you install patches, you should start by doing a full backup of your system.

2. Obtain the most recent Support Plus CD for your version of HP-UX. The CDs are available for 11.00 and 11i. Load the appropriate CD in your CDROM drive.
   If you don't have the Support Plus CD, you can obtain the same patch bundles from HP's **http://software.hp.com** website. If you choose to take this approach, download the patch bundle from the website and skip to step 8 below.

3. Use the **ioscan** command to determine your CDROM drive's device file name.

   ```
   # ioscan -funC disk
   ```


© 2004 Hewlett-Packard Development Company, L.P.

4. Create a mount point for the CD.

   ```
   # mkdir /patchCD
   ```

5. Mount the CD using the device filename obtained in step 3 and the mount point directory created in step 4.

   ```
   # mount /dev/dsk/cxtxdx /patchCD
   ```

6. List the contents of the CD. The CD will likely contain a PDF file describing HP-UX patch management, some diagnostic tools that may be loaded on your system, and **.readme** files describing the patches in each patch bundle, and several patch depots.

   ```
   # ls /patchCD
   ```

   See the table above for explanations of the various patch depot types.

7. View the **.readme** file associated with the bundle you wish to install. The **readme** file describes which patches are included in the bundle.

   ```
   # more /patchCD/GOLDQPK11i.readme
   ```

8. Use the **swinstall** command shown on the slide to install the matching patches from the appropriate bundle! If you prefer to manually select the patches to install via the **swinstall** GUI, run swinstall without the **-x** options.

   ```
   # swinstall -s /patchCD/GOLDQPK11i \
               -x autoreboot=true \
               -x patch_match_target=true
   ```

   If you want to manually select which patches should be installed, replace the **-x patch_match_target=true** argument with a space-separated list of patches you wish to install.

   ```
   # swinstall -s /patchCD/GOLDQPK11i \
               -x autoreboot=true \
               PHCO_1000 PHCO_2000
   ```

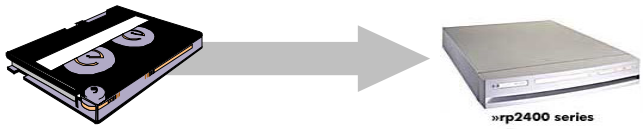

© 2004 Hewlett-Packard Development Company, L.P.

21–14. SLIDE: Installing Patches from Tape

Installing Patches from Tape

1. Do a full backup.
2. View the patch readme files from the depot:
```
# swlist -a readme -s /dev/rmt/0m
```
3. Install the patches
```
# swinstall  -s /dev/rmt/0m \
             -x autoreboot=true \
             -x patch_match_target=true
```

Depot Format Patch Tape



Student Notes

If you need to download a large number of patches from the ITRC, but don't have a high-bandwidth connection to the Internet, support customers can request a custom patch tape. Once you receive the tape, you can install the patches using **`swinstall`**, as shown on the slide.

1. Do a full backup.
2. View the patch **`readme`** files from the depot.

```
# swlist -a readme -s /dev/rmt/0m
```

3. Use the **swinstall** command shown on the slide to install the matching patches from the tape!

```
# swinstall -s /dev/rmt/0m \
            -x autoreboot=true \
            -x patch_match_target=true
```


© 2004 Hewlett-Packard Development Company, L.P.

If you want to manually select which patches should be installed, replace the **`-x patch_match_target=true`** argument with a space-separated list of patches you wish to install.

```
# swinstall -s /dev/rmt/0m  \
             -x autoreboot=true \
             PHCO_1000 PHCO_2000
```

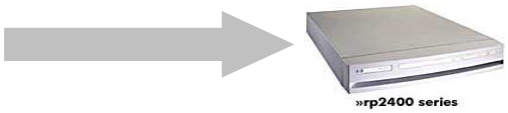

© 2004 Hewlett-Packard Development Company, L.P.

21–15. SLIDE: Installing Patches from a Depot Server

Installing Patches from a Depot Server

1. Do a full backup.
2. View the patch readme files from the depot:
   `# swlist -a readme -s svrname:/depotpath`
3. Install the patches
   ```
   # swinstall  -s svrname:/depotpath \
                -x autoreboot=true \
                -x patch_match_target=true
   ```

SD-UX Depot Server



Student Notes

In order to simplify patch management, many shops choose to configure an SD-UX depot server. Administrators then simply install patches from the depot server rather than tapes, CDROMs, and **`.depot`** files.

Installing patches from a depot server is a simple three step process.

1. Do a full backup.
2. View the patch **`readme`** files from the depot.

   ```
   # swlist -a readme -s svrname:/depotpath
   ```

3. Use the **`swinstall`** command shown on the slide to install the matching patches from the depot! If you prefer to manually select the patches to install via the **`swinstall`** GUI, run **`swinstall`** without the **`-x`** options.

   ```
   # swinstall -s svrname:/depotpath \
               -x autoreboot=true \
               -x patch_match_target=true
   ```


© 2004 Hewlett-Packard Development Company, L.P.

If you want to manually select which patches should be installed, replace the **-x patch_match_target=true** argument with a space-separated list of patches you wish to install.

```
# swinstall -s svrname:/depotpath \
            -x autoreboot=true \
            PHCO_1000 PHCO_2000
```

Installing Patches and Software Concurrently

If you install software products from a depot that also contains patches, SD-UX will automatically select and install the patches, too. If *svrname:/depotpath* contains both FooProd and patches associated with FooProd, the following command will install the product and its patches in a single **swinstall** operation:

```
# swinstall -s svrname:/depotpath \
            -x autoreboot=true \
            FooProd
```

If you prefer not to install patches at the same time that you install the product, add the **-x autoselect_patches=false** option.

```
# swinstall -s svrname:/depotpath \
            -x autoreboot=true \
            -x autoselect_patches=false \
            FooProd
```


© 2004 Hewlett-Packard Development Company, L.P.

21–16. SLIDE: Listing Patches with `swlist`

Listing Patches with `swlist`

```
# swlist -l patch LVM   (11.x)
# LVM                 B.11.00    LVM
# LVM.LVM-KRN                    LVM.LVM-KRN
  PHKL_15956.LVM-KRN,l=/,r=1.0   LVM.LVM-KRN   applied
  PHKL_16433.LVM-KRN,l=/,r=1.0   LVM.LVM-KRN   applied
  PHKL_20186.LVM-KRN,l=/,r=1.0   LVM.LVM-KRN   applied
```

```
# swlist -l product 'PH*'  (10.x or 11.x)
PHCO_19479      1.0    LVM commands cumulative patch
PHKL_15956      1.0    LVM Cumulative PVLinks patch
PHKL_16433      1.0    LVM commands hang
PHKL_19287      1.0    LVM failover to alternate path
PHKL_20186      1.0    LVM Cumulative patch
```

Student Notes

You can determine which patches are already installed on your system via the **`swlist`** command. At version 11.00 and beyond, **`swlist`** includes a special option specifically designed for listing patches: (the patch version numbers have been truncated from this listing to improve readability)

```
# swlist -l patch
# Contacting target "na168w1"...
#
# Target:  localhost:/
#
# LVM                    B.11.00         LVM
# LVM.LVM-KRN
  PHKL_15956.LVM-KRN     LVM.LVM-KRN     applied
  PHKL_16433.LVM-KRN     LVM.LVM-KRN     applied
  PHKL_20186.LVM-KRN     LVM.LVM-KRN     applied
```


© 2004 Hewlett-Packard Development Company, L.P.

`swlist` tells you which products and filesets have patches applied (LVM and LVM.LVM-KRN in this case), and the "state" of each patch. The patch state can be

Applied — The patch is currently installed.

Committed — The patch has been "committed" (patch committal will be described in detail on the next slide).

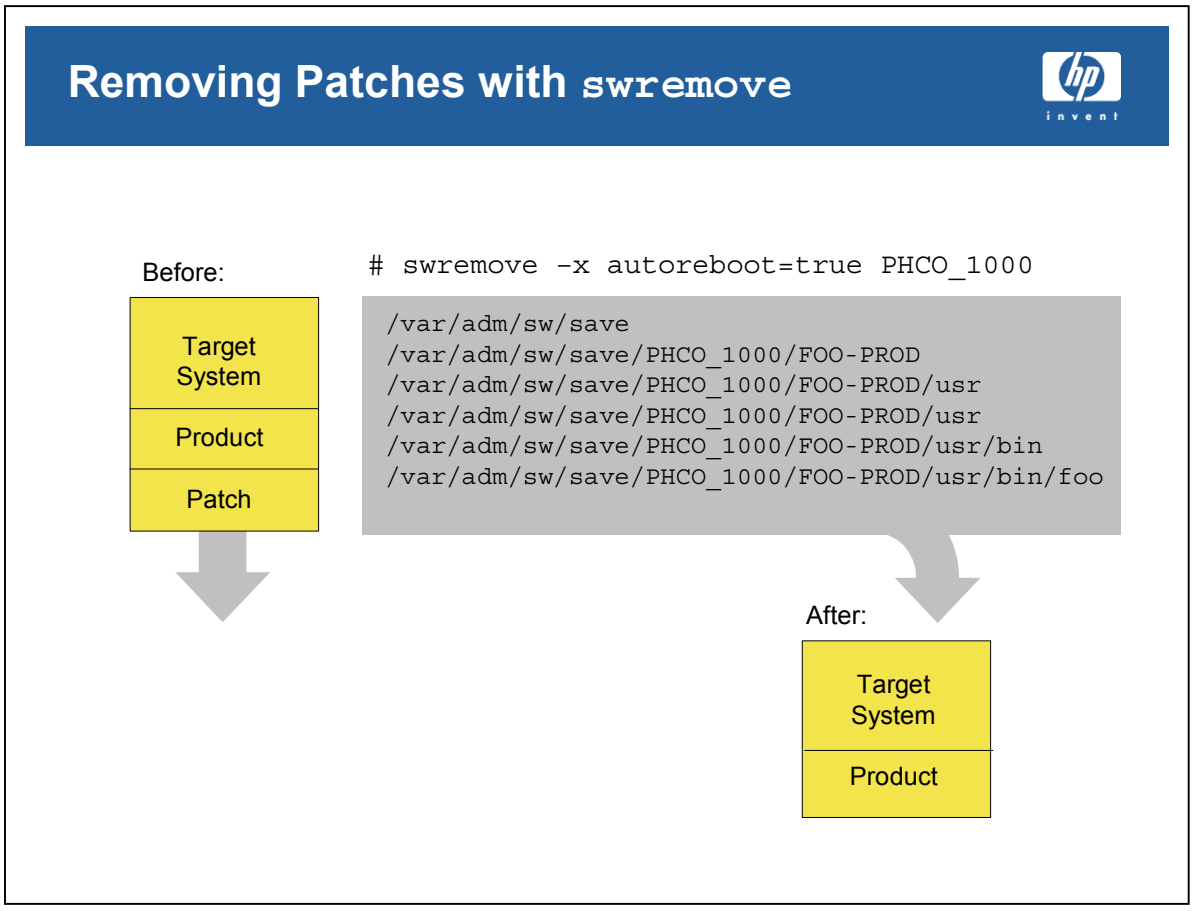
HP-UX Release 10.x

In HP-UX Release 10.x, **`swlist`** didn't support a **`-l patch`** option. In order to list patches installed on a 10.x system, simply list the products that begin with PH:

```
# swlist -l product 'PH*'
# Contacting target localhost...
#
# Target:  localhost:/
#
  PHCO_19479    1.0     LVM commands cumulative patch
  PHKL_15956    1.0     LVM cumulative PVLinks patch
  PHKL_16433    1.0     LVM commands hang
  PHKL_19287    1.0     LVM failover to alternate path
  PHKL_20186    1.0     LVM cumulative patch
```


© 2004 Hewlett-Packard Development Company, L.P.

21–17. SLIDE: Removing Patches with `swremove`



Student Notes

Occasionally, a new patch may conflict with existing patches or products on the system. For this reason, SD-UX makes it possible to remove a patch.

When **`swinstall`** installs a patch, it automatically copies the original pre-patched files to a special directory under **`/var/adm/sw/save/`**.

```
# find /var/adm/sw/save
/var/adm/sw/save
/var/adm/sw/save/PHCO_1000/FOO-PROD
/var/adm/sw/save/PHCO_1000/FOO-PROD/usr
/var/adm/sw/save/PHCO_1000/FOO-PROD/usr
/var/adm/sw/save/PHCO_1000/FOO-PROD/usr/bin
/var/adm/sw/save/PHCO_1000/FOO-PROD/usr/bin/foo
```

These backup files make it possible to restore files to their pre-patched state when a patch is removed. The command below removes patch PHKL_1000 and restores the original files as they existed before PHKL_1000 was first installed:

```
# swremove -x autoreboot=true PHCO_1000
```


© 2004 Hewlett-Packard Development Company, L.P.

Committing Patches via `swmodify`

If you install many patches on your system, the **`/var/adm/sw/save/`** directory may grow quite large and eventually fill the **`/var`** file system.

If you are certain that a patch is stable and will never be removed, the patch can be "committed" to reclaim disk space from **`/var/adm/sw/save/.`**

```
# swmodify -x patch_commit=true 'PHCO_1000.*'
```

Beware, however, that once a patch has been committed, it can't be removed unless you remove the product associated with the patch. HP doesn't recommend committing patches.

Committing Patches via `swinstall`

If you know at patch install time that you intend to commit a patch immediately, you can commit the patch as part of the **`swinstall`** process by including the **`-x patch_save_files=false`** option.

```
# swinstall -s /dev/rmt/0m \
            -x autoreboot=true \
            -x patch_match_target=true \
            -x patch_save_files=false
```

Rules Concerning Patch Rollback and Removal

Patch removal and rollback is performed in accordance with the following rules:

- Removal of the base product fileset of a patch fileset using **`swremove`** removes all patches that have been applied to that fileset , too.
- Files saved for rollback are also removed when their corresponding base product fileset is removed.
- An installed patch that has been superseded may not be rolled back unless the superseding patch is also rolled back.
- Committed patches cannot be removed unless the associated base product is also removed.

© 2004 Hewlett-Packard Development Company, L.P.

21–18. LAB: Installing Patches with SD-UX

Directions

Part 1: Preliminary Steps

1. Run the **`swinstall`** utility to install the EchoApp utility on your machine. EchoApp should be available in a local depot on your machine called **`/labs/depots/echoapp.depot`**.

   ```
   # swinstall -s /labs/depots/echoapp.depot \
               -x autoreboot=true \
               EchoApp
   ```

2. List the products installed on your host and verify that EchoApp is installed? Try running **`echoapp`** to see what it does.

   ```
   # swlist EchoApp
   # /opt/echoapp/bin/echoapp
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Installing and Removing Patches

1. Oftentimes administrators download and install patches from HP's SupportLine web site. For this lab exercise, that won't be necessary. The patch you will install, PHSS_01111, is already in your **/labs** directory. It is embedded in a patch package called **/labs/patches.tgz**. To start, copy the patch package to the **/tmp** directory.

   ```
   # cp /labs/patches.tgz /tmp
   ```

2. **cd** to the **/tmp** directory, unzip, and untar the patch package. What files are created as a result?

   ```
   # cd /tmp
   # gzip -d /tmp/patches.tgz
   # tar -xvf /tmp/patches.tar
   ```

3. Unpack the patch's shar archive. Read the **.text** file. What problem does this patch fix? Does the patch require a reboot?

   ```
   # cd /tmp
   # sh PHSS_01111
   # more PHSS_01111.text
   ```

4. Follow the directions in the patch's **.text** file to install the patch. Since we may need to remove this patch eventually, do NOT to use the **patch_save_files=false** option. Try running **echoapp** again. Did the patch seem to work?

   ```
   # swinstall -x autoreboot=true \
               -x patch_match_target=true \
               -s /tmp/PHSS_01111.depot

   # /opt/echoapp/bin/echoapp
   ```


© 2004 Hewlett-Packard Development Company, L.P.

5. **swinstall** keeps copies of all files that have been replaced by patches in the **/var/adm/sw/save** directory. Use the **find** command to take a look at the PHSS_01111 files under this directory. Which two files were replaced by the PHSS_01111 patch?

```
# find /var/adm/sw/save/PHSS_01111
```

6. Now run **swremove** to remove the patch. Again look at the files under **/var/adm/sw/save**. What changed?

```
# swremove -x autoreboot=true PHSS_01111
# find /var/adm/sw/save/PHSS_01111
```

7. Now remove the EchoApp product as well.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Installing and Removing Patches with Applications

1. Let's try installing EchoApp again, but this time install the product from **`/labs/depots/echoapp+patch.depot`**. Note that this depot contains both the EchoApp product and the PHSS_01111 patch.

   ```
   # swinstall -s /labs/depots/echoapp+patch.depot \
               -x autoreboot=true \
               EchoApp
   ```

2. Run **`echoapp`**. Based on this experiment, what is the advantage of having products and their associated patches in the same depot? (Note: This feature is new at 11.x.)

   ```
   # /opt/echoapp/bin/echoapp
   ```

3. Now remove the EchoApp product. Afterwards, use **`swlist`** to see if PHSS_01111 is still installed on your system. When you remove a product, what happens to the patches associated with that product?

   ```
   # swremove -x autoreboot=true EchoApp
   ```

4. Confirm your conclusion in the previous question by running **`swlist`**.

   ```
   # swlist EchoApp PHSS_01111
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Committing Patches

1. Install EchoApp and the PHSS_01111 patch again from the **echoapp+patch.depot**.

```
# swinstall -s /labs/depots/echoapp+patch.depot \
            -x autoreboot=true \
            EchoApp
```

2. As you install patches on your system, the **/var/adm/sw/save** directory can consume a significant amount of disk space. You can save some space in **/var/adm/sw/save** by committing a patch. Try committing the PHSS_01111 patch. What effect does the **swmodify** command have on the **/var/adm/sw/save** directory? Use **find** and **du** to find out.

```
# find /var/adm/sw/save/PHSS_01111
# du -sk /var/adm/sw/save
# swmodify -x patch_commit=true 'PHSS_01111.*'
# find /var/adm/sw/save/PHSS_01111
# du -sk /var/adm/sw/save
```

3. Try to remove the patch. What appears to be the downside of committing a patch?

```
# swremove -x autoreboot=true PHSS_01111
```

4. Can you still remove the product associated with the patch? Try it. Then **swlist** both the EchoApp product and the patch.

```
# swremove -x autoreboot=true EchoApp
# swlist EchoApp PHSS_01111
```


© 2004 Hewlett-Packard Development Company, L.P.

Module 22 — Installing the OS with Ignite-UX

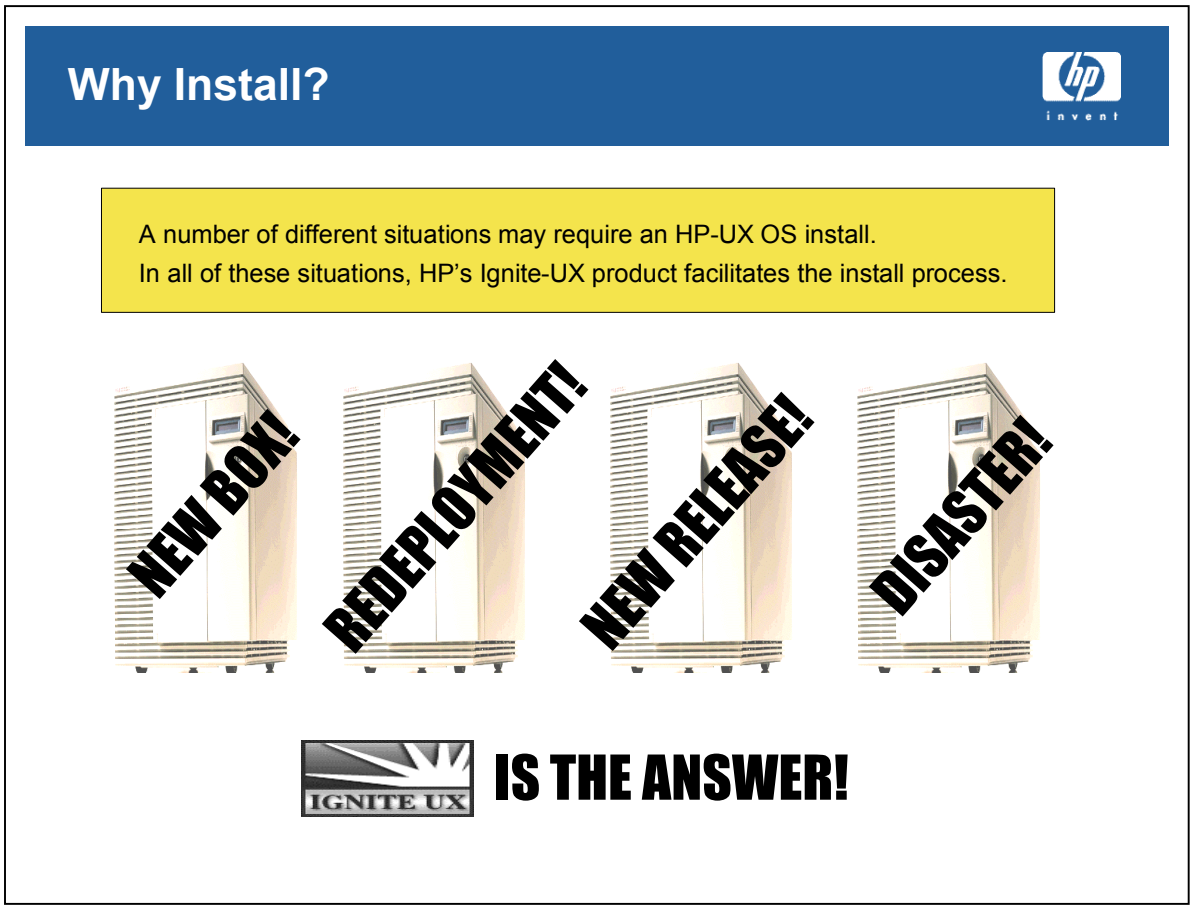
Objectives

Upon completion of this module, the student will be able to do the following:

- List four situations in which an install may be required.
- Describe the five HP-UX 11i Operating Environments.
- Summarize the functionality provided by Ignite-UX.
- Describe the differences between "Pull," "Push," and local media installs.
- Install HP-UX from CD or from an already configured Ignite-UX server.

© 2004 Hewlett-Packard Development Company, L.P.

22–1. SLIDE: Why Install?



Student Notes

A number of different situations may require an HP-UX OS install. In all of these situations, HP's Ignite-UX product facilitates the install process.

Installing a New Box

When you receive a new HP box fresh from the factory, you may need to "cold install" HP-UX on the new machine. Cold installs may be pulled from either the HP-UX "Core" CDs or from an Ignite-UX network install server elsewhere on the network.

Many system administrators need to install multiple systems with the same LVM configuration, file systems, software, and patch set. Ignite-UX makes it possible to construct a prototype of the desired image, then propagate this "Golden Image" to multiple machines. The "Golden Image" can be stored on an Ignite-UX install tape or on an Ignite-UX network install server.

Re-deploying an Already Installed System

Occasionally, you may wish to re-deploy a machine for use by another department or perhaps for use by another application. In this situation, you may wish to re-install HP-UX.

© 2004 Hewlett-Packard Development Company, L.P.

Re-deploying the box with a fresh image of the operating system gives you an opportunity to change the layout of the root disk, change file system types, and eliminate unneeded software.

Ignite-UX is the ideal tool for this purpose. You may either re-install HP-UX from CDs, or, if the machine already has a functional version of HP-UX up and running, you may "push" an install of the OS to the host from an Ignite-UX server elsewhere on the network.

Moving to a Major, New Release of HP-UX

When a new version of HP-UX is released, some administrators choose to update their machines using the **`update-ux`** command, but others choose to do a cold install. A cold install offers an opportunity to change the logical volume and file system configuration.

Recovering from a Disaster

The **`make_tape_recovery`** and **`make_net_recovery`** utilities, which are included with Ignite-UX, can be used to create a bootable system recovery tape. If your system's boot disk becomes seriously corrupted, the recovery tape may be used to restore the LVM configuration, file systems, and data to the root volume group. The procedure for making and using a system recovery tape was already discussed in the backup and recovery chapter.

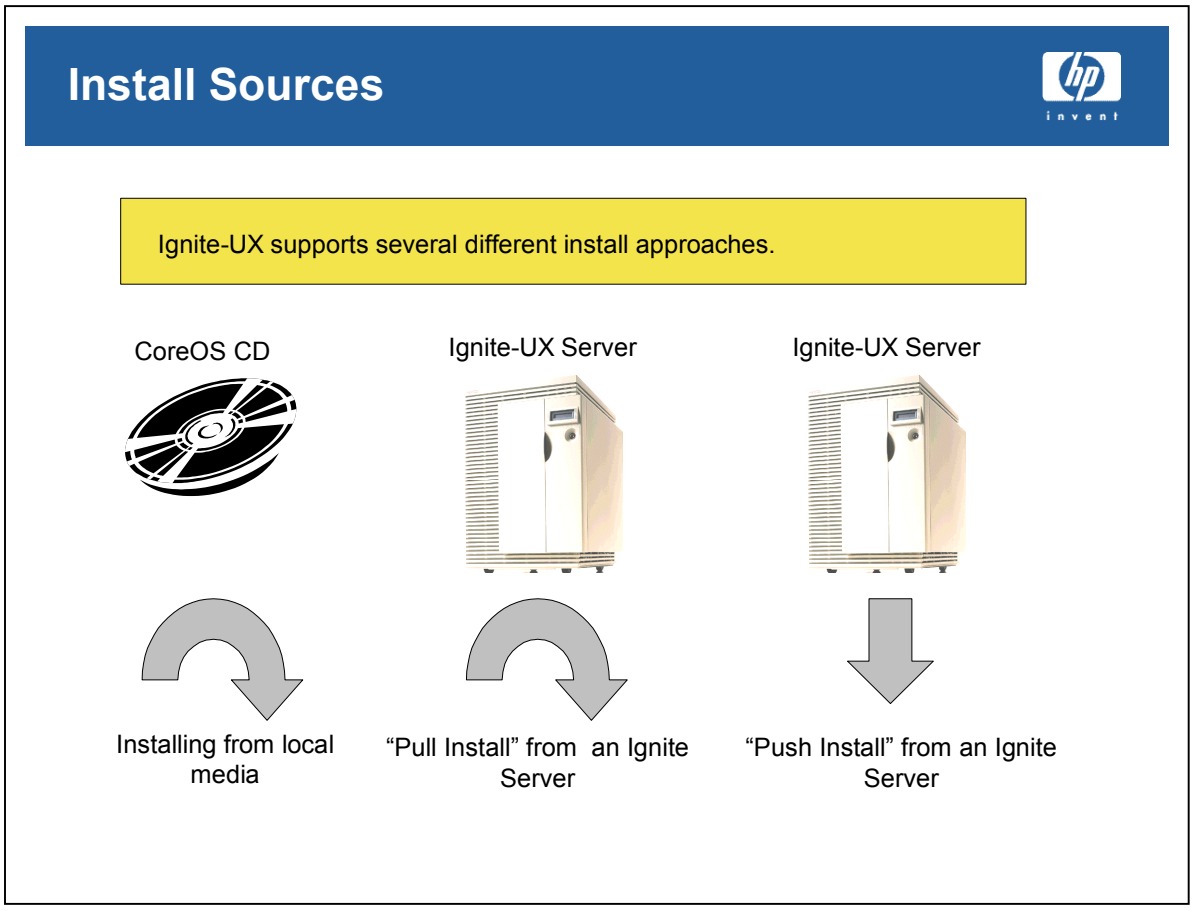
What Happens when you Install the OS?

Installing the HP-UX operating system will:

- Destroy any existing data on the root disk.
- Create a new root volume group.
- Create a new HP-UX file system on the root disk(s).
- Install the core HP-UX filesets.
- Build an initial HP-UX kernel.
- Create a login for the system administrator.

© 2004 Hewlett-Packard Development Company, L.P.

22–2. SLIDE: Install Sources



Student Notes

Using Ignite-UX, you can install HP-UX several different ways.

Installing from Local Media

In its simplest form, Ignite-UX can be used to install HP-UX from a locally connected HP-UX CD-ROM using CDs or an Ignite-UX "Golden Image" tape. The CD or tape will contain everything needed to install the OS. Booting from the CD or tape loads the Ignite-UX interface, which then prompts the administrator through a series of menus that determine how the system will be installed.

Initiating a "Pull" Install from an Ignite Server

You may wish to configure an Ignite-UX network install server. After configuring an Ignite server, clients can install HP-UX without ever loading a CD-ROM; all the needed install utilities and OS software can be "pulled" across the network from the Ignite server.

Initiating a "Push" Install from an Ignite Server

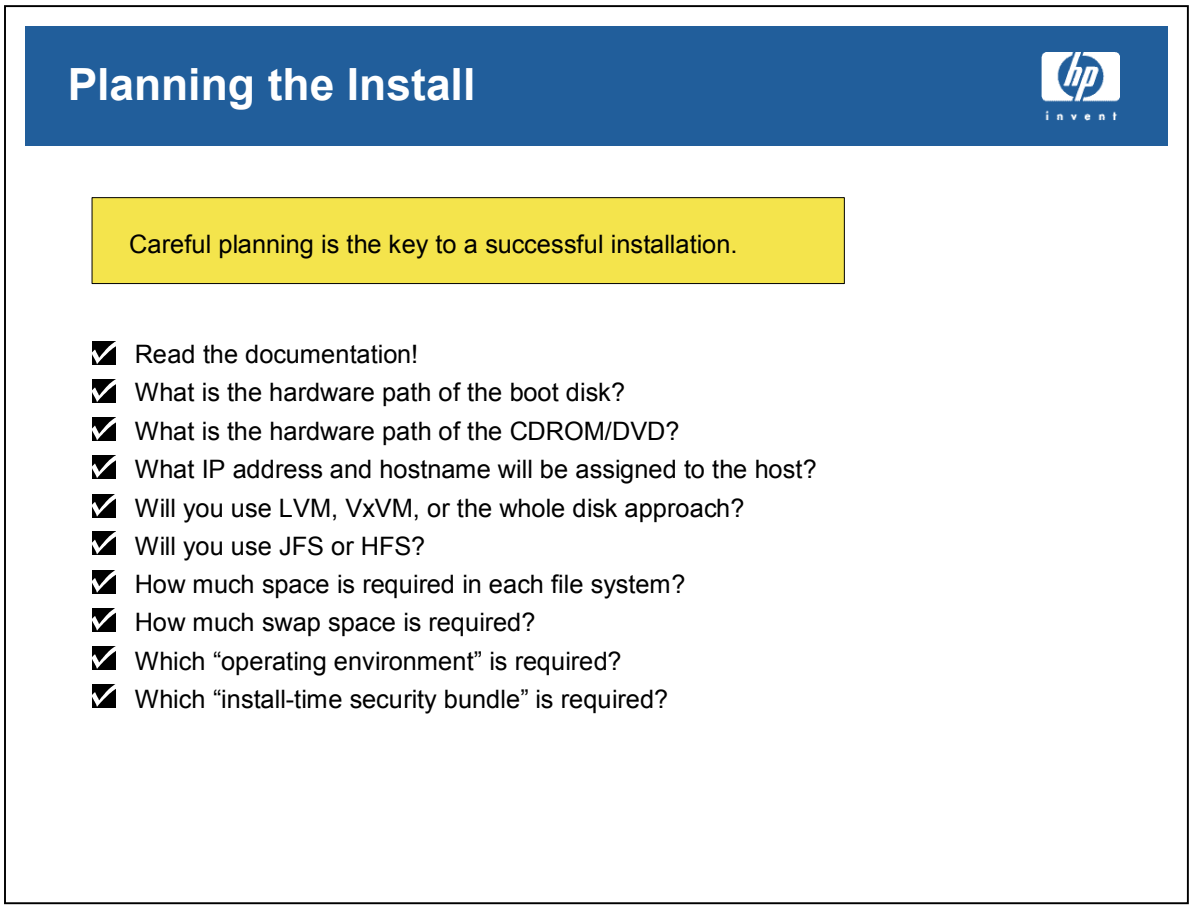
Alternatively, one can even "push" an install from the Ignite-UX server out to one or more clients. All configuration is thus performed via a GUI interface on the Ignite-UX server - no

© 2004 Hewlett-Packard Development Company, L.P.

interaction is required on the target client machine consoles! In order to push an install out to a client, the client machine must already be running some version of HP-UX, and the administrator on the Ignite server must know the client's root password. "Push" installs will not be covered in this course.

© 2004 Hewlett-Packard Development Company, L.P.

22–3. SLIDE: Planning the Install



Student Notes

Before beginning the installation process, you should carefully consider your system configuration. The "Installing HP-UX 11.x" manuals are a good place to begin this process. If you don't have a printed copy of the appropriate manuals, you can view the books in electronic form at **`http://docs.hp.com`**.

Be sure to determine the following:

- The hardware address of the device containing the install media.
- The hardware addresses of the desired root disk.
- Your hostname, IP address, netmask, and default gateway.
- Whether you wish to use LVM, VxVM, or the whole disk layout approach.
- Whether you wish to use HFS or JFS.
- How much space is required in each file system.

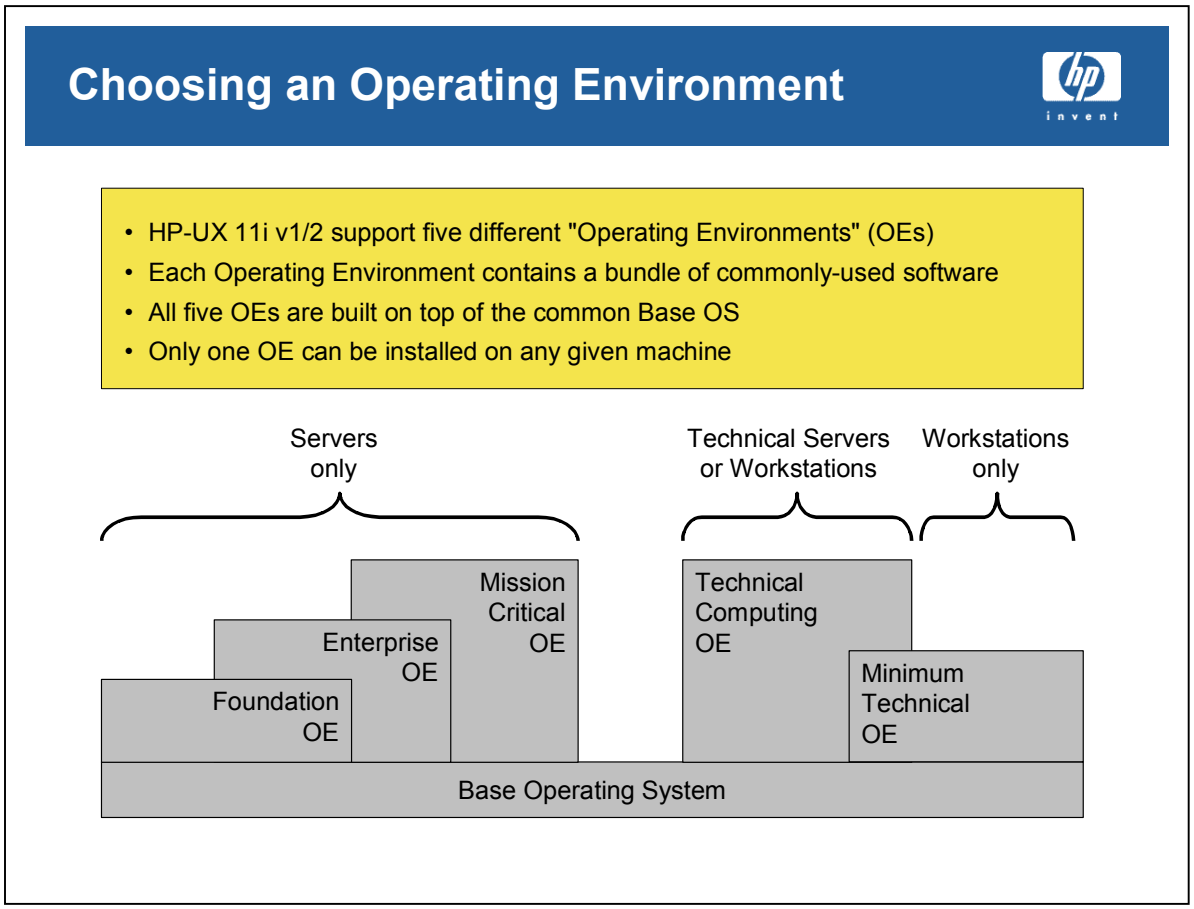
© 2004 Hewlett-Packard Development Company, L.P.

- How much swap space is required?
- Which applications are required?

During the install process you will also be asked to choose an “operating environment” and install-time security bundle. Operating environments and install-time security bundles are described on the next two slides.

© 2004 Hewlett-Packard Development Company, L.P.

22–4. SLIDE: Choosing an Operating Environment



Student Notes

HP supports five different HP-UX 11i "Operating Environments" (OEs). An Operating Environment is a bundle of software products that are licensed, installed, and managed together as a unit.

All of the 11i OEs start with a common Base OS, which includes the core kernel functionality, I/O subsystem, memory management subsystem, LVM functionality, and other critical components of HP-UX. Each OE bundles a slightly different group of applications with this base functionality. Only one Operating Environment can be installed at any given time, though you can purchase additional software components to supplement your OE at any time.

- The 11i Minimum Technical Operating Environment (MTOE) includes the base OS and some core graphics libraries. This OE is only supported on workstations.

- The 11i Technical Computing Operating Environment (TCOE) includes the full contents of the 11i Technical Computing Operating Environment as well as some 3D graphics libraries, the MLIB math library, and other software components that would be most beneficial for systems running mathematical and engineering applications. This OE is supported on technical servers and workstations.

© 2004 Hewlett-Packard Development Company, L.P.

- The 11i Foundation Operating Environment includes the Apache Webserver, CIFS file server, and other applications that would be required on a web, application, or file server. This is the standard offering for servers. The 11i Foundation OE isn't supported on workstations.

- The 11i Enterprise Operating Environment is a superset of the 11i Foundation OE. In addition to the applications included in the 11i OE, the Enterprise OE includes MirrorDisk/UX and Online JFS for maximizing file system uptime, GlancePlus and PRM performance tools, and other applications that are required to run an enterprise level commercial server. The 11i Enterprise OE is only supported on servers.

- The 11i Mission Critical Operating Environment is a superset of the 11i Enterprise OE. In addition to Online JFS, MirrorDisk/UX, GlancePlus, and PRM, the Mission Critical OE also includes MC/Service Guard and other tools required to run a true 24x7 commercial server. The 11i Mission Critical OE is only supported on servers.

The Operating Environment you purchase determines what software is available on the DVDs that accompany your system. Some components of each OE are required, but others are customer-selectable at install time.

You can also purchase additional software components to supplement your Operating Environment. For instance, a workstation administrator may choose to purchase Online JFS for a workstation installed with the Technical Computing Operating Environment.

For a complete list or software included in each OE bundle, see the "HP-UX 11i Operating Environments" PDF document on **`http://docs.hp.com`**..

Operating Environment Advantages

HP's 11i OE bundling strategy greatly simplifies HP-UX software management:

- When ordering new systems, it is no longer necessary to list part numbers for common individual applications; simply select the operating environment containing the applications you require.

- Purchasing a common combination of applications as part of an OE is cheaper than purchasing those same products a la carte.

- When installing software, it is no longer necessary to manage separate codewords for each application you wish to install. Each OE license contains a license for the OS as well as the applications that are included in the OE.

Viewing and Managing Operating Environments

An operating environment can be selected at install time, but you can upgrade or downgrade your Operating Environments post-install, too. The next few paragraphs introduce a few commands you can use to view and manage your operating environments.

You can use the **`swlist`** command to determine which base and OE bundles are currently installed on your system:

```
# swlist -l bundle HPUXBase* HPUX11i-OE*
```


© 2004 Hewlett-Packard Development Company, L.P.

You can also use **swlist** to determine what software was installed as part of your operating environment:

```
# swlist –l product HPUX11i-OE*
```

If you wish to upgrade to a higher-level OE, you will need to determine which OEs are available on your core depot:

```
# swlist -l bundle -s /cdrom HPUX11i-OE*
```

To upgrade 11i to a higher-level OE, say, from 11i OE to the Mission Critical OE, you must first purchase the Mission Critical OE media kit from HP, then use **update-ux** to add the Mission Critical OE software bundle to your server from those CDs. It may be necessary to add additional disk space to one or more file systems in order to complete the update. Refer to the HP-UX 11i install/update manual on **http://docs.hp.com** for specific disk space requirements. When the update is complete, the Mission Critical OE will be installed and the original HP-UX 11i OE bundle wrapper will be removed.

```
# /usr/sbin/update-ux -s /cdrom HPUX11i-OE-MC
```

If you wish to downgrade your system from a higher-level OE, say, from the Mission Critical OE to the 11i Foundation OE, first use **update-ux** to install the smaller OE from the latest set of HP-UX 11i CDs, then use **swremove** to remove the unwanted OE bundle and its software. The **swremove** command removes the IPD entries and software for the additional software in the Mission Critical OE, leaving the default HP-UX 11i OE software intact.

```
# /usr/sbin/update-ux -s /cdrom -y HPUX11i-OE
# /usr/sbin/swremove HPUX11i-OE-MC
```

Adding and Removing Individual OE Products

HP-UX 11i OEs have been packaged and tested as complete solutions. A mechanism for handling subsets of an OE, such as individual application products, is not available with 11i OEs. If you wish to delete a product from an installed OE, or install single products rather than a complete OE, you must deal with individual filesets. For more information on installing and removing individual OE products, see the "Adding and Removing HP-UX 11i Operating Environment Products" white paper on **http://docs.hp.com**.

© 2004 Hewlett-Packard Development Company, L.P.

22–5. SLIDE: Choosing an Install-Time Security Bundle

Choosing an Install-Time Security Bundle

During the HP-UX 11iv2 install process, it is now possible to select an install-time security bundle, which uses HP Bastille to automatically lock down a variety of vulnerable network services

Bundle Name	Config File	Effect on System Behavior
Sec00Tools	N/A	Install security infrastructure without applying security
Sec10Host	HOST.config	Host-based lockdown: no firewall; telnet and FTP remain enabled, but less common network services are disabled in `/etc/inetd.conf`.
Sec20MngDMZ	MANDMZ.config	Lockdown: IPFilter firewall blocks all incoming connections except common, secured, management protocols
Sec30DMZ	DMZ.config	Full lockdown: IPFilter blocks all incoming connections except HP-UX Secure Shell

Student Notes

During the 11i v2 install/update process, you will have an opportunity to select an "install-time security bundles" to install on your system.

Bastille

The install-time security bundles utilize the HP-UX Bastille host hardening tool to automatically lockdown vulnerable network services during the install process. Bastille was originally developed by the open source community for use on Linux systems, but is now supported by HP for use on HP-UX systems, too.

When executed in interactive mode by typing **`/opt/sec_mgmt/bastille/bin/bastille`**, Bastille interactively presents the administrator with a list of security measures commonly used to "harden" UNIX systems. The utility asks which measures should be applied on the administrator's system, then records the administrator's responses in an ASCII file, typically called **`/etc/opt/sec_mgmt/bastille/config`**. Running **`bastille -b`** then automatically applies the requested changes.

© 2004 Hewlett-Packard Development Company, L.P.

Install-time Security Configuration Bundles

When you choose to install one of the install-time security bundles at install time, Ignite-UX automatically installs the Bastille product, and optionally runs **`bastille -b`** using one of three pre-configured Bastille configuration files. Using one of the install-time security configuration bundles applies a default security profile. This dramatically improves system security, and is highly recommended, particularly for systems that are exposed to the public Internet. The bundle you select determines the level of security that Bastille enforces; higher-level bundles imply higher-levels of security. Each bundle is described below.

Sec00Tools
The Sec00Tools security infrastructure bundle is default-installed on your system. While this bundle does not implement any security changes at install- or update-time, it does ensure that the Bastille software is installed so you can run the tool interactively after the install/update is complete.

Sec10Host
Implements host-based lockdown. **`telnetd`** and **`ftpd`** remain enabled, but less commonly used network services such as **`fingerd`** and **`tftpd`** are disabled. NFS server and client daemons are disabled, too. See **`/etc/opt/sec_mgmt/bastille/HOST.config`** for a complete list of security enhancements implemented by this security bundle.

Sec20MngDMZ
Implements all of the security policies included in the Sec10Host security bundle, but also configures an IPFilter firewall to block all incoming connections except common, secured, management protocols (HP-UX Secure Shell, HIDS agent, WBEM, web admin and web admin autostart). See **`/etc/opt/sec_mgmt/bastille/MANDMZ.config`** for a complete list of security enhancements implemented by this security bundle.

Sec30DMZ
Configures IPFilter to block all incoming connections except HP-UX Secure Shell. See **`/etc/opt/sec_mgmt/bastille/DMZ.config`** for a complete list of security enhancements implemented by this security bundle.

Selecting Multiple Security Configuration Bundles

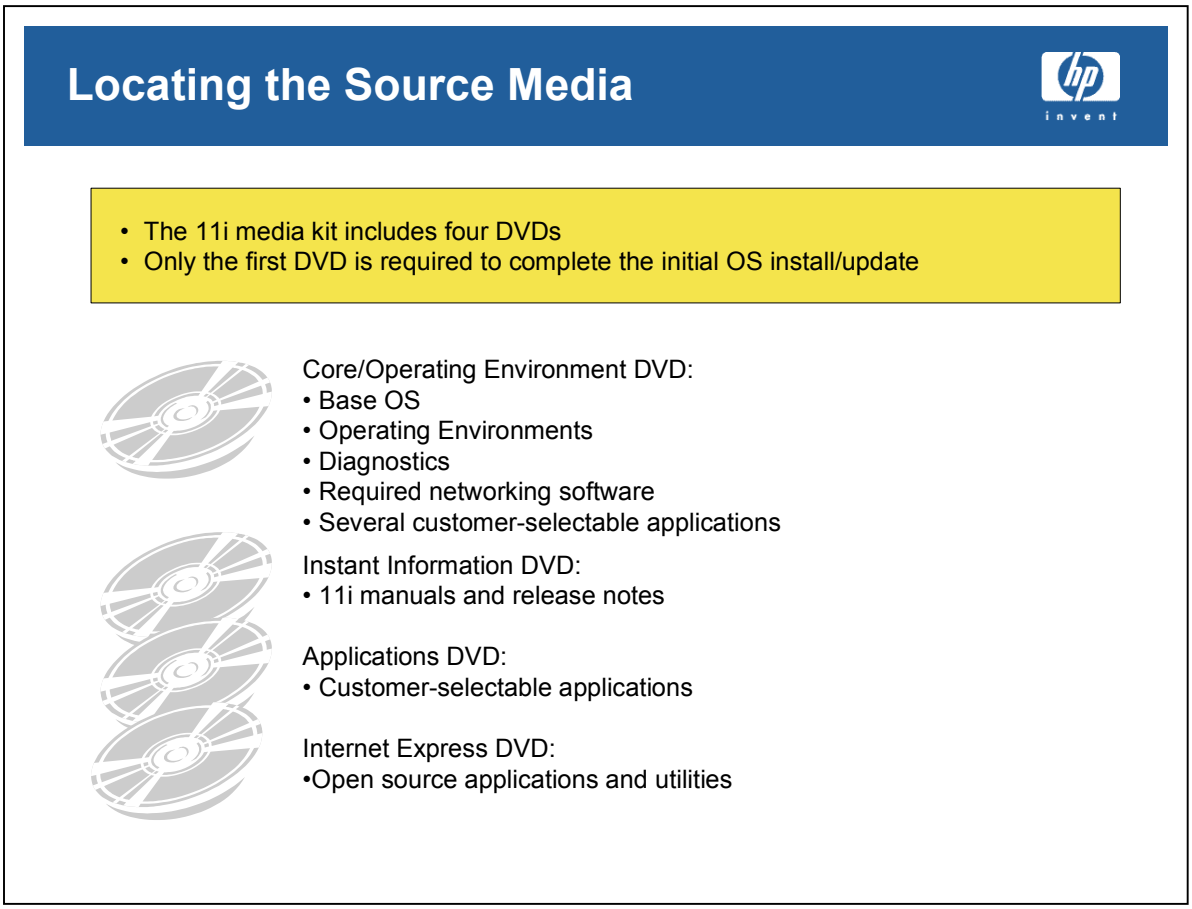
If multiple Security Configuration bundles are selected, the highest-level security selected determines the Bastille security level of the system.

Security Configuration Bundles

Security level bundles determine the level of security that is installed on the system. Installing a higher-level bundle ratchets system security up, and once it's been raised, it cannot easily be ratcheted down. It is not possible to uninstall a Security Level bundle and install a lower-level one, but it is possible to install a higher-level bundle. To revert to a lower-level of security, or remove Bastille security altogether, use the **`bastille -r`** command, then install the lower-level Security Configuration bundle via the **`swinstall`** command.

© 2004 Hewlett-Packard Development Company, L.P.

22–6. SLIDE: Locating the Source Media



Student Notes

The HP-UX 11i media kit includes several DVDs that you will need during the installation process. Only the first DVD is required to complete the initial OS install/update.

Core/Operating Environment DVD

The Core/Operating Environment DVD includes the Base OS software, Operating Environment software, Ignite-UX, diagnostics, and networking software to support the most common network interface cards.

Instant Information DVD

The Instant Information DVD contains a complete documentation set for HP-UX 11i v2.

Applications DVD

The Applications DVD includes additional software that can be installed to supplement the operating environment software. Many products on the Applications DVD are codeword protected. Contact your HP sales representative if you wish to purchase a license to install these products.

© 2004 Hewlett-Packard Development Company, L.P.

Internet Express DVD

The Internet Express DVD contains a number of pre-compiled open source products that are often used on HP-UX systems. Note that many of these products are unsupported. As this workbook goes to print, the following products are included on the Internet Express CD:

Ant	Jetty	Qpopper
Calamaris	Libpcap	Snort
Curl	Majordomo	SOAP
CyrusIMAP	MySQL	Squid
CyrusSASL	Net-SNMP	Stunnel
DanteSOCKS	OpenJMS	Sudo
Ethereal	OpenLDAP	Tcpdump
Fetchmail	OpenSSL	UDDI4J
Horde	Perl-LDAP	UW-IMAP
Hsqldb	Pine	Xalan-C
Hypermail	PostgreSQL	Xerces-C
IMP	ProcMail	Xinetd
Jabber	ProFTPD	

Support+ DVD

HP-UX 11i v1 media kits include a Support+ CDROM containing bundles of HP-recommened patches.

As this workbook went to press, Quality Pack patch bundles haven't yet been released for 11i v2. Future media kits may contain Quality Pack patch bundles similar to the 11iv1 Quality Pack patch bundles.

© 2004 Hewlett-Packard Development Company, L.P.

22–7. SLIDE: Initiating a PARISC Install

Initiating a PARISC Install

To initiate a PARISC system install,
simply use the BCH `boot` command to boot from DVD or an Ignite install server.

- If installing from DVD, CDROM, or tape, load the appropriate media
- Access the console interface
- Initiate the boot sequence with the `MP:CM> rs` command
- Press Escape to interrupt the autoboot sequence and access the BCH main menu
- From the BCH main menu, use the `search` and `boot` commands to boot from DVD

```
Main Menu: search ipl
Main Menu: boot p1
```

- Or, boot from the Ignite-UX install server

```
Main Menu: boot lan.10.1.1.1 install
```

- Don't interact with the ISL/IPL
- Wait for the Ignite-UX menu interface to appear

Student Notes

Once you have planned your new system configuration, you are ready to begin the installation process. Several steps are required to initiate the install process on a PARISC system:

1. If installing from DVD, CDROM, or tape, load the appropriate media. If you are installing from DVD, there should be just one core/OE DVD. If installing from CDROM, load the first core/OE CDROM. You will be prompted to swap CDROMs partway through the install. If you are re-installing from a recovery tape, make sure the tape is write-protected and load it in the tape drive.

2. Login via the MP console interface. Console access is required to install the OS.

3. Initiate the boot sequence with the **`MP:CM> rs`** command.

4. Press Escape to interrupt the autoboot sequence and access the BCH main menu.

5. From the BCH main menu, use the **`search`** command to identify the DVD/CDROM/tape device path, and boot from that device via the BCH **`boot`** command.

© 2004 Hewlett-Packard Development Company, L.P.

```
Main Menu: search ipl
Main Menu: boot p1
```

Or, boot from the Ignite-UX install server. Replace 10.1.1.1 with the Ignite server's IP address in the command below, or leave off the IP address if you want to boot from the first available server.

```
Main Menu: boot lan.10.1.1.1 install
```

6. When asked if you wish to interact with the ISL/IPL, answer **no**.

7. Wait for the Ignite-UX menu interface to appear. You may have to wait several minutes.

© 2004 Hewlett-Packard Development Company, L.P.

22–8. SLIDE: Initiating an Integrity Install

Initiating an Integrity Install

To initiate an Integrity system install,
simply use the EFI shell to boot from DVD or an Ignite install server.

- If installing from DVD or tape, load the appropriate media
- Access the console interface
- Initiate the boot sequence with the `MP:CM> rs` command
- Press Escape to interrupt the EFI autoboot sequence and access the EFI shell
- From the EFI shell, identify the DVD device and load the `install.efi` program

```
SHELL> map -r -fs -b
SHELL> fs0:\install.efi
```

- Or, run the EFI `lanboot` command to install from an Ignite server

```
SHELL> lanboot select
01  Acpi(HWP0002,0)/Pci(3|0)/Mac(00306E1E7EE0)
02  Acpi(HWP0002,100)/Pci(2|0)/Mac(00306E1E9EA9)
Select Desired LAN: 1
```

- Wait for the Ignite-UX menu interface to appear

Student Notes

Once you have planned your new system configuration, you are ready to begin the installation process. Several steps are required to initiate the install process on a PARISC system:

1. If installing from DVD, load the Core/OE DVD. If you are re-installing from a recovery tape, make sure the tape is write-protected and load it in the tape drive.

2. Login via the MP console interface. Console access is required to install the OS.

3. Initiate the boot sequence with the **`MP:CM> rs`** command.

4. Press Escape to interrupt the EFI autoboot sequence and access the EFI shell.

```
EFI Boot Manager ver 1.10 [14.60] Firmware ver 1.61 [4241]
Please select a boot option
    HP-UX Primary Boot
    HP-UX Alternate Boot
    EFI Shell [Built-in]
    Boot option maintenance menu
    Security/Password Menu
```


© 2004 Hewlett-Packard Development Company, L.P.

5. From the EFI shell, identify the DVD device and load the **`install.efi`** program

```
SHELL> map –r –fs -b
SHELL> fs0:\install.efi
```

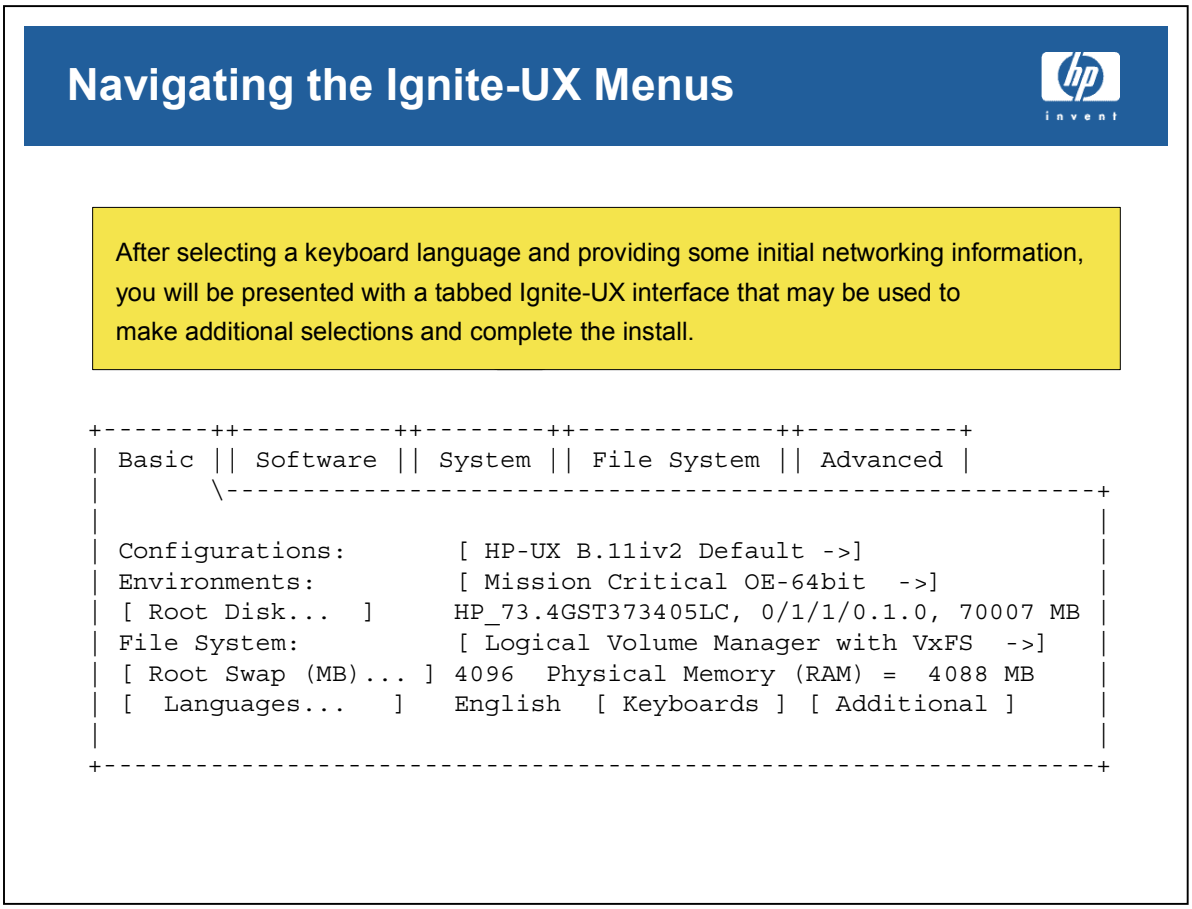
   Or, run the EFI **`lanboot`** command to install from an Ignite server. You will be prompted to select the LAN card you wish to use to initiate the request. Unlike PARISC systems, though, on Integrity systems use a BOOTP broadcast mechanism to identify an appropriate Ignite server. It isn't possible to specify an Ignite server IP address.

```
SHELL> lanboot select
01  Acpi(HWP0002,0)/Pci(3|0)/Mac(00306E1E7EE0)
02  Acpi(HWP0002,100)/Pci(2|0)/Mac(00306E1E9EA9)
Select Desired LAN: 1
```

6. Wait for the Ignite-UX menu interface to appear. You may have to wait several minutes.

© 2004 Hewlett-Packard Development Company, L.P.

22–9. SLIDE: Navigating the Ignite Menus



Student Notes

After the Ignite-UX interface has been loaded, you will be led through a series of menus to determine your desired system configuration.

Choosing a Keyboard Language

After the Ignite-UX interface has been loaded, you will be led through a series of menus to determine your desired system configuration. The first screen asks you to select a keyboard language.

```
=============================================================================
In order to use a keyboard on this interface, you must specify
a language mapping which will be used by X windows and
the Internal Terminal Emulator (ITE).
The characters "1234567890" will appear as "!@#$^&*()"
on keyboards that use the shift key to type a number.
Your choice will be stored in the file /etc/kbdlang

 1) USB_PS2_DIN_Belgian          2) USB_PS2_DIN_Belgian_Euro
 3) USB_PS2_DIN_Danish           4) USB_PS2_DIN_Danish_Euro
```


© 2004 Hewlett-Packard Development Company, L.P.

```
 5) USB_PS2_DIN_Euro_Spanish     6) USB_PS2_DIN_Euro_Spanish_Euro
 7) USB_PS2_DIN_French           8) USB_PS2_DIN_French_Euro
 9) USB_PS2_DIN_German          10) USB_PS2_DIN_German_Euro
11) USB_PS2_DIN_Italian         12) USB_PS2_DIN_Italian_Euro
13) USB_PS2_DIN_JIS_109         14) USB_PS2_DIN_Korean
15) USB_PS2_DIN_Norwegian       16) USB_PS2_DIN_Norwegian_Euro
17) USB_PS2_DIN_S_Chinese       18) USB_PS2_DIN_Swedish
19) USB_PS2_DIN_Swedish_Euro    20) USB_PS2_DIN_Swiss_French2_Euro
21) USB_PS2_DIN_Swiss_German2   22) USB_PS2_DIN_Swiss_German2_Euro
23) USB_PS2_DIN_T_Chinese       24) USB_PS2_DIN_UK_English
25) USB_PS2_DIN_UK_English_Euro 26) USB_PS2_DIN_US_English
27) USB_PS2_DIN_US_English_Euro

Enter the number of the language you want: 26
================================================================
```

Reviewing the Hardware Configuration

After asking you to choose a keyboard language, Ignite displays a summary of the hardware identified on your system:

```
================================================================================
                          Welcome to Ignite-UX!

Use the <tab> key to navigate between fields, and the arrow keys
within fields.  Use the <return/enter> key to select an item.
Use the <return> or <space-bar> to pop-up a choices list.  If the
menus are not clear, select the "Help" item for more information.

Hardware Summary:      System Model: 9000/712/80
+-----------------------------------------------------+ [ Scan Again ]
| Disks: 5  ( 5.8GB) | Floppies: 0 | LAN cards:   2   |
| CDs:   1           | Tapes:    1 | Memory:    256Mb |
| Graphics Ports: 0  | IO Buses: 3 | CPUs:        2   | [ H/W Details ]
+-----------------------------------------------------+

                  [      Install HP-UX       ]

                  [   Run a Recovery Shell   ]

                  [    Advanced Options      ]

         [  Reboot  ]                                [  Help  ]

================================================================================
```

If any disks or other devices appear to be missing, ensure that the cables are firmly connected, check your SCSI addresses, and tab to the **[Scan Again]** button. You can also view more detailed information about your hardware configuration by clicking on the **[H/W Details]** button.

Once you are satisfied that Ignite has successfully identified all of your hardware devices, tab to **[Install HP-UX]** and press Return.

© 2004 Hewlett-Packard Development Company, L.P.

Choosing a User Interface

The next screen lets you specify the install source that you wish to use for the remainder of the install process. You can choose to install solely from local media, from local media in conjunction with network SD-UX depots on other hosts, or solely from an Ignite server on the network.

You also get to choose the Ignite interface that should be used for the remainder of the install. The "Guided Installation" gives you an opportunity to set some basic system parameters via a menu interface on the target host, but doesn't allow you to tweak the LVM and file system configuration.

- The "Advanced Installation" provides a great deal of flexibility: you are given an opportunity to fine tune the LVM and file system configurations, define a variety of system networking parameters, and select software and patches to include in the initial system install.

- If you choose "Remote graphical interface running on the Ignite Server", then the remainder of the configuration must be done on the Ignite server rather than the target host.

- Choose whichever combination of options you prefer, then tab to [OK] and hit return.

```
=============================================================================

                      User Interface and Media Options

   This screen lets you pick from options that will determine if an
   Ignite-UX server is used, and your user interface preference.

Source Location Options:
   [ * ]  Media only installation
   [   ]  Media with Network enabled (allows use of SD depots)
   [   ]  Ignite-UX server based installation

User Interface Options:
   [   ]  Guided Installation   (recommended for basic installs)
   [ * ]  Advanced Installation (recommended for disk and FS management)
   [   ]  Remote graphical interface running on the Ignite-UX server

Hint: If you need to make LVM size changes, or want to set the
      final networking parameters during the install, you will
      need to use the Advanced mode (or remote graphical interface).

   [   OK   ]               [ Cancel ]                     [  Help  ]

=============================================================================
```

Entering Network Information

If you selected Media with Network enabled on the previous screen, Ignite will prompt you for an IP address, hostname, netmask, and default gateway to be used during the install

© 2004 Hewlett-Packard Development Company, L.P.

process. If you have a DHCP server on the network, you may not need to enter this networking information manually.

```
=====================================================================

                        NETWORK CONFIGURATION

                                This system's hostname: ____________

Internet protocol address (eg. 15.2.56.1) of this host: ____________

     Default Gateway routing internet protocol address: ____________

     The subnet mask (e.g. 255.255.248.0 or 0xfffff800: ____________

                 IP address of Ignite-UX server system: ____________

         Is this networking information only temporary?     [ No ]

    [  OK  ]                   [ Cancel ]                   [  Help  ]
=====================================================================
```

Navigating the Tabbed Interface

After setting your network parameters, Ignite allows you to set other system parameters via a tabbed install configuration interface:

```
=====================================================================

/------\/---------\/--------\/-------------\/----------\
|Basic ||Software || System || File System || Advanced |
|      \-------------------------------------------------------------\
|                                                                     |
| Configurations:   [ HP-UX B.11.11 Default  ->] [Description... ]    |
|                                                                     |
| Environments:     [ Mission Critical OE-32bit ->]  (HP-UX B.11.11)  |
|                                                                     |
| [Root Disk ...  ] SEAGATE_ST32272WC 8/4.5.0 2033MB                  |
|                                                                     |
| File System:      [ Logical Volume Manager (LVM) with VxFS   ->]    |
|                                                                     |
| [ Root Swap (MB) ... ] 512     Physical Memory (RAM) = 256MB        |
|                                                                     |
| [ Languages ... ] English_____  [ Keyboards... ]  [Additional... ]  |
|                                                                     |
\---------------------------------------------------------------------/
[ Show Summary... ]          [ Save As... ]     [ Reset Configuration ]
[ Go!  ]                       [ Cancel ]                    [  Help  ]

=====================================================================
```

Use the fields on the **`Basic`** tab to select your basic system configuration. What version of HP-UX do you wish to install? Which Operating Environment do you wish to install? Which disk should serve as the root disk? Do you want to use LVM? JFS? How much primary swap do you wish to configure?

© 2004 Hewlett-Packard Development Company, L.P.

The **`Software`** tab allows you to select additional software to include in the initial install. You can choose to simply install the core operating system, or you may want to install additional software from one or more additional network depots. Note that the 11i "Operating Environment" field on the **`Basic`** tab doesn't, by default, automatically select all of the software included on your OE CDROMs. Some of the OE software is customer-selectable. The install-time security bundles, for instance, must be manually selected on the **`Software`** tab screen if you wish to include it in your install. If you forget to install a selectable product at install-time, you can always add it later via the **`swinstall`** GUI.

Go to the **`System`** tab to configure the host's networking parameters, date, time zone, and root password.

On the **`File System`** tab, you can modify the LVM and file system configuration. You can add and remove disks in the root volume group, and define the size of each logical volume.

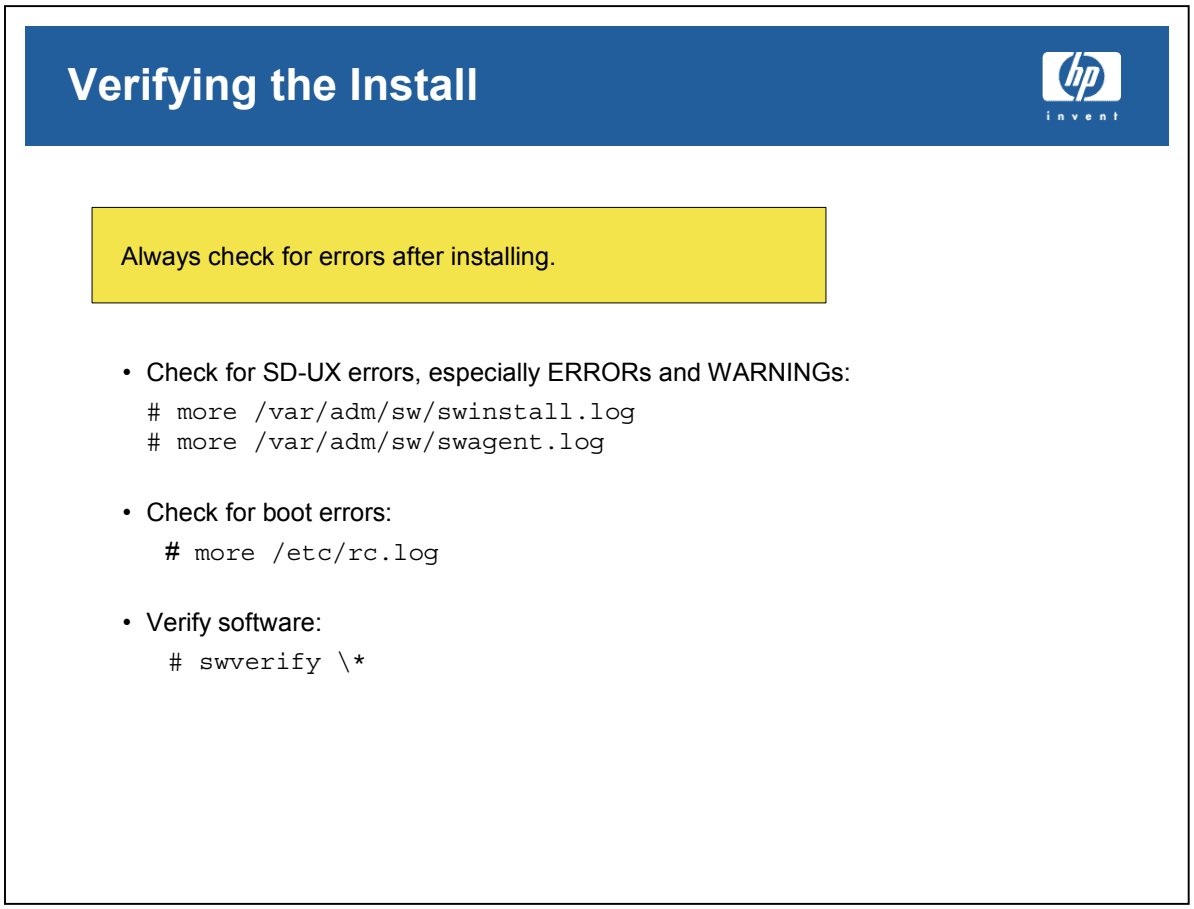
If you are installing from an Ignite-UX server, the Ignite administrator may have written some custom post-install scripts. Go to the **`Advanced`** tab to select the custom scripts from the Ignite server that you wish to execute as a part of the install process.

After navigating through each of the Ignite menus, tab to **`[Go!]`** and press return to continue the install process. If any warning boxes appear, read them carefully, and click either [Go!] to proceed, or **`[Cancel]`** to abort the install.

If you are installing HP-UX 11i from the CDROMs, you will be prompted to insert the second "Core" CD partway through the install.

© 2004 Hewlett-Packard Development Company, L.P.

22–10. SLIDE: Verifying the Install



Student Notes

After the install or update is complete, carefully check the **`swinstall.log`**, **`swagent.log`**, and **`rc.log`** files.

```
# more /var/adm/sw/swinstall.log
# more /var/adm/sw/swagent.log
# more /etc/rc.log
```

Expect to see a very large number of NOTE messages in the log. WARNING messages should be reviewed; some WARNINGS are harmless, but others may report potentially dangerous conditions in the system. Read ERROR messages very carefully. Some errors may cause the update process to fail. If this occurs, address the issues reported in the log files, and re-start the install/update.

If you have any problems during the install/update, consult the troubleshooting appendix at the end of the *HP-UX 11i Installation and Update Guide* on **`http://docs.hp.com`**.

© 2004 Hewlett-Packard Development Company, L.P.

Verify the State of the Packages

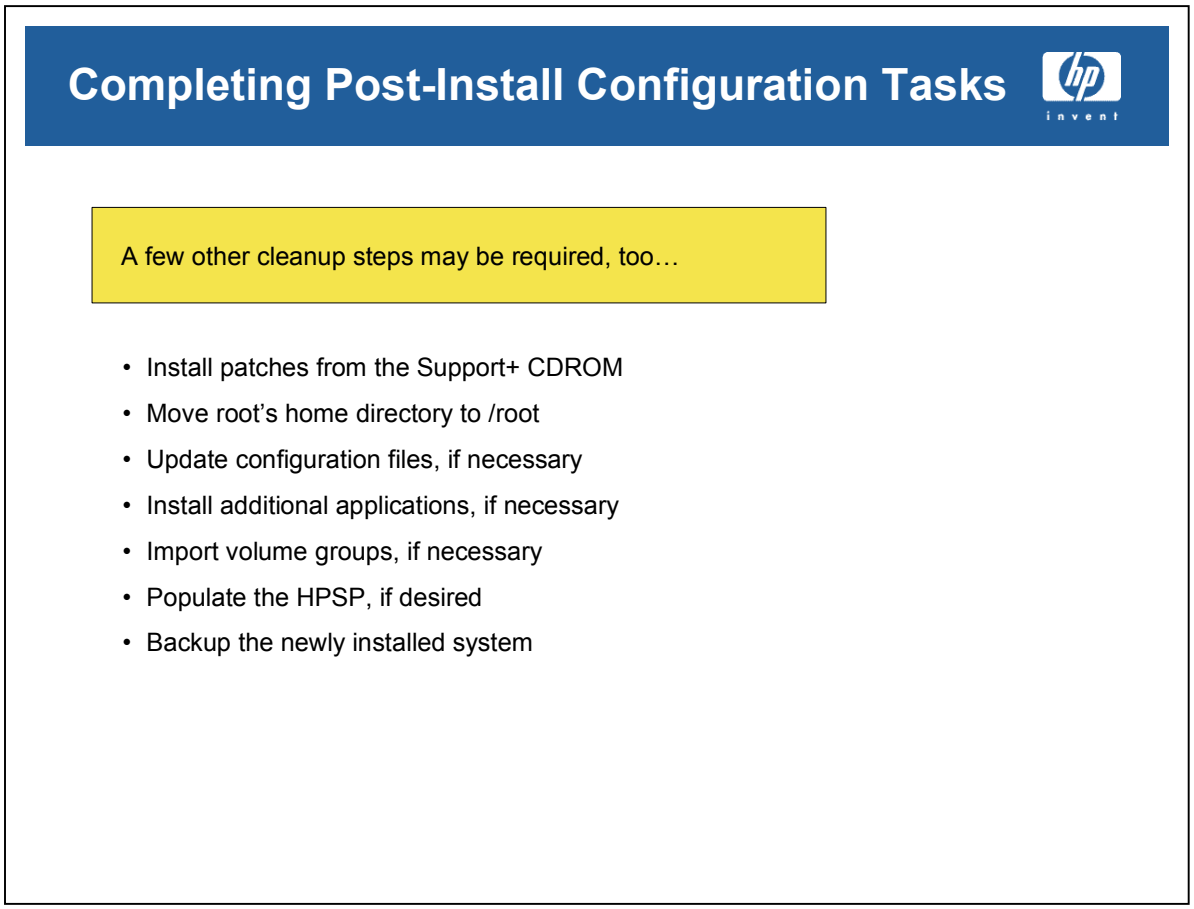
To verify that all installed products are compatible with the new OS, and properly installed, run **swverify**.

```
# swverify \*
```

Watch for errors and warnings.

© 2004 Hewlett-Packard Development Company, L.P.

22–11. SLIDE: Completing Post-Install Configuration Tasks



Student Notes

A few final post-install/update steps may be required to bring your system to a fully functional state.

Install Patches

Be sure to install the latest patches from the Support+ CD, and any additional point patches that may be applicable to your system. See the patch management chapter earlier in the course for more information.

Move Root's Home Directory to `/root`

By default, Ignite-UX configures root's home directory in `/`. It's a good idea to move root's home directory to a more secure location with 700 permissions (for example: **`/root`**):

```
# mkdir /root
# chown root:sys /root
# chmod 700 /root
# mv /.profile /.shrc /.sh_history /root
# usermod -d /root root
# vi /root/.profile   # update HISTFILE variable if necessary
```


© 2004 Hewlett-Packard Development Company, L.P.

Update Configuration Files, If Necessary

Configure **/etc/passwd**, **~/.profile**, and other configuration files as desired.

Install Additional Applications, If Necessary

The install process only installs operating system software. If you use other HP or third party applications, they may need to be updated, too. In many cases, this can be accomplished via **swinstall**.

Import Volume Groups, If Necessary

If you have any volume groups from other systems that you wish to move to the new system, those volume groups must be imported to your local **/etc/lvmtab** file.

On the other system, use the **vgexport** command to create a "map file" recording the volume group's logical volume names and volume group ID string.

```
# vgchange –a n vg01
# vgexport –s –v –m /etc/lvmconf/vg01.map vg01
```

Move the disks to the new system. In some cases, this may require powering off the devices and moving cables. If you are connected a Storage Area Network, you may be able to make the disks available to the new system by modifying your SAN configuration. Copy the map file to the new system, too.

```
# rcp otherhost:/etc/lvmconf/vg01.map /etc/lvmconf/vg01.map
```

Create a device file for each volume group that you plan to import (the volume group name and minor number need not match the volume group's old name and minor number).

```
# mkdir /dev/vg01
# mknod /dev/vg01/group c 64 0x010000
# chmod 640 /dev/vg01/group
# chown root:sys /dev/vg01/group
```

Now import and activate the volume groups.

```
# vgimport –s –v -m /etc/lvmconf/vg01.map vg01
# vgchange -a y vg01
# vgcfgbackup vg01
```

Add the new file systems to **/etc/fstab** if necessary.

Populate the HPSP, If Desired

Recall that the HPSP (HP Service Partition) may be used to store and run offline diagnostic utilities that may be useful if your system becomes unbootable. Ignite-UX creates an HPSP partition, but you have to manually copy the e-Diag utilities to the partition from the Diagnostic and Utilities CDROM that comes with your system.

Several steps are required.

1. Insert the HP diagnostics and utility CDROM in the CDROM drive.

© 2004 Hewlett-Packard Development Company, L.P.

2. Reboot the system and access the EFI shell.

3. View available devices, and look for the DVD/CDROM device.

```
SHELL> map -b –r -fs
```

4. Run the diagnostics utility from the DVD/CD.

```
SHELL> fsn:\efi\boot\launchmenu\launchmenu.efi
```

Follow the instructions to copy the utilities to the service partition, then reboot.

Backup the Newly Installed/Updated System

After the install or update process is complete, make a backup copy of your newly installed system using **fbackup** or your favorite backup utility. Also, create a new **make_tape_recovery** tape or **make_net_recovery** image. By backing up your system, you will have a reliable system image to return to if subsequent system changes or a system failure corrupts your system.

```
# fbackup -f /dev/rmt/0m -i / -v
# make_tape_recovery -A -d /dev/rmt/0m
```


© 2004 Hewlett-Packard Development Company, L.P.

22–12. SLIDE: For Further Study

For Further Study

To learn more about installing and updating HP-UX, read the following manuals on http://docs.hp.com:

- *HP-UX 11i Version 2 Release Notes*
- *HP-UX 11i Version 2 Installation and Update Guide*
- *Read Before Installing or Updating to HP-UX 11i Version 2*
- *Software Distributor Administration Guide for HP-UX 11i v2*
- *Ignite-UX Administration Guide, Edition 14*
- *Managing Systems and Workgroups: A Guide for HP-UX System Administrators*

Student Notes

To learn more about installing and updating HP-UX, read the following manuals on **`http://docs.hp.com`**:

- *HP-UX 11i Version 2 Release Notes*
- *HP-UX 11i Version 2 Installation and Update Guide*
- *Read Before Installing or Updating to HP-UX 11i Version 2*
- *Software Distributor Administration Guide for HP-UX 11i v2*
- *Ignite-UX Administration Guide, Edition 14*
- *Managing Systems and Workgroups: A Guide for HP-UX System Administrators*

© 2004 Hewlett-Packard Development Company, L.P.

22–13. LAB: Installing HP-UX

Directions

Your instructor will provide you with a new IP address, hostname, and netmask. You should also be given the IP address of an Ignite Server. Record these addresses below:

IP Address: _____

Hostname: _____

Netmask: _____

Ignite Server's IP: _____

1. This lab requires you to interact with the boot menus, which can only be accomplished on the system console. If you are using remote lab equipment, access your system's console interface via the GSP/MP.

2. Install the "HP-UX Mission Critical OE" from the Ignite server suggested by your instructor. If DHCP doesn't provide your network configuration information for you, use the parameters you recorded above.

© 2004 Hewlett-Packard Development Company, L.P.

Appendix A — Connecting to a Network

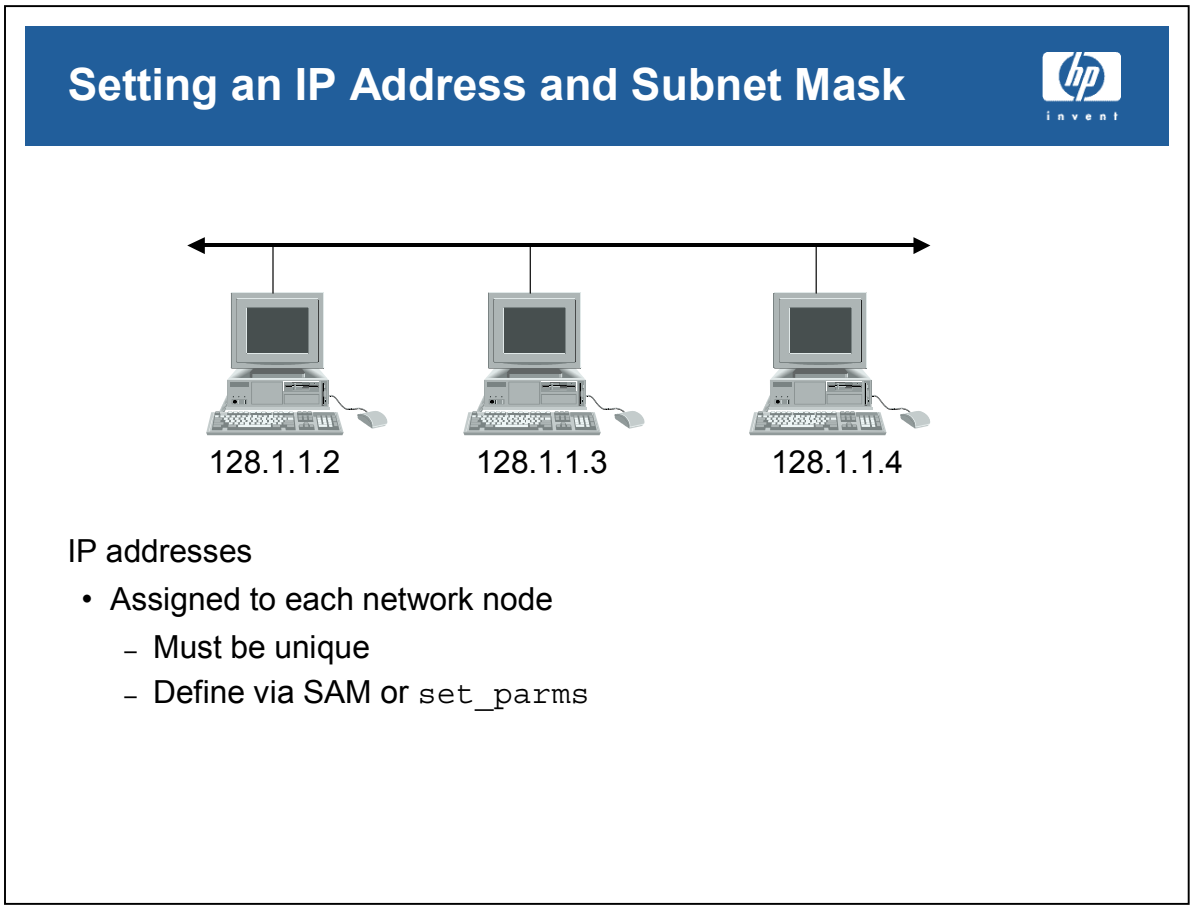
Objectives

Upon completion of this module, you will be able to do the following:

- Physically connect an HP-UX machine to an existing LAN.
- Configure a host name using **`set_parms`** or SAM.
- Configure an IP address using **`set_parms`** or SAM.
- Configure a default route using **`set_parms`** or SAM.
- Configure a host as a DNS client using **`set_parms`** or SAM.
- Configure a host as an NIS client using **`set_parms`** or SAM.
- Test connectivity with ping and **`nslookup`** or SAM.

© 2004 Hewlett-Packard Development Company, L.P.

A–1. SLIDE: Setting an IP Address and Subnet Mask



Student Notes

What Is an IP address?

Each machine on a local area network (LAN), whether it is a laser printer, workstation, or server, is assigned a unique Internet protocol (IP) address. Every packet of data sent across the network contains a destination IP address which determines the recipient of that packet of data, and the route taken to reach that destination.

An IP address consists of four dot (.) separated integers in the range 0-255.

IP Address Examples

125.67.12.43
15.34.67.9
212.1.1.34

Your network administrator or ISP should assign you an IP address, or multiple IP addresses if you have multiple LAN interface cards in your machine.

© 2004 Hewlett-Packard Development Company, L.P.

What Is a Subnet Mask?

Many companies these days have large networks that are divided into "subnetworks". Dividing a larger network into smaller "subnets" can greatly improve performance and reliability on large LANs. If your company has a subnetted network, your network manager will assign you a "subnet mask". The subnet mask, in conjunction with your IP address, determines which subnet you are attached to, and your unique host address on that subnet. In order to successfully communicate with other nodes on the LAN, you must define the subnet mask for your machine.

Subnet Mask Examples

255.255.0.0
255.255.255.0
255.255.254.0

Setting an IP Address and Subnet Mask with set_parms

When you connect your node to the LAN, your ISP or network manager will assign your host a unique IP address according to the address scheme implemented in your shop. You should also be given a subnet mask. When you boot your system for the first time, the **set_parms** utility automatically prompts for an IP address and several other network parameters. You can explicitly run **set_parms** any time to change your network parameters by typing:

```
# set_parms initial
```

Setting an IP Address and Subnet Mask with SAM

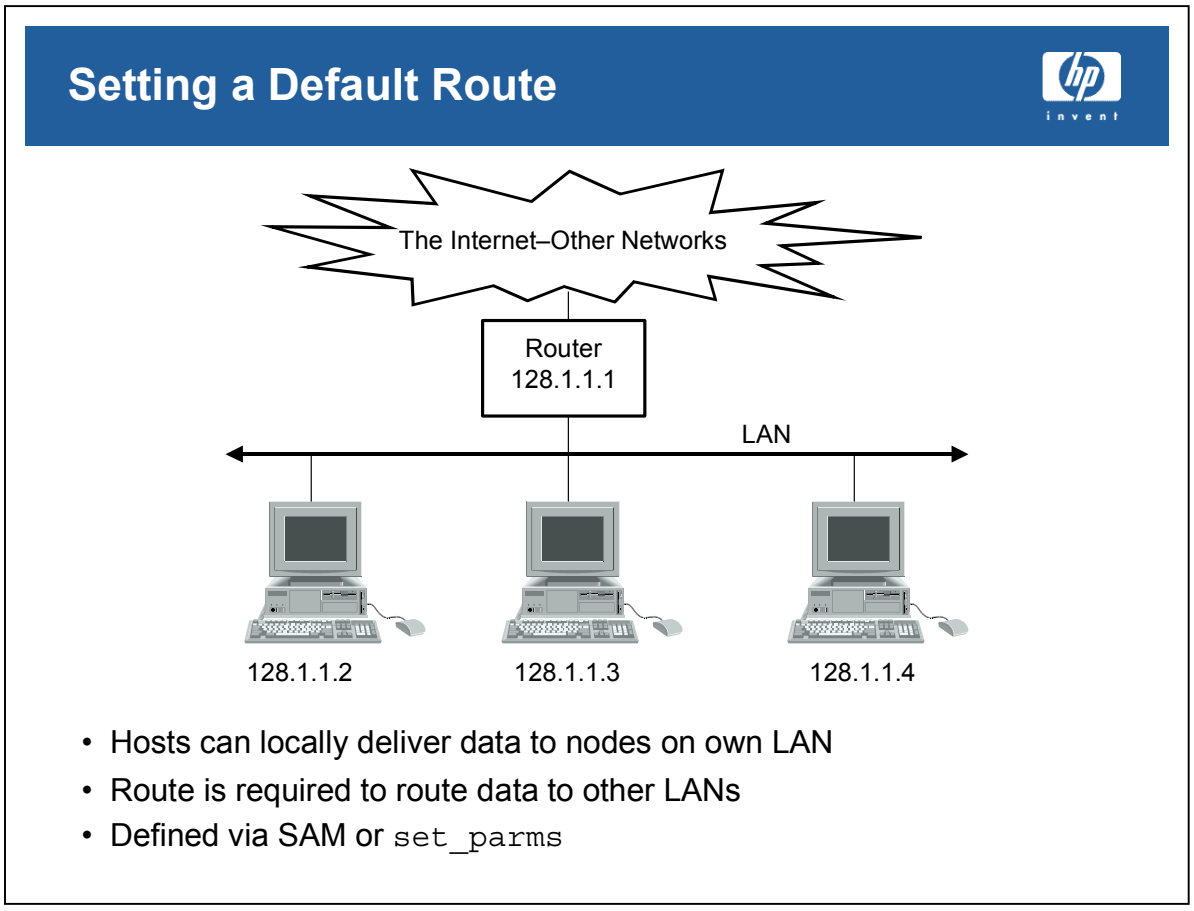
SAM also provides an intuitive interface for setting or changing your IP address and subnet mask:

```
# sam --> Networking & Communications
      --> Network Interface Cards
          (Select an interface card)
          Actions --> Configure
```

SAM displays a list of your system's LAN cards including the LAN card type, name, hardware path, status, and IP address. To change your subnet mask or IP address, select the interface card to change, select Actions --> Configure, and answer the questions in the dialog box that follows.

© 2004 Hewlett-Packard Development Company, L.P.

A–2. SLIDE: Setting a Default Route



Student Notes

What Is a Default Route?

Without any further configuration, HP-UX should be able to deliver data to any node on your local network once your IP address and subnet mask have been set. However, if you wish to access nodes beyond your local network, your machine must have access to a *router*. A router is a device that is able to route data to other networks.

Your network manager should provide you with the IP address of a router on your LAN. Your host should pass all data destined to other networks through the default router specified by your network manager.

© 2004 Hewlett-Packard Development Company, L.P.

Defining a Default Route

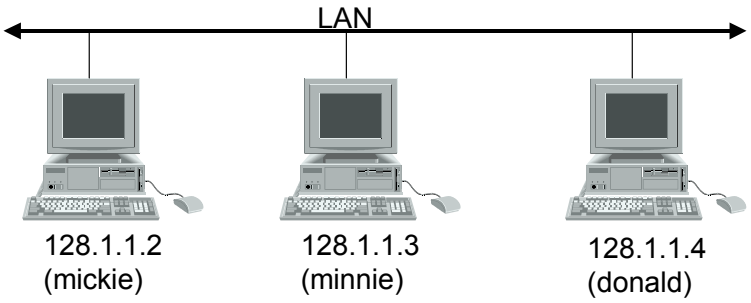
The default router address may be set via either **`set_parms`** or SAM. Note that SAM uses the term *default gateway* interchangeably with *default router*:

```
# set_parms initial
# sam --> Networking & Communications
      --> Hosts
      --> Local Hosts File
          Actions --> Configure Default Gateway
```


© 2004 Hewlett-Packard Development Company, L.P.

A–3. SLIDE: Setting a System Hostname

Setting a System Hostname



Hostnames

- Mnemonic name assigned to each node
- Can be based on
  - Purpose of node
  - Primary users of node
  - Any other naming convention
- Defined via `set_parms`

Student Notes

What Is a Hostname?

Since many users find it difficult to remember IP addresses, most machines are also assigned a "hostname". A hostname is simply a unique alphanumeric "nickname" assigned to each host on the network. Hostnames are assigned based on any of the following:

- the purpose of the node
- the primary user of the node
- any other naming convention chosen by the network manager

© 2004 Hewlett-Packard Development Company, L.P.

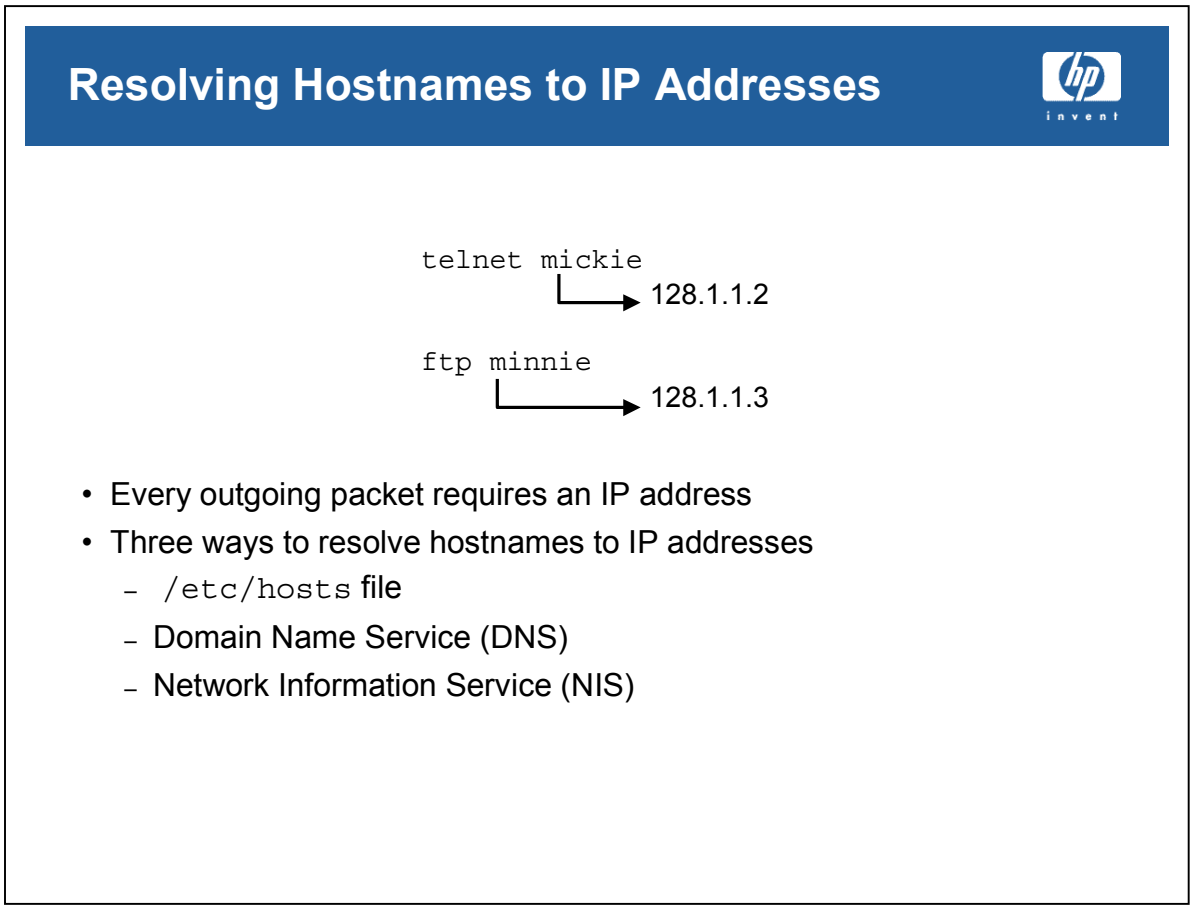
Setting Your System Hostname

Your network manager will assign you a hostname, which must be defined on your machine using **`set_parms`** or SAM:

```
# set_parms initial

# sam --> Networking & Communications
      --> Lan Interface Cards
          Actions --> Modify System Name
```


© 2004 Hewlett-Packard Development Company, L.P.

A–4. SLIDE: Resolving Hostnames to IP Addresses



Student Notes

Although hostnames are more memorable than IP addresses, a router must know a packet's IP address to successfully deliver data to a destination. For this reason, if a user or application specifies a packet's destination as a hostname, the system must translate the destination hostname to a destination IP address.

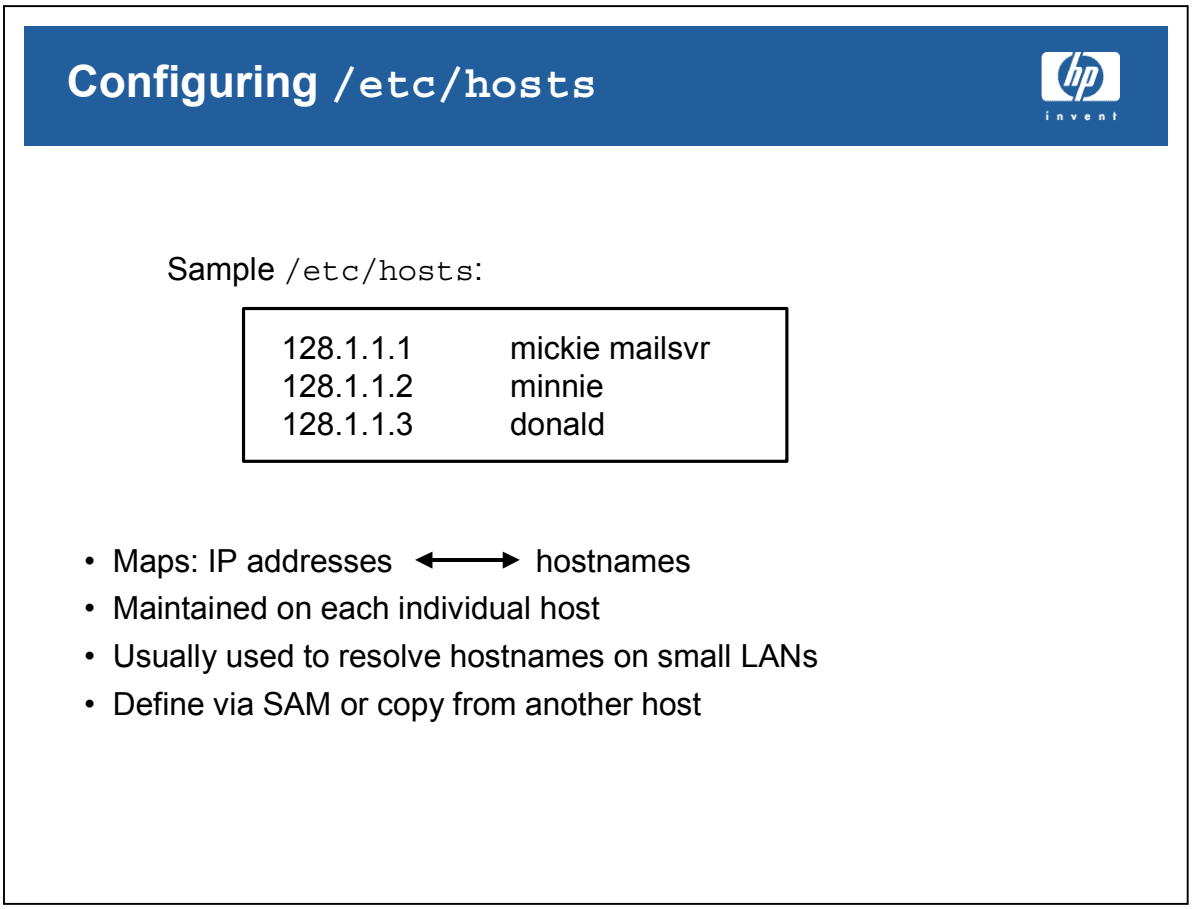
There are three ways to resolve hostnames to IP addresses on an HP-UX system:

- via the **`/etc/hosts`** file.
- via the Domain Name Service (DNS) .
- via the Network Information Service (NIS).

We will consider each method in the pages that follow.

© 2004 Hewlett-Packard Development Company, L.P.

A–5. SLIDE: Configuring `/etc/hosts`



Student Notes

What Is the /etc/hosts File?

If you need to communicate only with hosts on your local network, your network manager may recommend using the **`/etc/hosts`** file to resolve hostnames to IP addresses. The **`/etc/hosts`** file is a configuration file maintained locally on each host to resolve hostnames to IPs.

Each line in the **`/etc/hosts`** file contains an IP address in the far left field, followed by the hostname associated with that IP address in the second field. In the example on the slide, hostname **`donald`** resolves to IP address 128.1.1.4. Hostname **`minnie`** resolves to IP address 128.1.1.3.

You may optionally define one or more **`aliases`** for each IP address, too. An alias is simply another name by which an IP can be referenced. In the example on the slide, both **`mailsvr`** and **`mickie`** resolve to IP 128.1.1.2.

Each line in **`/etc/hosts`** can contain a comment preceded by a **`#`** sign as well.

© 2004 Hewlett-Packard Development Company, L.P.

Configuring `/etc/hosts`

The **`/etc/hosts`** file must be modified anytime an IP address or hostname changes. Note that **`/etc/hosts`** is maintained locally on each host, so changes must be propagated to all hosts if you wish to maintain consistency across all machines on your network.

You can modify **`/etc/hosts`** several different ways:

- Modify the local hosts file using SAM.
- Modify the local hosts file directly with vi.
- FTP an up-to-date **`/etc/hosts`** file from another host.

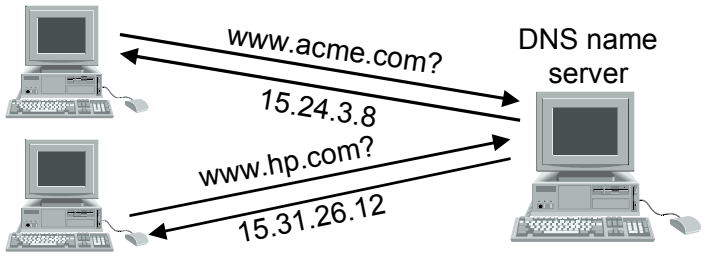
To get to the SAM interface for modifying your hosts file, select

```
# SAM --> Networking and Communications
      --> Hosts
      --> Local Hosts File
          Actions --> Add/Modify/Remove
```


© 2004 Hewlett-Packard Development Company, L.P.

A–6. SLIDE: Configuring a DNS Client

Configuring a DNS Client



- Used to resolve hostnames on larger networks
- Clients request lookups from DNS servers
- To prevent duplicate names, each host assigned to a “domain”
- Must define the following via SAM or `set_parms`
  - Your DNS server address
  - Your DNS “domain” name

Student Notes

What Is DNS?

Although the **`/etc/hosts`** file is adequate for hostname resolution on small networks, your network manager will likely recommend using the Domain Name Service (DNS) for name resolution if you wish to access the outside Internet or other networks within your company.

Instead of referencing a locally stored **`/etc/hosts`** file that quickly becomes outdated, hosts using DNS send name resolution requests to a specially configured *DNS name server*. Your network manager can provide the IP addresses of a DNS name server that your host uses.

In addition to defining your local name server's address, you need to define your *host's DNS domain*. The slide shows hosts in two domains:

```
hp.com
acme.com
```


© 2004 Hewlett-Packard Development Company, L.P.

Each host on the Internet can be uniquely identified by the concatenation of the host's hostname and domain. If two hosts have the same hostname, the domain name appended to the end of the hostname determines which host you wish to reference.

The example on the slide shows two hosts with hostname **www**. The two are differentiated from one another by the **hp.com** and **acme.com** domain names appended after **www**.

Specifying a DNS Nameserver

Configuring your host to use DNS is most easily accomplished via SAM. Go to:

```
# SAM --> DNS (BIND)
      --> DNS Resolver
          Actions --> Specify Name Servers
```

Up to three DNS nameservers may be defined. If one nameserver is unavailable, your system will automatically try another. Use the name server IP addresses provided by your network manager.

Specifying a DNS Domain Search List

While in SAM, you should also define your DNS domain. You may optionally list up to five other DNS domains to search when resolving hostnames. Defining additional search domains saves your users the trouble of typing long and cumbersome domain names.

Example

Without a domain search list defined, users in the **hp.com** domain may need to type:

```
# telnet corp.acme.com
```

Adding **acme.com** to the domain search list allows users to access **corp.acme.com** by simply typing:

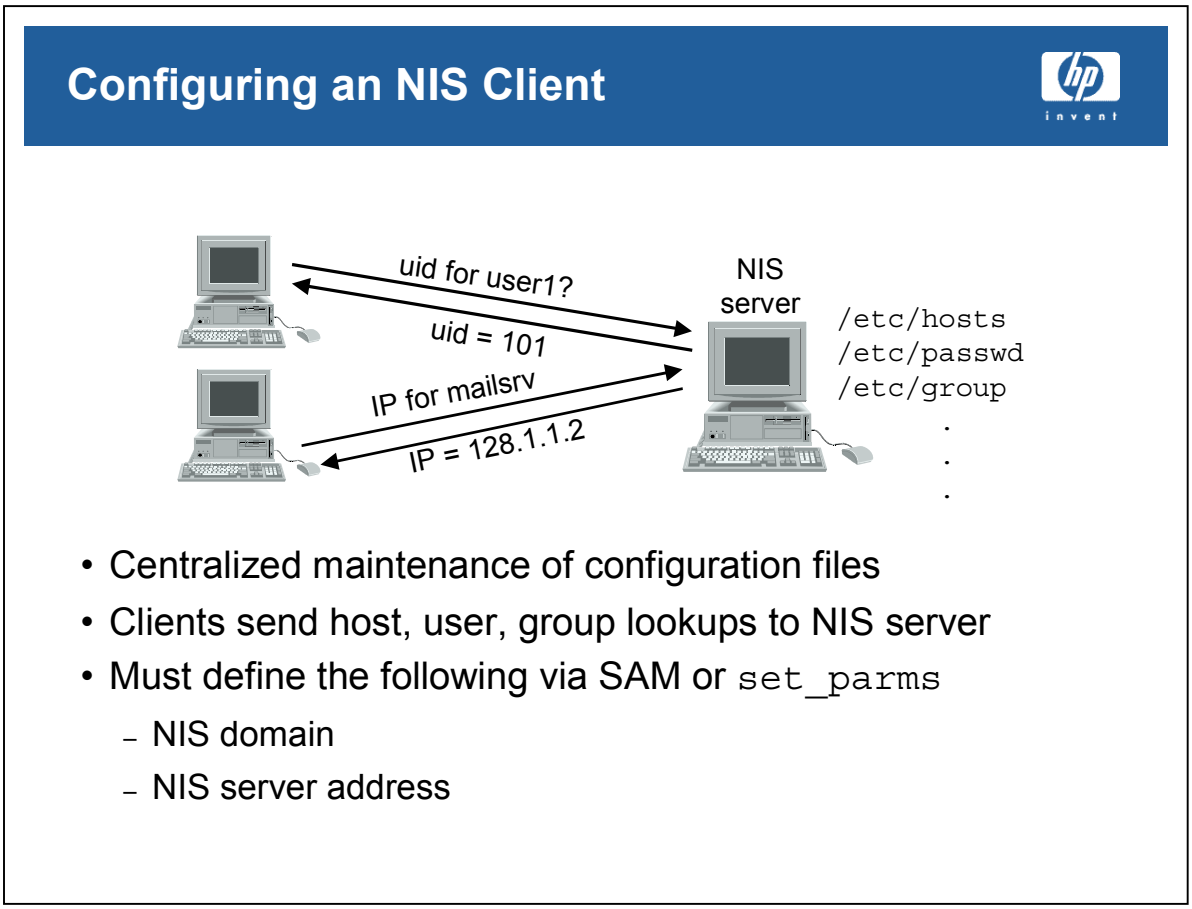
```
# telnet corp
```

SAM provides an intuitive interface for defining your own domain and a domain search list:

```
# SAM --> DNS (BIND)
      --> DNS Resolver
          Actions --> Set Default Domain
```


© 2004 Hewlett-Packard Development Company, L.P.

A–7. SLIDE: Configuring an NIS Client



Student Notes

UNIX hosts are configured via a series of configuration files in the **/etc** directory. These files must be constantly updated. Adding a new host on the network may require an additional line in every host's **/etc/hosts** file. Adding a new user may require an additional entry in every host's **/etc/passwd** file. In large network environments, maintaining these files on multiple hosts becomes tedious and time consuming.

The Network Information Service (NIS) simplifies configuration file updates significantly. Instead of managing copies of the system configuration files on every host, NIS makes it possible to manage **/etc/passwd**, **/etc/hosts**, and other configuration files from a single NIS server. NIS clients, then, simply forward user and host lookup requests to the NIS server. Thus, NIS provides a central point of administration for common configuration files, and preserves consistency of these configuration files across all hosts on the network.

Just one piece of information is required to configure an NIS client: the *NIS Domain Name*. An NIS domain is a logical grouping hosts that share a common set of configuration files. For example, all hosts in the marketing department may be placed in the marketing domain, while hosts in the accounting department may be placed in the accounting domain. If your organization uses NIS, you will need to obtain your NIS domain name from your organization's NIS administrator.

© 2004 Hewlett-Packard Development Company, L.P.

Configuring an NIS Client

When you first boot a newly-installed system, **`set_parms`** provides an opportunity to define the NIS domain name.

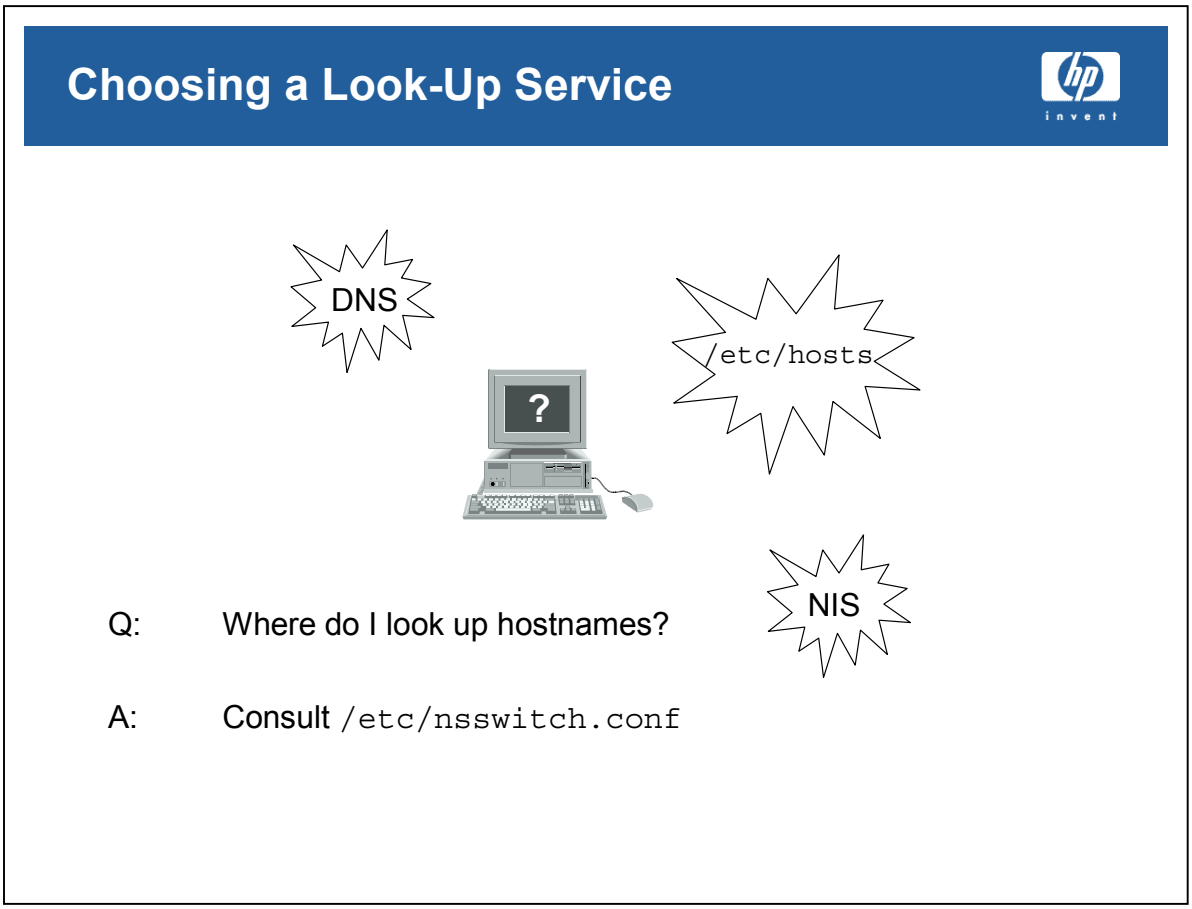
You can also configure NIS client functionality via SAM:

```
# sam --> Networking & Communications
      --> NIS
          Actions --> Set Domain Name
          Actions --> Enable Client
```

To learn how to configure an NIS server, return for System & Network Administration II!

© 2004 Hewlett-Packard Development Company, L.P.

A–8. SLIDE: Choosing a Look-Up Service



Student Notes

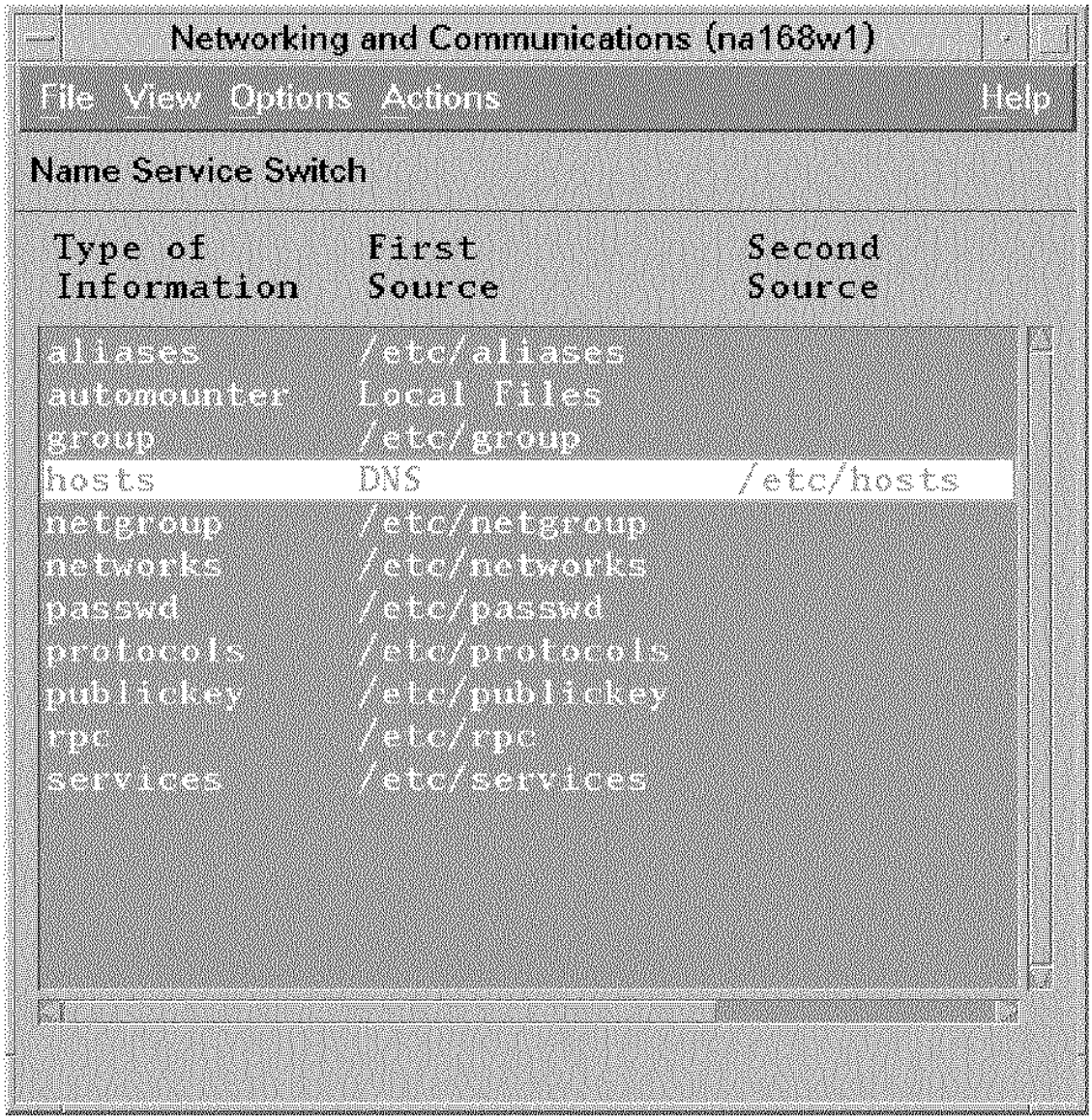
We've seen that hostnames may be resolved via the local **`/etc/hosts`** file, DNS or NIS. How does the system decide, then, which source to use? The **`/etc/nsswitch.conf`** file determines which source should be used for look-ups. This file can be edited manually, or can be configured via SAM:

```
# sam --> Networking and Communications
      --> Name Service Switch
          (Select a lookup type)
          Actions --> Configure Name Service Switch
```

The sample system that provided the screen shot below is configured to use DNS as the primary source for hostname resolution. If the DNS server doesn't respond or isn't configured, the system resorts to using the local **`/etc/hosts`** file instead.

© 2004 Hewlett-Packard Development Company, L.P.

SAM Name Service Switch

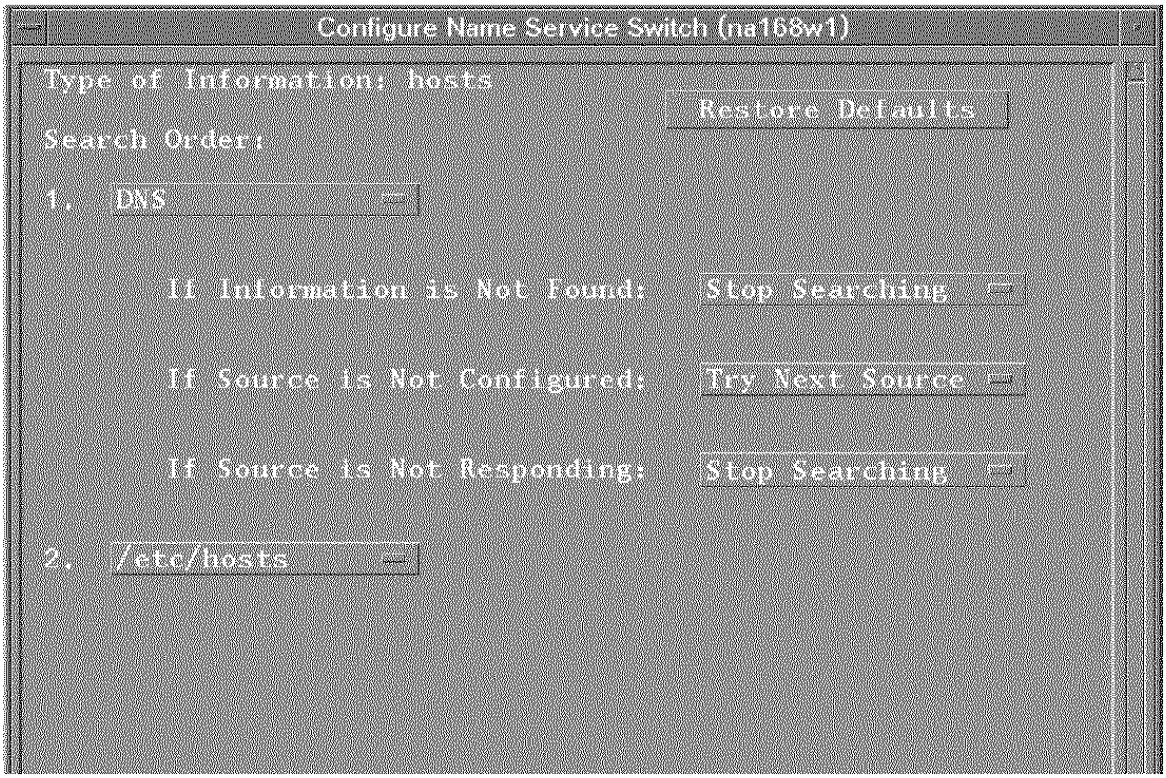


a669139

SAM also lets you change the lookup source order. For instance, to force your system to resolve hostnames first from **/etc/hosts**, then from DNS, select the hosts line from the object list and click **Actions --> Configure Name Service Switch**. The screen that follows lets you determine which service is used when.

© 2004 Hewlett-Packard Development Company, L.P.

SAM Configure Name Service Switch

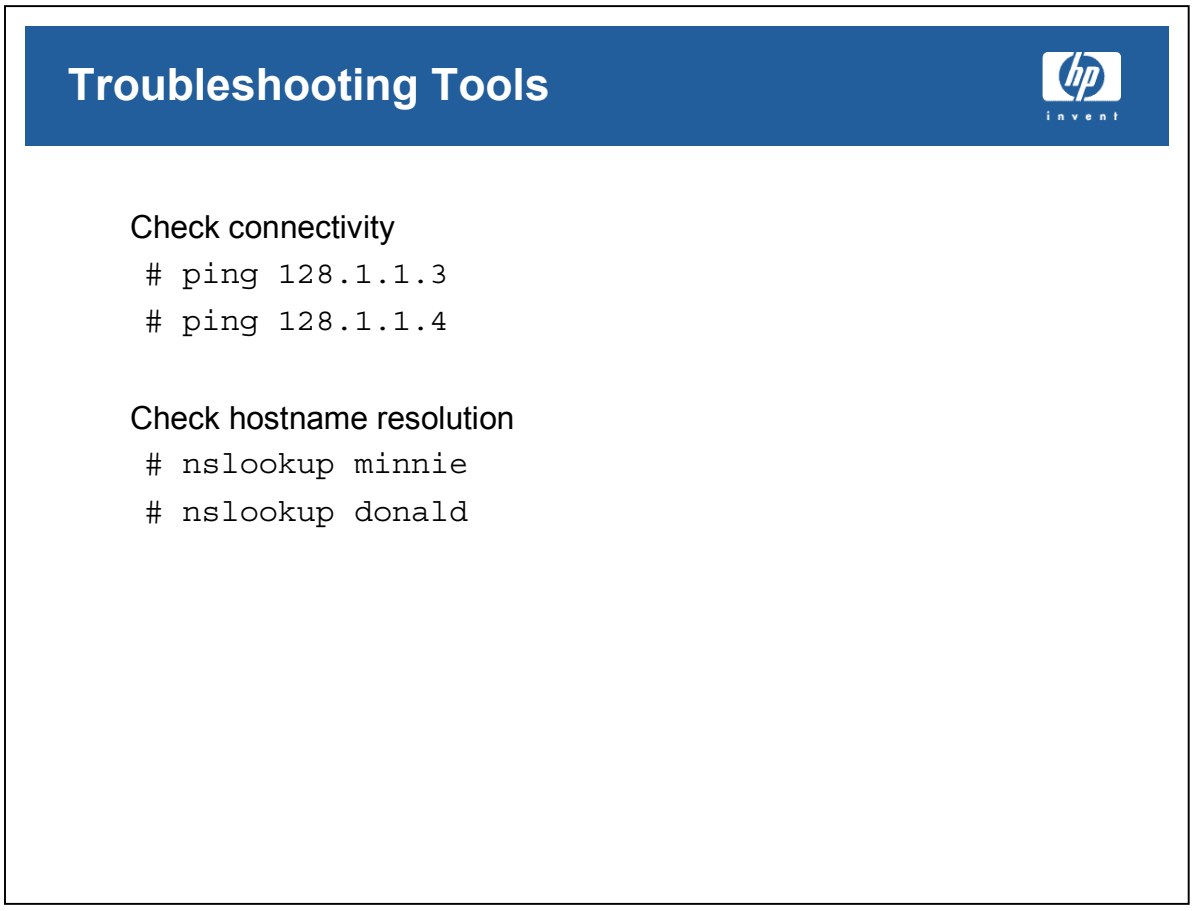


a669140

The **/etc/nsswitch.conf** file is used to configure the source order for password, group, and other types of look-ups as well, if you have access to a network information service (NIS) server.

© 2004 Hewlett-Packard Development Company, L.P.

A–9. SLIDE: Troubleshooting Tools



Student Notes

After configuring your network connection, check to ensure your configuration worked using **`ping(1m)`** and **`nslookup(1)`**.

Checking Connectivity with ping

The **`ping`** command is used to test connectivity to any IP address.

Examples

```
# ping 128.1.1.3
# ping 128.1.1.4
```

`ping` sends a series of test packets to the specified IP and waits for a response. After a few seconds, hit CTRL + c and check to ensure that the packet loss shows a value of 0.

© 2004 Hewlett-Packard Development Company, L.P.

Try several pings:

1. **ping** yourself.
2. **ping** another host on your local network.
3. **ping** your default router.
4. **ping** a host on another network.

If **ping** fails, check the following:

- Is the LAN card installed properly?
- Is the LAN cable connected properly?
- Did you set your IP properly?
- Did you configure a default gateway/router?
- Did you configure your subnet mask properly?

Check Hostname Resolution with `nslookup`

The **nslookup** utility can be used to ensure that your host is resolving hostnames properly. Given a hostname as an argument, **nslookup** responds with the corresponding IP address. **nslookup** also notes the source providing the IP address.

Example (on a host using the local `/etc/hosts` file)

```
# nslookup minnie
Using /etc/hosts on mickie

looking up FILES
Name:         minnie
Address:      128.1.1.3
```

If **nslookup** responds with an incorrect IP address, check the following:

1. Did you configure the name service switch properly?
2. If you are using DNS, is the name server IP address correct?
3. If you are using NIS, is the NIS domain name correct?
4. If you are using **/etc/hosts**, is the **/etc/hosts** file entry correct?

© 2004 Hewlett-Packard Development Company, L.P.

A–10. LAB: Connecting to the Network

Directions

In the exercise that follows, you will have an opportunity to explore the **set_parms** and SAM screens used to set network parameters on your system. Back in your shop, your IT department or ISP should provide you with your basic networking parameters. For this lab exercise, your instructor will provide the addresses you need:

```
                          Hostname:
                        IP address:
                       Subnet mask:
Def. Router/Gateway IP address:
                        DNS domain:      (optional)
                DNS server address:      (optional)
  Secondary DNS server address:          (optional)
```

Part 1: Preliminary Steps

1. In just a moment, you will be asked to run a script which will temporarily un-configure your system's default LAN interface. If you are using remote lab equipment, log into your system's console interface via the GSP/MP to ensure that you are able to maintain connectivity throughout the lab.

2. There should be a shell script on your system called **/labs/clobbernet.sh**. Run the script. The script clobbers your network configuration files and reboots your machine. When your machine comes back up again, it will automatically run the **set_parms** utility, just as it would if you had just installed the operating system.

   ```
   # /labs/clobbernet.sh
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 1: Configuring an IP Address and Hostname

1. **`set_parms`** first asks if you want to connect to the network. For this lab exercise, we do want to connect to the network, so click "OK."

2. Next, **`set_parms`** asks if you wish to obtain your network parameters via DHCP. DHCP is a service that can automatically provide your machine with an IP address and hostname, if a DHCP server exists on your network. For this exercise, click "NO".

3. Answer the prompts in the screens that follow to set your hostname, time zone information, and IP address. When asked if you wish to set additional network information (for example, a default gateway, netmask, and DNS and NIS information), Answer NO. We will configure these parameters later via SAM.

NOTE:	If you make a mistake while entering any of the network parameters, you can run **`set_parms`** again any time by typing **`set_parms initial`**.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Checking Your IP Address and Setting Your Netmask

1. After running **`set_parms`** and logging in, go to:

   ```
   SAM --> Networking and Communications --> Network Interface Cards
   ```

   to ensure that your IP is set correctly. What hardware slot number is your lan0 card installed in? Is lan0 enabled for use? What is lan0's IP address? Are there any other LAN cards on your machine?

2. While still at the Network Interface Card screen, set your subnet mask. Without the subnet mask, you may not be able to communicate with other nodes on your LAN. Set your subnet mask for lan0 via SAM.

   Each machine on a network, regardless of the number of LAN interface cards, has a single hostname. If you have multiple LAN cards on a host, SAM requires you to configure a unique "Host Name Alias" for each LAN card. This is useful in troubleshooting situations where you may wish to access a specific LAN card on a target machine rather than the general hostname. If SAM asks you to set an alias, simply use your hostname with an "a" appended to the end.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Testing Your Network Connection

1. Try pinging your own IP address to see if your IP address is set properly. Did this work?

2. Next, try pinging a couple of neighboring machines' IP addresses. Did this work?

3. Next, try pinging a neighboring machine using its hostname. Can you guess why this fails? (Hint: Look in the **`/etc/hosts`** file.)

© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Configuring `/etc/hosts`

You won't be able to access other hosts by hostname until you add entries for those machines to your **`/etc/hosts`** file. SAM offers the easiest method for viewing and modifying your hosts file. Use SAM to Answer the questions that follow.

1. Which hosts' entries are configured so far?

2. Add entries in your hosts file for the other machines in your row.

3. Try pinging your neighbors by hostname again. Does this work?

© 2004 Hewlett-Packard Development Company, L.P.

Part 5 (Optional): Configuring a Default Gateway and DNS

If you want to be able to access other networks or the Internet, you will have to define a default gateway and DNS server address. Both of these addresses can be configured via SAM.

1. Use SAM to configure your default gateway.

2. Use SAM to configure your default DNS domain name. Don't configure any "Other Domains to Search".

3. Now use SAM to configure the IP addresses of your DNS nameservers.

© 2004 Hewlett-Packard Development Company, L.P.

Part 6 (Optional): Configure the Name Service Switch

1. Did your DNS configuration work? If DNS is configured properly on your machine, and the DNS server is functioning, you should be able to lookup hostnames of machines outside your local network. Try an **`nslookup`** on **`www.hp.com`**. What happens?

2. Look carefully at the output from the **`nslookup`** you did in the previous exercise. Can you explain why you were unable to find an IP address for www.hp.com? (Hint: What lookup source is **`nslookup`** using at this point?)

3. Configure the name service switch via SAM to ensure that your system consults the DNS server if a hostname isn't found in the local **`/etc/hosts`** file.

4. After changing the nameservice switch, SAM reboots your machine. After the reboot, do an **`nslookup`** on **`www.hp.com`** again. Are you able to resolve **`www.hp.com`** now?

© 2004 Hewlett-Packard Development Company, L.P.

Solutions

© 2004 Hewlett-Packard Development Company, L.P.

1–7. LAB: SAM

Part 1: Exploring SAM's Functional Areas

Our goal in this first portion of the lab exercise is to become more familiar with the SAM interface, and to explore some of SAM's functional areas. Typically when using SAM you will use the following procedure:

1. Run SAM.
2. Navigate functional areas to the proper object list.
3. Sort and filter the object list to your liking.
4. Select an object or objects with the mouse or space bar.
5. Choose an action from the **`Actions`** menu to perform on the selected object(s).

Let's try it!

1. Log onto your workstation or server and launch the SAM GUI or TUI. The first screen in SAM lists a number of functional areas.

   Answer

   # sam

2. Which functional area would you choose to view a list of user accounts on your system? Select the appropriate area, and determine the number of accounts on your system. (Note: In some cases, a functional area may contain one or more sub-areas. You may have to drill down through several menus to get to the information you are looking for.) Explore some of the other SAM functional areas to find answers to the questions that follow.

   Answer

   The functional area that lists user accounts is:

   ```
   Accounts for Users and Groups -> Users
   ```

3. Without selecting a user from the object list, pull down the Actions menu. Can any actions be performed here without selecting a user?

   Answer

   Several actions are available, even before selecting a user. For example, you can add a new user account.

4. The **`Add`** action allows you to create a new user account. Select **`Actions ->Add`**. This should bring up a dialog screen requesting information about the new account. Set **`operator`** as the user name for the new account, then click the **`OK`** button so SAM will take the defaults for the rest of the screen. After SAM prompts you for a password for the new account, check the object list to see what happened.

© 2004 Hewlett-Packard Development Company, L.P.

Answer

After following the suggested steps and entering a password, an operator user is added to the object list.

5. Choose **Actions -> Add** again and enter user name **dbmgr**. Instead of pressing **OK**, try the **Apply** button. What is the difference between the **OK** and **Apply** buttons in a dialog box? Can you think of a situation where **Apply** might be a more efficient choice than **OK**?

   Answer

   The difference between **Apply** and **OK** is that **Apply** leaves the dialog box open to add another user, whereas OK creates the requested user and closes the dialog box. This might be useful if you were creating many new user accounts and didn't want to repeatedly choose **Add** from the **Actions** menu.

6. Now try performing an action on an existing object. Using the mouse (GUI) or space bar (TUI), select **user1** from the object list. Now pull down the Actions menu. Now that you have selected a user, there should be many more actions to choose from. Choose **Modify user's password** and answer the questions that follow.

7. In some cases, the dialog boxes that result from SAM actions may reference terms or concepts you haven't encountered before. Select **user1** from the object list again, then choose **Actions -> Modify**. This should open the **Modify a User** dialog box. The dialog box allows you to change the user's Primary group and Start-up Program. For more information about terms and concepts you encounter in any SAM window, look for a Help button or Help menu on the menu bar. Click on the **Help** button at the bottom of the dialog box window. Skim the text, and try clicking on a couple of the underlined phrases. (In the TUI, select one of the "See also" topics instead.) Any underlined phrase in a SAM help window is a link to additional information. When you have finished with the Help window, click **Close** to return to the **Modify a user** dialog box. We didn't really want to modify **user1**'s account after all, so click on **Cancel**.

8. Occasionally, SAM will complain about an action you request. Now select the root user account. Choose **Actions -> Remove**. Can you explain SAM's response to this action? Do you consider this to be a *feature* or a *bug*?

   Answer

   A dialog box appears noting that SAM cannot remove root. This is a feature; removing root could be disastrous.

9. In other situations, SAM simply provides a warning message when you attempt a risky action. Try another experiment: From the Users screen, select the root user account, then choose **Actions -> Modify**. You will get a warning box, to which the most prudent response at this point is **No**; do not modify root's account.

10. SAM keeps a record of all actions that you perform. You can view this log via the SAM log viewer. Access the SAM log by selecting **View SAM Log** from the Options menu.

© 2004 Hewlett-Packard Development Company, L.P.

11. Experiment with some of the buttons in the dialog box that appears. Which Message level provides the most detail? When does your SAM log begin?

Answer

The Commands Only message level gives the most detail. The box at the top of the screen indicates when the SAM log began.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Using the Restricted SAM Builder

By default, SAM may only be run by user root. Administrators in large shops often have the luxury of a staff of operators to assist with system administration tasks. So how can we allow multiple operators, and perhaps even regular users, to access some of the functionality in SAM? Sharing the root password probably isn't the best solution. The **restricted SAM builder** makes it possible to grant non-root users access to selected functional areas in SAM. Our goal in the exercise that follows is to allow the operator user that you created earlier in the lab to manage processes via SAM and run automated backups, while denying them access to all other areas of SAM.

1. Choose **File -> Exit** to exit your current SAM session. Then type **sam -r** to start the restricted SAM builder. (Note: you must be logged in as root to run the restricted SAM builder.)

2. The SAM builder should display a list of user names and templates. Just accept the defaults for now and click **OK**.

3. Next, you will see a window listing each of the SAM functional areas. Note that some functional areas are marked to be enabled (indicated by a green icon), some are disabled (indicated by a red icon), and some are partially enabled (indicated by a yellow icon). In the TUI interface, the status of a functional area is displayed in text form (enabled, disabled, partially enabled). Start by choosing **Actions -> Disable all**. What happens to the functional area icons?

   Answer

   All the icons turn red.

4. Next, select the Process management icon with a single-click (GUI) or the space bar (TUI). Choose **Actions -> Enable**. What happens to the Process Management icon?

   Answer

   The icon turns green.

5. Next, we need to enable automated backups. Go to the **Backup and recovery** functional area, and select the Automated backup icon. Again go to the Actions menu and choose: **Actions -> Enable**. Go back up to the main functional area window. What happened to the Backup and recovery icon? Why is this icon marked as partially enabled?

   Answer

   The Backup and Recovery icon turns yellow. This indicates that some of the Backup and Recovery functionality is enabled, but some functionality is disabled.

6. You can enable and disable as many functional areas as you wish in the restricted SAM builder. Once you have enabled all the desired icons, choose **Actions -> Save privileges**.

© 2004 Hewlett-Packard Development Company, L.P.

7. The "Save privileges screen defines which user(s) will have access to the functional areas you have selected. Select the **operator** user from the list, and click **OK**. (If you had multiple operators, you could select multiple user names from the list while holding the shift key. If the names are not together in the list, hold the control key and select the names.) SAM saves the privileges you selected, then returns you to the icon window. Exit the restricted SAM builder, then log off your workstation or server.

8. Log in as operator and try running SAM by typing:**/usr/sbin/sam.** Which functional areas are available for use by your operator user?

   Answer

   Only the two functional areas we enabled appear when operator runs SAM.

9. Exit SAM, then try running SAM again by just typing **sam**. You will probably get an error message, **sh: sam: not found**. Try typing **/usr/sbin/sam**. Can you explain why SAM wouldn't start in the first case, but would start in the second?

   Answer

   The **/usr/sbin** directory is not included in the user's PATH variable.

10. Log out as user operator, then log back in again as user root.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Customizing SAM (Optional)

Although SAM has a fair amount of built-in functionality, you may have some database utilities or custom shell scripts that you would like to be able to execute directly from SAM. SAM allows you to do this. In the exercise that follows, you will create a custom SAM icon that could be used. In the exercise that follows, you will create a custom SAM icon that could be used to execute the **/usr/sbin/whodo** command to view a description of who is doing what on your system.

1. From the functional area launcher, choose `Actions –> Add Custom Application`.

2. Choose a name for your icon in the Label field.

3. Enter the full path name for the **whodo** command, **/usr/sbin/whodo**, in the **Command** field.

4. **whodo** is not a graphical utility, so choose **Terminal Environments** as the user interface.

5. Skip the optional fields and click **OK**.

6. Try your new icon.

7. Navigate to SAM's Time functional area. Create another custom SAM application that automatically runs the **/usr/bin/cal** program for you to display a calendar of the current month.

   Answer

   Go to: **Sam -> Time**.

   Choose: **Actions -> Add Custom Application**.
   Fill in the fields as follows:

Label:	Calendar
Command:	**/usr/bin/cal**
Execute using:	User's ID
Supported interfaces:	Terminal

© 2004 Hewlett-Packard Development Company, L.P.

2–12. LAB: Hands-On Adding Users

Perform the following tasks. Record the commands you use, and Answer any questions.

Part 1: Creating and Modifying Users and Groups

1. Use SAM to create a user account for user **mickey** on your system. Let SAM choose a UID for you. Assign the user to the **users** group, accept the default home directory name, and use the Posix shell as the startup shell. You may use any password you wish.

   Answer

   SAM -> Accounts for Users and Groups -> Users
   Actions -> Add...

2. What did SAM add to the **/etc/passwd** file?
   Is the new user listed in the **/etc/group** file? Explain!

   Answer

   There should be a new entry in the **/etc/passwd** file for the new user.
   However, "mickey" is not listed in **/etc/group**. In HP-UX 11.x, a user's *primary* group membership is recorded in the **/etc/passwd** GID field; **/etc/group** only records secondary group memberships.

3. Oops! We misspelled **mickey**'s username. Can you use SAM to change **mickey**'s username to **mickie**? Try it! Then do an **ll** of the **/home** directory. Did SAM automatically modify Mickey's home directory, too? If not, make the change manually.

   Answer

   SAM -> Accounts for Users and Groups -> Users
   Select "**mickey**"
   Actions -> Modify

   ```
   # ll /home
   ```

   Changing the user's username in **/etc/passwd** does not change the name of the user's home directory. Assuming the UID remains the same, however, the owner field in the **ll** output should be updated automatically. You can change the name of **mickie**'s home directory with the **mv** and **usermod** commands.

   ```
   # mv /home/mickey /home/mickie
   # usermod -d /home/mickie mickie
   ```


© 2004 Hewlett-Packard Development Company, L.P.

4. Create another user account for user **donald**, but this time don't use SAM. Make sure you include the option to create a home directory for the new user; otherwise, you may accept the defaults.

   Answer

   ```
   # useradd -m donald
   ```

5. Can **donald** log in at this point? What else needs to be done to make his account fully functional? Make it so!

   Answer

   donald needs a password before he can login:

   ```
   # passwd donald
   ```

6. Oops! **donald** wants to use the Korn shell as his default startup program. Make this change without using SAM.

   Answer

   ```
   # usermod -s /usr/bin/ksh donald
   ```

7. **mickie** and **donald** need to collaborate on a project, and have asked you to create a group ID for them. Create a group called **project**, and ensure that **mickie** and **donald** both have access to the group. Their primary group IDs, however, should remain set to **users**. Again, use the command line rather than SAM.

   Answer

   ```
   # groupadd project
   # usermod -G project mickie
   # usermod -G project donald
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Deactivating and Removing User Accounts

1. Deactivate/lock **mickie**'s account, but leave all of his files on the system.
   Remove **donald**'s account, but leave his files on the system, too.
   You may use either SAM or the command line.

   Answer

   ```
   # passwd -l mickie
   # userdel donald
   ```

2. What changed in the **/etc/passwd** file because of the commands in the previous question?

   Answer

   donald's **/etc/passwd** entry disappeared entirely. **mickie**'s password field is set to "*****" to indicate that his account is disabled.

3. What happens now when **donald** and **mickie** attempt to log in? **telnet** to your local host, and try to login using both usernames. What happens?

   ```
   # telnet localhost
   ```

   Answer

   Both login attempts should fail.

4. What happened to **donald** and **mickie**'s home directories? Do a long listing of **/home**. Can you explain what you see?

   ```
   # ll /home
   ```

   Answer

   Both directories are still there, but the owner field for **donald**'s directory lists a number rather than **donald**'s name. Internally, HP-UX identifies file ownership by UID rather than username. **ll** attempts to resolve these UIDs into usernames. However, since **donald** is no longer listed in **/etc/passwd**, the **ll** command has no way of determining which username is associated with the **/home/donald** directory.

5. Re-enable mickie's account. Choose a password for him, then do whatever is necessary to force him to choose a new password at next login.

   Answer

   ```
   # passwd mickie
   # passwd –f mickie
   ```


© 2004 Hewlett-Packard Development Company, L.P.

6. Verify your configuration by **telnet**'ing to your **localhost** using username **mickie**.

```
# telnet localhost
```

Answer

The login should succeed immediately prompt you to choose a new password.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Implementing Shadow Passwords

1. Run **pwconv** to create the **/etc/shadow** file.

   a. What is in the password field in **/etc/passwd** now?

   b. What fields are populated in **/etc/shadow**?

   c. What are the permissions on **/etc/shadow**? Why is this significant?

   Answer

   The password fields in **/etc/passwd** should contain x's.

   Each **/etc/shadow** entry should contain a user name, an encrypted password, and a timestamp field that indicates when the password was last changed. The other fields should be empty.

   The permissions on **/etc/shadow** should be **r--------**, so hackers can't view user password information.

2. Will **useradd** add new users be added to the **/etc/passwd** file, the **/etc/shadow** file, or both? Is the **passwd** command shadow aware? Try it! Use the **useradd** command to create an account for user25, then use the **passwd** command to set a null password for the user and force a password change at next login. Then check the **/etc/passwd** and **/etc/shadow** files to see what happened.

   Answer

   ```
   # useradd -m user25
   # passwd -d user25
   # passwd -f user25
   # grep user25 /etc/passwd /etc/shadow
   ```

   user25 should be included in both files now.

3. Enable shadow password aging on the user1 account.

   a. Ensure that the password is changed at least twice per year.

   b. Ensure that users wait at least one week between password changes.

   c. Provide a one-week warning before the user's password expires.

   Answer

   ```
   # passwd -x 180 -n 7 -w 7 user1
   ```


© 2004 Hewlett-Packard Development Company, L.P.

4. Before you continue on to the next part, revert to a non-shadowed password file.

Answer

```
# pwunconv
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: (Optional) Automating User Account Creation

1. Pretend for a moment that you are a system administrator at a large university. Fifty students have just enrolled to start classes, and you need to create user accounts for them. Can you write a simple shell script to automatically create the user accounts? Initially, you can assign the students null passwords, but force them to change their passwords after their first successful login.

   Hint: Try running the sample shell script below. What must be changed in the shell script to automatically create the desired accounts?

   ```
   #!/usr/bin/sh
   n=1
   while ((n<=50))
   do
      echo stud$n
      ((n=n+1))
   done
   ```

   Answer

   ```
   #!/usr/bin/sh
   n=1
   while ((n<=50))
   do
      echo stud$n
      useradd -m stud$n
      passwd  -d stud$n
      passwd  -f stud$n
      ((n=n+1))
   done
   ```


© 2004 Hewlett-Packard Development Company, L.P.

3–6. LAB: Customizing User Accounts

Directions

Part 1: Customizing Accounts with `.profile`

1. Modify the appropriate configuration file so that root will have access to command line editing at next login.

   Answer

   ```
   # vi .profile
     export EDITOR=vi
     export HISTFILE=~/.sh_history
     export HISTSIZE=50
   ```

2. Modify the appropriate configuration file so that the system administrator's shell prompt displays the present working directory and user name after the next login.

   Answer

   ```
   # vi .profile
     export PS1='$LOGNAME:$PWD #'
   ```

3. If your system is running CDE, modify root's CDE login script to ensure that the system consults your **`.profile`** at next login.

   Answer

   ```
   # vi .dtprofile
     DTSOURCEPROFILE=true
   ```

4. Linux users often use the bash shell rather than HP-UX. Try running the bash shell. What happens? Why?

   ```
   # bash
   ```

   Answer

   You should see an **`sh: bash: not found`** message since the directory containing the bash executable isn't included in the **`PATH`** variable.

5. Where is the bash shell executable on the system? How can you find out?

   Answer

   ```
   # whereis bash
   ```

   The bash shell is in **`/usr/local/bin/.`**

© 2004 Hewlett-Packard Development Company, L.P.

6. What do you need to do to ensure that you can run bash by simply typing the command name?

   Answer

   You need to add **/usr/local/bin/** to the **PATH** variable in **~/.profile**:

```
# vi ~/.profile
  PATH=$PATH:/usr/local/bin
```

7. Shell aliases are sometimes used to abbreviate commonly used commands. For instance the **alias** below automatically pipes the output from **ps -ef** to the **grep** command. Try it!

```
# alias psgrep="ps -ef | grep"
# psgrep nfs
# psgrep init
```

   Add this useful alias to your personal POSIX shell startup script. If the script doesn't exist yet, create it! Make sure that **.profile** calls your personal POSIX shell startup script by adding the following line to **.profile**: **export ENV=~/.shrc**

   Answer

```
# vi ~/.profile
  export ENV=~/.shrc
# vi ~/.shrc
  alias psgrep="ps -ef | grep"
```

8. Log out, then log back in again to see if your changes were successful.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Customizing New Accounts via `/etc/skel`

1. New users added to your system may appreciate having access to the same functionality you configured for **`root`** in Part 1 of this lab. How can you ensure that all new user accounts have the same custom functionality you configured for **`root`** in Part 1? Make it so.

   Answer

   ```
   # vi /etc/skel/.profile
     export EDITOR=vi
     export HISTFILE=~/.sh_history
     export HISTSIZE=50
     export PS1='$LOGNAME:$PWD #'
     export PATH=$PATH:/labs
     export ENV=~/.shrc
   ```

   (Note: Don't simply copy **`~root/.profile`** to the **`/etc/skel`** directory. The PATH variable in **`~root/.profile`** includes directories such as **`/sbin`** that regular users shouldn't normally access.)

   ```
   # cp ~/.shrc /etc/skel
   ```

2. **`.profile`** will be read only if your new users have a modified **`.dtprofile`**, too. How can you ensure that new home directories automatically get a copy of the modified **`.dtprofile`** that you created in Part I? Make it so.

   Answer

   ```
   # cp -p /usr/dt/config/sys.dtprofile /etc/dt/config/sys.dtprofile
   # vi /etc/dt/config/sys.dtprofile
   ```

3. Create a new user account using SAM or **`useradd`**; then log in as that new user and see if your customizations worked.

   Answer

   ```
   # useradd -m mickie
   # passwd mickie
   # login mickie
   # exit
   ```


© 2004 Hewlett-Packard Development Company, L.P.

4. Do changes to **/etc/skel** affect already-existing accounts? Try logging in as **user24** to find out. Does **user24** have the custom prompt you configured in **/etc/skel/.profile**?

   Answer

   ```
   # login user24
   # exit
   ```

   user24 was not affected by the new **/etc/skel/.profile**. The files in **/etc/skel** are copied only to new home directories.

© 2004 Hewlett-Packard Development Company, L.P.

4–5. LAB: HP-UX File System Hierarchy

Directions

Answer all the questions below.

1. Which of the following directories are dynamic?

   /etc

   /usr

   /sbin

   /dev

   /tmp

   Answer

   /etc

   /dev

   /tmp

2. Viewing a report on your disk space usage, you note that **/usr**, **/var**, and **/opt** are all nearing 90% capacity. Which of these directories should you be most concerned about? Why?

   Answer

   /var deserves the most attention here because it is a dynamic file system that could grow quite quickly in case of an error condition that creates entries in the system log files. **/usr** and **/opt** are static file systems that are less likely to cause problems.

3. Match the directory with its contents:

1. **/usr/share/man**	A. kernel, boot loader
2. **/stand**	B. system configuration files
3. **/var/adm**	C. shareable operating system commands
4. **/etc**	D. man pages
5. **/usr**	E. application directories
6. **/opt**	F. common admin files and logs

   Answer

1. **/usr/share/man**	D. man pages
2. **/stand**	A. kernel, boot loader
3. **/var/adm**	F. common admin files and logs

© 2004 Hewlett-Packard Development Company, L.P.

4. **/etc**	B. system configuration files
5. **/usr**	C. shareable operating system commands
6. **/opt**	E. application directories

4. Where would you expect to find the **cp** and **rm** OS user executables? See if you are correct.

   Answer

   Both are in **/usr/bin**, along with all the other user executables.

5. Where would you expect to find the **sam**, **useradd**, and **userdel** executables? See if you are correct.

   Answer

   All three are in **/usr/sbin** along with many other administrative utilities.

6. The **pre_init_rc** utility executes in the early stages of the system start-up procedure to check for file system corruption. Where would you expect to find this executable? See if you are correct.

   Answer

   pre_init_rc is in the **/sbin** directory, along with other files used during the boot process.

7. There is a system log file that maintains a record of system shutdowns. Where would you expect to find the shutdown log file? See if you are correct.

   Answer

   The full path name is **/var/adm/shutdownlog**.

   Most OS log files are kept in **/var/adm**.

8. In which directory would you expect to find the "hosts" configuration file, which contains network host names and addresses? See if you are correct.

   Answer

   The path name for the hosts file is **/etc/hosts**.

9. Though many utilities and daemons maintain independent log files, many daemons and services write their errors and other messages to a log file called **syslog.log**. See if you can find the path for this file, then check to see if any messages have been written to the file in the last day.

   Answer

   ```
   # more /var/adm/syslog/syslog.log
   ```


© 2004 Hewlett-Packard Development Company, L.P.

10. Find all of the directories (if any) under **/home** that are owned by root.

    Answer

    ```
    # find /home -user root
    ```

11. (Optional) Find all the files under /tmp that haven't been accessed within the last day.

    Answer

    ```
    # find /tmp -atime +1 –type f
    ```

12. (Optional) Find all the files on your system that are greater than 10000 bytes in size. If you needed to make some disk space available on your system, would it be safe to simply remove these large files?

    Answer

    ```
    # find / -size +10000c –type f
    ```


© 2004 Hewlett-Packard Development Company, L.P.

5–27. LAB: Viewing the Hardware Configuration with `ioscan`

Directions

The **`ioscan`** command is a powerful tool for exploring your system's hardware configuration.

The sample **`ioscan`** output shown below was taken from an rp8400 server. Use the sample output to answer the questions that follow.

```
# ioscan
H/W Path      Class             Description
============================================================================
0             cell
0/0              ioa               System Bus Adapter (804)
0/0/0               ba             Local PCI Bus Adapter (782)
0/0/0/0/1              tty         PCI Serial
0/0/0/1/0              lan         HP A3639-60019 1000Base-T Built-in I/O
0/0/0/2/0              ext_bus     SCSI C87x Fast Wide Single-Ended
0/0/0/2/0.6               target
0/0/0/2/0.6.0                disk  HP 36.4GST336704LC
0/0/0/2/0.7               target
0/0/0/2/0.7.0                ctl   Initiator
0/0/0/2/1              ext_bus     SCSI C87x Fast Wide Single-Ended
0/0/0/2/1.2               target
0/0/0/2/1.2.0                disk  HP DVD-ROM 305
0/0/0/2/1.7               target
0/0/0/2/1.7.0                ctl   Initiator
0/0/0/3/0              ext_bus     SCSI C896 Fast Wide Single-Ended
0/0/0/3/0.6               target
0/0/0/3/0.6.0                disk  HP 36.4GST336704LC
0/0/0/3/0.7               target
0/0/0/3/0.7.0                ctl   Initiator
0/0/0/3/1              ext_bus     SCSI C896 Fast Wide LVD
0/0/0/3/1.7               target
0/0/0/3/1.7.0                ctl   Initiator
0/0/1               ba             Local PCI Bus Adapter (782)
0/0/1/0/0              lan         HP A5230A PCI 10/100Base-TX Addon
0/0/2               ba             Local PCI Bus Adapter (782)
0/0/2/0/0              ext_bus     SCSI C87x Fast Wide Differential
0/0/2/0/0.7               target
0/0/2/0/0.7.0                ctl   Initiator
0/0/4               ba             Local PCI Bus Adapter (782)
0/0/4/0/0              ext_bus     SCSI C87x Fast Wide Differential
0/0/4/0/0.7               target
0/0/4/0/0.7.0                ctl   Initiator
0/0/4/0/1              ext_bus     SCSI C87x Fast Wide Differential
0/0/4/0/1.3               target
0/0/4/0/1.3.0                disk  HP 36.4GST336704LC
0/0/4/0/1.6               target
0/0/4/0/1.6.0                disk  HP 36.4GST336704LC
0/0/4/0/1.7               target
0/0/4/0/1.7.0                ctl   Initiator
0/0/6               ba             Local PCI Bus Adapter (782)
0/0/6/0/0              fc          HP Tachyon Fibre Channel Adapter
```


© 2004 Hewlett-Packard Development Company, L.P.

```
0/0/8                ba             Local PCI Bus Adapter (782)
0/0/8/0/0               fc          HP Tachyon Fibre Channel Adapter
0/0/10               ba             Local PCI Bus Adapter (782)
0/0/10/0/0              fc          HP Tachyon Fibre Channel Adapter
0/0/12               ba             Local PCI Bus Adapter (782)
0/0/12/0/0              fc          HP Tachyon Fibre Channel Adapter
0/0/14               ba             Local PCI Bus Adapter (782)
0/0/14/0/0              fc          HP Tachyon Fibre Channel Adapter
0/5               memory            Memory
0/10              processor         Processor
0/11              processor         Processor
0/12              processor         Processor
0/13              processor         Processor
2              cell
2/5               memory            Memory
2/10              processor         Processor
2/11              processor         Processor
2/12              processor         Processor
2/13              processor         Processor
```

1. How many cell boards are available on this system?

   Answer

   There are two cell boards: one at hardware path 0, and one at hardware path 2.

2. How many CPUs are configured on this system?

   Answer

   There are eight CPUs: four on the cell board at address 0, and four on the cell board at address 2.

3. How many LAN interfaces are available? What is the hardware path of the high-speed LAN interface?

   Answer

   There is a 1000BaseT (Gigabit) interface at address 0/0/0/1/0, and 100BaseT interface at 0/0/1/0/0.

4. When you install HP-UX from CD, you will need to know the hardware path of the system's DVD/CD drive. What is the DVD/CD hardware path on this system?

   Answer

   The hardware path appears to be 0/0/0/2/1.2.0.

© 2004 Hewlett-Packard Development Company, L.P.

5. How many SCSI interfaces are available on this system? Which card would provide best performance for an Ultra-2 SCSI LVD device?

   Answer

   There are seven SCSI interfaces on this system:

```
0/0/0/2/0 = SCSI C87x Fast Wide Single-Ended
0/0/0/2/1 = SCSI C87x Fast Wide Single-Ended
0/0/0/3/0 = SCSI C896 Fast Wide Single-Ended
0/0/0/3/1 = SCSI C896 Fast Wide LVD
0/0/2/0/0 = SCSI C87x Fast Wide Differential
0/0/4/0/0 = SCSI C87x Fast Wide Differential
0/0/4/0/1 = SCSI C87x Fast Wide Differential
```

   LVD SCSI devices should be attached to the 0/0/0/3/1 card.

6. If an administrator chose to put a tape drive on the 0/0/0/2/1 interface, which target addresses could be assigned to the drive?

   Answer

   Single-ended SCSI supports addresses ranging from 0 to 7. Address 2 has already been taken by the DVD. Address 7 has already been taken by the initiator. Thus, addresses 0-1 and 3-6 are the only available target addresses.

7. If an administrator chose to put a tape drive with target address 0 on the 0/0/0/2/1 interface card, what would the tape drive's hardware path be?

   Answer

   0/0/0/2/1.0.0

8. Assuming there are no redundant physical links on this system, if the SCSI interface card at address 0/0/4/0/1fails, will any disks be affected? If so, which ones?

   Answer

   There are two disks connected to that interface card:

   0/0/4/0/1.3.0
   0/0/4/0/1.6.0

9. How many fibre channel interfaces are available on this system? Are any of these interfaces connected to disk arrays currently?

   Answer

   There are five **`fc`** class interfaces. None of them appear to be connected to disk arrays currently.

© 2004 Hewlett-Packard Development Company, L.P.

6–4. QUESTIONS: Listing Device Files with `lssf`

The following are answers to the teaching questions in the topic:

1. Which command would you use to list device files for the tape drive at hardware address 2/0/1.0.0?

   Answer

   ```
   # ioscan -funH 2/0/1.0.0
   ```

2. Which command(s) could you use to list all of your disk device files?

   Answer

   ```
   # ioscan -funC disk

   # ll /dev/*dsk
   ```

3. Which command would you use to determine the hardware path of the device accessed by **`/dev/rmt/c0t0d0BEST`**?

   Answer

   ```
   # lssf /dev/rmt/c0t0d0BEST
   ```

4. Which command will tell you the device-specific options used by the **`/dev/rmt/c0t0d0BESTnb`** device file?

   Answer

   ```
   # lssf /dev/rmt/c0t0d0BESTnb
   ```


© 2004 Hewlett-Packard Development Company, L.P.

6–6. QUESTIONS: Device File Naming Convention

The following are answers to the teaching questions in the topic:

```
# ioscan -fun

Class       I  H/W Path    Driver      S/W State H/W Type  Description
=============================================================================
ext_bus     0  8/12        c720        CLAIMED   INTERFACE GSC Fast/Wide SCSI
disk        0  8/12.5.0    sdisk       CLAIMED   DEVICE    SEAGATE ST32171W
disk        1  8/12.6.0    sdisk       CLAIMED   DEVICE    SEAGATE ST32171W
ext_bus     1  8/16/5      c720        CLAIMED   INTERFACE Built-in SCSI
disk        2  8/16/5.3.0  sdisk       CLAIMED   DEVICE    SEAGATE ST32171W
```

1. What is the interface card instance number of the disk at 8/12.5.0?

   Answer

   Looking at the sample **ioscan** output above, disk 8/12.5.0 is attached to the interface card at 8/12. The instance number for this interface card is "0".

2. What is the target address of the disk at 8/12.5.0?

   Answer

   The target address is the second to last component of a SCSI device's hardware path. Thus, the SCSI ID appears to be "5" in this case.

3. What is the interface card instance number of the disk at 8/16/5.3.0?

   Answer

   According to the **ioscan** output, disk 8/16/5.3.0 is attached to the interface card at 8/16/5, which has been assigned instance number 1.

4. What is the target address of the disk at 8/16/5.3.0?

   Answer

   Since the SCSI target address is typically the second-to-last component of a SCSI device's hardware path, it appears as if the SCSI target address is "3" in this case.

© 2004 Hewlett-Packard Development Company, L.P.

6–7. QUESTIONS: Tape Device File Names

The following are answers to the teaching questions in the topic:

1. What would be the device file name for the disk labeled "C" on the slide?

   Answer

   Disk "C" is at target address "6" on the SCSI card with I=1. Thus, the disk may be referenced as either **`/dev/dsk/c1t6d0`** or **`/dev/rdsk/c1t6d0`**.

2. What would be the device file name for the CD-ROM labeled "D" on the slide?

   Answer

   CD-ROM "D" is at target address "4" on the SCSI card with I=1. Thus, the CD-ROM may be referenced as either **`/dev/dsk/c1t4d0`** or **`/dev/rdsk/c1t4d0`**.

© 2004 Hewlett-Packard Development Company, L.P.

6–8. QUESTIONS: Tape Device File Names

The following are answers to the teaching questions in the topic:

1. What device file would you use to access tape "B" in the diagram on the slide using the best available density and compression features?

   Answer

   Tape "B" is at target address "0" on the SCSI card with I=1. Thus, to access the tape drive using the "best" density, one would use the **`/dev/rmt/c1t0d0BEST`** device file.

2. What device file would you use to access tape "B" in the diagram on the slide using the best available density and compression features, and with the auto-rewind feature disabled?

   Answer

   Tape "B" is at target address "0" on the SCSI card with I=1. Thus, to access the tape drive using the "best" density available with the autorewind feature disabled, one would use the **`/dev/rmt/c1t0d0BESTn`** device file.

© 2004 Hewlett-Packard Development Company, L.P.

6–17. LAB: Device Files

Directions

Part 1: Viewing and Interpreting Device Files

1. Use **ioscan** to find the names of the device files for all of your disk class devices.

   Answer

   ```
   # ioscan -fnC disk
   ```

2. Use **ioscan** to find the names of the device files for your system's LAN card(s).

   Answer

   ```
   # ioscan -fnC lan
   ```

3. Choose one of the disk device files in your **/dev/dsk** directory. Execute the command that lists the characteristics of this device file. What kernel driver is associated with this device file? What is the hardware path of the disk associated with this device file?

   Answer

   ```
   # lssf /dev/dsk/cxtxdx
   ```

4. How does **lssf** know which kernel driver is associated with each device file?

   Answer

   lssf determines the driver associated with a device file by examining the device's major number. **lssf** determines the hardware path of the device file by examining the device file's minor number.

5. (Optional) If you have a tape drive on your system, run **lssf /dev/rmt/***. Which device file(s) access your tape drive with the no-rewind feature?

   Answer

   ```
   # lssf /dev/rmt/*
   ```

   Note: File name typically ends in **n** or **nb**

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Creating Device Files with SAM

You should have at least two available serial ports on your machine. In this portion of the lab, you will create device files to support a modem on one of your serial ports, and a hard-wired ASCII terminal on the other. SAM provides a simple interface for creating device files for both of these devices.

1. Go to **`SAM --> Peripheral Devices --> Terminals and Modems`**. Are there any modems or terminals currently?

2. First, configure a modem on your first available serial port.

   a) Choose **`Actions --> Add Modem`**.
   b) Choose a serial port/MUX interface hardware path.
   c) Click Port Number to choose an available MUX port (Choose 0 on workstations).
   d) Choose any baud rate. (It doesn't really matter since we don't have a real modem.)
   e) Choose to support both incoming and outgoing calls.
   f) Click OK.

   Which device files does SAM create for you?

   Answer

   SAM should have created three device files:

   ```
   /dev/cul0p0
   /dev/ttyd0p0
   /dev/cua0p0
   ```

3. Now configure a hard-wired terminal on another available serial port.

   a) Choose **`Actions --> Add Terminal`**.
   b) Choose a serial port/MUX hardware path.
   c) Click Port Number to choose an available MUX port.
   d) Choose "H" as the baud rate.
   e) Click OK.

   Which device files does SAM create for you?

   Answer

   SAM should create a single device file for you: **`/dev/tty1p0`** (The actual device file name will vary, depending on the hardware path and port number you chose for your terminal.)

© 2004 Hewlett-Packard Development Company, L.P.

4. (Optional) If you have a tape drive on your machine, use SAM to create a non-standard device file for it. Specify DDS1 density, no compression, and Berkeley-style semantics. Let the other options default.

   This DDS1 device file may be useful if you create tapes that may need to be read on older tape drives that don't support the high-density format used by "BEST" device files.

   What device file does SAM create?

   Answer

   - Go to **`SAM --> Peripheral Devices --> Tape Drives`**.
   - Select a tape drive.
   - Choose **`Actions --> Create Device Files --> Create Custom Device File`**.
   - Click on the Density button and select DDS1.
   - Click 'Berkeley' Semantics
   - Uncheck 'Compressed Mode'
   - Take the defaults for the rest of the dialog box.
   - SAM should create a device file called **`/dev/rmt/c0t0d0DDS1b`**.
   - Click OK.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Creating Device Files with `insf`

Though SAM must be used when you want to create serial-device device files or non-standard device files, most device files can be created automatically with **`insf`**. In this exercise you will have a chance to try using **`insf`**.

1. Remove all of the device files for one of the non-vg00 disks on your system.

   ```
   # rmsf -a /dev/dsk/cxtxdx
   ```

2. Run **`ioscan -funC disk`**. Do you see the disk's device files? Do you see the disk's hardware path?

   ```
   # ioscan –funC disk
   ```

   Answer

   Since we executed **`rmsf`** with the **`-a`** option, you should no longer see the disk's hardware path or device file.

3. Force the system to recognize the disk by running **`ioscan`** without the –u option. Does this recreate the device files?

   ```
   # ioscan –fnC disk
   ```

   Answer

   The hardware path should appear, but the device file wasn't recreated. Device file auto-configuration only occurs at boot time.

4. Recreate the disk device file with **`insf -C disk`**.

   ```
   # insf -C disk
   ```

5. Verify that the device file was recreated.

   ```
   # ioscan -funC disk
   ```

   Answer

   `ioscan` should show that your disk device files were recreated.

© 2004 Hewlett-Packard Development Company, L.P.

7–4. SLIDE: LVM Device Files

The following are answers to the teaching questions in the topic:

1. What directory would contain the logical volumes' device files?

   Answer

   All device files for logical volumes in **vg02** should be **/dev/vg02**.

2. What would be the name of the volume group's device file?

   Answer

   The volume group's device file name should be **/dev/vg02/group**.

3. What would be the name of the first logical volume's device file?

   Answer

   The first logical volume's raw device file would be **/dev/vg02/rlvol1**.

4. Overall how many device files should you find overall in **/dev/vg02**?

   Answer

   There should be three raw logical volume device files, three block logical volume device files, and the "group" device file for the volume group as a whole.

5. What would be the minor number of the third logical volume's device file?

   Answer

   The minor number for the third logical volume in **vg02** would be **0x020003**.

© 2004 Hewlett-Packard Development Company, L.P.

7–5. SLIDE: LVM Extents

The following are answers to the teaching questions in the topic:

1. Assuming a volume group uses the default extent size, how many logical extents would you expect to find in an 88-MB logical volume?

   Answer

   Since the default extent size is 4 MB, there should be 22 extents in an 88-MB logical volume.

2. Assuming a volume group uses the default extent size, how many physical extents would you expect to find on a 400-MB physical volume?

   Answer

   Since the default size is 4 MB, a 400-MB disk should have 100 physical extents.

© 2004 Hewlett-Packard Development Company, L.P.

7–10. LAB: Logical Volume Manager

Directions

In this exercise you will have an opportunity to create several physical volumes, volume groups, and logical volumes. Except where noted, do all of the exercises from the command line. You will have a chance to create some logical volumes via SAM in a later chapter.

Part 1: Choosing a Disk for Use in a Volume Group

1. How many disks do you have on your system? Determine each disk's hardware path and device file names.

   Answer

   ```
   # ioscan -funC disk
   ```

2. Use **vgdisplay** to determine which of the disks on your system are already members of active volume groups. You should have at least one disk that isn't currently in a volume group. Note the free disk's device file name; you will create a new volume group using your free disk in the next part of the lab exercise.

   Answer

   The free disk's hardware path and device file name will vary from system to system. For the sake of the solution that follows, we will assume that the device file name is **/dev/dsk/*cxtxdx***. The **vgdisplay** command will show you which disks are already members of active volume groups on your system. **# vgdisplay –v**

3. Before adding a disk to a volume group, you may want to check the size of the disk. This is accomplished via the **diskinfo** command. How large is your spare disk?

   ```
   # diskinfo /dev/rdsk/cxtxdx
   ```

   Answer

   ```
   # diskinfo /dev/rdsk/cxtxdx
   ```

   The disk size will vary

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Creating a Physical Volume, Volume Group, and Logical Volumes

1. Configure your free disk as an LVM physical volume.

   Answer

   ```
   # pvcreate /dev/rdsk/cxtxdx
   ```

2. Can you **pvdisplay** your disk at this point? Try it.

   Answer

   ```
   # pvdisplay /dev/dsk/cxtxdx
   ```

   Fails. You can't **pvdisplay** a disk until it is a member of a volume group.

3. Create a new **vg01** volume group using your newly created physical volume.

   Answer

   ```
   # mkdir /dev/vg01
   # mknod /dev/vg01/group c 64 0x010000
   # vgcreate vg01 /dev/dsk/cxtxdx
   ```

4. Use **vgdisplay** and **pvdisplay** to check the status of your new physical volume and volume group. How many physical volumes are in the volume group at this point? How many logical volumes are in the volume group at this point? What is the extent size?

   Answer

   ```
   # vgdisplay -v vg01 | more
   # pvdisplay /dev/dsk/cxtxdx
   ```

   Currently there should be just one PV in the volume group, and no LVs. The PE size should be 4 MB (the default).

5. Create two 24-MB logical volumes in your new volume group. Name the first logical volume **cad** and the second **cam**.

   Answer

   ```
   # lvcreate -L 24 -n cad vg01
   # lvcreate -L 24 -n cam vg01
   ```


© 2004 Hewlett-Packard Development Company, L.P.

6. Use **vgdisplay** and **lvdisplay** to ensure that your new logical volumes were actually created.

   Answer

   ```
   # vgdisplay -v | more
   # lvdisplay -v /dev/vg01/cad
   # lvdisplay -v /dev/vg01/cam
   ```

7. Do an **ll** of the **/dev/vg01** directory. What is the name of the volume group device file for your new volume group? Each of your logical volumes should have two device files. Why?

   Answer

   ```
   # ll /dev/vg01
   ```

   The volume group device file should be called **/dev/vg01/group**. Each logical volume requires both a raw and a block device file. Some commands used to access logical volumes require a block device file, while others require a character device file. Both device file types are required for every device file.

8. Use SAM to remove your **vg01** volume group in preparation for the next portion of this lab. Volume groups can be managed in SAM from the Volume Groups object list:

   ```
   SAM --> Disks and File Systems --> Volume Groups
   ```

9. Try a **vgdisplay** back at the command line to ensure the volume group is gone. Did SAM remove the volume group device files, too?

   Answer

   ```
   # vgdisplay -v vg01
   ```

   The volume group is not found. In fact, SAM even removed the volume group device files.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: (Optional) LVM "What-Ifs"

1. What happens if you **pvcreate** a disk that is already in use by another active volume group? Try it. See what happens if you **pvcreate** your boot disk in **vg00**. Did this work? Try again using the **-f** "force" option on **pvcreate**. What is the result?

   Answer

   ```
   # vgdisplay -v vg00   (Determine the name of your boot disk.)
   # pvcreate/dev/rdsk/cytydy        (Use your boot disk here)
   # pvcreate –f /dev/rdsk/cytydy   (Use your boot disk here)
   ```

   Both **pvcreates** should give you error messages. LVM won't let you **pvcreate** a disk that's already claimed by another active volume group, even with **-f**.

2. Re-**pvcreate** the spare disk you used in the previous part of the lab. (This **pvcreate** should work.)

   Answer

   ```
   # pvcreate /dev/rdsk/cxtxdx
   ```

3. For the sake of variety, create a new volume group called **vg02** using the spare disk you just **pvcreate**d.

   Answer

   ```
   # mkdir /dev/vg02
   # mknod /dev/vg02/group c 64 0x020000
   # vgcreate vg02 /dev/dsk/cxtxdx
   # pvdisplay /dev/dsk/cxtxdx
   # vgdisplay -v vg02
   ```

4. Now that you have a volume group, try creating a few logical volumes. Create a logical volume called **test1** in **vg02**. This time, though, don't specify a size for your logical volume. Based on the result of this experiment, what is the default logical volume size?

   ```
   # lvcreate -n test1 vg02
   # vgdisplay -v vg02
   ```

   Answer

   The new logical volume is created with size 0 MB. The default logical volume size appears to be 0 MB.

© 2004 Hewlett-Packard Development Company, L.P.

5. What happens if you don't specify a logical volume name when you **lvcreate**? Try it. Create two new logical volumes of size 12 MB and 16 MB, leaving off the **-n** option in both cases. What names did LVM assign to your new logical volumes? Why?

   Answer

   ```
   # lvcreate -L 12 vg02
   # lvcreate -L 16 vg02
   # vgdisplay -v vg02
   ```

   By default, LVM uses the following naming convention for new logical volumes: **lvol1**, **lvol2, lvol3,...**

   In this case, since these were the second and third logical volumes in **vg02**, LVM named them **lvol2** and **lvol3**. The number after **lvol** should match the last couple of digits of the logical volume device file's minor number.

6. See what happens if you attempt to create an 11 MB logical volume in **vg02** called test2. Watch the output from **lvcreate** carefully. What size is your new logical volume? Explain.

   Answer

   ```
   # lvcreate -L 11 -n test2 vg02
   # vgdisplay -v vg02
   ```

   The new logical volume is 12 MB, not 11 MB. Space is allocated to logical volumes in units of extents. If you choose a size that is not a multiple of the extent size, LVM rounds up to the nearest extent boundary.

7. At some point in your UNIX career, you will almost certainly use **-l** instead of **-L** on **lvcreate** accidentally. What appears to be the difference between these two options? Try it and find out.

   ```
   # lvcreate -l 12 -n test3 vg02
   # lvcreate -L 12 -n test4 vg02
   # vgdisplay -v vg02 | more
   ```

   Answer

   The **-L** option defines a logical volume's size in megabytes, while the **-l** option defines a logical volume's size in extents. Thus, using **lvcreate -l** 12 results in a logical volume that is 48 MB instead of 12 MB.

8. Once again, use SAM to remove the volume group you just created.

   ```
   SAM --> Disks and File Systems --> Volume Groups
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: (Optional) Exploring VxVM

As of HP-UX 11i, there is an alternative to the Whole Disk and LVM disk management solutions: VxVM. This optional portion of the lab gives you an opportunity to explore some of the basic VxVM functionality.

1. VxVM refuses to manage disks that contain LVM header information. Use the **dd** command to clobber any remnants of the LVM headers on your spare disk. Make sure you choose the right disk! If you accidentally overwrite the LVM headers on your boot disk, your system may be left in an unbootable state!

   ```
   # dd if=/dev/zero of=/dev/rdsk/cxtxdx bs=1024 count=1024
   ```
   Use your spare disk's device filename!

2. Verify that VxVM is installed on your system.

   ```
   # swlist Base-VXVM
   ```

3. Run the **vxinstall** program to initialize VxVM. You only need to run this utility once, when you first install VxVM.

   ```
   # vxinstall
   ```

4. When asked if you wish to enter a license key, answer “n”. We will only use the “base” VxVM functionality, which doesn’t require a license.

   ```
   Are you prepared to enter a license key [y,n,q,?] n
   ```

5. When asked if you wish to use enclosure based names for your disks, answer “n”. Enclosure-based names are most useful in disk array and SAN environments.

   ```
   Do you want to use enclosure based names for all disks ? n
   ```

6. The program will interactively walk you through a series of menus. Whenever **vxinstall** asks if you wish to continue, press Press **[Return]**. If you make a mistake, you can press **^C** to break out and start over again at any point.

   a) The program will search for and report your system's disk enclosures and/or controller cards. Press **[Return]**.

   b) Next, **vxinstall** asks how you wish to configure the disks on your system. You can choose either a "Quick Installation" (in which VxVM initializes all currently unused disks with little interaction from the administrator) or "Custom Installation" (in which you get to choose which disks should and shouldn't be configured). Choose "Custom Installation".

   c) If you chose "Custom Installation", **vxinstall** should list all of the currently unused disks on the system, and ask if you wish to initialize them. You can initialize selected disks, or all of the currently unused disks. For the sake of this exercise, choose "Install all disks as new disks".

© 2004 Hewlett-Packard Development Company, L.P.

d) Finally, VxVM gives you an opportunity to assign a meaningful name to each disk. These names will be used in place of device file names throughout the menus that follow. Accept the default disk names for the selected disks.

```
Use default disk names for these disks? [y,n,q,?] y
```

e) When asked to confirm that you wish to destroy the selected disks, answer "y".

```
Are you sure you want to destroy the contents of this disk? y
```

f) Proceed through the remaining confirmation screens.

7. VxVM can be managed from the command line, or via the "Veritas Enterprise Administrator" (**vea**) GUI-based management tool. In this lab, we will use **vea**. VEA is a client/server application. After you initially configure VxVM with **vxinstall**, it is necessary to stop and restart the VEA server daemon, **vxsvc**.

```
# /opt/VRTS/bin/vxsvc -k
# /opt/VRTS/bin/vxsvc
```

It may take several minutes for the daemon to fully launch. You can monitor its progress via the **vxsvc -m** command. When **vxsvc -m** reports that the daemon is Running, proceed on to the next step.

```
# /opt/VRTS/bin/vxsvc -m
Current state of server: RUNNING
```

8. Now launch the VEA client GUI, **vea**.

```
# /opt/VRTSob/bin/vea
```

9. On the VEA Connection screen, enter hostname **localhost**, and the root username and password. Click [OK]. Be patient! It may take several minutes for VEA to download all the necessary plugins.

10. The left pane displays a hierarchical tree of VxVM objects. Objects can be expanded and collapsed by clicking the "**+**" and "**-**" icons.

a) On the left pane, click the "**+**" sign to the left of **localhost** to expand the list of VxVM objects on your localhost.

b) You should now see a list of VxVM objects available on localhost such as **Controllers**, **Disk Groups**, and **Volumes**. Some of these objects can be further expanded. Click on the "**+**" sign to the left of **Disk Groups**.

11. Clicking on an object in the left pane (not the "**+**" sign preceding the object) displays that object's properties in the right pane.

a) Click on **Controllers** to view a list of disk controllers.

b) Click on **Disks** to view a list of disks on your system.

© 2004 Hewlett-Packard Development Company, L.P.

c) Click on **File Systems** to view a list of file systems.

d) Click on **Disk Groups** to view a list of VxVM disk groups. Disk groups are conceptually very similar to LVM volume groups. The **vxinstall** program should have automatically created a disk group for you called **rootdg**.

12. The **vea** interface may be used to view your VxVM configuration, but can also be used to modify your configuration. Let's create a VxVM volume in the **rootdg** disk group. VxVM volumes are conceptually very similar to LVM logical volumes." At this point, you shouldn't have any VxVM volumes configured.

a) To configure a VxVM volume, left-click on Disk Groups in the left pane to display a list of available disk groups in right pane.

b) In the right pane, right click on **rootdg**. A menu of available actions should appear. Choose **New Volume ...**.

c) In the **New Volume** wizard dialog, click **Next>** to proceed to the **Select attributes** screen.

d) On the **Select attributes** screen, enter the following information:

Disk Group Name:	**rootdg**
Volume Name:	**data**
Comment:	**Contains application data**
Size:	**12MB**

Accept the defaults for all other fields.

e) Click **Next>** to proceed to the **Select disks** screen.

f) On the **Select disks** screen, click the **Let Volume Manager decide what disks to use** radio button.

g) Click **Next>** to proceed to the **Create file system** screen.

h) On the **Create file system** screen, click the "Create a file system" radio button, then enter the following information:

File system type: VxFS
Mount point: **/data**

i) Accept the defaults for the rest of the screen, then click **Next>** to proceed to the final confirmation screen.

j) Click **Finish** to create the volume and file system.

k) Click the **Volumes** object in the left pane to display a list of volumes in the right pane. Verify that your volume was created properly.

© 2004 Hewlett-Packard Development Company, L.P.

13. Verify that your file system was created properly. In the left pane, left-click **File Systems** to display a list of file systems in the right pane.

14. VxVM makes it very easy to make your volumes and file systems larger and smaller, too.

    a) In the left pane, left-click **File Systems** to display a list of file systems in the right pane.

    b) In the right pane, right click on **/data**, and select **Resize File System** from the resulting menu.

    c) Set the **New Size** to 24 MB.

    d) Click **OK**.

    e) Use the same procedure to resize the Volume back to 12M.

15. Can you guess what procedure is required to remove a Volume?

    a) In the left pane, left-click **Volumes** to display a list of volumes in the right pane.

    b) In the right pane, right click on the **data** volume, and select **Delete Volume** from the resulting menu.

    c) Confirm that you desire to delete the volume and file system.

16. If time permits, explore some of the other **vea** menus on your own.

17. If you create any additional volumes, remove them before proceeding.

18. Choose the **Exit** option from the **File** menu to exit out of **vea**.

19. Some disks on your system may be managed by VxVM, while others may be managed by LVM. However, no disk on the system can be managed by both VxVM and LVM concurrently. Your student workstation probably only has one disk, which will be needed for the last part of this lab exercise. Before proceeding, clobber the VxVM headers on your spare disk so it can again be used by LVM.

    Normally, **vea** makes it very easy to remove disks from VxVM disk groups. However, after running **vxinstall**, a minimum of one disk must remain under the control of VxVM. The only way to reclaim your spare disk for use in the LVM labs that remain in the next few modules is to resort to the command line. Replace ***cxtxdx*** with the device name of the disk in your **rootdg**.

    ```
    # vxconfigd -k -r reset -m disable
    # vxdg init rootdg
    # vxdisk rm cxtxdx
    # vxdctl stop
    # dd if=/dev/zero of=/dev/rdsk/cxtxdx bs=1024 count=1024
    ```


© 2004 Hewlett-Packard Development Company, L.P.

NOTE:	This lab only demonstrated VxVM's basic functionality. For more information on VxVM, attend one of HP's VxVM training courses, or take a look at the Veritas Volume Manager documentation on **`http://docs.hp.com`**.

© 2004 Hewlett-Packard Development Company, L.P.

Part 5: Creating One Last Volume Group

1. In preparation for the discussion of file systems and swap in the following chapters, create a volume group **vg01** with your spare disk. Create three 12 MB logical volumes in **vg01** called **data**, **app**, and **tables**.

   Answer

```
# pvcreate /dev/rdsk/cxtxdx (device file names may vary)
# mkdir /dev/vg01
# mknod /dev/vg01/group c 64 0x010000
# vgcreate vg01 /dev/dsk/cxtxdx
# lvcreate -L 12 -n data vg01
# lvcreate -L 12 -n app vg01
# lvcreate -L 12 -n tables vg01
```


© 2004 Hewlett-Packard Development Company, L.P.

8–5. SLIDE: HP-UX Hard Links

The following are answers to the teaching questions in the topic:

1. Why isn't it possible to link across file system boundaries?

   Answer

   Each file system has a separate inode table. Thus, each file system has a separate inode #1, #2, #3, and so on. If HP-UX allowed hard links across file system boundaries, a link to inode #1234, for example, would be ambiguous. Is that a link to #1234 in the **`/tmp`** file system, or #1234 in the **`/home`** file system? To avoid this ambiguity, HP-UX doesn't allow hard links across file system boundaries.

2. How would **`/mydir/file1`** be affected if you were to remove **`/mydir/f1`**?

   Answer

   Try it! As it turns out, an **`ll`** shows that the link count in the inode is decremented by one each time a link to a file is removed. However, the data remains in tact on disk until the link count reaches "0". Thus, in this case, removing **`/mydir/f1`** has little effect on **`/mydir/file1`**.

© 2004 Hewlett-Packard Development Company, L.P.

8–6. SLIDE: HP-UX Symbolic Links

The following are answers to the teaching questions in the topic:

1. Why is it possible to create symbolic links, but not hard links, across file system boundaries?

Answer

A single inode number (such as #1234) may represent a different file in each file system. Since hard links are based on inode numbers, linking across file systems would lead to ambiguous references.

Symbolic links, however, link by pathname rather than inode number. Since each pathname identifies a unique file/directory on a system-basis rather than a file-system basis, HP-UX *does* allow symbolic links across file system boundaries.

© 2004 Hewlett-Packard Development Company, L.P.

8–9. SLIDE: JFS Structural Overview

The following are answers to the teaching questions in the topic:

1. What metadata structures does JFS have in common with HFS?

   Answer

   Comparing slides 9:7 and 9:12, your students should note that JFS and HFS both have inodes, data blocks, free space maps, and a superblock. You might also note that the JFS allocation unit is conceptually similar to an HFS cylinder group.

2. What metadata structures and features distinguish JFS from HFS?

   Answer

   Unlike HFS, JFS has an intent log that it uses to record its "intent" to change file system metadata. Some students may also note that JFS allocates and accesses files in units of blocks and extents, rather than in units fragments and blocks (More about this on slide 9:11).

© 2004 Hewlett-Packard Development Company, L.P.

9–9. LAB: Creating File Systems

Directions

Record the commands used to complete the tasks below, and answer all of the questions.

Part 1: Preliminary Steps

1. The exercises in this lab assume that you already have the following three logical volumes:

   ```
   /dev/vg01/data
   /dev/vg01/app
   /dev/vg01/tables
   ```

   If you already have these logical volumes, you can skip ahead to Part 2 of the lab. Otherwise, create the **`vg01`** volume group and all three of the logical volumes listed above. Make each of the logical volumes 12 MB.

   Answer

   ```
   # pvcreate /dev/rdsk/c0t5d0
   # mkdir /dev/vg01
   # mknod /dev/vg01/group c 64 0x010000
   # vgcreate vg01 /dev/dsk/c0t5d0
   # lvcreate -L 12 -n data vg01
   # lvcreate -L 12 -n app vg01
   # lvcreate -L 12 -n tables vg01
   # vgdisplay -v vg01
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Creating File Systems from the Command Line

1. Create an HFS file system in the **data** logical volume. Create a JFS file system in the **app** logical volume. Don't mount your file systems yet.

   Answer

   ```
   # newfs -F hfs /dev/vg01/rdata
   # newfs -F vxfs /dev/vg01/rapp
   ```

2. Do a **mount -v**. Why don't your new file systems appear at this point?

   Answer

   ```
   # mount –v
   ```

   The new file systems don't appear in the **mount -v** output because we haven't mounted them yet.

3. Create a mount point for each of your new file systems. Use **/data** as the mount point for the file system in the **data** logical volume, and **/app** as the mount point for the app logical volume. Again, don't mount your file systems, yet.

   Answer

   ```
   # mkdir /data
   # mkdir /app
   ```

4. Add your new file systems to the **/etc/fstab** file so they will be mounted automatically at each system boot. Again, do not mount your file systems yet.

   Answer

   ```
   # vi /etc/fstab
   /dev/vg01/data /data hfs  defaults 0 2
   /dev/vg01/app  /app  vxfs defaults 0 2
   ```


© 2004 Hewlett-Packard Development Company, L.P.

5. Go ahead and mount your file systems now by doing a **mount -a**. Watch the resulting messages carefully. You should see several error messages indicating that **/dev/vg00/lvol1** and several other file systems are already mounted. Do the **mount -a** output messages offer any indication that your new file systems were successfully mounted?

   Answer

   ```
   # mount –a
   ```

   The output from **mount -a** notes that several other file systems are "already mounted", but there is no mention of the two new file systems. Often times in UNIX, in the absence of an error message, you may assume that a command has succeeded. **mount -a** exemplifies this philosophy. Because **mount -a** didn't complain about your new file systems, you can assume that they mounted successfully.

6. Execute **mount -a** a second time and note the output messages again. Why did **mount -a** mention your new file systems in its output this time, but not when you did a **mount -a** in the previous exercise?

   Answer

   In question 5, **mount -a** successfully mounted the new file systems as they weren't yet mounted. Executing **mount -a** a second time, however, generates an error message because the new file systems have already been mounted.

7. Use **mount -v** to see what file systems are now mounted. Did your new file systems mount properly? What other information can you glean from the **mount -v** output about your mounted file systems? List three fields presented in the **mount -v** output.

   Answer

   Your file systems should all be mounted at this point.

   mount -v presents several fields of information including:

   - Device name
   - Mount point
   - File system type
   - Mount options
   - Mount time

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Experimenting with `mount` and `umount`

1. The exercises in this section will be more meaningful if you have some real files in the **/data** and **/app** file systems. Towards that end, create a few files in these file systems using the following commands:

```
# cd /data; touch d1 d2 d3
# cd /app;  touch a1 a2 a3
# ls /data /app
# cd /
```

NOTE: You may notice a **lost+found** directory in your new file systems. **newfs** creates this directory for you automatically. This special directory serves as a home for files that are victims of file system corruption. This directory will be discussed in some detail in a later chapter.

---

2. Can you access files in a file system that is unmounted? Try an experiment to find out: Unmount **/data**, then do an **ls** of **/data**. Does the mount point still exist? Can you access **/data/d1** and **/data/d2**? Why or why not? Do whatever is necessary to regain access to **/data/d1** and **/data/d2**.

   Answer

```
# umount /data
# ls /data
# mount –a
```

   After unmounting the file system, the mount point is still there, but the files in the file system are no longer visible under the mount point. The files in the file system cannot be accessed again until the file system is remounted.

3. Can you unmount a file system that is still in use? Try an experiment to find out: Open a second window on your screen. In this second window, **cd** to the **/data** file system. Back in your original window try unmounting the **/data** file system. What message do you get? Why?

   Answer

   In a second window:

```
# cd /data
```

   In your original window:

```
# umount /data
```

   The umount fails. The **/data** file system cannot be unmounted until the processes using **/data** die.

© 2004 Hewlett-Packard Development Company, L.P.

4. Before you can unmount a file system, you will have to kill all processes accessing the file system. HP-UX provides a command to solve this very problem. Try the following command:

```
# fuser -u /dev/vg01/data
```

Each entry in the **fuser** output lists the PID of a process accessing the file system, a single letter code indicating how the process is using the file system ("c" indicates that a user has changed to a directory in the file system), and the name of the user that owns the offending process. **Fuser** can also kill all of the offending processes:

```
# fuser -ku /dev/vg01/data
```

The effect of this command should be pretty dramatic. What happens? Now try unmounting **/data**.

Answer

```
# fuser -u /dev/vg01/data      # lists at least one process using the LV
# fuser -ku /dev/vg01/data     # kills the shell/window that was using /data
# umount /data                 # should work now.
```

5. What happens if you mount a file system on a directory that already contains files? Try an experiment to find out:

While the **/dev/vg01/data** file system is still unmounted, touch a few files under the **/data** mount point:

```
# touch /data/junk1 /data/junk2
# ls /data
```

Now mount the **/dev/vg01/data** file system and list the contents of **/data** again:

```
# mount /data
# ls /data
```

Do you still see **junk1** and **junk2**? Unmount **/data** again and check to see if **junk1** and **junk2** still exist. Why are **junk1** and **junk2** hidden while the data file system is mounted? Explain.

NOTE: File systems should always be mounted on empty mount point directories.

Answer

While **/data** is mounted, **junk1** and **junk2** are hidden.

When **/data** is unmounted, **junk1** and **junk2** become visible again.

© 2004 Hewlett-Packard Development Company, L.P.

6. Remove the junk files you created in the previous exercise, and remount all of your file systems again.

   Answer

   ```
   rm /data/junk*
   mount –a
   ```

7. One last experiment: Can you unmount all of your file systems? Try a **`umount -a`** and explain the result.

   Answer

   ```
   # umount –a
   ```

   Though a couple of your file systems may successfully unmount, most will fail because they are in use by the system daemons.

8. Remount all of your file systems before continuing.

   Answer

   ```
   # mount -a
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Experimenting with `newfs`

1. HP-UX supports two different types of disk-based file systems: HFS and JFS. In all of the examples you have tried so far, you have specified your desired file system type via the **`-F`** option. What happens if you forget the **`-F`**? Try it: do a **`newfs`** on the **`/dev/vg01/tables`** logical volume, but leave off the **`-F`** option. Look at the **`newfs`** output carefully. How does the system determine your default file system type? What is your default file system type?

   Answer

   ```
   # newfs /dev/vg01/rtables
   ```

   The default file system type is defined in **`/etc/default/fs`**. At version 11.x, **`vxfs`** is usually the default file system type.

2. The **`newfs`** command is a powerful tool for creating file systems, but it must also be treated with some respect. What happens if you accidentally **`newfs`** a logical volume or disk that already contains a file system? Try an experiment to find out: What happens if you attempt to create a new **`hfs`** file system on the data logical volume with the following command:

   ```
   # newfs -F hfs /dev/vg01/rdata
   ```

   Answer

   It fails. **`newfs`** will not overwrite a mounted file system.

3. What happens if you try to **`newfs`** a logical volume that contains an unmounted file system? Try it.

   ```
   # umount /data
   # newfs -F hfs /dev/vg01/rdata
   # mount /data
   # ls /data
   ```

   Did you encounter any problems? What happened to the d1, d2, d3 files you created earlier? Always use **`newfs`** with care.

   Answer

   This sequence of commands should succeed without any problems. **`newfs`** *will* overwrite a file system in a logical volume if the file system is unmounted.

© 2004 Hewlett-Packard Development Company, L.P.

Part 5: Using SAM to Create Logical Volumes and File Systems

1. So far, you have created several file systems from the command line. File systems can be created and managed within SAM as well. Open SAM, then do the following:

   a. Select: **SAM --> Disks and file systems --> Logical volumes**
   b. Select: **Actions --> Create**
   c. Click: **Select a Volume Group** (Choose **vg01**)
   d. Click: **Define New Logical Volumes**
   e. Fill in the blanks in the dialog box that appears:

      LV name: **app2**
      LV size (MB): 12
      Usage: File system
      Mount Dir: **/app2**

   f. Click: **Modify FS defaults**.
   g. Fill in the blanks in the dialog box that appears:

      i) Create a Journaled file system.
      ii) Choose **Now** AND **Every system boot** options under **When to mount**.
      iii) For now, take the defaults for the rest of the screen, and click **OK**.

   h. You will be returned to the **Define new logical volumes** screen.
   i. Click **Add** to add your new logical volume to the list of new LVs to create.
   j. Click **OK** to go back to the **Create new logical volumes** step menu.
   k. Click **OK** to create the new logical volume.

2. Do a **mount -v** from the command line. Is the new file system mounted?

   Answer

   Yes. The file system should be mounted.

3. Look at **/etc/fstab**. Did SAM ensure that the new file system will be mounted with every system boot?

   Answer

   Yes. SAM automatically added the file system to **fstab**.

© 2004 Hewlett-Packard Development Company, L.P.

Part 6: Cleaning Up

1. Before moving on to the next chapter, use SAM to remove the file systems and logical volumes you created during this lab exercise:
   a. Select: SAM –> Disks and file systems –> Logical volumes
   b. While holding down CTRL, select each of the vg01 LVs. Do not remove any of the logical volumes in vg00.
   c. Select: Actions –> Remove
   d. When asked for confirmation, Answer Yes.
2. What did SAM do on your behalf?

   Are the file systems unmounted?

   Are the file systems still listed in **`/etc/fstab`**?

   If you do a **`vgdisplay`**, do the logical volumes still appear?

   Do the mount points still exist?

   Answer

   SAM unmounted the file systems, removed them from **`/etc/fstab`**, and removed the logical volumes. The mount points, however, remain in tact.

3. You should also remove volume group vg01 before continuing on to the next chapter. Again, use SAM:

   Select: **`SAM --> Disks and file systems --> Volume groups`**

   Select **`vg01`**

   Select: **`Actions --> Remove`**

© 2004 Hewlett-Packard Development Company, L.P.

10–8. LAB: File System Management

Directions

Perform the tasks suggested below. Record the commands you use and the answers to all of the questions.

Part 1: Purging and Extending File Systems

1. Occasionally, the **/tmp** file system fills up, causing problems on a system. Using SAM or the command line, list all of the files in **/tmp** that haven't been accessed within the last 2 days.

   Answer

   ```
   # find /tmp -atime +2
   ```

   or

   ```
   SAM --> Routine Tasks --> Selective File Removal
   ```

2. **/var** is another file system that sometimes reaches 100%. Often this is a result of log files that haven't been properly trimmed. Trim the following log files back to size 0:

   ```
   /var/adm/btmp
   /var/adm/wtmp
   ```

   Answer

   ```
   # >/var/adm/btmp
   # >/var/adm/wtmp
   ```

3. The questions that follow give you an opportunity to fix a full file system. You should have a script on your system called **/labs/fillfs.sh**. The **fillfs.sh** script will fill one of your file systems to capacity. Run **fillfs.sh**. As the script executes, you may see a number of file system full messages scroll across your screen. Don't worry -- yet!

   Answer

   ```
   # /labs/fillfs.sh
   ```

4. Which file system appears to be full? How many KB were allocated for this file system? What percent of the space in that file system is in use?

   Answer

   ```
   # bdf
   ```

   bdf will show the status of each of your file systems. The **/home** file system should be nearly 100% full.

© 2004 Hewlett-Packard Development Company, L.P.

5. What happens at this point if a user tries to copy a file to the full file system? Is anything recorded in **syslog.log**? Copy a large file (such as **/stand/vmunix**) to **/home** to find out.

   Answer

   ```
   # cp /stand/vmunix /home
   # tail /var/adm/syslog/syslog.log
   ```

   The **cp** command will generate an error message on the screen, and the administrator will see a message in the **syslog**, too.

6. List two possible solutions to this full file system problem.

   Answer

   Add additional disk space to the **/home** file system.
   Remove core files.
   Purge large, unneeded files from **/home**.

7. Are there any core files in the problem file system that could be removed? If so, remove them. What commands did you use to find the core files?

   Answer

   ```
   # find /home -name core
   # rm /home/user5/core
   ```

8. Which directory under **/home** is taking the most space? Which command can you use to find out? Mail a message to the culprit asking him or her to purge some files.

   Answer

   ```
   # du -sk /home/* | sort -rn
   # echo "remove some files, user5!" | mail user5
   ```

9. You could wait for your users to purge some old files, but in many cases, you will eventually need to add some additional space to the full file system. **pvcreate** your free disk and add it to vg00. Use **vgdisplay** to ensure that the new disk was successfully added to the volume group.

   Answer

   ```
   # pvcreate /dev/rdsk/c0t5d0 # disk device filename may vary.
   # vgextend vg00 /dev/dsk/c0t5d0
   # vgdisplay v vg00
   ```


© 2004 Hewlett-Packard Development Company, L.P.

10. Double the size of the logical volume containing **/home**. Which disk contains the new extents? What command did you use to find out?

    Answer

    ```
    # lvdisplay -v /dev/vg00/lvol4    # lvol name may vary
    # lvextend -L 48 /dev/vg00/lvol4
    # lvdisplay -v /dev/vg00/lvol4
    # note the extents used by lvol4
    ```

    The logical volume should use the first free extents available in the volume group.

11. Using **bdf**, check to see if the **/home** file system contained in the logical volume you just extended increased in size. Does extending a logical volume automatically extend the file system within the logical volume?

    Answer

    bdf suggests that the home file system hasn't changed in size. The file system won't take advantage of the additional space in the logical volume until you do an **extendfs**.

12. Execute the commands necessary to extend the **/home** file system to take advantage of the additional space in the logical volume.

    Answer

    ```
    # umount /home
    # extendfs /dev/vg00/rlvol4 # lv name may vary
    # mount /home
    ```

13. Unmount the file system and try doing **extendfs** on the logical volume containing **/home** again. Explain the resulting message.

    Answer

    ```
    # umount /home
    # extendfs /dev/vg00/rlvol4 # lv name may vary
    # mount /home
    ```

    The file system can't be extended any further because there isn't any more space in the logical volume. However, no harm was done.

14. Before moving on, remove all the "bigfiles" from user5's home directory.

    Answer

    ```
    # rm /home/user5/bigfile*
    ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Defragmenting File Systems

1. Run a directory and extent fragmentation report on the **/var** file system. At this point, don't do anything to repair the fragmentation. Record the following values:

   a. Directories to reduce:
   b. Average number of extents per file:

   Answer

   ```
   # fsadm -F vxfs -DE /var
   ```

   Results will vary.

2. Defragment the **/var** file system.

   Answer

   ```
   # fsadm -F vxfs -de /var
   ```

3. How successful was the defragmentation utility? Run a second fragmentation report and compare the results to the values you recorded above in question 1. Do you see any improvement?

   Answer

   ```
   # fsadm -F vxfs -DE /var
   ```

   Most likely, your file system had very little fragmentation to begin with. At a bare minimum, however, you should see the "Dirs to reduce" parameter decrease.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Repairing HFS File Systems

In the next couple portions of the lab, you will have an opportunity to run **fsck** on a number of corrupted file systems via an **fsck** simulation program. You may run the simulator by typing: **/labs/simfsck**. The **simfsck** simulator contains five file systems:

```
/dev/vg00/lvol1     HFS     /
/dev/vg01/app       HFS     /app
/dev/vg01/cad       HFS     /cad
/dev/vg02/db        HFS     /db
/dev/vg02/data      VXFS    /data
```

The simulator recognizes all of the UNIX commands necessary to run **fsck** on the above file systems. Note that the simulator only recognizes the most common options on each of these commands.

```
bdf
cat
cd
file
fsck
ls
ll
mount
pwd
strings
umount
```

Run the simulations suggested below. Record the commands you use, and Answer all the questions. After gaining some experience with **fsck** in the simulator, the last exercise asks you to run **fsck** on some "real" file systems as well.

1. Once in the simulator, choose simulation #1 from the menu. Simulation #1 asks you to run **fsck** on the **/app** file system. You will need to know the name of the logical volume containing the **/app** file system when you run **fsck**, so start by viewing a list of the currently mounted file systems.

   Answer

   ```
   # /labs/simfsck
   Enter choice: 1
   Sim: mount -v
   ```

2. File systems must be unmounted before running **fsck**. Unmount the **/app** file system.

   Answer

   ```
   Sim: umount /app
   ```


© 2004 Hewlett-Packard Development Company, L.P.

3. Now run **fsck** on the raw logical volume containing **/app**. Be sure to specify the file system type. **fsck** will check the file system and ask which problems should be corrected.

   Although one could conceivably Answer "no" in response to **fsck's** questions, you should generally Answer "yes" -- otherwise you will be left with a still-corrupted file system. Take note of the problems identified by **fsck**, but go ahead and Answer "yes" to all of the prompts.

   Answer

   ```
   Sim: fsck -F hfs /dev/vg01/rapp
   ```

4. Once **fsck** terminates, remount the file system.

   Answer

   ```
   Sim: mount /app
   ```

5. After mounting the file system, you should check the **lost+found** directory to see if **fsck** identified any orphaned files. Are there any orphaned files in this case?

   Answer

   ```
   Sim: ls /app/lost+found
   ```

   In this case there were no orphaned files.

6. Did the **fsck** prompts indicate that **fsck** "REMOVE"d any files from **/app**? If you have a tape backup of the file system, you should restore these files at this point. In the case of our simulator; however, we don't have a tape backup so any removed files are lost forever.

   Answer

   Yes. **fsck** noted an unallocated I=4, and prompted for permission to remove **importantfile**. This file should be restored from a tape backup.

7. Go back to the main simulator by typing: "menu".

   Answer

   ```
   Sim: menu
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Repairing HFS File Systems, Revisited

1. Choose simulation #2, which asks you to run **fsck** on the **/db** file system.
   Enter choice: 2

2. Following the same procedure used in the previous exercise, **fsck** the **/db** file system and fix all the corruption identified.

   Answer

   ```
   Sim: mount -v
   Sim: umount /db
   Sim: fsck -F hfs /dev/vg02/rdb
   Sim: mount /db
   ```

3. Are there any files in **/db/lost+found** that need attention? If there are any files in **lost+found**, what can you do to find the file's owner and file name?

   Answer

   ```
   Sim: cd /db/lost+found
   Sim: ll \#0004
   Sim: file \#0004
   Sim: cat \#0004
   ```

   There does appear to be one file in **lost+found** with inode #4. The **file** command tells us that the file contains ASCII text, **ll** shows us the owner is **root**, and the **cat** command shows that the file contains a list of key card holders. We can surmise, perhaps, that this file should be named **keycards**.

4. Did **fsck** "REMOVE" any files this time?

   Answer

   No. No **REMOVE**? prompts appeared while running **fsck**.

5. Return to the simulator menu by typing “menu”.

   Answer

   ```
   Sim: menu
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 5: Repairing JFS File Systems

1. Run simulation #3, which asks you to run **fsck** on the **/data** file system. Enter choice: 3

2. What happens if you specify the wrong file system type to **fsck**? Unmount the **data** file system, and try running **fsck -F hfs /dev/vg02/rdata**. What happens? Despite the worrisome message **fsck** offers here, the superblock really isn't damaged. Can you guess why you get this message?

   Answer

   ```
   Sim: mount -v
   Sim: umount /data
   Sim: fsck -F hfs /dev/vg02/rdata
   ```

   fsck was expecting an HFS-style superblock, but instead encountered a JFS-style superblock in the rdata logical volume. **fsck** interprets this inconsistency as a corrupted superblock.

3. Try running **fsck** again with the **-F vxfs** option. How does running **fsck** on a VxFS file system differ from running **fsck** on an HFS file system?

   Answer

   ```
   Sim: fsck -F vxfs /dev/vg02/rdata
   ```

   By default, **fsck** simply replays the intent log rather than doing a full check of the file system.

4. After running **fsck**, remount the file system and return to the main simulator menu.

   Answer

   ```
   Sim: mount /data Sim: menu
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 6: Repairing JFS File Systems, Revisited

1. Run simulation #4, which asks you to run a "full" **fsck** on the **/data** file system. Menu choice: 4

2. You saw in the previous simulation that, by default, **fsck** simply replays the JFS intent log. After a system crash, an intent log replay is all that is required. An intent log replay simply scans the intent log completes any pending transactions; it does not check the consistency of your superblock, inodes, and allocation units.

   If you suspect more serious file system corruption in a JFS file system, you can perform a full check of the file system. Try the following:

   ```
   Sim: mount -v
   Sim: umount /data
   Sim: fsck -F vxfs -o full /dev/vg02/rdata
   Sim: mount /data
   Sim: ll /data/lost+found
   ```

3. How did this **fsck** differ from the **fsck** in the previous JFS simulation?

   Answer

   Instead of simply replaying the intent log, **fsck** appears to do a multipass check of the file system's inodes, directories, superblock, and resource maps. After making several corrections, **fsck** clears the intent log.

© 2004 Hewlett-Packard Development Company, L.P.

Part 7: Repairing Real File Systems

1. Now that you have had an opportunity to run **fsck** in the simulator a few times, try running **fsck** on some real file systems. Do a **mount -v** to determine which file systems are mounted where on your system.

   Answer

   ```
   # mount –v
   ```

2. You should have an HFS file system mounted on **/stand**, and a JFS file system mounted on **/home**. Unmount both of these file systems, and run the appropriate **fsck** commands to check them both. Does **fsck** identify any problems?

   Answer

   ```
   # umount /stand
   # fsck -F hfs /dev/vg00/lvol1
   # mount /stand
   # umount /home
   # fsck -F vxfs /dev/vg00/lvol4
   ```

   There shouldn't be any problems in either file system.

3. If you haven't already remounted your file systems, do so now.

   Answer

   ```
   # mount –a
   ```


© 2004 Hewlett-Packard Development Company, L.P.

11–11. LAB: Backup and Recovery

Directions

Perform the tasks suggested in the questions that follow. Record the commands you use, and answer all the questions.

Since your class machine may not have a tape drive, perform the backups to a series of fake tape files called tape1, tape2, etc. Note that on production machines you would more commonly use a true tape device file (e.g. **/dev/rmt/0m**).

Part 1: Preliminary Steps

1. If you don't already have a **/var/adm/fbackupfiles** directory on your machine, create it. By default, this is the directory where **fbackup** maintains the "dates" log file of backups completed on your system. Make this directory

   Answer

   ```
   # mkdir -p /var/adm/fbackupfiles
   # cd /var/adm/fbackupfiles
   ```

2. Run the **/labs/corp1.sh** script, which will create a few directories that you can practice backing up. Note the directories that are created by the script.

   Answer

   ```
   # /labs/corp1.sh
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Full and Incremental Backups

1. Create a graph file that includes everything in the **/corp** directory.

   Answer

   ```
   # vi graph
     i /corp
   ```

2. Perform a full backup of **/corp** using the graph file you just created. Write the backup to a file called **tape1**, write an index of the files included in the backup to **index1**, and use the **-u** option to ensure that the **dates** file records the time stamp of your backup.

   Answer

   ```
   # fbackup -f tape1 -u0g graph -I index1
   ```

3. What was recorded in the **dates** file as a result of your **fbackup**?
   Were all of the files under **/corp** backed up?
   Look at the **index1** file that was created, and explain what you see.

   Answer

   ```
   # cat dates
   # cat index1
   ```

   There should be a single entry in the **dates** file indicating the time at which the level 0 backup of "graph" was performed.

   index1 should show that all of the files under **/corp** were backed up.

4. Run **/labs/corp2.sh**. This script should create a few new files under the **/corp** directory. Note the changes.

   Answer

   ```
   /labs/corp2.sh
   ```

5. Do a level 1 backup of **/corp** using your graph file. Write the backup to **tape2**, create an index of your backup in **index2**, and use the **-u** option to ensure that the **dates** file is updated again.

   Answer

   ```
   # fbackup -f tape2 -u1g graph -I index2
   ```


© 2004 Hewlett-Packard Development Company, L.P.

6. What was recorded in your **dates** file as a result of your backup?
   Were all of the files under **/corp** backed up?
   Look at the **index2** file and explain what you see.

   Answer

   ```
   # cat dates
   # cat index2
   ```

   A new line should have been added to your **dates** file indicating that a level 1 backup of the **graph** file was performed. In the **index2** file, note that only **/corp** and the new files under **/corp** were included in this backup. Since this was level 1 incremental backup, only the files that changed were included in the backup.

7. Run the **/labs/corp3.sh** script. Note the newly created files.

   Answer

   ```
   # /labs/corp3.sh
   ```

8. Run another level 1 backup. Write the backup to **tape3**. Ensure that the **dates** file is updated and create an index file called **index3**.

   Answer

   ```
   # fbackup -f tape3 -u1g graph -I index3
   ```

9. What changed in your **dates** file as a result of your backup? Explain.
   Which files and directories under **/corp** backed up? Explain.

   Answer

   ```
   # cat dates
   # cat index3
   ```

   The previous level 1 backup in the dates file was replaced by this new level 1 backup. The previous level 1 backup is no longer needed to ensure a full recovery since the new level 1 includes ALL the files created or modified since the last FULL backup.

10. Run the **/labs/corp4.sh** script. Note that this script doesn't touch any new or existing files.

© 2004 Hewlett-Packard Development Company, L.P.

11. Now try running a level 2 backup of **/corp** using your **graph** file. Ensure that **fbackup** updates the **dates** file and creates an index file called **index4**. What files and directories are included in this backup? Explain.

Answer

```
# fbackup -f tape4 -u2g graph -I index4
```

fbackup adds an entry to the **dates** file to indicate that a level 2 backup was performed. However, since no files have been created or changed since the last level 1, no files were included in the backup, though the **/corp** directory itself is included.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Restoring Files with `frecover`

1. Run the **`/labs/corp5.sh`** script, and note the effect this script has on your **`/corp`** directory.

   Answer

   ```
   # /labs/corp5.sh
   ```

   The script destroyed **`/corp`** and everything under it.

2. You did a total of four backups in the previous part of this lab. Is it necessary to restore all four tapes to fully recover your system? Explain.

   Answer

   It is only necessary to restore the most recent backup at each backup level. The most recent level 0 backup is on tape1, the most recent level 1 backup is on tape3, and the most recent level 2 backup is on tape4. It is not necessary to restore tape2.

3. Do whatever is necessary to fully restore **`/corp`**.

   Answer

   ```
   # frecover -f tape1 –rv        # restore the most recent level 0
   # frecover -f tape3 –rv        # restore the most recent level 1
   # frecover -f tape4 –rv        # restore the most recent level 2
   # find /corp                   # ensure that the recovery worked.
   ```

4. Occasionally, users accidentally remove a file and expect the administrator to be able to restore the missing file from tape. Remove **`/corp/reporta`**. How can you restore this single file? Do it.

   Answer

   ```
   # rm /corp/reporta
   # frecover -f tape3 -i /corp/reporta –xv
   ```

5. Will **`frecover`** overwrite newer data in a file with older data from a tape archive? Try it and find out.

   Answer

   ```
   # vi /corp/reporta                  # Make a change to reporta
   # ll /corp/reporta                  # Note the time stamp
   # frecover -f tape3 -i /corp/ reporta –xv
   # cat /corp/ reporta                # Were your changes overwritten by
                                         frecover?
   ```

   Note that **`frecover`** does *not* overwrite the newer version of the file with the older version of the file from the tape archive. By default, **`frecover`** always compares the time stamp of the archived version against the time stamp of the file on disk.

© 2004 Hewlett-Packard Development Company, L.P.

12–2. SLIDE: Understanding the `cronfile`

The following are answers to the teaching questions in the topic:

What values would you include in the **`cron`** file time fields if you wanted to execute a command:

1. Every day at 6:30 pm?

2. Every weekday at 6:30 pm?

3. Every Monday, Wednesday, and Friday at 6:30 pm?

4. Every 10 minutes, every hour, every day?

Answer

```
# min  hour  date   month  day    command
  30   18    *      *      *      myprog   #every day at 6:30pm
  30   18    *      *      1-5    myprog   #every weekday at 6:30pm
  30   18    *      *      1,3,5  myprog   #every M,W,Fr at 6:30pm
  0,10,20,30,40,50,60 * * * *     myprog   #every 10 minutes
```


© 2004 Hewlett-Packard Development Company, L.P.

12–5. LAB: cron

Directions

In the exercises in Part 1, you will schedule a simple **cron** job that repeatedly displays the output from the date command in the system console window. To ensure that you see these messages, open a console window by typing:

```
# dtterm -C -name console &
```

Any output sent to **/dev/console** now appears in this window.

Part 1: Managing Scheduled cron Jobs

1. **cron** should start automatically during the boot process. Check to ensure that the **cron** daemon is running on your machine.

   Answer

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

2. Create and submit a **cron** job to display the current time and date to the console every minute.

   Answer

   ```
   # crontab -e
   * * * * * /usr/bin/date > /dev/console
   ```

3. Add another **cron** job that sends the output from the **who** command to the console every 10 minutes.

   Answer

   ```
   # crontab -e
   * * * * * /usr/bin/date > /dev/console
   0,10,20,30,40,50 * * * * /usr/bin/who  > /dev/console
   ```

4. List your scheduled **cron** jobs. Are they both there?

   Answer

   ```
   # crontab -l
   ```

   Both jobs should appear.

© 2004 Hewlett-Packard Development Company, L.P.

5. Edit your **cron** file again. See what happens if you precede the **/usr/bin/date** line with a **#** sign. Save your changes. What effect does the **#** sign have?

   Answer

```
# crontab -e
# *               * * * * /usr/bin/date > /dev/console
0,10,20,30,40,50 * * * * /usr/bin/who  > /dev/console
```

   After saving the change, the **date** command stops executing.

6. Why would commenting a line out of the **cron** file be preferable to simply removing the line?

   Answer

   Simply commenting the **date** line out of the **cron** file makes it easier to reschedule the job later: just delete the **#** sign.

7. How can you remove ALL of your scheduled **cron** jobs? Do it.

   Answer

```
# crontab -r
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 2: (Optional) Scheduling Backups with `cron`

Your goal in this part of the exercise is to schedule full and incremental backups of your home file system to occur automatically via **`cron`**. If you have a tape drive on your student system, write your backups to the tape drive device file. If you don't have a tape drive, write your backups to /tmp/tape.

1. Create a **`graph`** file that includes **`/home`**, but nothing else. Name your graph file **`/home.graph`**.

   Answer

   ```
   # vi /home.graph
   i /home
   ```

2. Schedule a full backup of **`/home.graph`** to occur every Sunday night, and an incremental backup to occur every weekday night. Schedule both backups to run at 11 pm. Redirect the error messages from your backups to a file called **`/home.err`**.

   Answer

   ```
   # crontab -e
   0 23 * * 0         /usr/sbin/fbackup -f /dev/rmt/0m -u0g /home.graph
   2>> /home.err
   0 23 * * 1,2,3,4,5 /usr/sbin/fbackup -f /dev/rmt/0m -u1g /home.graph
   2>> /home.err
   ```

3. List your **`cron`** jobs to ensure they are properly scheduled.

   Answer

   ```
   # crontab -l
   ```

4. If you have a tape drive on your system, leave your scheduled **`cron`** jobs in place, and check your **`/home.err`** file tomorrow morning to see if your backup succeeded. If you don't have a tape drive on your system, remove all of your scheduled cron jobs to avoid filling the /tmp file system.

   Answer

   ```
   # crontab -r
   ```


© 2004 Hewlett-Packard Development Company, L.P.

13–10. LAB: Swap

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. How much physical memory does your training system have?
   How much available memory?
   How much lockable memory?

   Answer

   All three questions may be Answered by looking for the physical memory line that appears on the system console during the system boot process. In most cases, this console message may be viewed after system boot via the **dmesg** command.

2. Is pseudoswap enabled on your system? Use the **kmtune** command (11i v1) or **kctune** command (11i v2) to view your kernel parameters and find out!

   ```
   # kmtune -q swapmem_on   (11i v1)
   # kctune swapmem_on      (11i v2)
   ```

   Answer

   The **swapmem_on** parameter should be set to "1", indicating that pseudoswap is on.

3. How much device swap is currently configured on your system?
   How much file system swap is currently configured?
   How much swap space has been reserved by processes on your system?

   Answer

   ```
   # swapinfo -d       Lists device swap areas.
   # swapinfo -f       Lists file system swap areas.
   # swapinfo          Lists all swap areas, including a "reserve" line.
   ```

   Your classroom system should have one device swap logical volume, but no file system swap at this point. The amount of memory currently reserved will vary, depending on the number of processes currently running on your system.

4. As you add new applications on your system, your application vendor may ask you to configure additional swap space. Create an additional 100 MB logical volume in **vg0**0 and enable it for use as device swap.

   Answer

   ```
   # lvcreate -L 100 -n swap vg00
   # swapon /dev/vg00/swap
   ```


© 2004 Hewlett-Packard Development Company, L.P.

5. In crunch periods, you may need to configure file system swap to supplement already configured device swap. Enable file system swap on **/home**. Set the priority level to 4 to ensure that this file system swap is only used when your device swap areas are full. Also ensure that no more than half of the file system will be used as swap.

   Answer

   ```
   # swapon -p 4 -l 24M /home  adjust your "limit" to match half of the /home LV
   ```

6. Execute **swapinfo** again to verify that your new swap areas are configured properly.

   Answer

   ```
   # swapinfo
   ```

7. Do whatever is necessary to ensure that your new swap areas are automatically enabled after the next reboot.

   Answer

   ```
   # vi /etc/fstab
     /dev/vg00/swap  .       swap   defaults 0 0
     /dev/vg00/lvol4/home    swapfs defaults 0 0
   # swapon -a
   ```

8. Can you unmount a file system that has file system swap enabled? Try it.

   Answer

   ```
   # umount /home
   ```

   Fails! After file system swap has been enabled on a file system, it can *never* be unmounted.

9. You should have discovered in the previous question that once swap is enabled in a file system, it is impossible to unmount that file system. Is this a problem? Explain.

   Answer

   Some administrative tasks require a file system to be unmounted. Enabling file system swap makes it very difficult to perform those tasks. File system swap should truly only be used as a last resort.

10. Try to view the man page for **swapoff**. What happens?

NOTE:	There is no **swapoff** command in HP-UX! To disable swap, edit the **/etc/fstab** file; comment out or delete the lines for the swap areas you wish to disable; and reboot. When the system reboots, the commented lines in **fstab** will be ignored. There is no way to disable swap without rebooting.

© 2004 Hewlett-Packard Development Company, L.P.

11. You need to disable both of your new swap areas before moving on to the next section. Remove your swap entries from **/etc/fstab**, then reboot using the command shown below:

```
# vi /etc/fstab              remove all swap entries
# shutdown -ry 0
```


© 2004 Hewlett-Packard Development Company, L.P.

14–12. LAB: Hands-On Adding Printers

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. If your system has a locally attached printer available, add that printer to your local spooler using SAM. If your system doesn't have a locally attached printer, run the **`/labs/fakeprinter.sh`** script. This script creates a terminal window at the bottom right of your CDE desktop that may be configured as a pseudo-printer.

   From the SAM functional area launcher, choose **`Printers and Plotters --> LP Spooler --> Printers and Plotters`**. Then select **`Actions --> Add Local Printer --> With Non-Standard Device File`**. When prompted for the device filename, use "**`/dev/fakeprinter`**". When prompted for the Model Interface Script to use, choose "dumb". Configure your fake printer as your default printer.

2. List the status of the LP scheduler. Is the scheduler running? What is your system default destination? Is your printer enabled? Is your printer accepting requests?

   Answer

   ```
   lpstat –t
   ```

   SAM should have started the scheduler, enabled your printer and issued the accept command for you.

3. Try to print to your printer. Issue the command:

   ```
   # banner success|lp -dyour_destination
   ```

   Try to print to the destination **`class1`**.

   Answer

   ```
   # banner success|lp –dsparky
   # banner This printer has class|lp -dclass1
   ```


© 2004 Hewlett-Packard Development Company, L.P.

4. Set the fence priority on your printer to a priority higher than your default priority. Issue the **lp** command without the **-p** option. Did the request print? Display the status of the scheduler. Do you see your request on the queue? Raise the priority on your request to make it print.

   Answer

```
# lpshut                              The scheduler must first be stopped.
# lpfence sparky 5                    Alter the priority.
# lpsched                             The scheduler must be started again.
# banner fenced in|lp -dsparky        This request will not print yet.
# lpstat -t                  Find the request id for your request.
# lpalt sparky-23 -p5        Substitute your request id for sparky-23.
```

5. Configure your neighbor's local printer as a remote printer for your machine. Print a copy of **/etc/passwd** on your new remote printer and see what happens.

   Answer

   Use SAM.

6. Use SAM to remove your printer and exit SAM when finished.

   Answer

   Select **Printers and Plotters** from the main **SAM** menu. Select **Printers and Plotters** from the submenu. Highlight your printer in the object list. Select **Remove** from the **Actions** menu. SAM will ask for confirmation. Press **Yes** to proceed.

7. If the equipment is available in the classroom, configure a network printer.

   Answer

   Use **SAM** or **jetadmin**.

© 2004 Hewlett-Packard Development Company, L.P.

15–22. LAB: Interacting with the MP

Directions

Many critical, low-level administration tasks may only be accomplished via the system console. On all of HP's current server models, a virtual system console is accessible via the Management Processor (or "Guardian Service Processor" on older systems). This lab provides an opportunity for you to explore the MP's features and commands.

Your instructor should have given you the connectivity information for your server's Management Processor. Record the username and network information below:

MP IP address: _____

MP hostname: _____

MP admin username: _____

MP admin password: _____

Assigned partition: _____ (only needed on nPar systems)

Some sites may prohibit student admin access to the MP. In this case, your instructor may simply demonstrate some portions of the lab.

© 2004 Hewlett-Packard Development Company, L.P.

Part 1: Accessing and Exploring the MP

This portion of the lab shows you how to connect to your system's MP. Operator access is sufficient to complete this portion of the lab.

1. Open a telnet session to your server's MP.

   Answer

   ```
   # telnet 192.168.1.1      # Use your MP IP here.
   ```

2. Login using the username and password suggested by your instructor.

   Answer

   ```
   MP login: Admin           # Use your MP login here.
   MP Password: Admin        # Use your MP password here.
   ```

3. Some versions of the MP firmware immediately display the MP main menu. Others immediately place MP users in the MP console interface. If your MP sent you directly to the MP console interface, press ^b to return to the MP main menu.

4. You should see several commands on the MP main menu. However, there are many other commands that don't appear in the main menu. Which command can you use to learn more about the features of the MP? Use this command to generate a list of supported MP commands and fill-in the blanks below. After you answer the questions, return to the MP main menu.

   a. The ____ command may be used to access the system console.
   b. The ____ command may be used to access the virtual front panel.
   c. The ____ command may be used to reset the system.
   d. The ____ command may be used to configure MP user accounts.
   e. The ____ command may be used to view the MP LAN configuration.
   f. The ____ command may be used to view the system's power status.
   g. The ____ command may be used to view a list of users currently using the MP.

   Answer

   ```
   MP> he
   MP:HE> li
   MP> q
   ```

   a. The **co** command may be used to access the system console.
   b. The **vfp** command may be used to access the virtual front panel.
   c. The **rs** command may be used to reset the system.
   d. The **uc** or so commands may be used to configure MP user accounts.
   e. The **ls** command may be used to view the MP LAN configuration status.
   f. The **ps** command may be used to view the system's power status.
   g. The **who** command may be used to view a list of users currently using the MP.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Accessing the System Console via the MP

Perhaps the most common reasons for accessing the MP is to access the system console. This is your chance to explore this functionality! Operator access is sufficient to complete this part of the lab.

1. Use the MP to access your system console.

   Answer

   ```
   MP> co
   Please select partition number: 0   If prompted, enter your partition# here
   ```

2. Login as HP-UX user root, and run the hostname command to verify that you accessed the right console.

   Answer

   ```
   Console Login: root
   Password: **        Enter your root password here
   # hostname
   ```

3. From the console interface, you can return to the MP main menu at any time by pressing **^b**. Without logging out from HP-UX, press ^b.

   Answer

   This should take you back to the MP.

4. Access the console interface again. Are you prompted for a Unix username and password this time?

   Answer

   ```
   MP> co
   Please select partition number: 0 If prompted, enter your partition# here
   ```

   No login or password is required!

5. Logout from HP-UX, and press **^b** to return to the MP.

   Answer

   ```
   # exit
   ^b
   MP>
   ```


© 2004 Hewlett-Packard Development Company, L.P.

6. Access the console interface again. What happens?

   Answer

   ```
   MP> co
   Please select partition number: 0 If prompted, enter your partition# here
   Console Login: root
   Password: **
   ```

   This time, you should be prompted for a UNIX username and password.

7. Why is it important to logout of HP-UX before using ^b to return to the MP main menu?

   Answer

   If an MP user doesn't logout of the console session, the next user to access the console can gain root access to the partition without entering a password!

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Rebooting the System via the MP Console

When you reboot the system, it is important to do so via the MP console interface to ensure that you don't miss any important messages. Try it! This portion of the lab only requires MP operator privileges.

1. If you aren't already logged in as root on the MP console, login now.

2. While logged in as root, type shutdown –hy 0 to halt your system or nPar. Watch the resulting shutdown messages.

   ```
   # shutdown –hy 0
   ```

3. When the shutdown appears to be complete, return to the MP main menu by pressing ^b.

4. What command can you use to reset your system and reboot? Make it so! (Note: Some systems automatically power-off when halted via shutdown –h. If your system is powered-off, power it back on again with the pc command. This should automatically boot the system, too).

   Answer

   ```
   MP> cm
   MP:CM> rs # on older systems, this command is available on the main menu
   ```

   If your system powered off, do this instead:

   ```
   MP> cm
   MP:CM> pc
   ```

5. Now monitor the system's self-tests via the VFP.

   Answer

   ```
   MP> cm
   MP:CM> pc # on older systems, this command is available on the main menu
   MP:CM> ma
   MP> vfp
   ```

6. Watch the VFP for a few minutes, then access the console log to see some of the console messages generated by the HP-UX startup process.

   Answer

   ```
   MP> cl
   ```

7. When your system finishes booting, access the console interface again and login as root to verify the boot succeeded.

   ```
   MP> co
   Console Login: root
   Password: **        Enter your root password here
   ```


© 2004 Hewlett-Packard Development Company, L.P.

8. Return to the MP main menu before proceeding.

```
# exit
^b
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Configuring and Using MP User Accounts

The MP must be carefully secured to control access to your system console. In this part of the lab, you will learn how to create and manage MP user accounts to prevent unauthorized access to your system. This portion of the lab requires MP admin privileges.

1. Which MP command may be used to manage MP access accounts? Read the full description of this command via the MP he (help) utility.

   Answer

   ```
   MP> he
   MP:HELP> li
   MP:HELP> so or uc
   MP:HELP> q
   ```

   On older models, use the **so** (security options) command.
   On newer models, use the **uc** (user configure) command.

2. Using the command you discovered in the previous question, create a new MP user account for yourself, then log out of the MP.

   - Use login name "student"
   - Also use "Student" as the user's real name
   - Use HPE as the user's organization name
   - Grant your new user "Operator" access to the MP
   - Configure the "Multiple Use" mode for the new account
   - Enable the account
   - Disable dialback access
   - Choose a password for the new user

   Answer

   If available, run the **uc** command, then follow the prompts to create your new user account. If your MP doesn't provide a **uc** command, use the so command instead. On some systems, you may need to navigate to the command (**cm**) menu to run these commands.

   ```
   MP> cm
   MP:CM> so or uc # on older systems, this command is available on the main menu
   MP:CM> ma
   ```


© 2004 Hewlett-Packard Development Company, L.P.

3. In the previous question, you granted the "student" user "operator" level access to the MP. Which of the following MP commands should the new MP user be able to access? (Don't try the commands yet. You will have a chance to try that in a moment.)

   a. **co**
   b. **vfp**
   c. **rs**
   d. **ps**
   e. ls
   f. **lc**
   g. **uc**/**so**

   Answer

   MP users with operator privileges should have access to all of the commands necessary to access and boot the system, and to view the MP configuration. Thus, the student user should be able to run the **co**, **vfp**, **rs**, **ps**, and **ls** commands. However, operator level users can't run any commands that might be used to change the MP configuration. Thus, the student user won't be able to run the **lc**, **uc**, or **so** commands.

4. Let's test your answers to the previous question. Without closing your original "admin" MP telnet session, open a second telnet window on your desktop, connect to the MP, and login as MP user "student" to verify that the new MP user account works.

   Answer

   ```
   telnet 192.168.1.1 # use your MP IP address here
   MP login: student
   MP password: **
   ```

   You should see the MP welcome screen, indicating that the login succeeded.

5. Can the original admin user tell that there is a second user logged into the MP? Run the MP **who** command in the original telnet window to find out! (Hint: On some systems, you may need to go to the **cm** command menu to access the **who** command.)

   Answer

   ```
   MP> cm
   MP:cm> who
   MP:cm> ma
   ```

   who should report two MP user logins.

© 2004 Hewlett-Packard Development Company, L.P.

6. What command can you use to warn other MP users that you are using the system? Make it so!

   Answer

   ```
   MP> cm
   MP:CM> te   # on older systems, this command is available on the main menu
   Hello fellow MP users!
   MP:CM> ma
   ```

7. What happens if the "student" user attempts to access the console interface at the same time as the "admin" user? Try it! Type the **co** command in both telnet windows. Explain the results.

   Answer

   Both users should be able to access the console interface simultaneously, but only one user at a time can control the console. The other user is given read-only access to the console. Any console user can unilaterally demand write access at any time by pressing **^e cf**.

8. Remove the "student" MP user account. If you made any other changes to the MP configuration undo them as well before moving on.

   Answer

   ```
   MP> cm
   MP:CM> so or uc # on older systems, this command is available on the main menu
   MP:CM> ma
   ```


© 2004 Hewlett-Packard Development Company, L.P.

16–7. LAB: Shutting Down and Rebooting Your System

Directions

Carefully follow the instructions below.

Part 1: Preliminary Steps

A portion of this lab requires you to interact with the ISL and boot menus, which can only be accomplished via a console login. If you are using remote lab equipment, access your system's console interface via the GSP/MP.

Part 2: System Shutdown

1. Currently, your system should be in multi-user mode. Note which file systems are mounted, and how many processes are running.

   Answer

   ```
   # mount –v
   # ps -ef | wc –l
   ```

2. Shut down to single-user mode. Then check to see what processes and mounted file systems remain. What differences do you see between single- and multi-user modes?

   Answer

   ```
   # cd /
   # shutdown -y 0
   # mount –v
   # ps -ef | more
   ```

   There are few file systems mounted in single-user mode, and few processes running. Single-user mode is a good place to do kernel configuration, file system maintenance with **`fsck`**, and other maintenance activities that require a quiet system.

3. From single-user mode, take your machine down to the halt state. What can you do in the halt state? Why might it be necessary to take your system down to the halt state?

   ```
   # reboot –h
   ```

   Answer

   Nothing is running in the halt state, and no logins are possible. It is necessary to bring the system to a halt state before powering off the system to install new peripherals and interface cards.

4. Use the MP **`rs`** command or the system power switch to reset your server before proceeding to the next part of the lab. Immediately begin pressing the [Escape] key to interrupt the boot process.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Booting the System

1. If you didn't already do so in the previous part of the lab, reboot your system and interrupt the autoboot process. Proceed to the PDC/BCH main menu.

2. Search for possible boot devices on your system. How many SCSI devices does your system have? If a system isn't booting properly, this is one way of determining if all your SCSI devices are properly connected and powered on.

   Answer

   ```
   Main Menu: Enter command or menu > search
   ```

   The number of devices on the system will vary.

3. Now search for disks that contain an IPL that you can boot from. How many of your disks appear to be bootable?

   Answer

   ```
   Main Menu: Enter command or menu > search ipl
   ```

   Most likely, there will be only one bootable disk.

4. Display your primary and alternate boot paths. Which disk is defined as your primary boot device?

   Answer

   ```
   Main Menu: Enter command or menu > path
   ```

   The primary and alternate boot paths will vary.

5. Boot to the ISL from your primary boot disk. Be sure to specify that you want to interact with the ISL/IPL.

   Answer

   ```
   Main Menu: Enter command or menu > boot pri isl
   ```

   Some models ask if you want to interact with the ISL. Answer "yes".

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

6. At the ISL prompt, get a list of valid ISL commands.

   Answer

   ```
   ISL> help
   ```


© 2004 Hewlett-Packard Development Company, L.P.

7. Interacting with the ISL may be useful if your primary kernel is corrupted and you have to boot from a backup kernel. Do you currently have a backup kernel in your /stand directory? (Kernel file names usually begin with vmunix.)

   Answer

   ```
   ISL> hpux ls
   ```

   There should be just one kernel at this point: **/stand/vmunix**.

8. By default, the system boots to multi-user mode using **/stand/vmunix**. Boot to single-user mode from the default kernel.

   Answer

   ```
   ISL> hpux –is
   ```

   OR

   ```
   ISL> hpux -is /stand/vmunix
   ```

9. Did the system prompt you for a password when you were brought to single-user mode? When might this be helpful?

   Answer

   No, single-user mode does not prompt for a password. The administrator is automatically logged in as root. This may be useful if the administrator forgets the root password.

10. Reboot your system again, but this time let it boot unattended using the default boot disk and kernel.

    Answer

    ```
    # reboot
    ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: (Optional) Mirroring the Boot Disk

In order to improve system stability, many administrators choose to mirror the system boot disk. Doing so guarantees that you system continues running, even if one of the boot disks or SCSI interface cards on your system is damaged. If time permits, follow the procedure below to mirror your student system's boot disk:

1. Use **ioscan** and **diskinfo** to find a disk large enough to mirror the existing boot disk.

   ```
   # ioscan -funC disk
   # diskinfo /dev/rdsk/boot
   # diskinfo /dev/rdsk/bootmirror
   ```

2. **pvcreate** the new disk and add it to the root volume group. The **-B** option on **pvcreate** reserves space for a "boot area" on the new boot disk. Files in the boot area are stored in a special format and may not be viewed or modified using the standard UNIX file system commands.

   ```
   # pvcreate -fB  /dev/rdsk/bootmirror
   # vgextend vg00 /dev/dsk/bootmirror
   ```

3. The new boot disk's boot area must have copies of the ISL and HP-UX utilities. These utilities may be created in the boot area using the **mkboot** command.

   ```
   # mkboot /dev/rdsk/bootmirror
   ```

4. Recall that the AUTO file is a special structure in the boot area at the top of each boot disk that determines how the kernel is loaded during the boot process. By default, an HP-UX system won't boot unless LVM "quorum" is met. Quorum is met when 51% of the disks in a volume group are present. If the boot disk is mirrored, we want the system to boot even if one of the mirrors in vg00 is unavailable. The two commands below modify the AUTO file on both boot disks to allow the system to boot even if vg00 doesn't meet the quorum restrictions during the boot process.

   ```
   # mkboot –a "hpux –lq" /dev/rdsk/boot
   # mkboot –a "hpux –lq" /dev/rdsk/bootmirror
   ```

5. Mirror all of the boot disk logical volumes to the new boot disk. Note that the names of the logical volumes on *your* boot disk may be different. Make sure you mirror all of the boot disk logical volumes! It may take several minutes to synchronize each mirror.

   ```
   # lvextend -m 1 /dev/vg00/lvol1 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol2 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol3 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol4 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol5 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol6 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol7 /dev/dsk/bootmirror
   # lvextend -m 1 /dev/vg00/lvol8 /dev/dsk/bootmirror
   ```


© 2004 Hewlett-Packard Development Company, L.P.

6. Update stable storage so your new mirror is listed as the alternate boot disk. Stable storage may be modified from the PDC menu, but it may also be modified from multi-user mode using the command shown below:

```
# setboot -a 2/0/1.5.0    use your mirrored disk's hw path here
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 5: (Optional) Testing the New Boot Mirror

After configuring the new mirrored boot disk, test it!

1. Verify that the new PV was added to the volume group properly.

```
# vgdisplay -v vg00
```

2. Check to see that the logical volumes are mirrored and current.

```
# lvdisplay -v /dev/vg00/lvol1
# lvdisplay -v /dev/vg00/lvol2
# lvdisplay -v /dev/vg00/lvol3
```

3. List the files in the mirror's boot area, then display the AUTO files from both disks using the commands shown below.

```
# lifls /dev/rdsk/bootmirror
# lifcp /dev/rdsk/bootmirror:AUTO -
# lifcp /dev/rdsk/primaryboot:AUTO -
```

4. Now for the real test: try to boot from your mirrored root disk! Do a shutdown, then interrupt the boot sequence. Use the PDC **search** command to see if your new boot disk is recognized. Assuming it is, go ahead and boot from it!

```
# shutdown -ry 0
[esc]
Main Menu: Enter command or menu > search ipl
Main Menu: Enter command or menu > boot alt
```


© 2004 Hewlett-Packard Development Company, L.P.

17–17. LAB: Booting Integrity Servers

Directions

Carefully follow the instructions below.

Part 1: Preliminary Steps

A portion of this lab requires you to interact with the EFI and HPUX interfaces, which can only be accomplished via a console login. If you are using remote lab equipment, access your system's console interface via the MP.

Part 2: Shutting Down the System

1. Currently, your system should be in multi-user mode. Note which file systems are mounted, and how many processes are running.

   Answer

   ```
   # mount –v
   # ps -ef | wc –l
   ```

2. Shut down to single-user mode. Then check to see what processes and mounted file systems remain. What differences do you see between single- and multi-user modes?

   Answer

   ```
   # cd /
   # shutdown -y 0
   # mount –v
   # ps -ef | more
   ```

   There are few file systems mounted in single-user mode, and few processes running. Single-user mode is a good place to do kernel configuration, file system maintenance with **`fsck`**, and other maintenance activities that require a quiet system.

3. From single-user mode, take your machine down to the halt state. What can you do in the halt state? Why might it be necessary to take your system down to the halt state?

   ```
   # reboot –h
   ```

   Answer

   Nothing is running in the halt state, and no logins are possible. It is necessary to bring the system to a halt state before powering off the system to install new peripherals and interface cards.

© 2004 Hewlett-Packard Development Company, L.P.

4. Use the MP **rs** command or the system power switch to reset your server before proceeding to the next part of the lab. Return to the console interface and immediately begin pressing the [Escape] key to interrupt the boot process.

   Answer

```
# ^b
MP> cm          # not necessary on low end systems
MP:CM> rs
MP:CM> ma
MP> co
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Interacting with the EFI Boot Manager and Shell

1. If you didn't already do so in the previous part of the lab, reboot your system and interrupt the EFI autoboot process. At this point, you should see the EFI boot manager menu:

```
EFI Boot Manager ver 1.10 [14.60] Firmware ver 1.61 [4241]

Please select a boot option

    HP-UX Primary Boot
    EFI Shell [Built-in]
    Boot option maintenance menu
    Security/Password Menu
```

2. Select the EFI shell option from the boot manager menu.

Answer

```
EFI Boot Manager ver 1.10 [14.60] Firmware ver 1.61 [4241]

Please select a boot option

    HP-UX Primary Boot
    HP-UX HA Alternate Boot
    EFI Shell [Built-in]
    Boot option maintenance menu
    Security/Password Menu
```

3. Use the EFI info –b all command to view your system configuration.

   a. What is your system product name / model string?

   b. How many CPUs do you have?

   c. How much memory do you have?

   d. How many bootable devices do you have?

   e. How many LAN interfaces do you have?

   f. What are their MAC addresses?

Answer

```
Shell> info –b all
```

The hardware configuration will vary from system to system.

© 2004 Hewlett-Packard Development Company, L.P.

4. Use the map –r –fs command to view a list of potential boot devices. Record the fs*n*: names below, and see if you can determine which hardware path each fs*n*: name represents. You may need to refer back to the EFI hardware addressing slide earlier in the book to interpret the hardware paths.

   Answer

   ```
   Shell> map –b –r –fs
   ```

   The actual fs*n*: labels and hardware paths will vary, but you should have at least two EFI file system partitions. The hardware paths that include the string Part1 usually represent OS partitions, and paths that include the string Part3 usually represent HPSP partitions.

5. Type fs*n*: to select the system partition on your primary boot disk, where fs*n*: is the drive name that you identified in the previous step.

   Answer

   Drive names will vary, but after selecting the desired drive name, your EFI shell prompt should display the drive that you're currently accessing.

6. Use the EFI cd and ls commands to explore the file system and find the hpux.efi OS loader and the auto file. What are the full pathnames for these files?

   Answer

   ```
   fsn:\> cd \efi\hpux
   fsn:\> ls
   ```

   \efi\hpux\hpux.efi is the full pathname for the HP-UX OS loader.
   \efi\hpux\auto is the full pathname for the auto file.

7. Which EFI shell command can be used to view the contents of the auto file? Try it!

   Answer

   ```
   fsn:\> cat -b \efi\hpux\auto
   ```

8. Recall that the bcfg command may be used to view and manage the EFI boot menu. Use the appropriate bcfg arguments to dump the current boot menu to the screen.

   Answer

   ```
   fsn:\> bcfg boot dump
   ```


© 2004 Hewlett-Packard Development Company, L.P.

9. Remove your primary boot disk from the EFI boot manager menu.

   Answer

   ```
   fsn:\> bcfg boot rm 2
   ```

10. Oops! We didn't really want to remove the primary boot disk from the boot manager. Add the primary boot disk back to the top of the boot menu. Label the menu item "HP-UX Primary Boot".

    Answer

    fsn:\> bcfg boot add 1 \efi\hpux\hpux.efi "HP-UX Primary Boot"

11. Dump the EFI boot menu to the screen again to verify your work.

    Answer

    ```
    fsn:\> bcfg boot dump
    ```

12. What command can you execute from the EFI shell to boot the system? Make it so! When given an opportunity to interrupt the HPUX kernel loader autoboot, press Escape.

    Answer

    ```
    fsn:\> \efi\hpux\hpux.efi
    Escape
    ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Interacting with the HPUX kernel loader

In this part of the lab, you will have a chance to explore the HPUX kernel loader.

1. If you didn't interrupt the autoboot process in the previous part of the lab, reboot again, and press Escape when the HPUX autoboot message appears.

```
HP-UX Boot Loader for IA64  Revision 1.71

Press Any Key to interrupt Autoboot
\EFI\HPUX\AUTO ==> boot vmunix
Seconds left till autoboot - 10  Escape
  Type 'help' for help
```

2. Type the help command to view a list of available commands.

   Answer

```
HPUX> help
```

3. Which command can you use to list the kernels available in /stand? Try it!

   Answer

```
HPUX> ll
```

4. Which command can you use to view the contents of the auto file? What is the purpose of the auto file?

   Answer

```
HPUX> showauto
```

   The auto file determines the default boot string.

5. Do whatever is necessary to boot vmunix to single-user mode.

   Answer

```
HPUX> boot vmunix -is
```

6. Did the system prompt you for a password when you were brought to single-user mode? When might this be helpful?

   Answer

   No, single-user mode does not prompt for a password. The administrator is automatically logged in as root. This may be useful if the administrator forgets the root password.

© 2004 Hewlett-Packard Development Company, L.P.

7. Reboot your system again, but this time let it boot unattended using the default boot disk and kernel.

   Answer

   ```
   # reboot
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 5: (Optional) Configuring a Partition Table for a Mirrored Boot Disk

Many administrators choose to mirror the boot disk to ensure that the failure of a single interface or controller card doesn't jeopardize a system's availability. The last few parts of the lab walk you through the process required to mirror an Integrity boot disk. First, you need to identify create three EFI partitions on the new boot disk mirror.

1. Use ioscan and diskinfo to identify a disk large enough to mirror the existing boot disk.

```
# ioscan -funC disk
# diskinfo /dev/rdsk/c0t1d0
```

2. Create the system, OS, and service partitions with the HP-UX idisk command. Allocate 500MB for the EFI system partition, 400MB for the HPSP partition, and all the remaining space for the OS partition. Replace c0t1d0 in the command below with your new mirror's block device file name.

```
# vi /tmp/idf
  3
  EFI 500MB
  HPUX 100%
  HPSP 400MB
# idisk –wf /tmp/idf /dev/rdsk/c0t1d0
```

3. Create device files for the new partitions.

```
# insf –eC disk
```

4. Verify the partition table. Near the end of the idisk output, you should see three partitions. Replace c0t1d0 with your new mirror's raw device file name.

```
# idisk /dev/rdsk/c0t1d0
```

5. Verify that the device files were created properly. There should be block and raw *cxtxdx*s1, *cxtxdx*s2, and *cxtxdx*s3 device files for the new boot mirror.

```
# ioscan –funC disk
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 6: (Optional) Configure the Mirrored Boot Disk's EFI System Partition

After you create the partition table, you can initialize and populate the EFI system partition.

1. Populate the /efi/hpux/ directory in the new EFI system partition with the mkboot command. This command copies the hpux.efi boot loader and other files to the system partition.

```
# mkboot –e –l /dev/rdsk/c0t1d0
```

2. By default , the LVM "quorum" rules require at least 51% of the disks in vg00 to be available at boot time. However, we want our system when just 50% of the mirrors in vg00 are available. Thus, we must disable the default quorum check by changing the auto file boot string to boot vmunix –lq on both disks. You could do this from the EFI shell, but we'll use the HP-UX efi_cp command to avoid rebooting.

```
# vi /tmp/auto
  boot vmunix –lq
# efi_cp –d /dev/rdsk/c0t0d0s1 /tmp/auto /efi/hpux/auto
# efi_cp –d /dev/rdsk/c0t1d0s1 /tmp/auto /efi/hpux/auto
```

3. Verify the contents of the system partition.

```
# efi_ls –d /dev/rdsk/c0t1d0s1
```

4. Verify the contents of the auto file in the system partition.

```
# efi_cp –d /dev/rdsk/c0t1d0s1 –u /efi/hpux/auto /tmp/efi.auto
# cat /tmp/efi.auto
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 7: (Optional) Configure the Mirrored Boot Disk's OS Partition

Next, configure the OS partition. Recall that the OS partition is essentially an encapsulated LVM boot disk.

1. Initialize and add the new OS partition to vg00. Be sure to specify s2, the OS partition!

```
# pvcreate -fB /dev/rdsk/c0t1d0s2
# vgextend vg00 /dev/dsk/c0t1d0s2
```

2. Mirror *all* vg00 logical volumes. If your vg00 has non-default logical volume names, you may need to manually lvextend each logical volume.

```
# for lv in /dev/vg00/lvol*
  do
     lvextend -m 1 $lv
  done
```

3. Add a line to /stand/bootconf so SDUX knows which disks are boot disks. When patches are swinstall'ed for LIF area utilities, SDUX consults /stand/bootconf to determine which disks have LIF areas that might need to be patched. Be sure to specify s2, the OS partition!

```
# vi /stand/bootconf
    l /dev/dsk/c0t0d0s2
    l /dev/dsk/c0t1d0s2
```

4. Verify that the disk was added to vg00, and that the logical volumes are sync'ed. The LV Status field for each logical volume should report syncd.

```
# vgdisplay -v vg00
```

5. Verify that the BDRA was updated properly. You should see references to both disks in the boot, root, and swap portions of the output.

```
# lvlnboot -v
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 8: (Optional) Add the Mirrored Boot Disk to the EFI Boot Manager Menu

Finally, add the new mirrored disk to the EFI boot manager menu. This ensures that the EFI will automatically attempt to boot the mirrored boot disk if the primary boot disk fails.

1. You can do this via the EFI bcfg command, but we'll use the HP-UX setboot command instead to avoid rebooting. Update the EFI boot manager primary, high availability, and alternate boot paths.

```
# setboot -p 0/1/1/0.0.0
# setboot -h 0/1/1/0.1.0
# setboot -a 0/0/2/0.0.0.0
```

2. Verify that the boot paths are configured properly in NVRAM.

```
# setboot
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 9: (Optional) Booting from the Mirrored Boot Disk

After you complete the mirroring process, it is a good idea to verify your work by booting from the new mirror.

1. Initiate the boot sequence.

```
# shutdown –ry 0
```

2. Interrupt the EFI boot manager autoboot.

```
EFI Boot Manager ver 1.10 [14.60] Firmware ver 1.61 [4241]
[Escape]
```

3. Select your mirrored disk from the boot manager selection menu.

```
EFI Boot Manager ver 1.10 [14.60] Firmware ver 1.61 [4241]
Please select a boot option
   HP-UX Primary Boot
   HP-UX Alternate Boot
   EFI Shell [Built-in]
```

4. Verify which disk you booted from when the boot process completes.

```
# grep "Boot device's HP-UX HW path" /var/adm/syslog/syslog.log
  vmunix: Boot device's HP-UX HW path is: 0.0.0.0.1.0
```


© 2004 Hewlett-Packard Development Company, L.P.

18–8. LAB: Kernel Configuration

Directions

Follow the directions below carefully.

Part 1: Preliminary Steps

A portion of this lab requires you to interact with the ISL and boot menus, which can only be accomplished via a console login. If you are using remote lab equipment, access your system's console interface via the GSP/MP.

Part 2: Managing Kernel Drivers and Subsystems with SAM

The kernel requires device drivers to communicate with devices on your system. Without the proper driver, the kernel is unable to access a device. As you add new devices and interface cards to your system, you may need to install new drivers. Removing unused drivers may save some space in memory.

1. SAM is the easiest tool for managing kernel drivers. Go to the SAM kernel driver screen by selecting: **`SAM --> Kernel Configuration --> Drivers`** This should give you a list of all the drivers currently installed on your system. The **`Current State`** column indicates which of the drivers are actually configured in your kernel. The **`Pending State`** column indicates which drivers will be included in the next kernel rebuild. Based on this list, can your kernel successfully communicate with a tape drive that requires the stape driver? Can you dynamically load or unload the stape driver without rebooting, or is it a **static** kernel driver that requires a reboot?

   Answer

   The **`Current State`** of the stape driver may be either In or Out. The "Type" column should show that this is a "Static" driver that would require a reboot.

2. Select a "static" driver that isn't yet configured in your kernel, and use the Actions menu to add the driver to the kernel.

   Answer

   Select **`lpr3`** (or any other static driver) from the driver object list.

   Select **`Actions --> Add Driver to Kernel`**.

3. What changed on the SAM screen to indicate that the static driver you selected will be included in the next kernel rebuild?

   Answer

   The **`Pending State`** changes from "Out" to "In". However, since this is a static driver, the "Current State" remains "Out" until the kernel is rebuilt.

© 2004 Hewlett-Packard Development Company, L.P.

4. Dynamically load the **krm** DLKM driver:

   - Select the **krm** driver from SAM's list of available drivers.
   - Select **Actions --> Add driver to kernel**.

   You should be sitting in the **Set Kernel Module Attributes** window at this point. DLKM drivers may be statically or dynamically loaded in the kernel. If you set the module type to Static, the new module will become available only at the next kernel build and reboot.

   If you choose **Loadable**, the driver becomes available immediately. If you choose **Load automatically at boot**, the kernel will load the DLKM automatically during the boot process. Otherwise, the kernel will load only the DLKM as needed.

   - Ensure that the Module Type is **Loadable**.
   - Ensure that Load Automatically at Boot is set to **No**.
   - If you changed either of the settings above, click **Modify**.
   - Click **OK** to make the changes take effect.

5. What changed on the SAM screen to indicate that the module is immediately available for use by devices on your system?

   Answer

   The Current State and Pending State columns both changed to "In."

6. You can also use SAM to manage kernel subsystems. Choose **List --> Subsystems**. Is the diagnostic subsystem configured? Is the fibre channel subsystem configured?

   Answer

   The **Current State** field for these subsystems may vary from system to system.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Tuning a Kernel Parameter

1. Choose **`List --> Configurable Parameters`**. What is the maximum number of processes your kernel can support? How many files can be open simultaneously?

   Answer

   Look for the **`nproc`** and **`nfile`** parameters. The parameter values will vary.

2. Double the value of the maxuprc kernel parameter, which defines the maximum number of concurrent processes a user can run. Which fields in the object list changed as a result of your actions? Current State? Pending State? Or both? Why?

   Answer

   Select **`maxuprc`**.
   Select **`Actions --> Modify Configurable Parameter`**.
   Double the parameter value.
   Click OK.

   Since **`maxuprc`** is a dynamically tunable parameter, both fields change immediately.

3. Look a bit further down the list of parameters until you find maxusers. Double the value of this kernel parameter. Does this change take effect immediately? Why?

   Answer

   Select **`Maxusers`**.
   Select **`Actions --> Modify Configurable Parameter`**.
   Double the parameter value.
   Click OK.

   The Pending Value changes immediately, but the Current Value doesn't since this is a static kernel parameter.

4. After changing the value of **`maxusers`**, note that the pending value for **`nproc`** changed as well. Select the **`nproc`** parameter and select **`Actions --> Modify Configurable Parameter`**. How does the definition of this parameter differ from the **`maxusers`** parameter? Why do you think **`nproc`** is defined in this manner?

   Answer

   `nproc` is a calculated value, based on the value of **`maxusers`**. Since the number of processes allowed on the system largely depends on the number of users, this scheme makes some sense. Note that you can modify or replace the formula if you wish.

© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Rebuilding the Kernel

1. Now that you have marked several pending changes to the kernel, choose **Actions --> Process New Kernel**. Before moving the kernel into place, look at the next question.

   Answer

   Select **Actions --> Process New Kernel**.

   Wait a few minutes while the new kernel is being processed.

2. Shortly, SAM will ask if you want to move the new kernel into place and reboot. SAM built your new kernel in the **/stand/build** directory. Which three files or subdirectories in **/stand/build** must be moved into place in **/stand** before rebooting with the new kernel?

   Answer

   ```
   /stand/build/system.SAM
   /stand/build/vmunix_test
   /stand/build/dlkm.vmunix_test
   ```

3. Allow SAM to move the new kernel into place and reboot.

   Answer

   Select **Move New Kernel into Place and Shutdown/Reboot System Now**.
   Check **Overwrite /stand/system**.
   Click **OK**.
   Click **OK** again to confirm that you wish to overwrite **/stand/system**.

4. When your system returns, try the following:

   ```
   # lsdev              # Lists currently configured static kernel drivers.
   # kmtune | more      # Lists your currently running kernel's tunable parameters.
   ```

   Was your new static kernel driver included in the new kernel? Did the value of **nproc** increase?

   Answer

   Hopefully your changes were successful.

© 2004 Hewlett-Packard Development Company, L.P.

Part 5: Booting from the Backup Kernel

Hopefully, your new kernel will boot successfully and bring you up to multiuser mode. In some cases, however, the parameters in your new kernel may conflict and cause the boot to fail. Fortunately, SAM keeps a backup of your previous kernel in **/stand** called **/stand/vmunix.prev.**

1. Reboot your system to single user mode using the backup kernel.

   Answer

   ```
   # shutdown -ry 0
   (press the [Escape] key to interrupt the boot process)
   Main Menu> boot pri isl
   Interact with IPL (Y, N, or Cancel)?> y
   ISL> hpux ls
   ISL> hpux -is /stand/vmunix.prev
   ```

2. Use **lsdev** and **kmtune** to check your parameter values and driver list again. Which kernel appears to be running? (Note: The **lsdev** and **kmtune** executables are in **/usr/sbin**, so you will need to do a **mount -a** first.)

   ```
   # mount -a
   # kmtune | more
   # lsdev
   ```

   Hopefully, the driver and parameter lists reflect the original kernel, not the kernel you tuned in the earlier part of this lab.

3. Move your backup kernel and associated files back into place as the default kernel and reboot again.

   Answer

   ```
   # kmupdate /stand/vmunix.prev
   # mv /stand/system.prev /stand/system
   # shutdown -ry 0
   ```


© 2004 Hewlett-Packard Development Company, L.P.

19–28. LAB: Configuring the HP-UX 11i v2 Kernel

Directions

Carefully follow the instructions below.

Preliminary Steps

1. Portions of this lab require console access. Login as root via your server's MP console interface.

2. Portions of this lab require access to a web browser, too. If you are accessing your server directly via a PC desktop, simply launch Internet Explorer from your Start menu. If you are accessing your server from another UNIX system, you can use your client's Mozilla browser.

3. If your instructor tells you to disable your browser's proxy settings, do so now. Ask your instructor for details.

Part 1: Configuring and using kcweb

Our goal in this portion of the lab is to configure, launch, and navigate the **kcweb** interface.

1. In your console window, enable the **kcweb** autostart functionality.

   Answer

   ```
   # waconf -a on
   ```

2. Point your desktop browser to the **kcweb** autostart URL, **http://*server*:1110/casey/top.cgi**.

   Answer

   This should work!

3. You may see a dialog box noting that the server is using a certificate issued by "a company you haven't chosen to trust". Click "Accept this certificate permanently" when asked if you wish to proceed.

4. When prompted for your username and password, enter the root username and password.

5. Logout out of the **kcweb** interface.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Managing Kernel Modules

When you add a new interface card or choose to implement a previously unused HP-UX feature, it sometimes becomes necessary to add a kernel module to your kernel configuration. This portion of the lab provides an opportunity to add and remove both static and dynamic kernel modules via **kcweb**.

1. Before you make any changes to the kernel configuration, it's a good idea to save a copy of your current configuration. Call the configuration copy "**OldConfig**".

   Answer

   ```
   # kconfig -s OldConfig
   ```

2. If you didn't launch **kcweb** in the previous part of the lab, do so now.

   Answer

   ```
   # mozilla http://server:1110/casey/top.cgi
   ```

3. View the list of kernel modules that are on your system.

   a. Does your kernel contain the **sdisk** module required to support SCSI disks?
   b. Does your kernel contain the **stape** module required to support SCSI tapes?
   c. Does your kernel contain the **nfs_server** module required to support NFS servers?

   Answer
   – From the kcweb navigation pane, click modules
   – Review the **state** column

   Most likely, all of these modules will be in the static state, which indicates that they are configured and available as part of the static kernel.

4. The **LCentIf** module supports parallel interfaces. You probably don't have a parallel interface on your Integrity server, so change its state to **unused** to remove it from the kernel.

   Answer

   – From the **kcweb** navigation pane, click **modules**
   – Click **LCentIf** from the module list
   – Click the **modify module state** button
   – In the resulting dialog box, click **unused** in the **next boot** field
   – Click the **ok** button to return to the module list

© 2004 Hewlett-Packard Development Company, L.P.

5. Was the module removed from the running kernel immediately? Explain.

   Answer

   No, the module's **state** remains unchanged, though its **next boot** state should change immediately. Since **LCentIf** is a static module, the module won't be removed until the next reboot.

6. The **cdfs** module, which supports the CDFS file system, is the first major module to fully implement HP-UX 11i v2's DLKM functionality. View the detailed description of the **cdfs** module by selecting it from the module list in **kcweb**.

   a. What is the current state of the module?
   b. What other states are supported?

   Answer

   – From the **kcweb** navigation pane, click **modules**
   – Click **cdfs** from the module list
   – Look for the capabilities field at the bottom of the screen

      a. Looking at the current **State** field, you should see that the current state is probably **auto**, which the module developer has chosen as the **best** default state for the module.
      b. The module also supports the **static**, **loaded**, and **unused** states.

7. The initial state of the module should be **auto**, which indicates that the module will automatically be loaded when needed. However, you can force the module to be loaded immediately by changing the state to **loaded**. Try it!

   Answer

   – From the **kcweb** navigation pane, click **modules**
   – Click **cdfs** from the module list
   – Click the **modify module state** button
   – In the resulting dialog box, click **loaded**
   – Click **change immediately**
   – Click the **ok** button to return to the module list

8. Check the state again. What is the current state now? Did you have to reboot? What is the current state cause?

   Answer

   Hopefully, the module state changed to **loaded** as a result of an **explicit** state change request from the system administrator (you).

© 2004 Hewlett-Packard Development Company, L.P.

9. Can you unload a module that's already loaded? Try it! Change the **cdfs** module's state to **unused**. Then review the verbose **kcmodule** output again. Can you think of a situation that might prevent the administrator from marking the **cdfs** module **unused**?

   Answer

   - From the **kcweb** navigation pane, click **modules**
   - Click **cdfs** from the module list
   - Click the **modify module state** button
   - In the resulting dialog box, click **unused**
   - Click **change immediately**
   - Click the **ok** button to return to the module list

   This should work! If, however, a CDROM was mounted, the command would have failed.

10. In the last two questions, you should have discovered that the **cdfs** module can be dynamically loaded and unloaded from the running kernel, without rebooting the system. Thus, if one had to install a CDFS patch on the system, one could temporarily unload the module from the running kernel, patch the module, then reload it – no reboot required!

    Return the **cdfs** module to the best/auto state.

    Answer

    - From the **kcweb** navigation pane, click **modules**
    - Click **cdfs** from the module list
    - Click the **modify module state** button
    - In the resulting dialog box, click **auto**
    - Click **change immediately**
    - Click the **ok** button to return to the module list

NOTE:	The next couple questions require a CD/DVD drive with a loaded CD/DVD. Only proceed to questions #10 & #11 if you have a CDROM/DVD drive with a CD/DVD. However, everyone should complete #12 before moving on to the next part of the lab.

11. (Optional) Currently, the **cdfs** module state should be **auto**. Mount the CDROM, then click the **Reload** or **Refresh** button in your web browser to see what happened! Note the current **state** and the **cause** field.

    Answer

    ```
    # ioscan -funC disk
    # mount /dev/dsk/cxtxdx /cdrom
     Reload or Refresh
    ```

    The module state changed to loaded! The **cause** should be **auto**.

© 2004 Hewlett-Packard Development Company, L.P.

12. (Optional) Can you change the **cdfs** module's state to **unused** while a CDROM is mounted? Try it!

Answer

- From the **kcweb** navigation pane, click **modules**
- Click **cdfs** from the module list
- Click the **modify module state** button
- In the resulting dialog box, click **unused**
- Click **change immediately**
- Click the **ok** button to return to the module list

The change is delayed until the next boot since the module is currently in-use.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Managing Tunable Parameters

This part of the lab gives you an opportunity to experiment with 11i v2 tunable parameters. You may complete this portion of the lab using **kcweb**.

1. Before you make any changes to the kernel configuration, it's a good idea to save a copy of your current configuration. Call the configuration copy "**OldConfig**". If you created an **OldConfig** backup earlier in the lab, you can skip this step.

   Answer

   ```
   # kconfig -s OldConfig
   ```

2. First let's consider the **nswapdev** kernel parameter. Have a look at the man page for this parameter. What does this parameter do? When might it be necessary to change it?

   Answer

   ```
   # man nswapdev
   ```

   This parameter determines the maximum number of devices that can be enabled for device swap. If you have many swap devices, it may be necessary to increase this parameter.

3. Increase the **nswapdev** parameter by 10. Does this change take effect immediately?

   Answer

   - From the **kcweb** navigation pane, click **tunables**
   - Click **nswapdev** from the tunable list
   - Click the **modify nswapdev** button
   - In the resulting dialog box, increase the **next boot** value by 10
   - Click **ok**

   The change is delayed until nextboot since this isn't an **immediate** tunable parameter.

4. Now review the man page for **maxuprc**. What does this parameter do? When might it be necessary to change it?

   Answer

   ```
   # man maxuprc
   ```

   This parameter determines the maximum number of processes a non-root user can run at any given time.

© 2004 Hewlett-Packard Development Company, L.P.

5. Increase the **maxuprc** parameter by 50. Does the change take effect immediately?

   Answer

   – From the **kcweb** navigation pane, click **tunables**
   – Click **maxuprc** from the tunable list
   – Click the **modify maxuprc** button
   – In the resulting dialog box, increase the **next boot** value by 50
   – Click **change immediately** in the mode of change field .
   – Click **ok**

   The change does take effect immediately if you specify **change immediately**.

6. What happens if you accidentally add a few extra zeros on the end of the **maxuprc** value you entered in the previous question? Try it!

   Answer

   – From the **kcweb** navigation pane, click **tunables**
   – Click **maxuprc** from the tunable list
   – Click the **modify maxuprc** button
   – In the resulting dialog box, change the **next boot** value to 500000
   – Click **change immediately** in the mode of change field.
   – Click **ok**

   HP-UX 11i v2 automatically does range checking to ensure that the new parameter value is within acceptable boundaries. **maxuprc** must be less than **nproc-5**, so this command will probably fail.

7. Since we don't have any applications running on the system, the current value of **maxuprc** is probably too high. Drop **maxuprc** back to its default value.

   Answer

   – From the **kcweb** navigation pane, click **tunables**
   – Click **maxuprc** from the tunable list
   – Click the **modify maxuprc** button
   – In the resulting dialog box, click the **default** button
   – Click **change immediately** in the mode of change field.
   – Click **ok**

8. Reboot your system to make sure the new kernel configuration boots successfully.

   Answer

   ```
   # shutdown -ry 0
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Managing Kernel Configurations

1. Assuming your updated kernel configuration booted properly, save a copy of the updated configuration. Call it **NewConfig**. Assign a title to the new configuration, too: "**Approved new kernel configuration**".

   Answer

   ```
   # kconfig –s NewConfig
   # kconfig –t NewConfig "Approved new kernel configuration"
   ```

2. After testing the configuration thoroughly, you may choose to apply the configuration to your production server. Export a system file based on your current configuration. Name the file **system.NewConfig**.

   Answer

   ```
   # kconfig –e /tmp/system.NewConfig
   ```

3. View the resulting system file with the **more** command.

   Answer

   ```
   # more /tmp/system.NewConfig
   ```

4. What would be required to apply this system on the production server? (Record the commands below, but don't actually execute them.)

   Answer

   One would have to import the file to create a new configuration on the production server, then load it. Here are the commands:

   ```
   # kconfig –i NewConfig /tmp/system.NewConfig
   # kconfig –l NewConfig
   ```

5. As it turns out, management decided not to install the new application after all. Load the **OldConfig** configuration that you created at the beginning of the exercise, then reboot.

   Answer

   ```
   # kconfig –l OldConfig
   # shutdown –ry 0
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 5: Logging and Troubleshooting

1. Let's undo all of the parameter changes you made in the previous parts of the lab. Reboot your system. When you get to the HPUX OS loader stage, terminate the autoboot sequence.

   Answer

   ```
   # cd /
   # shutdown -ry 0
   HP-UX Boot Loader for IA64  Revision 1.71
   Press Any Key to interrupt Autoboot
    \EFI\HPUX\AUTO ==> boot vmunix
    Seconds left till autoboot - 10  Escape
    Type 'help' for help
   HPUX>
   ```

2. Boot to tunable maintenance mode. What distinguishes tunable maintenance mode from a normal boot?

   Answer

   ```
   HPUX> boot vmunix -tm
   ```

   When booting to tunable maintenance mode, all tunable parameters are set to the failsafe values defined by the module developers. These failsafe parameters are designed to ensure a successful boot to single-user mode so the administrator can troubleshoot kernel problems.

3. Once you reach tunable maintenance mode, mount all of your file systems. Then use the **`more`** command to review recent kernel changes that were recorded in the kernel change log.

   Answer

   ```
   # mount -a
   # more /var/adm/kc.log
   ```

4. Reboot your system again. This time, interrupt the autoboot process, list the contents of **`/stand`**, and boot from the **`OldConfig`** configuration that you created back in Part 2.

   Answer

   ```
   # reboot
   HP-UX Boot Loader for IA64  Revision 1.71
   Press Any Key to interrupt Autoboot
    \EFI\HPUX\AUTO ==> boot vmunix
    Seconds left till autoboot - 10  Escape
    Type 'help' for help
   HPUX> ll
   HPUX> boot OldConfig
   ```


© 2004 Hewlett-Packard Development Company, L.P.

20–9. LAB: Hands-On, Using the Software Distributor

Directions

Your instructor will tell you the address of a depot that you should use for this lab exercise, and a product that you should install. Record this information below:

Depot hostname: _____

Depot pathname: _____

Perform all the tasks suggested below. Record the commands you use, and answer all the questions.

1. Remove the "perl" product from your student machine. Any errors or problems?

   Answer

   ```
   # swremove
     View --> Change Software View --> Start with Products
     (Select the suggested product)
     Actions --> Mark for Remove
     Actions --> Remove Analysis
   ```

   (After the analysis completes, click OK to remove the product.)

2. Do an **`swlist`** to see if the perl product was successfully removed.

   Answer

   ```
   # swlist -l product | grep -i perl
   ```

3. Now reinstall the perl product from the depot server suggested by your instructor. Will the install require a reboot? Must any dependencies be met?

   Answer

   ```
   # swinstall -s servername:/mnt/depotpath
   View --> Change Software View --> Start with Products
   (Select "perl" from the software list)
   Actions --> Show Description of Software


   (Close the description window)
   Actions --> Mark For Install
   Actions --> Install Analysis

   (After the analysis, click OK and start the install)
   File --> Exit
   ```


© 2004 Hewlett-Packard Development Company, L.P.

4. Verify that perl is correctly installed on your machine.

   Answer

   ```
   # swverify perl
   ```

5. Create your own local perl depot by **swcopy**ing the perl product from the depot suggested by your instructor to your own local **/var/spool/sw/perldepot** depot.

   Answer

   ```
   # swcopy -s depothost:/depotpath

   (enter the info requested in the source and target dialog boxes)
   View --> Change Software View --> Start with Products

   (Select "perl" from the software list)
   Actions --> Show Description of Software

   (Close the description window)
   Actions --> Mark For Copy
   Actions --> Copy Analysis

   (After the analysis, click OK and start the install)
   File --> Exit
   ```

6. Did **swcopy** automatically register the new depot? **swlist** the depots on your machine to find out!

   Answer

   ```
   # swlist -l depot
   ```

7. What products are included in your perl depot? How can you find out?

   Answer

   ```
   # swlist -l product -d @ /var/spool/sw/perldepot
   ```


© 2004 Hewlett-Packard Development Company, L.P.

21–18. LAB: Installing Patches with SD-UX

Directions

Part 1: Preliminary Steps

1. Run the **swinstall** utility to install the EchoApp utility on your machine. EchoApp should be available in a local depot on your machine called **/labs/depots/echoapp.depot**.

```
# swinstall -s /labs/depots/echoapp.depot \
            -x autoreboot=true \
            EchoApp
```

2. List the products installed on your host and verify that EchoApp is installed? Try running **echoapp** to see what it does.

```
# swlist EchoApp
# /opt/echoapp/bin/echoapp
```


© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Installing and Removing Patches

1. Oftentimes administrators download and install patches from HP's SupportLine web site. For this lab exercise, that won't be necessary. The patch you will install, PHSS_01111, is already in your **/labs** directory. It is embedded in a patch package called **/labs/patches.tgz**. To start, copy the patch package to the **/tmp** directory.

   ```
   # cp /labs/patches.tgz /tmp
   ```

2. **cd** to the **/tmp** directory, unzip, and untar the patch package. What files are created as a result?

   ```
   # cd /tmp
   # gzip -d /tmp/patches.tgz
   # tar -xvf /tmp/patches.tar
   ```

   Answer

   You should see the following files:

   ```
   /tmp/PHCO_01111
   /tmp/README_hp-ux
   /tmp/create_depot_hp-ux_11
   ```

3. Unpack the patch's shar archive. Read the **.text** file. What problem does this patch fix? Does the patch require a reboot?

   ```
   # cd /tmp
   # sh PHSS_01111
   # more PHSS_01111.text
   ```

   Answer

   The patch makes the **echoapp** output a little more interesting.
   The patch does not require a reboot.

4. Follow the directions in the patch's **.text** file to install the patch. Since we may need to remove this patch eventually, do NOT to use the **patch_save_files=false** option. Try running **echoapp** again. Did the patch seem to work?

   ```
   # swinstall -x autoreboot=true \
               -x patch_match_target=true \
               -s /tmp/PHSS_01111.depot

   # /opt/echoapp/bin/echoapp
   ```

   Answer

   The output should look a little different.

© 2004 Hewlett-Packard Development Company, L.P.

5. **swinstall** keeps copies of all files that have been replaced by patches in the **/var/adm/sw/save** directory. Use the find command to take a look at the PHSS_01111 files under this directory. Which two files were replaced by the PHSS_01111 patch?

   ```
   # find /var/adm/sw/save/PHSS_01111
   ```

   Answer

   /echoapp/doc/echoapp_pic and **/opt/echoapp/bin/echoapp** were replaced by the patch.

6. Now run **swremove** to remove the patch. Again look at the files under **/var/adm/sw/save**. What changed?

   ```
   # swremove -x autoreboot=true PHSS_01111
   # find /var/adm/sw/save/PHSS_01111
   ```

   Answer

   Note that the **/var/adm/sw/save/PHSS_01111/** directory is gone.

7. Now remove the EchoApp product as well.

   Answer

   ```
   # swremove -x autoreboot=true EchoApp
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Installing and Removing Patches with Applications

1. Let's try installing EchoApp again, but this time install the product from **`/labs/depots/echoapp+patch.depot`**. Note that this depot contains both the EchoApp product and the PHSS_01111 patch.

   ```
   # swinstall -s /labs/depots/echoapp+patch.depot \
               -x autoreboot=true \
               EchoApp
   ```

2. Run **`echoapp`**. Based on this experiment, what is the advantage of having products and their associated patches in the same depot? (Note: This feature is new at 11.x.)

   ```
   # /opt/echoapp/bin/echoapp
   ```

   Answer

   When installing a product, **`swinstall`** automatically selects and installs the associated patches, too.

3. Now remove the EchoApp product. Afterwards, use **`swlist`** to see if PHSS_01111 is still installed on your system. When you remove a product, what happens to the patches associated with that product?

   ```
   # swremove -x autoreboot=true EchoApp
   ```

   Answer

   When a product is removed with **`swremove`**, all patches associated with that product are automatically removed as well.

4. Confirm your conclusion in the previous question by running swlist.

   ```
   # swlist EchoApp PHSS_01111
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Committing Patches

1. Install EchoApp and the PHSS_01111 patch again from the **echoapp+patch.depot**.

```
# swinstall -s /labs/depots/echoapp+patch.depot \
            -x autoreboot=true \
            EchoApp
```

2. As you install patches on your system, the **/var/adm/sw/save** directory can consume a significant amount of disk space. You can save some space in **/var/adm/sw/save** by committing a patch. Try committing the PHSS_01111 patch. What effect does the **swmodify** command have on the **/var/adm/sw/save** directory? Use find and **du** to find out.

```
# find /var/adm/sw/save/PHSS_01111
# du -sk /var/adm/sw/save
# swmodify -x patch_commit=true 'PHSS_01111.*'
# find /var/adm/sw/save/PHSS_01111
# du -sk /var/adm/sw/save
```

Answer

Committing the patch removes all of the saved files for the patch, thus saving disk space in **/var**.

3. Try to remove the patch. What appears to be the downside of committing a patch?

```
# swremove -x autoreboot=true PHSS_01111
```

Answer

The command fails! After a patch is committed, it can no longer be removed.

4. Can you still remove the product associated with the patch? Try it. Then **swlist** both the EchoApp product and the patch.

```
# swremove -x autoreboot=true EchoApp
# swlist EchoApp PHSS_01111
```

Answer

After a patch is committed, you can't remove the patch. However, you can still remove the product, which removes the patch as well.

© 2004 Hewlett-Packard Development Company, L.P.

22–13. LAB: Installing HP-UX

Directions

Your instructor will provide you with a new IP address, hostname, and netmask. You should also be given the IP address of an Ignite Server. Record these addresses below:

IP Address: _____

Hostname: _____

Netmask: _____

Ignite Server's IP: _____

1. This lab requires you to interact with the boot menus, which can only be accomplished on the system console. If you are using remote lab equipment, access your system's console interface via the GSP/MP.

2. Install the "HP-UX Mission Critical OE" from the Ignite server suggested by your instructor. If DHCP doesn't provide your network configuration information for you, use the parameters you recorded above.

© 2004 Hewlett-Packard Development Company, L.P.

A–10. LAB: Connecting to the Network

Directions

In the exercise that follows, you will have an opportunity to explore the **set_parms** and SAM screens used to set network parameters on your system. Back in your shop, your IT department or ISP should provide you with your basic networking parameters. For this lab exercise, your instructor will provide the addresses you need:

```
                          Hostname:
                        IP address:
                       Subnet mask:
Def. Router/Gateway IP address:
                        DNS domain:      (optional)
                DNS server address:      (optional)
  Secondary DNS server address:          (optional)
```

Part 1: Preliminary Steps

1. In just a moment, you will be asked to run a script which will temporarily un-configure your system's default LAN interface. If you are using remote lab equipment, log into your system's console interface via the GSP/MP to ensure that you are able to maintain connectivity throughout the lab.

2. There should be a shell script on your system called **/labs/clobbernet.sh**. Run the script. The script clobbers your network configuration files and reboots your machine. When your machine comes back up again, it will automatically run the **set_parms** utility, just as it would if you had just installed the operating system.

   ```
   # /labs/clobbernet.sh
   ```


© 2004 Hewlett-Packard Development Company, L.P.

Part 1: Configuring an IP Address and Hostname

1. set_parms first asks if you want to connect to the network. For this lab exercise, we do want to connect to the network, so click "OK."

   Answer

   Click **OK**.

2. Next, **set_parms** asks if you wish to obtain your network parameters via DHCP. DHCP is a service that can automatically provide your machine with an IP address and hostname, if a DHCP server exists on your network. For this exercise, click "NO".

   Answer

   Click **NO**.

3. Answer the prompts in the screens that follow to set your hostname, time zone information, and IP address. When asked if you wish to set additional network information (for example, a default gateway, netmask, and DNS and NIS information), Answer NO. We will configure these parameters later via SAM.

NOTE:	If you make a mistake while entering any of the network parameters, you can run **set_parms** again any time by typing **set_parms initial**.

   Answer

   Click **NO**.

© 2004 Hewlett-Packard Development Company, L.P.

Part 2: Checking Your IP Address and Setting Your Netmask

1. After running **set_parms** and logging in, go to:

   SAM --> Networking and Communications --> Network Interface Cards

   to ensure that your IP is set correctly. What hardware slot number is your lan0 card installed in? Is lan0 enabled for use? What is lan0's IP address? Are there any other LAN cards on your machine?

   Answer

   Answers will vary, depending on the system configuration.

2. While still at the Network Interface Card screen, set your subnet mask. Without the subnet mask, you may not be able to communicate with other nodes on your LAN. Set your subnet mask for lan0 via SAM.

   Each machine on a network, regardless of the number of LAN interface cards, has a single hostname. If you have multiple LAN cards on a host, SAM requires you to configure a unique "Host Name Alias" for each LAN card. This is useful in troubleshooting situations where you may wish to access a specific LAN card on a target machine rather than the general hostname. If SAM asks you to set an alias, simply use your hostname with an "a" appended to the end.

   Answer

   ```
   SAM --> Networking and Communications --> Network Interface Cards
   ```

   Select **lan0**.

   ```
   Actions --> Modify
   ```

   Enter the subnet mask.

   Click **OK**.

   SAM may still prompt you to add a hostname alias as described above.

© 2004 Hewlett-Packard Development Company, L.P.

Part 3: Testing Your Network Connection

1. Try pinging your own IP address to see if your IP address is set properly. Did this work?

   Answer

   ```
   # ping 128.1.1.1   # use your IP address.
   ```

   This should work.

2. Next, try pinging a couple of neighboring machines' IP addresses. Did this work?

   Answer

   ```
   # ping 128.1.1.2   # use your neighbors' IP addresses.
   # ping 128.1.1.3
   ```

   This should work, too.

3. Next, try pinging a neighboring machine using its hostname. Can you guess why this fails? (Hint: Look in the **`/etc/hosts`** file.)

   Answer

   The **`/etc/hosts`** file doesn't contain an entry for your neighbor's machine yet.

© 2004 Hewlett-Packard Development Company, L.P.

Part 4: Configuring `/etc/hosts`

You won't be able to access other hosts by hostname until you add entries for those machines to your **`/etc/hosts`** file. SAM offers the easiest method for viewing and modifying your hosts file. Use SAM to Answer the questions that follow.

1. Which hosts' entries are configured so far?

   Answer

   ```
   SAM --> Networking and Communications
       --> Hosts
       --> Local Hosts File
   ```

   `localhost` is a special hostname that always references your local machine. This entry should exist in every **`/etc/hosts`** file.

   The other entry in your **`/etc/hosts`** file defines your own hostname and IP address. Since this entry exists, you should be able to ping yourself by hostname.

2. Add entries in your hosts file for the other machines in your row.

   Answer

   ```
   Actions --> Add
   ```

   Enter your neighbor's IP and hostname.

   Click **`OK`**.

3. Try pinging your neighbors by hostname again. Does this work?

   Answer

   ```
   # ping neighbor      # Neighbor's hostname will vary.
   ```

   This should work.

© 2004 Hewlett-Packard Development Company, L.P.

Part 5 (Optional): Configuring a Default Gateway and DNS

If you want to be able to access other networks or the Internet, you will have to define a default gateway and DNS server address. Both of these addresses can be configured via SAM.

1. Use SAM to configure your default gateway.

   Answer

   ```
   SAM     --> Networking and Communications
           --> Hosts

   Actions --> Configure Default Gateway
   ```

   Enter the Default Gateway address suggested by your instructor

   Optionally, you can enter the hostname of the gateway.

   Click **OK**.

2. Use SAM to configure your default DNS domain name. Don't configure any "Other Domains to Search".

   Answer

   ```
   SAM     --> Networking and Communication
           --> DNS/BIND
           --> DNS Resolver

   Actions --> Set Default Domain
   ```

   Enter the DNS Domain Name suggested by your instructor

   Don't enter any Other Domains to Search

   Click **OK**

3. Now use SAM to configure the IP addresses of your DNS nameservers.

   Answer

   ```
   SAM     --> Networking and Communications
           --> DNS/BIND
           --> DNS Resolver

   Actions --> Specify Name Servers
   ```

   Enter the IP address(es) of the DNS server(s) suggested by your instructor.

   Click **OK**.

© 2004 Hewlett-Packard Development Company, L.P.

Part 6 (Optional): Configure the Name Service Switch

1. Did your DNS configuration work? If DNS is configured properly on your machine, and the DNS server is functioning, you should be able to lookup hostnames of machines outside your local network. Try an **nslookup** on **www.hp.com**. What happens?

   Answer

   ```
   # nslookup www.hp.com
   ```

   Fails! "No address information available."

2. Look carefully at the output from the **nslookup** you did in the previous exercise. Can you explain why you were unable to find an IP address for www.hp.com? (Hint: What lookup source is **nslookup** using at this point?)

   Answer

   nslookup is using the **/etc/hosts** file on your machine. The local hosts file doesn't contain an entry for **www.hp.com**, so the lookup fails.

3. Configure the name service switch via SAM to ensure that your system consults the DNS server if a hostname isn't found in the local **/etc/hosts** file.

   Answer

   Select **SAM** --> **Networking and Communications**.

   Select --> **Name Service Switch**.

   Select **Hosts**.

   Select **Actions** --> **Configure Name Service Switch**.

   Set **/etc/hosts** as the first service.

   Set **DNS** as the second service.

4. After changing the nameservice switch, SAM reboots your machine. After the reboot, do an **nslookup** on **www.hp.com** again. Are you able to resolve **www.hp.com** now?

   Answer

   ```
   # nslookup www.hp.com
   ```

   This should succeed, assuming your DNS server is configured properly.

© 2004 Hewlett-Packard Development Company, L.P.

© 2004 Hewlett-Packard Development Company, L.P.