HP Computer Systems
Training Course

HP-UX 10.0 System Administration for HP 9000 Systems

Student Workbook

HEWLETT
PACKARD

Version B 8/95

H6296-90001
Printed in USA

Notice

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

Trademarks

Product Development
Professional Services Division
100 Mayfield Avenue
Mountain View, CA 94043 U.S.A.

Contents

Contents

Module 3 — Users and Their Environment

Module 4 — HP 9000 Hardware Overview

Contents

Contents

Contents

Contents

Module 13 — File System Management

Module 14 — Managing Swap Space

Module 15 — Reconfiguring the Kernel

Contents

Contents

Contents

Contents

Appendix C — DTC Management

Appendix D — Disk Quotas

Solutions

Glossary

Tables

Contents

Overview

Course Description

System administration is one of the most complex jobs on an HP-UX system, and a good system administrator is essential for the system to run efficiently and reliably. This course is designed to prepare you to be a successful system administrator of an HP 9000 server or workstation. This is a 5-day lecture and lab course.

Student Performance Objectives

Module 1 — Introduction to HP-UX Administration

- Identify responsibilities of a system administrator.
- Identify the HP-UX documentation targeted towards helping system administrators.
- Identify three sources of information for system administrators.

Module 2 — Overview of SAM

- Use SAM to locate various information regarding system configuration and resources.
- Customize SAM for special pre-defined action.
- Create your own functions in the Other Utilities Menu.

Module 3 — Users and Their Environment

- Add users to the system.
- Maintain the `/etc/passwd` and `/etc/group` files.
- Identify required entries in `/etc/passwd` and `/etc/group`.
- Change user passwords.
- Specify a user's default environment.
- Communicate with users.

Module 4 — HP9000 Hardware Overview

- List features of PA-RISC Architecture.
- Describe the components of an HP9000 PA-RISC system.
- Describe types of I/O interfaces and their functions.
- Determine the hardware configuration of the local system.

Overview

Module 5 — Device Addresses and Special Files

- Use the appropriate device file to access a peripheral.
- Describe the autoconfiguration process.
- Describe the function of a device file.
- Differentiate between block and character I/O systems.
- Determine the conventional name for a specific device.
- Describe major and minor numbers.
- Create device files with `mksf` and `insf`.
- Decide when to use the `insf` versus the `mksf` command.
- List device files with `lssf`.
- Remove device files with `rmsf`.

Module 6 — Booting Your HP-UX System

- Describe the LIF files and utilities found in the boot area.
- Describe the system boot ROM startup sequence.
- Explain the function of the ISL.
- Differentiate between autoboot and manual boot.
- Boot HP-UX in Manual mode.
- Change primary and alternate boot paths.

Module 7 — System Startup

- Explain the purpose and syntax of the `/etc/inittab` file.
- Add an entry to `/etc/inittab`.
- Explain how `init` reads the `inittab` file.
- Identify the default run-levels supplied with an HP-UX system.
- Change the default system run-level.
- List and describe the functions of these system startup files:

```
/sbin/bcheckrc
/sbin/rc
/sbin/lvmrc
```

Module 8 — System Shutdown

- Describe the importance of a proper system shutdown.
- Explain the differences between `shutdown` and `reboot`.
- Use the `shutdown` command.

- Use the `reboot` command.

Module 9 — File System Concepts

- List the different types of file system available.
- List the dynamic information in an HFS superblock.
- Describe a cylinder and a cylinder group.
- List the contents of the cylinder group information.
- Describe the inode table.
- List the parts of an inode.
- Explain three different ways that an inode references a data block.
- Explain fragment and block allocation.
- Explain the link process.

Module 10 — Managing Disk Space

- Describe disk partitioning and how it relates to the "sectioning" method of disk management.
- Describe the general features and benefits of LVM.
- Differentiate among volume groups, logical volumes, and physical volumes, and explain how they relate to each other.
- Create physical volumes, volume groups and logical volumes.

Module 11 — Creating and Using File Systems

- Create a file system.
- Mount or unmount a file system.
- Automatically mount a file system.

Module 12 — Maintaining the File System

- Explain how an HFS and a JFS file system handles file modifications.
- Identify three causes of file system corruption.
- Explain the purpose of the `sync` command.
- Check for file system corruption.
- Repair file system corruption.

Module 13 — File System Management

- Check available free disk space.
- Implement some space management procedures.
- Moving data from one disk to another.

- Extend file systems.
- Move a volume group from one system to another.
- Remove unused software.

Module 14 — Managing Swap Space

- Explain the concept of physical and virtual memories as applied to a demand page virtual memory system.
- Explain the difference between physical memory, available memory, and lockable memory.
- Identify the amount of memory your system has and how much of that memory is lockable.
- Define swap space and describe why it is necessary.
- Define various terms that describe swap types.
- Evaluate your own swap space needs.
- Set up device swap space on your system.
- Set up file system swap space on your system.
- Describe two methods for managing swap space on your system.

Module 15 — Reconfiguring the Kernel

- Describe reasons why it might be necessary to reconfigure the kernel.
- Perform the steps to reconfigure the kernel using HP-UX commands and SAM.
- Reconfigure the kernel for an additional device or subsystem.
- Describe some important tunable parameters.

Module 16 — System Backup

- Explain why backups are important, and what should be backed up.
- Describe several backup strategies for a system.
- Differentiate between different backup and restore methods.
- Backup and restore files with `fbackup/frecover`, `cpio`, `tar`, and `dd`.
- Find files with the `find` command.
- Backup the LVM configuration.

Module 17 — Job Scheduling

- Use the `at` and `batch` commands.
- Schedule programs for repetitive invocation with `cron`.

Overview

Module 18 — Printers and the LP Spooler

- Activate the LP (line printer) spooler.
- Add a serial printer to your system.
- Add a local and remote printer to the LP spooler system.
- Start and stop the LP spooler.
- Remove a printer or class of printers from your system.
- Check LP spooler status.
- Enable and disable a printer class.
- Set the default printer.
- Move printing requests to other destinations.
- Explain what print classes are.
- Describe what priority fences are and differentiate them from printer priorities and print job priorities.

Module 19 — Terminals and Modems

- List three ways to connect terminals to a system.
- Explain the function of the `getty` process.
- Configure a Modem for dialin and dialout.

Module 20 — LAN Setup Cookbook

- Describe the functionality of the LAN.
- Describe important networking terms.
- State the steps for configuring a LAN.
- Describe Basic LAN Troubleshooting techniques.

Module 21 — Installing HP-UX

- Differentiate between an installation and an update.
- Perform the steps to install HP-UX 10.0 on an HP 9000 workstation or server.

Module 22 — Installing Additional Software

- Describe the steps you should take before installing additional software.
- Perform the steps necessary to install software on your HP-UX system using the `swinstall(8)` command for an install using local media.
- List the files and directories to look through after a software installation.
- Perform the steps necessary to install software on your HP-UX system using the `swinstall(8)` command for an install from a Network Installation Server.
- Perform the steps needed to configure a Network Installation Server.

Overview

Module 23 — Introduction to HP VUE Administration

- Describe the basics of the HP Visual User Environment.
- Differentiate between HP VUE and HP VUE Lite.
- Outline the component parts of HP VUE.
- Describe the HP VUE architecture and explain the need for a Broadcast Message Server.
- Describe the system wide directory structure.
- Describe the home directory structure.
- Describe how applications are started.
- Describe and customize the HP VUE Login Manager.
- Describe and customize the HP VUE Session Manager.
- Add system-wide actions to HP VUE and HP VUE Lite.
- List basic troubleshooting files and tools provided with HP VUE.

Module 24 — Connecting an X Station

- Connect an X station to your system.
- Sett up the X station user's environment.

Module 25 — DTC Management

- Explain the function of a DTC.
- Configure a terminal to a DTC terminal port.
- Configure a printer to a DTC terminal port.
- Describe the functions of DTC Device File Access Utilities (DDFA).
- Configure a port using DTC Device File Access Utilities (DDFA).

Module 26 — Disk Quotas

- Implement disk quotas.

Student Profile and Prerequisites

This course is designed for the individual who is responsible for performing the system administration tasks on an HP 9000 server or workstation. The student should be an experienced HP-UX user and have a working knowledge of shell programming.

The student should have completed *HP-UX Fundamentals*, Course number 51434S.

Overview

Reference Documentation

- *HP-UX System Administration Tasks*, P/N B2355-90079.
- *Configuring HP-UX for Peripherals*, P/N B2355-90053.
- *Installing HP-UX 10.0 and Updating HP-UX 10.0 version B.10.00 to B.10.01*, P/N B2355-90078.
- *Upgrading from HP-UX 9.x to 10.0 Version B.10.01*, P/N B2355-90083.
- *Managing HP-UX Software with SD-UX*, P/N B2355-90080.
- *HP-UX Documentation: What is Available and How to Order It*, P/N B2355-90085.
- *Installing and Administering LAN/9000*, P/N 98194-90050.
- *Installing and Administering Internet Services*, P/NB1030-90000.
- *Installing and Administering NFS Services*, P/N B1031-90000.
- *Using Internet Services*, P/N B1030-90001.
- *HP VUE 3.5 User's Guide*, P/N B1171-90079.
- *HP VUE 3.5 Quick Start Card*, P/N B1171-90080.

Curriculum Path for System Administrators and Managers

```
HP-UX Fundamentals (5-days) ---> HP-UX System Administration (5-days)
```

Overview

Module 1 — Introduction to HP-UX Administration

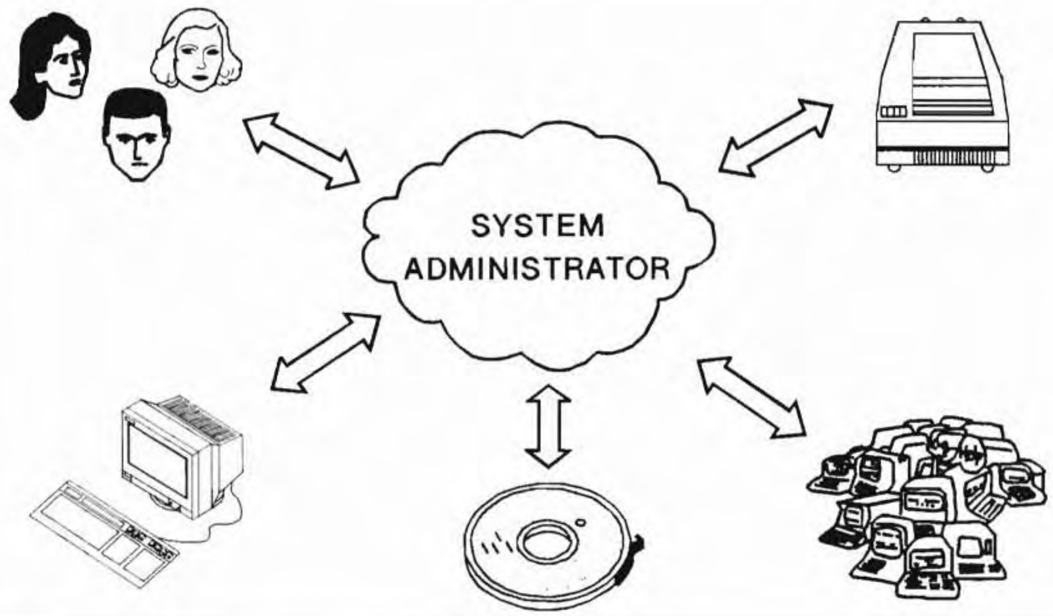
Objectives

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

- Identify responsibilities of a system administrator.
- Identify the HP-UX documentation targeted towards helping system administrators.
- Identify three sources of information for system administrators.

1-1. SLIDE: The Role of the System Administrator

The Role of the System Administrator



H6296 1-1 1

Student Notes

The system administrator is responsible for setting up and maintaining the system. Not only must the administrator understand both hardware and software, but he/she must also understand the needs of the user community.

Since many of the tasks associated with these responsibilities require access to commands that should not be available to everyone, the system administrator needs special access to the system. This access is called **superuser** or **root access**.

1-2. SLIDE: Hardware Responsibilities

Hardware Responsibilities

- Create and maintain a hardware diagram of the system
- Verify that peripherals are installed correctly and tested
- Monitor performance of hardware components
- Arrange for repair in event of hardware failure

H6296 1-2 2

Student Notes

The system administrator of an HP-UX system is responsible for configuring and managing the system hardware. The administrator may not be the person who actually installs the hardware. Often a Hewlett-Packard Customer Engineer will perform the installation of the hardware. Once the system is operational, the administrator must monitor the performance of the various hardware components. If a hardware failure occurs, the administrator should attempt to isolate the problem as much as possible. Depending upon the service agreements in place, the administrator may schedule a customer engineer to make necessary repairs.

The system administrator must know some basic things about the system hardware to be effective in his job.

1-3. SLIDE: Software Responsibilities

Software Responsibilities

- Install and configure the HP-UX operating system.
- Create file systems.
- Manage the integrity of file systems.
- Monitor system resource usage.
- Design and implement backup and recovery routines.
- Configure and maintain printer spooler software.
- Install and maintain network communication software.
- Update the HP-UX operating system for new releases.
- Install and update application software.

H6296 1-3 3

Student Notes

You may need to install the HP-UX operating system software, if it is not preinstalled on your system. The operating system is supplied on a distribution medium that can be one of many forms. The software is distributed on a series of magnetic tapes or on optical CD-ROM discs.

The distribution media may be:

- Compact Disc - Read Only Memory (CD-ROM) disc
- Digital Data Storage (DDS) tapes based on Digital Audio Tape (DAT) technology
- Quarter-Inch Cartridge Tape (QIC), (Servers only)

The type of media used varies based on the system type; we will discuss this in detail later. The system software must be installed onto a hard disk by the system administrator.

Module 1 — Introduction to HP-UX Administration

Once the HP-UX system has been installed, it will be necessary to configure the software by altering certain values or parameters. This is done to allow the operating system (HP-UX) to recognize additional devices or to increase the efficiency of the system.

HP-UX utilizes a data organization scheme called a file system. Since the file system is where all of the system and user data is stored, it is quite important to ensure that the integrity of the file system is maintained. The administrator should implement procedures that will detect any errors or corruption in the file system. If problems are found, the administrator must ensure that corrective action is taken.

A file system is of a finite size and usually resides on a locally connected disk drive. The available space in a file system must be monitored by the system administrator. Procedures should be employed to archive and/or remove obsolete and unused files so the available free space is not completely consumed. Files that tend to grow in size should also be monitored.

System resources include not only disk storage space, but memory, peripheral devices, and kernel data structures, all of which can be customized and monitored to some extent by the System Administrator.

It is the administrator's job to ensure the security of data on the system. Regular data backups are created and maintained. If there is a loss of data due to either user error or hardware failure, recovery procedures can be employed.

Most HP-UX systems include peripheral devices such as line printers and laser printers for hard copy output. The system administrator must manage the software that sends output to the printing devices.

Different HP-UX systems communicate with one another across an electronic communications mechanism called a network. The network allows electronic mail and files to be transmitted from machine to machine. Networking software must be installed, configured, and monitored by a system administrator.

Hewlett-Packard periodically releases an update to the HP-UX operating system software, and to many of the subsystems and applications programs. An update may enhance or modify existing system features or add new capabilities. The system administrator is responsible for installing each software update so the HP-UX system available to the user community contains the latest version of the software.

A new HP-UX release is not necessarily installed as soon as it is received. It is your responsibility to ensure that interactions and dependencies between applications and system software are maintained. A new HP-UX release is often followed several months later by releases for applications that make them compatible with the new HP-UX release. Installing the new HP-UX release too soon may break an application that worked on an older release. Obtaining and sharing this information is your responsibility.

1-4. SLIDE: Responsibilities to The Users

Responsibilities to The Users

- Allow user access to the system as required.
- Evaluate user needs.
- Plan for future system growth/change.
- Provide assistance to the user community.
- Implement the policies and procedures of your company/organization regarding the use of the computer system and network.

H6296 1-4 4

Student Notes

Once the HP-UX system has been installed, certain modifications are required to allow a user to access the system. These modifications must be performed by the system administrator.

The administrator must, to the greatest extent possible, tailor the system to the needs of the user community. The system administrator should analyze the intended use of the system, and should be aware of the number of users on the system, the characteristics of each user, the system resources and peripherals required by each user, and the data/programs that must be shared by various user groups.

As system administrator you will be looked upon as the resident HP-UX expert. Many users will assume that you know everything about the system and will view you as a "guru." This occurs independently of your knowledge level. To many users, the fact that you have been trained, in their minds, means that "you must know" more than they do.

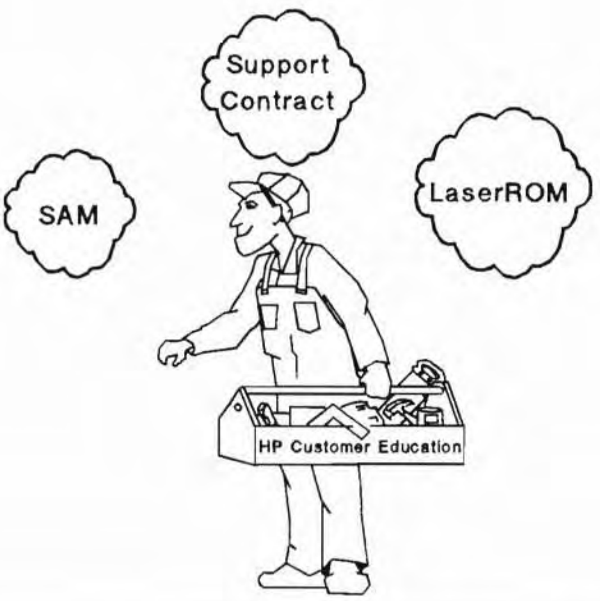
Module 1 — Introduction to HP-UX Administration

You may be asked many questions such as, "How do I do ... ?" and receive comments such as "My terminal is broken" and "I forgot my password." The problem solving and consulting aspects of the administrator's role can be the most challenging as well as the most enjoyable parts of the job. An important message to convey early on is that the HP-UX reference manuals are available on-line on an HP-UX system. Get the users used to at least trying to "look it up for themselves" before coming to you. This can minimize interruptions in your daily activities, and give the users greater confidence in using the system.

Since you will implement the policies and procedures of your company/organization, be aware that these policies and procedures take precedence over the things that HP will tell you in this class. We can only recommend certain administration practices. If our recommendations are in conflict with your company practices, clearly you have to follow your company guidelines.

1-5. SLIDE: The System Administrator's Toolkit

The System Administrator's Toolkit



H6296 1-5 5

Student Notes

The System Administrator is responsible for the smooth day-to-day operation of the system, as well as for responding to and correcting large and small emergencies that may occur. In addition, the System Administrator is usually the person responsible for making sure that in the event of a disaster of small or large magnitude, there is a plan in place to recover the system and the data.

Although this can seem like a monumental task, there are several tools available that can make the System Administrator's job easier. Some of them are noted on the slide.

SAM — The **S**ystem **A**dministration **M**anager is a user interface for performing most routine administrative tasks without using the underlying HP-UX commands. SAM can save you keystrokes. However, it is important that you understand the nature of the task you are performing, whether you use SAM or the HP-UX commands. This course will teach the concepts needed to administer an HP-UX system, and the HP-UX commands used to perform those tasks. You will also be allowed to explore using SAM to perform these same tasks.

LaserRom

LaserRom is a CD-ROM product that contains the HP-UX manuals on a CD. You can search by keyword in a set of manuals that you specify, or you may choose to browse through a specified manual or manuals.

Support Contract

Your support contract covers the hardware as well as the software on your system. There are several levels of support, which specify such things as the standard on-site hardware support, software and network assistance from the HP Response Center, and possibly other services such as assigned Account Teams, patch management assistance, operational reviews, system release planning, and assistance with software updates. If you are not familiar with the terms and features of your HP System Support Contract, contact your local HP office and ask for the Contracts Coordinator. You should know which services you are entitled to.

Shell Scripts

Shell scripts are written by the System Administrator to automate frequently performed tasks. Shell scripts are tailored specifically to your site. You can add your shell scripts to SAM if you wish.

1-6. SLIDE: HP-UX Manuals For System Administrators

HP-UX Manuals For System Administrators

Common to all HP-UX systems

- *HP-UX System Administration Tasks*
- *Installing HP-UX 10.0 Version B.10.01 and Updating HP-UX 10.0 Version B.10.00 to Version B.10.01*
- *Upgrading from HP-UX 9.x to Version B.10.01*
- *Managing HP-UX Software with SD-UX*
- *Configuring HP-UX for Peripherals*
- *HP-UX Reference*, an on-line resource

Hardware-Specifics

- *Using Your HP Workstation*
- *Owner's Guide: HP 9000 Series 800 Model E Computers*
- *Owner's Guide: HP 9000 Series 800 Model F, G, H, I Computers*
- *Owner's Guide: HP 9000 Series 800 Model J Computers*

H6296 1-6 6

Student Notes

Often during the administration of your system, you will need to reference the documentation. There are many HP-UX manuals. How do you know which manual to check? The slide lists manuals specifically targeted towards administrators. You should be familiar with these manuals and their content as they can be of tremendous help to you.

HP-UX Reference (on-line)

The **manpages** (as they are popularly known) are intended for all HP-UX systems. You may view reference material on any command, system call, subroutine, device file, or file format by using the `man(1)` command. The manpages are intended as reference material and are most useful to experienced users. They are not designed to serve as learning tools for beginners.

The manpages are divided into eight sections:

- Section 1: User Commands
- Section 1M: System Administration Commands
- Section 2: System Calls
- Section 3: Subroutines
- Section 4: File Formats
- Section 5: Miscellaneous Facilities
- Section 7: Device Files
- Section 9: Glossary

Section 1M contains information on those commands that are used primarily by a system administrator. Section 4, File Formats, is also invaluable to the system administrator as it contains information on most of the configuration files that you will be responsible for maintaining.

HP-UX System Administration Tasks Manual

This manual covers most system administration tasks in detail. Topics covered include constructing and customizing an HP-UX system, updating HP-UX, starting and stopping HP-UX, managing run-levels, managing groups, users, file systems, and clusters, and setting up and administering backups and the LP spooler.

Configuring HP-UX for Peripherals

This manual provides step-by-step instructions on how to configure peripherals such as terminals, printers, plotters, as well as disk and tape drives.

Installing HP-UX 10.0

This manual covers what its title states: how to install and modify the HP-UX operating system.

1-7. TEXT PAGE: Manual Part Numbers

This is a list of the primary manuals referenced in this class.

Table 1-1. Manuals Referenced in this Class

Part Number	Manual Title
B2355-90078	*Installing HP-UX 10.0 Version B.10.01 and Updating HP-UX 10.0 Version B.10.00 to B.10.01*
B2355-90083	*Upgrading from HP-UX 9.x to 10.0 Version B.10.01*
B2355-90079	*HP-UX System Administration Tasks*
B2355-90053	*Configuring HP-UX for Peripherals*
98194-90050	*Installing and Administering LAN/9000*
B1030-90000	*Installing and Administering Internet Services*
B1031-90000	*Installing and Administering NFS Services*
B1030-90001	*Using Internet Services*
B1171-90079	*HP VUE 3.5 User's Guide*
B1171-90080	*HP VUE 3.5 Quick Start Card*
B2355-90080	*Managing HP-UX Software with SD-UX*
B2355-90085	*HP-UX Documentation: What is Available and How to Order It*

1-8. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions in the space provided.

1. Describe the role of the System Administrator.

2. What must the System Administrator understand in order to perform her/his duties appropriately?

3. What are the three main categories of System Administration responsibilities?

4. What are two items in the System Administrator's toolkit?

Module 2 — Overview of SAM

Objectives

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

- Use SAM to locate various information regarding system configuration and resources.
- Customize SAM for special pre-defined action.
- Create your own functions in the Other Utilities Menu.

Module 2 — Overview of SAM

2-1. SLIDE: The System Administration Manager


```
SAM     Accounts for Users and Groups ->
SAM     Auditing and Security         ->
SAM     Backup and Recovery           ->
SAM     Disks and File Systems        ->
SAM     Kernel Configuration          ->
SAM     Networking and Communications ->
SAM     Peripheral Devices            ->
SAM     Printers and Plotters         ->
SAM     Process Management            ->
SAM     Routine Tasks                 ->
SAM     Run SAM on Remote Systems
SD-UX   Software Management           ->
SAM     Time                          ->
```

H6296 2-1 7

Student Notes

The System Administration Manager (SAM) is a menu-driven tool designed to perform typical system administration tasks without using the underlying HP-UX commands. The slide shows some of the tasks SAM can perform.

SAM has two user interfaces, an X Window System interface and a text terminal interface. The differences are the screen appearance and the keyboard/mouse interactions.

There are many benefits of using SAM:

- Instead of executing commands from a shell, you work through menus that guide task selection and facilitate data entry.
- Tasks are easier to perform because you need not remember (or type) complex commands.

- You get a rich set of functions, and those functions provide significant options and control.
- You can use SAM on any HP 9000 system without relearning anything.

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:

- Use SAM to administer your system whenever it has the capability you need.
- Since SAM does not accommodate every task you need to perform, learn the HP-UX commands for performing a task.
- While performing a task with SAM, if you encounter a situation that SAM cannot accommodate, exit SAM and perform the task using HP-UX commands.
- Use HP-UX commands when SAM cannot perform a task or you know (as an expert) how you want to customize a functionality.

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.

Note

SAM is an optionally loadable part of HP-UX. If you have not loaded SAM onto your system, you will not be able to use it. To use SAM on a workstation or X-terminal that is running the X Window System, you must also have loaded the necessary `X11*` filesets.

The next page shows a complete list of the tasks that SAM can perform.

2-2. TEXT PAGE: The System Administration Manager Roadmap

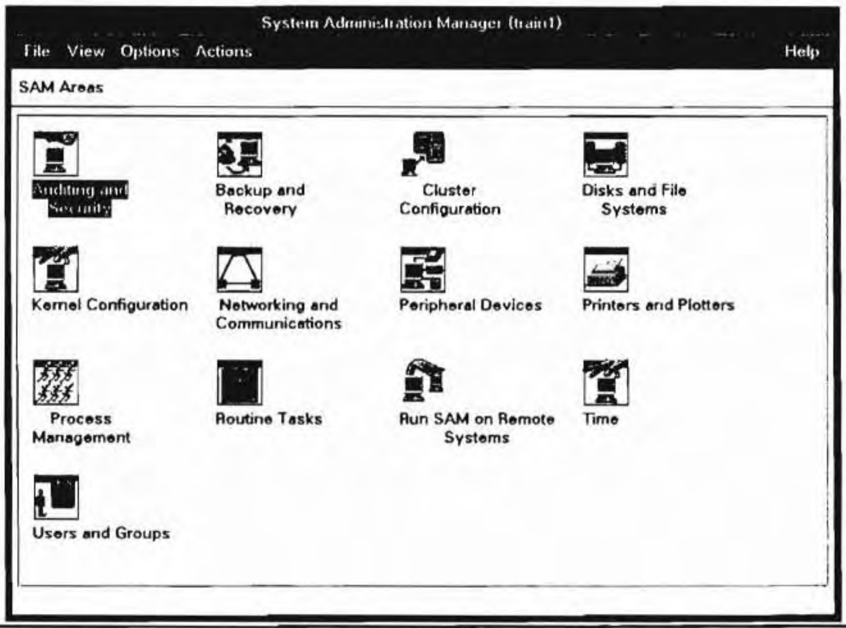
This list shows how the SAM functional areas and subareas are organized. Some of these areas are optional, and may not appear on your system if they are not configured or available on your system.

```
SAM      Accounts for Users and Groups ->
        SAM      Groups
        SAM      Users
SAM      Auditing and Security         ->
        SAM      Audited Events
        SAM      Audited System Calls
        SAM      Audited Users
        SAM      System Security Policies
SAM      Backup and Recovery           ->
        SAM      Automated Backups
        SAM      Interactive Backup and Recovery
SAM      Disks and File Systems        ->
        SAM      Disk Devices
        SAM      File Systems
        SAM      Logical Volumes
        SAM      Swap
        SAM      Volume Groups
SAM      Kernel Configuration          ->
        SAM      Configurable Parameters
        SAM      Drivers
        SAM      Dump Devices
        SAM      Subsystems
SAM      Networking and Communications ->
        SAM      Bootable Devices
        SAM      Internet Addresses
        SAM      Network Interface Cards
        SAM      Network Services
        SAM      Networked File Systems  ->
                SAM      Exported Local File Systems
                SAM      Mounted Remote File Systems
        SAM      System Access           ->
                SAM      Internet Services
                SAM      Remote Logins
        SAM      UUCP                    ->
                SAM      UUCP Devices
                SAM      UUCP Remote Systems
SAM      Peripheral Devices            ->
        SAM      Cards
        SAM      Device List
        SAM      Disks and File Systems          ->
                SAM      Disk Devices
                SAM      File Systems
                SAM      Logical Volumes
                SAM      Swap
```

```
                SAM      Volume Groups
        SAM      Instruments
        SAM      Printers and Plotters          ->
                SAM      Print Requests
                SAM      Printers and Plotters
                SAM      Save/Restore Spooler Configuration
        SAM      Tape Drives
        SAM      Terminals and Modems
        SAM      Uninterruptable Power Supplies
SAM      Printers and Plotters          ->
        SAM      Print Requests
        SAM      Printers and Plotters
        SAM      Save/Restore Spooler Configuration
SAM      Process Management             ->
        Other    Performance Monitors ->
                Other    Disk and Terminal Activity
                Other    Inter-Process Communication Facility Status
                Other    Processes With Highest CPU Usage
                Other    System Activity
                Other    Virtual Memory Activity
        SAM      Process Control
        SAM      Scheduled Cron Jobs
SAM      Routine Tasks                  ->
        SAM        Backup and Recovery            ->
                SAM      Automated Backups
                SAM      Interactive Backup and Recovery
        freedisk   Find and Remove Unused Filesets
        SAM        Selective File Removal
        SAM        System Log Files
        SAM        System Shutdown
SAM      Run SAM on Remote Systems
SD-UX    Software Management            ->
        SD-UX    Copy Software to Local Depot
        SD-UX    Install Software to Local Host
        SD-UX    List Software                  ->
                SD-UX    List Depot Software
                SD-UX    List Installed Software
        SD-UX    Remove Software                ->
                SD-UX    Remove Software from Local Depot
                SD-UX    Remove Software from Local Host
SAM      Time                           ->
        SAM      NTP Broadcasting
        SAM      NTP Network Time Sources
        SAM      System Clock
```

2-3. SLIDE: How to use SAM on the X Window System

How to use SAM on the X Window System



H6296 2-3 8

Student Notes

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 logon.

To view the current environment variable values for all shells, 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 `.vueprofile`
C shell	`setenv DISPLAY=`*hostname*`:0.0`	`.login`

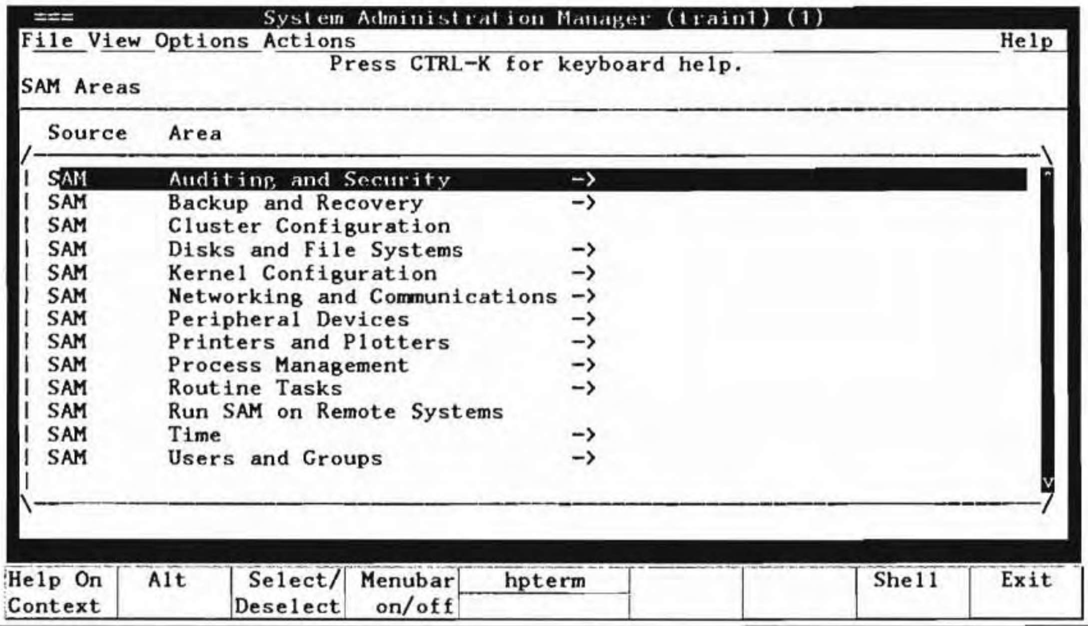
To run SAM, simply log in as root and type:

```
# sam
```

The SAM Control window is shown on the slide. An arrow at the end of an option means you get another list. Each of these selections takes you to a Functional Area.

2-4. SLIDE: SAM on Terminals

SAM on Terminals



H6296 2-4 9

Student Notes

SAM has a special interface 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" a different item on a 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 (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.

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

To move to the menubar:

1. Press (F4) (or (Tab)).
2. Use (◀) and (▶) keys to move to the menu you wish to open, then press (Spacebar).
3. Use (▲) and (▼) to move the highlight to the desired menu item, then press (Spacebar) or type the mnemonic.

   In addition, many terminals have function keys which 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 menubar
`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

2-5. SLIDE: The SAM Object-Action Paradigm

The SAM Object-Action Paradigm

1. Choose a Functional Area
2. Customize the contents or format of an object list:
   - Filter out some objects
   - Sort the objects
   - Rearrange the columns
3. Choose Object from the object list
4. Choose an action
5. Fill in the dialog boxes
6. Proceed with the task by activating the (OK) screen button

H6296 2-5 10

Student Notes

With the object-action paradigm, you can first view a list of objects (for example, file systems), select one or more of them, and then select the action to apply. The object-action paradigm has the advantage of providing data visibility: system state is readily visible through object lists and status information.

2-6. SLIDE: Types of SAM Windows

Types of SAM Windows

- Control Window
- Object-List Screen
- Task Dialog
- Step Menu

H6296 2-6 11 © 1995 Hewlett-Packard Co.

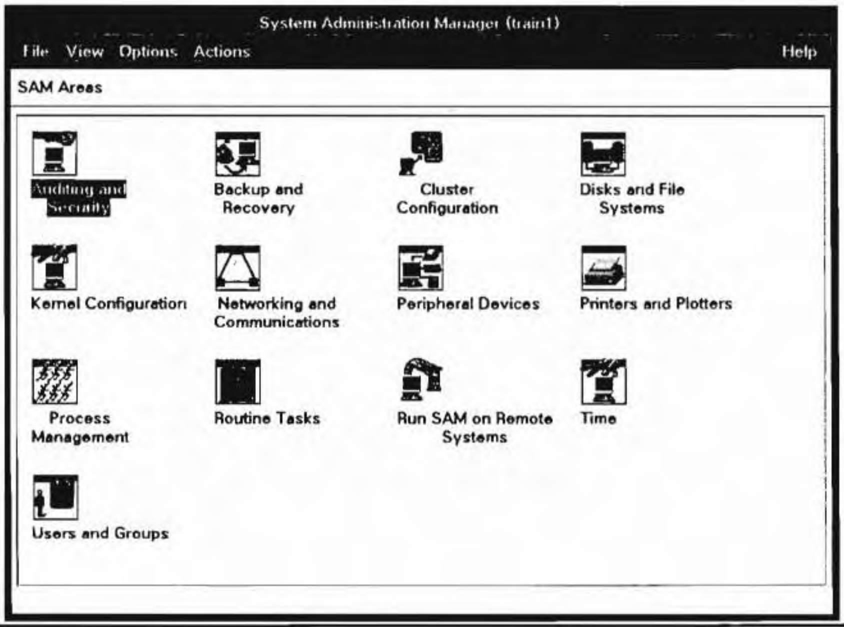
Student Notes

The SAM interface has four main types of windows, not counting message dialogs:

Control Window	Initial display after invoking SAM. Contains a list of Functional Areas in which you can choose to perform tasks.
Object-List Screen	Lists managed objects and provides a menu of actions that apply to them.
Task Dialog	A dialog box used to collect the information to complete a task. Used for doing one-step tasks.
Step menu	A window containing push buttons which each represent a set in some task. Used for organizing and presenting complex tasks.

2-7. SLIDE: SAM Control Window

SAM Control Window



H6296 2-7 12

Student Notes

The SAM control window permits selection of a functional area and subarea. The following actions are also available from the control window:

- Set a size limit for the SAM log file.
- View the SAM log file.
- Add custom actions to SAM's menu system.
- Exit SAM, prompting for kernel regeneration and reboot if necessary.
- Access the help system.

Set SAM log options

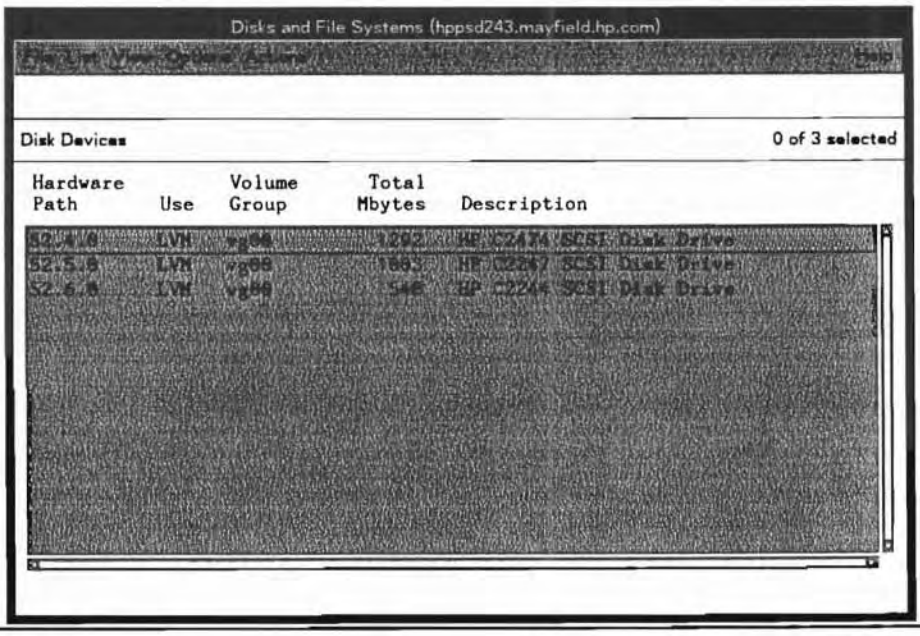
Pressing the (Options...) screen button gives you the ability to set SAM log options. These include:

- Auto-start the SAM Log Viewer. Ordinarily, when you start SAM, you won't see the logfile. This option arranges for a Logging Window to be opened the next time you start SAM. Within this Logging Window, you can set a level of the log display:
  - detail
  - summary
  - verbose
  - commands only
- Trim SAM log file automatically (default) or manually (the system administrator is responsible for its size).
- Maximum logfile size (in bytes). When SAM starts up, it checks the logfile size. If the logfile size is greater than the specified maximum, the logfile will be moved to `samlog.old` and a new logfile will be started.

The SAM control window will not be available when a functional area is open.

2-8. SLIDE: Object-List Screens

Object-List Screens



H6296 2-8 13

Student Notes

Select objects and actions from an **object-list screen**. This is the main type of screen for most functional areas.

The Object List

The object list displays a list of the objects being managed. Typically, each object has a number of attributes, some or all of which are displayed on columns in the list with attribute titles at the top of each column. You can scroll the list horizontally or vertically.

Select one or more objects, and then choose an action from the `Selected` menu.

The Status Area

One line of the status area on object-list screen displays status about the screen: which object-list is current and whether the list is filtered or not. The upper line is optional, and may be used for information that is specific to the functional area, such as whether the print spooler (`lpsched`) is running or stopped.

Functional Area Status Line	If used, may contain status information relevant to the functional area, such as `Print Spooler: RUNNING`.
UI Status Line	User Interface Status. Displays the name of the current subarea, and the number of items currently selected.

The Menubar

Some menubar items are standard across all object-list screens in SAM; other items are specific to the functional area or subarea.

The `File` Menu	Allows printing the current view to a printer and exiting the screen.
The `View` Menu	Allows you to manipulate your view of the object list by filtering out some objects, sorting the objects, or arranging the columns.
The `Options` Menu	Lets you select options global to the object-list screen.
The `Actions` Menu	Contains the actions that apply to a given subarea or to all the subareas.
The `Selected` Menu	Allows you to manipulate the selected object(s). The menu items change with every subarea.
The `Help` Menu	Provides context-sensitive Help to assist you in accomplishing System Administration tasks.

Customizing the Object List

You can customize the following areas of the Object List:

- The object list contents
- The format of the object list (columns and justification)
- Display a subset of objects that meet particular criteria
- Order of the object list.

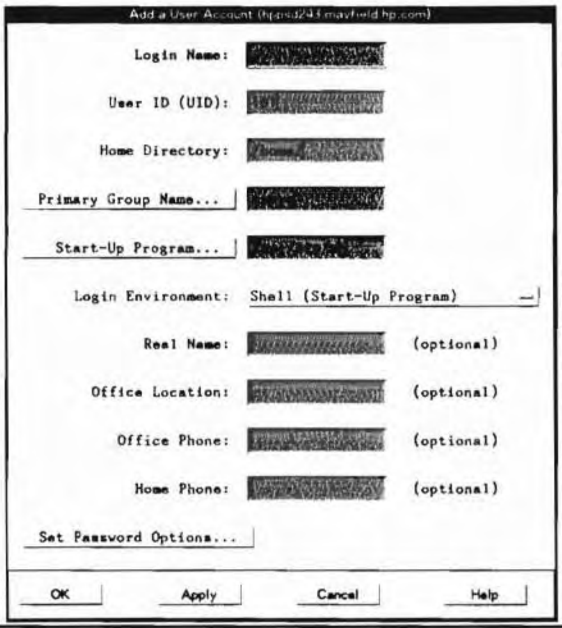
You change the format of the Object List by choosing `Columns` from the `View` menu. You can define numeric column position, justification (left or right), and width in inches. You can preserve this view of the objects by choosing `Save View As Default` from the `View` menu.

To display a subset of objects that meet particular criteria, choose `Filter` from the `View` menu. You define criteria based on field values. You enter an attribute value, then choose menu buttons that define operators (matches, not, greater than, or less than). You can define conditions based on single or multiple attribute values. You can preserve this view of the objects by choosing `Save View As Default` from the "View" menu.

To change the object list order, choose `Sort` from the `View` menu. You specify sort direction as ascending or descending. To sort on multiple attributes, you specify the priority. You can preserve this view of the objects by choosing `Save View As Default` from the `View` menu.

2-9. SLIDE: Task Dialog Screens

Task Dialog Screens



H6296 2-9 14

Student Notes

Task dialogs can appear when you:

- Select an action from the `Actions` or `Selected` menu that requires parameters to be specified or approved
- Select a functional area from the SAM control window that, for some reason, does not make sense to initiate with another type of window
- Press a push-button in another task dialog
- Select a step from a step menu that requires parameters to be specified or approved

Task dialogs are divided into two parts: components for the display and collection of data, and control buttons for closing the dialog, performing the task, or getting help.

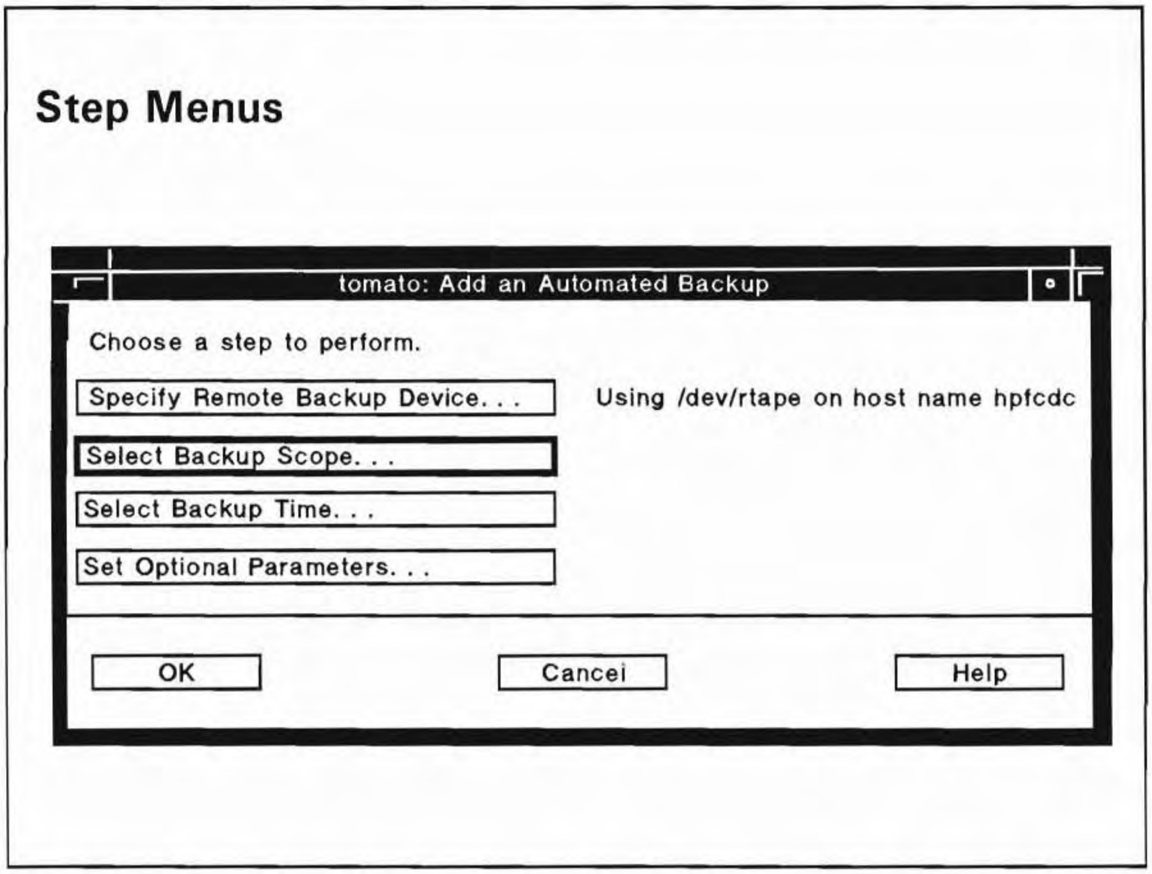
There are three types of task dialogs, each having a different set of control buttons:

Standard	(OK) (Cancel) (Help)
Read-only	(OK) (Help)
Repeatable	(OK) (Apply) (Cancel) (Help)

Each control button has a distinct action:

(OK)	Executes the task and closes the window
(Apply)	Executes the task and presents the task dialog again
(Cancel)	Cancels the task and closes the window
(Help)	Presents help on the task

2-10. SLIDE: Step Menus



H6296 2-10 15

Student Notes

Step menus provide a way for you to perform complex procedures requiring multiple steps.

Step menus consist of push-buttons labeled with the steps and adjacent status areas that provide feedback on which steps have been visited, options, what the prerequisites, if any, are, etc. The steps will actually modify the system as each is performed, or they may merely collect parameters and leave the system state unmodified until (OK) is pressed.

The slide shows a step menu where the steps take effect only after (OK) is pressed. This is also referred to as "late binding."

2-11. SLIDE: SAM Messages

SAM Messages

- Confirmation Messages
- Error Messages
- Progress Messages, transient and persistent
- Information Messages

H6296 2-11 16

Student Notes

SAM uses four different types of messages:

Confirmation Messages

Confirmation messages present some information along with a yes/no question. You must answer before you can continue with the application.

Error Messages

Error messages indicate that a mistake has been made and what should be done to repair it. You must acknowledge an error message before you can continue with the application.

Progress Messages

Transient progress messages are used to reassure you that the system is doing something (initialization, for example); without the message, you might assume that the system is hung. Transient messages are displayed at the bottom of the screen while the activity is occurring.

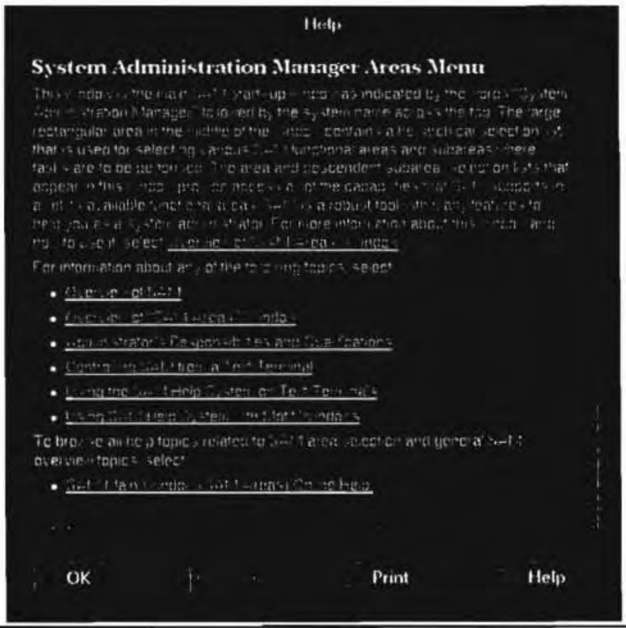
Persistent progress messages tell you what the system is doing, but remain on the screen for you to dismiss. The messages appear in a scrolling area in the progress dialog box.

Information messages

Information messages convey information, and must be dismissed before you can continue with the application.

2-12. SLIDE: SAM's Help System

SAM's Help System



H6296 2-12 17

Student Notes

The help window will contain a help text viewing area, a list of related topics, and the ability to navigate to those related topics or back to previously visited topics.

If you are using SAM's graphical interface under the X Window System, the VUE Help system will be used. VUE Help does not depend on VUE being present or running. VUE Help has two types of windows: a general browsing window and a "quick help" window.

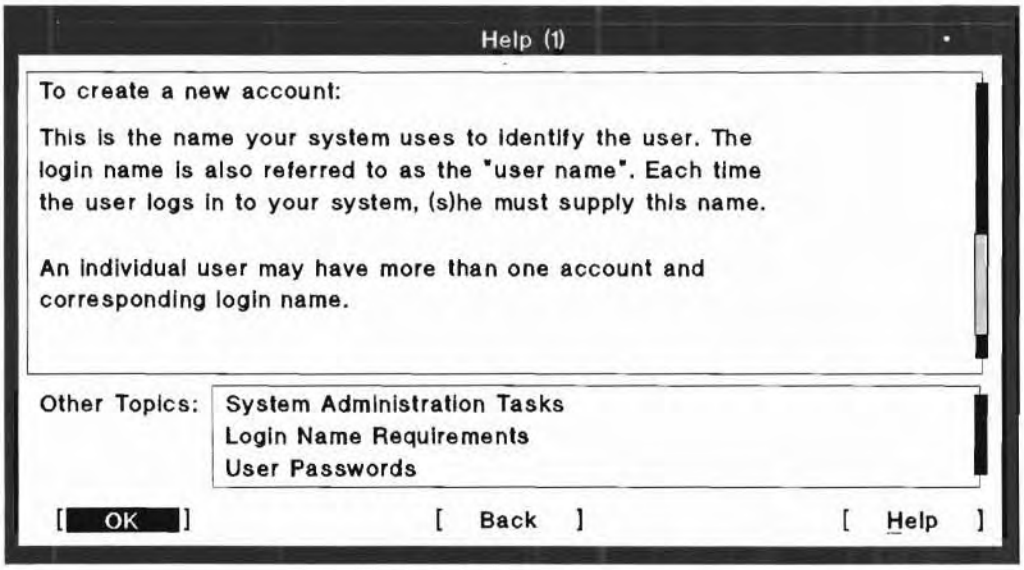
To use the browsing window, press the (Help) button in the Control Box or choose `Overview` form `Help` on the menubar. The slide shows the appearance of the browsing window.

All other areas of SAM use the "quick help" window.

Quick Help

Help on Terminals

The help window contains a help text viewing area, a list of related topics, and the ability to navigate to those related topics or back to previously visited topics.



Help on Terminals

2-13. SLIDE: Customizing SAM

Customizing SAM

Task Customization

- Additional steps to perform before or after processing

Build Restricted SAM

- Different capabilities for different users

Add Customer Utilities

- Access your own executable programs from within SAM
- Add menu items
- Add your own help screens

H6296 2-13 18

Student Notes

You can customize SAM in three ways:

- Task Customization "Action"

  Customize selected tasks by specifying steps to be performed before and/or after SAM's normal processing for that task.

- Customer Utilities

  Add new functionality to SAM by including standalone programs and scripts in the menu system.

- Restricted SAM

Build a restricted version of SAM that can be run be specified non-root users. You can create different versions of SAM, with different capabilities and for different users.

2-14. SLIDE: Customizing SAM Tasks

Customizing SAM Tasks

Task Customization

- Add a New User Account
- Remove a User Account

H6296 2-14 19

Student Notes

You can customize the following SAM tasks:

- Add a New User Account to the System.
- Remove a User Account from the System.

For each of these tasks, you can specify steps you want performed before and/or after SAM does its processing for the task. Before SAM performs one of the tasks, it checks to see if a pre-task step (executable file) has been defined. If so, SAM invokes the executable, passing it a set of parameters, and waits for its completion. After SAM has finished processing, it checks for a post-task step, performing the same type of actions as for the pre-task step.

There are three requirements for these executable programs and scripts:

- Must be owned by `root`.
- Must be executable only by `root`, and if writable, only by `root`.
- Must reside in a directory path where all the directories are writable only by `root`.

SAM accomplishes this customization through the passing of parameters to and from the executable file. Each of the customizable tasks has its own specific parameters which are passed to the program based on how you supply information on the SAM task screen.

The following files contain examples of how to process these parameters:

Sample File	Task Being Customized
`/usr/sam/lib/ct_adduser.ex`	Add a New User Account to the System
`/usr/sam/lib/ct_rmuser.ex`	Remove a User Account from the System

Parameters for Adding Users

These parameters are used for customizing the SAM task "Adding Users"

```
-l login_name
-v user_id
-h home_directory
-g group
-s shell
-p password
-R Real_name
-L office_location
-H home_phone
-O office_phone
```

Parameters for Removing Users

These parameters are used for customizing the SAM task "Removing Users"

`-f` *user_name*	Option supplied when all of *user_name*'s files are being removed.
`-h` *user_name*	Supplied when *user_name*'s home directory and files below it are being removed.
`-n` *new_user_name*	Supplied when *user_name*'s files are being assigned to *new_owner*.

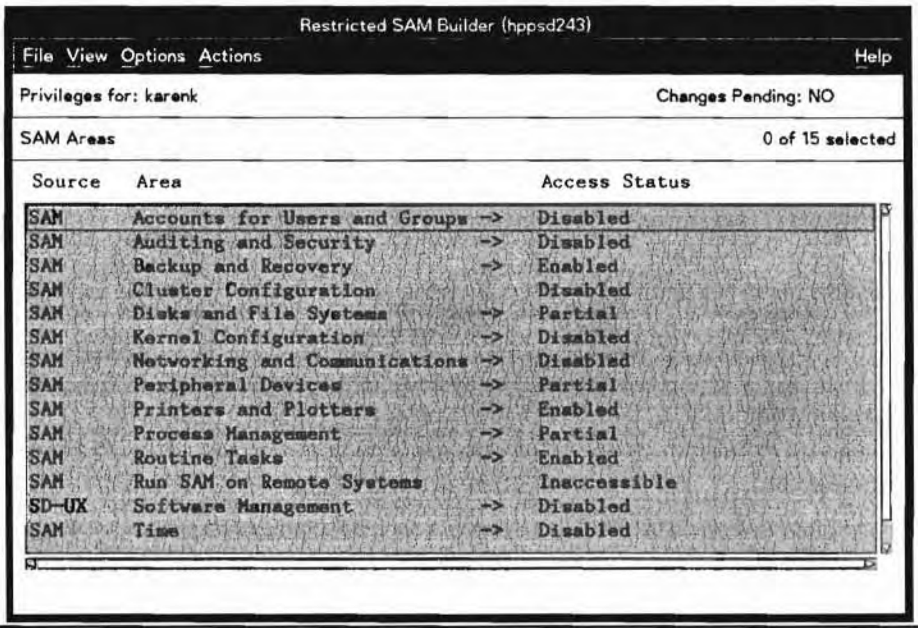
Example of Customizing a Task

For example, you might maintain a file for each group that has all the mailstops and telephone extensions of employees in that group. You want to have the file for the user's primary group (the group specified in the `/etc/passwd` file) copied to the user's home directory.

You would create a shell script file that contains the commands to copy the appropriate group file name to the user's home directory, place it in a directory that is writable by `root` (`/etc/sam/custom` is a good choice), chmod the file to make sure it is executable only by `root`, and then choose `Task Customization` from the `Actions` menu in the `Users` Functional Area.

2-15. SLIDE: Restricted SAM Builder

Restricted SAM Builder



H6296 2-15 20 © 1995 Hewlett-Packard Co.

Student Notes

Restricted SAM Builder

The Restricted SAM Builder empowers system administrators to give non-root users access to limited SAM functionality. The superuser can delegate to specified users some of the tasks normally performed only by root.

This is done by starting SAM from the command line with the `-r` option:

```
# sam -r
```

SAM displays a list of all valid user accounts. Mark the user who is to gain access to SAM, then press (OK).

SAM displays its list of functional areas, along with information about the accessibility of those areas. The menu system is accessible, but no tasks can be performed.

The system administrator can now decide whether each menu item is accessible or not. Items which can be enabled or disabled are the menu items shown in the System Administration Manager Roadmap.

Denying access to menu items containing submenus automatically disables all submenus of the item. A warning message will be displayed.

After you have completed denying access to menus, select the **Save** item from the **Actions** menu to save the current state of SAM in a file. This state is written to a file containing the users name in its filename. For example, for the user `oper` the file will be `/etc/sam/custom/oper.cf`. Now the user `oper` is allowed to start SAM and will have the ability to execute any tasks that were enabled by the superuser.

Each time SAM is started, it tests whether the user has root permissions or not. If not, SAM looks for a custom file with the user's name in `/etc/sam/custom`. If it is there, the file's contents are read and SAM displays only the allowed menus. Otherwise SAM displays an error message and exits.

Alternatives for Certain Tasks

Beginning with HP-UX Release 10.0, the systems may have special user accounts to perform certain tasks (for example, shutting down the system or adding hard disk drives). This requires the Bellcore Standard Operating Environment (SOE) to be installed. If SOE is not on your system, use `swinstall` to add it.

The predefined users (in `/etc/passwd`) are:

`checkfsys` — An interactive program that uses a SAM-like interface to provide access to `fsck`. When a user logs into the machine as `checkfsys` or a privileged user invokes the `checkfsys` command, a screen will appear giving the user a list of local file systems that can be checked.

`makefsys` — Provides access to a screen within SAM. When a user logs into the machine as `makefsys` or a privileged user invokes the `makefsys` command, a screen will appear giving the user a list of hardware devices. The user can add another device and at that time create a file system.

`mountfsys` — Provides access to a screen within SAM. When a user logs into the machine as `mountfsys` or a privileged user invokes the `mountfsys` command, a screen will appear giving the user a list of local file systems. The user can add disks that are not currently used but have a file system on them.

`umountfsys` — It provides access to a screen within SAM. When a user logs into the machine as `umountfsys` or a privileged user invokes the `umountfsys` command, a screen will appear giving the user a list of local file systems. The user can select one of the local file systems to remove.

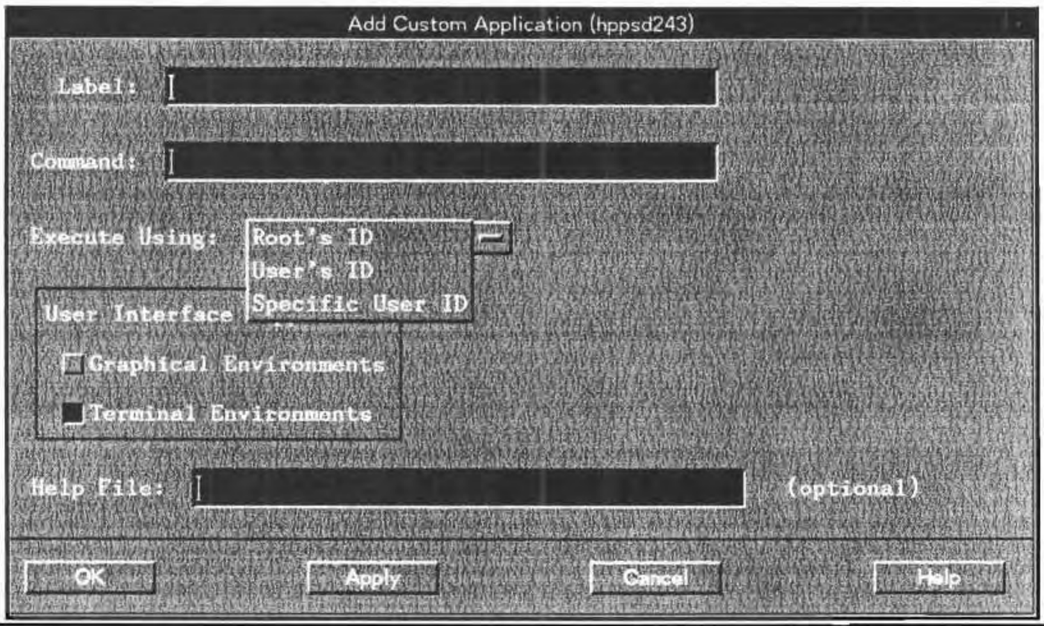
`powerdown` — It provides access to a screen within SAM. When a user logs into the machine as `powerdown` or a privileged user invokes the `powerdown` command, a screen will appear giving the user the choice of halting, rebooting, or shutting down to single user mode the system.

`sysadm` It is an alias for the System Administration Manager (SAM). When a user logs into the machine as `sysadm` or a privileged user invokes the `sysadm` command, SAM will be invoked.

All logins can be used in an X11/VUE environment or an HP-supported terminal.

2-16. SLIDE: Adding Custom Utilities and Associated Help

Adding Custom Utilities and Associated Help



H6296 2-16 21

Student Notes

You can easily add standalone programs and scripts to SAM. With the **Actions** - **Add Custom Menu Item** or **Add Custom Menu Group** menu, you can add your own hierarchy of menus and utilities in the SAM (sub-) menu you would like to put it in.

The executables can use both terminal and graphics environment.

You can also write your own help for each menu item you create.

Steps to perform to add your own utility to SAM:

1. Install your own utility, (which may be a shell script, a binary file or an application using the Window system), into the file system.

2. Optionally, write a help file for your utility.
3. Put both files in a subdirectory which is writable only by `root`, e.g. in `/etc/sam/custom`, and give write and execute rights of the utility file only to `root`.
4. Enter SAM
5. Walk through the menu system to the submenu where you want to put your utility.
6. If you want to create your own hierarchy with sub-items, choose **`Add Custom Menu Group`** from the `Actions` menu.
7. Choose **`Add Custom Menu Item`** from the `Actions` menu.
8. Fill in the filenames using absolute path names and mark whether the utility supports graphical or terminal environment.
9. Select (OK) or (Apply).

All your modifications will appear marked `Custom`, so it's easy to find out whether a menu item originally included in SAM or created by the System Administrator. The `Custom` menu items can be deleted by selecting the menu point, then choose **`Remove from SAM`** from the `Actions` menu. This will only remove the menu item from SAM. It will not remove your files.

2-17. LAB: Hands-On with SAM

This is an opportunity for you to explore SAM. You will be using SAM throughout this course to perform certain tasks, and you are also encouraged to use SAM to perform routine System Administrative tasks on your own system.

Directions

Follow the steps below to gain some initial exposure to SAM and explore its capabilities.

1. Log on and run SAM.

2. Select the `Users and Groups` Functional Area from the SAM Control window.

   Display a list of the users currently set up on your system.

   Use the `Column` editor to right-justify the `Login Name` column.

   Next, put all the entries in order by user ID.

   Now, display only the entries for users with user IDs of less than 100.

3. How much swap space is configured on the system you are using in class? (Hint: swap space is on disks)

4. How many local file systems are on the system you are using in class? (Hint: file systems are on disks).

5. How many backup devices are configured on the system you are using in class?

6. Is there currently an Automated Backup schedule for this system? If so, what is the schedule?

7. List three drivers that are configured in the kernel.

8. Create an executable script in `/etc/sam/custom` to add a functionality to SAM. The script should show you who is doing what (Hint : use `/usr/sbin/whodo`). Then add your own script to the "Users and Groups" submenu. Verify the functionality.

Module 3 — Users and Their Environment

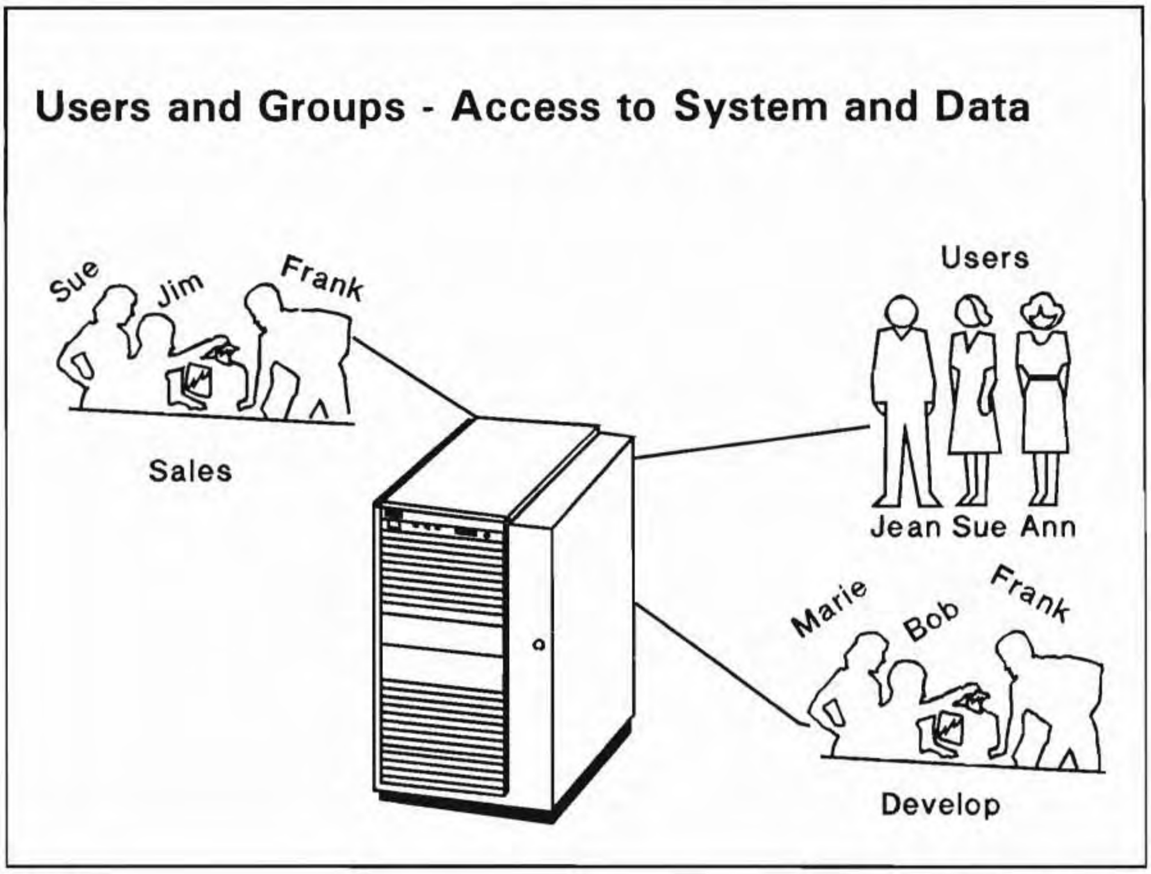
Objectives

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

- Add users to the system.
- Maintain the `/etc/passwd` and `/etc/group` files.
- Identify required entries in `/etc/passwd` and `/etc/group`.
- Change user passwords.
- Specify a user's default environment.
- Communicate with users.

Module 3 — Users and Their Environment

3-1. SLIDE: Users and Groups - Access to System and Data



H6296 3-1 22 © 1995 Hewlett-Packard Co.

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 your system can be divided into various working groups, so that files owned by members of a given group can be shared and yet protected from access by users who are not members of the group. A user can be a member of more than one group, but that user can only function as a member of one group at any given time, and only one of those groups is considered to be the user's primary or default group.

You assign users to groups based on their need to access, or share, files on the system. When a user requests to perform an operation on a file (that is, read, write, or execute), that user's identity (both user and group) is used by the operating system to evaluate whether that user is the owner, a group member, or "other" with respect to the file's ownership and group affiliation.

Module 3 — Users and Their Environment

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 group.

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.

3-2. SLIDE: Adding Users

Adding Users

1. Add an entry to `/etc/passwd`
2. Add an entry to `/etc/group` (optional)
3. Add an entry to `/etc/logingroup` (optional)
4. Create a home directory and set permissions to 755

```
# mkdir /home/maya
# chmod 755 /home/maya
```

5. Create login (shell initialization) files for the user:

```
# cp /etc/skel/d.profile /home/maya/.profile
```

6. Change the owner and group of the home directory and default files

```
# chown -R maya:users /home/maya
```

H6296 3-2 23

Student Notes

The administrator of an HP-UX machine is responsible for maintaining the accounts of the users. This entails adding new user accounts when necessary, removing or disabling the accounts of users who should no longer be permitted to access the machine, and setting up the user's default working environment.

To add a new user, there are only a few things to remember. Any new user account requires an entry in `/etc/passwd`. In addition, an entry for the new user should be made in `/etc/group`.

A home directory must be created for the new user using the `mkdir` command. The permissions on this home directory should be 755 (`drwxr-xr-x`).

After creating the home directory, create the shell initialization files in that directory. The particular initialization files depend on the type of shell the user will be using. These startup files serve as a basis for allowing the new user to further customize their operating environment.

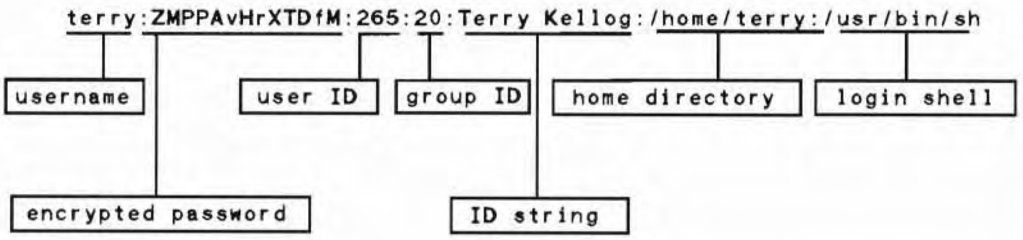
3-3. SLIDE: The /etc/passwd File

The /etc/passwd File

Example:

```
root:mAj8as.,ofads::0:3::/:/sbin/sh
daemon:*:1:5::/:/sbin/sh
date:r.c7.0x4/,hGJq:20:1::/:/usr/bin/date
qrik:.r.ca8/,f2i5y:204:20::/home/erik:/usr/bin/sh
```

Use /usr/sbin/vipw to edit
Use /usr/sbin/pwck



H6296 3-3 24

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 (:).

username — The user name that is used when a user logs in. 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.

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 should be numeric or a special character. If the password field is empty, then no password is associated with the login name. You should never leave the password field empty as it 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.

user ID — The user ID (UID). UID zero (0) is reserved for root. Typically, values 1-99 are reserved for the system. Use numbers 100-60000 as UIDs for your users. Each UID must be unique. When adding users with SAM, it will use UIDs starting from 101. You can choose to use sequential UIDs for particular groups. For example:

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

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 is unable to log in.

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 (Korn) shell for the Super User. 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 could 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, and 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`.

Required Entries in /etc/passwd

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

```
# vipw
```

This command puts a lock on the `/etc/passwd` file by copying `/etc/passwd` to `/etc/passwd.tmp`. If a user attempts to change her password, 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 `/etc/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.

3-4. SLIDE: Changing Passwords

Changing Passwords

```
$ passwd                         Users can change their own password (must
                                 know current password)
Changing password for bugs
Old password:
New password:
Re-enter new password:
$
# passwd bugs                    Root can change any user's password
New password:
Re-enter new password:
#
# passwd                         Root can change the root password
Changing password for root
New password:
Re-enter new password:
#
```

H6296 3-4 25 © 1995 Hewlett-Packard Co.

Student Notes

Any time a password needs to be changed, whether by a regular user or the superuser, the `passwd` command is used. An ordinary user on the system can change his or her own password (but nobody else's). When invoked, the user will be prompted to enter the existing password. Upon entering the correct password, the user will be prompted to enter the new password. After the new password is entered, the user is prompted to enter it again. This is done for verification purposes and to ensure the user didn't make a typing error. If the second password does not match the first, the password is not changed and the user is returned to the shell.

If a user forgets his or her password, the user must seek the assistance of the administrator. The administrator can change any other user's password by invoking `passwd` with an argument of the user's login name.

If, for some reason the root password needs to be changed, the administrator should invoke `passwd` while logged in as the superuser.

If you are changing/assigning a password as a user:

- Passwords must contain at least 6 characters. Though a password may be assigned more characters, only the first eight are significant.
- Password must contain at least two alpha characters (upper or lower case) and at least one numeric or special character. This enforces a certain level of security within the password structure.

Note

When you use the `passwd` command, a copy of the old `/etc/passwd` file is saved in `/etc/opasswd`.

3-5. SLIDE: The /etc/group File

The /etc/group File

group_name:password:group_id:group_list

Example:

```
other::1:root,daemon,uucp,sync
users::20:
develop::30:bugs,daffy
sales::40:bugs,daffy,elmer,marvin
```

Use /usr/sbin/grpck to check

H6296 3-5 26

Student Notes

The /etc/group file is used to define groups. The fields are delimited by a colon (:).

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. This number is shared by all group members.

group_list is a list of user names. These user names *are not* necessarily members of the group. Instead, this list defines the users who may become members of the group using the newgrp command.

Look at the example below to see how the *group_id* is the same in the /etc/passwd and /etc/group files.

The /etc/passwd entry

```
bugs:A.oxqZ1n7w4j.0.:101:20:Bugs Bunny:/home/bugs:/usr/bin/ksh
```

The /etc/group entry

```
users::20:bugs,daffy,elmer,marvin
```

Note that a user can be a member of more than one group. A user can use the newgrp command to change to a different group.

$ newgrp *group_name*

The new group is referred to as the **effective group** of the user. Changing to a new group does not alter the user's primary group entry in the /etc/passwd file, it only alters the user's group association for any files he accesses or creates after executing the newgroup command. Executing the newgroup command with no parameters returns the user to the group to which he is assigned in the /etc/passwd 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.

The /etc/logingroup file

The /etc/logingroup file has the same structure as the /etc/group file and can be a copy of, or a link to, that file.

The /etc/logingroup file is checked when a user attempts to access files that are associated with a different group that he is currently identified with. If the user's user name appears in a group entry in the /etc/logingroup file, then that user can access files from the second group without having to execute the newgrp command.

For example, if this entry appears in both the /etc/group and /etc/logingroup files:

```
groupa:*:21:herbert,monika,suzy
groupb:*22:herbert,terry,don
```

then the user herbert can access files in either groupa or groupb without having to use the newgrp command. If these same entries appeared only in the /etc/group file, and the /etc/logingroup file did not exist, then user herbert would have to execute the newgrp command to change his group affiliation to whichever group whose files he wished to access.

You may check to which groups you belong by using the groups (1) command.

Checking the /etc/group file

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

3-6. SLIDE: Removing Users

Removing Users

- Decide which to do:
  - Remove the user's files and directories from the system
  - Assign the user's files and directories to another user
  - A combination of the above
- Remove the user's name from all entries in `/etc/group`
- Remove or inactivate the user's entry in `/etc/passwd`
- Locate all files owned by the user
- Remove or move and change ownership of the files
- SAM can help

H6296 3-6 27

Student Notes

You may remove a user from the system because that person has left the department or company, because you are getting a new system to support that user, or for other reasons.

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.

- Decide whether to remove the user's files, assign them to another user, or a combination of the two. Your decision here will affect your actions in the next steps.
- Locate all files owned by the user

  find / -user *username* > listfile

- Remove the user's name from all entries in /etc/group. Also remove the user's name from all entries in /etc/logingroup if it exists.
- Remove or inactivate the user's entry in /etc/passwd. You can either delete the entry entirely, or deactivate it by placing an asterisk (*) in the *password* field. This will render the login name unusable.
- Remove or move and change ownership of the files. If some of the user's files are executable programs shared by several users, you will need to decide whether to leave them in their existing directory or move them. You will also need to decide which user name to assign as the new owner. Once you have decided, you will have to use some combination of the rm, mv , chown and possibly chmod commands. If the number of files is large, you may decide to move them using cpio. If you decide to remove the files entirely, you may wish to make a tape archive of them before removing them

SAM Can Help

When removing users or files from a system, there is always the potential for error in removing the wrong user, or removing files inadvertently during the removal process.

For example, user bin is the owner of (from the operating system's perspective) the majority of the executable commands on the system. Removing this user would obviously be disastrous. On the other hand, suppose user joe owns all of the files comprising the test suite for a project. It may be appropriate to remove joe, but the test suite should be left intact and assigned to a new owner.

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.

Action

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:

```
/usr/bin
/usr/sbin
/dev
/etc
/usr
/var
/stand/vmunix
/stand/vmunix_alt
```

3-7. SLIDE: How SAM Can Help

H6296 3-7 28

Student Notes

SAM can make adding users to your system very easy. SAM will use certain defaults such as the next available user ID number, the home directory of `/home/`*username*, the primary group name of `users`, and `/usr/bin/sh` as the start-up program and login environment.

You can override any of these defaults if you wish. SAM lessens some of the potential for error that is associated with manually editing files, creating directories, etc.

Once you understand what SAM is asking for, adding users is a simple matter.

Change User's UID Number

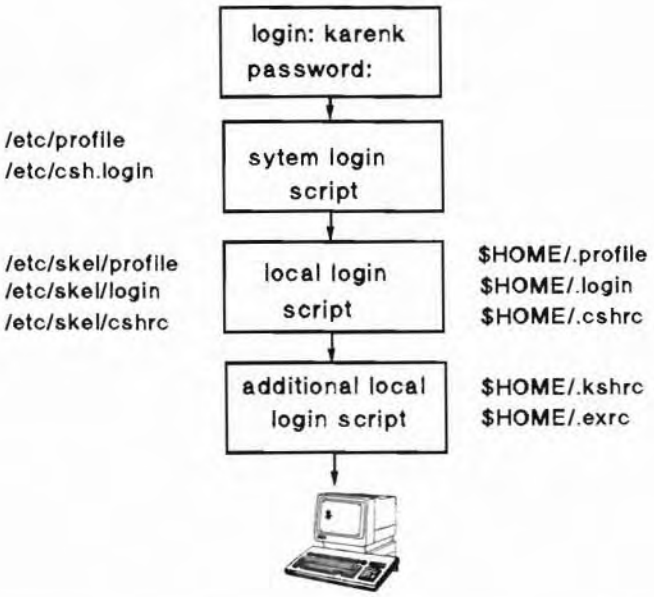
The user identity (UID) is a number used by the file system to identify users. Owners of files in a file system are determined by the UID numbers, not by the login names of the users.

The UID is automatically assigned when the user is added to the system. You might want to give a user a specific UID for the following reasons:

- Give the user `root` permission (`UID 0`).
- Make the user's UID matches that user's UID on other networked file systems.

If for any reason you want to change the user's UID, you must also change the ownership of any of the files currently owned by that user to the new UID. If you do not do this, then the user will no longer be the "owner" of the files. If you use SAM to change the UID, SAM will automatically reassign the new UID to all files previously owned by the user.

3-8. SLIDE: What Happens When a User Logs In?



H6296 3-8 29

Student Notes

System login scripts contain commands to set a user's environment. System login scripts run *before* a user's local login script.

Local login scripts can override or modify the system defaults for individual users. When you add a new user to the system, you should be sure that the user has a local login script. HP-UX provides four default local login scripts that you may copy to the user's home directory.

```
/etc/skel/.profile
/etc/skel/.login
/etc/skel/.cshrc
/etc/skel/.exrc
```

You may add files and/or directories to the /etc/skel directory. The complete contents of /etc/skel (which provides a *skeleton* for a new user's home directory) is copied to the new user's home directory, reflecting the structure found in /etc/skel.

The Shell Environment Initialization Sequence

To be fully functional, a shell must first go through a series of steps to initialize its environment:

1. The shell runs the appropriate *system login script*, which initializes the user's environment. The system login scripts define a *default* environment, and can be customized by the System Administrator.

If the Shell is ...	The System Login Script is ...
Bourne (/usr/old/bin/sh)	/etc/profile
Korn (/usr/bin/ksh)	/etc/profile
Posix (/usr/bin/sh	/etc/profile
Restricted (/usr/bin/rsh, /usr/bin/rksh)	/etc/profile
C (/usr/bin/csh)	/etc/csh.login

   As shipped, these scripts define and export for shell use the environment variables PATH, TZ, and TERM. Inside these scripts, the files /etc/PATH, /etc/MANPATH, /etc/TIMEZONE are sourced. Since the system login scripts are run for all users at login, the system administrator can modify these files to set global defaults for all users. This is useful for ensuring that each user runs essential commands at login.

2. Displays the contents of the /etc/copyright and /etc/motd file.
3. Notifies the user of unread news with the prompt:

   news: *news_filename*

4. The shell runs the user's *local login script* (if it exists) in the user's home (login) directory:

If the Shell is ...	The Local Login Script is ...
Bourne (/usr/old/bin/sh)	.profile
Korn (/usr/bin/ksh)	.profile
Posix (/usr/bin/sh	.profile
Restricted (/usr/bin/rsh,/usr/bin/rksh)	.profile
C (/usr/bin/csh)	.login

Note

Typically, the system administrator initially creates a local login script for each user. If SAM is used to add a user, it copies the default local login script (the `/etc/skel/`.*file*) to the user's home directory. Users can further customize their environments by modifying these files to suit their needs.

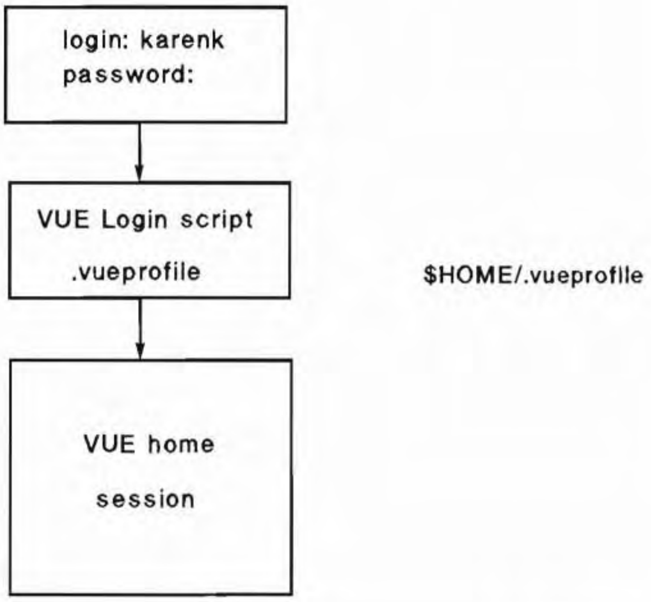
In addition to the above scripts, the Korn and C shells may (and usually do) have additional local login scripts:

- Korn shell - if the environment variable is defined, the Korn shell runs the file defined by ENV (typically, `.kshrc`) whenever a new Korn shell is started. Many programs (for example, `vi` and `mailx`) allow users to start a shell from within the program; this is called a *shell escape*. The ENV file is re-run for a shell escape, whereas `.profile` is run only at login.
- C shell - Runs the `.cshrc` file whenever a new C shell is started. This is similar to how the Korn shell ENV file works. The `.login` file is run only at login, whereas `.cshrc` is rerun for every new C shell.

5. Once all initialization is complete, the shell displays a prompt and waits for input from the user.

3-9. SLIDE: What Happens at VUE Login?

What Happens at VUE Login?



H6296 3-9 30

Student Notes

The files `/etc/profile` and `$HOME/.profile` are not read by VUE because they may contain terminal I/O based commands inappropriate for a graphical interface. Users should set up `.vueprofile` with personal environment variables for their VUE session. Personal environment variables can be set in the script file `$HOME/.vueprofile`.

VUE sets the following environment variables for each user:.

DISPLAY	set to the value of the first field in the Xservers file
EDITOR	set to the HP VUE default editor
ENV	set to `$HOME/.kshrc`
HOME	set to the user's home directory (from `/etc/passwd`)
KBD_LANG	set to the value of `$LANG` for certain languages

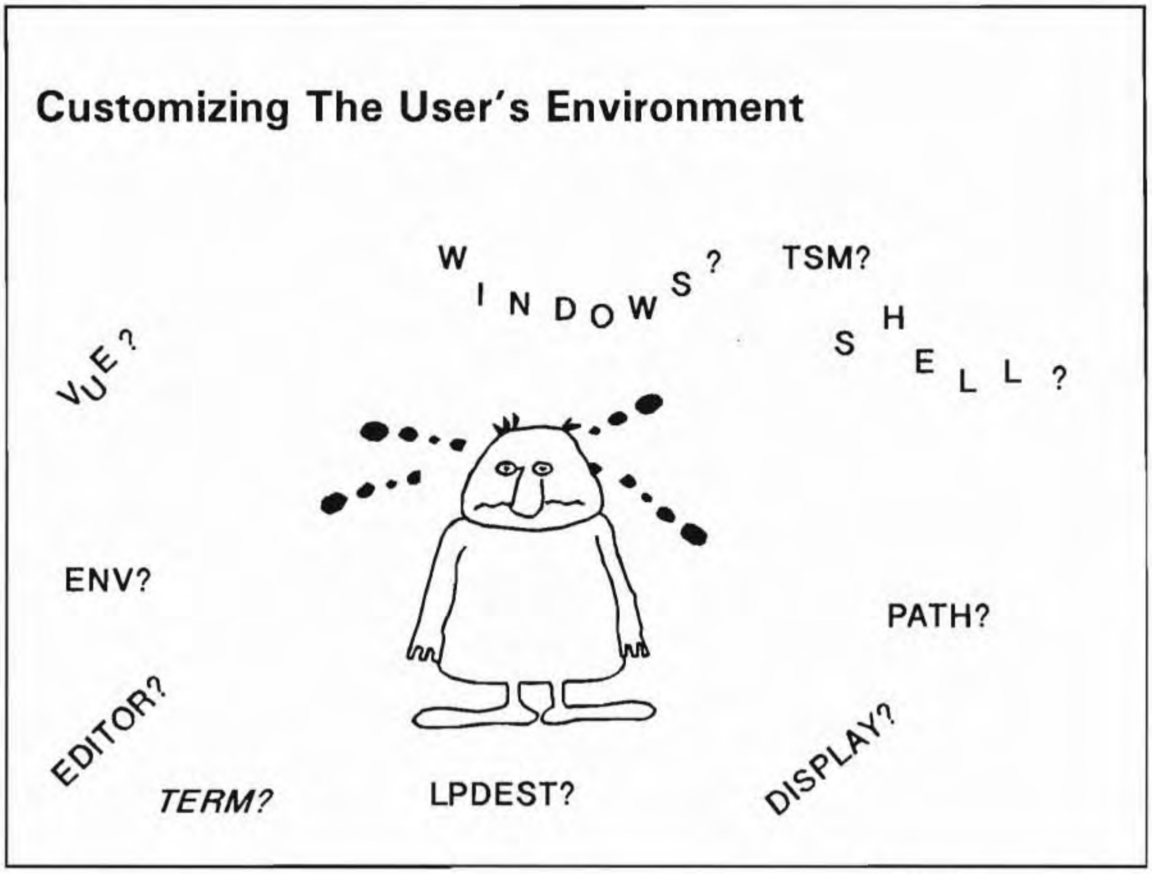
LANG	set to the display's current NLS language (if any)
LC_MESSAGES	set to the value of $LANG
LOGNAME	set to the user name
MAIL	set to /usr/bin/mail/$USER
PATH	set to the value of the Vuelogin userPath resource
SESSION_SVR	set to the name of the session server (uname -n)
SHELL	set to the user's default shell (from /etc/passwd)
TERM	set to xterm
TZ	set to the value of the Vuelogin timeZone resource
USER	set to the user name

It is possible to adapt `$HOME/.profile` or `.login` so that it will be read at VUE login. Group the statements not allowed for VUE (such as those that perform terminal I/O, for example `tset` or `stty`), into one section and enclose them with an `if` statement that checks for the setting of the VUE environment variable. From then on, changes need only be made to `$HOME/.profile` or `.login`.

Here is an example suitable for addition to a user's `.profile` login script:

```
#
#  commands and environment variables used when logging in
without VUE
#
if [ ! "$VUE" ]; then
stty ...
 tset ...
DISPLAY=mydisplay:0.0
MAIL=/var/mail/$USER
EDITOR=/usr/bin/vi
             ...
fi

#
# environment variables common to both VUE and non-VUE
#
PATH=$HOME/bin:$PATH
```

3-10. SLIDE: Customizing The User's Environment



H6296 3-10 31 © 1995 Hewlett-Packard Co.

Student Notes

A user account can be configured so that when the user logs in to the system, one of the following three environments is automatically started (as long as VUE is not in use):

- A shell
- The X Window System
- Terminal Session Manager

The Shell Environment

A user's environment consists of variables used by the shell. Environment variables define various characteristics of the environment in which the shell runs. Environment variables consist of a name and a value (*NAME=value*). The assigned values of environment variables are used by the shell and passed to each process created during a session.

There are two variables that are especially important for you to understand and set for users, based on your computing environment. These are the `PATH` and the `TERM` variables.

PATH

When you type a command, HP-UX searches all the directories specified by the `PATH` variable until it finds the command. If the command is not in a directory specified in `PATH`, the system displays:

```
sh: command_name:   not found
```

The value of the `PATH` variable is specified as a list of directories to search, separated by colons. There should be no spaces surrounding the colons. These directories will be searched in the order they are specified in the `PATH` variable. In general, you should put the most frequently used directories first in the path.

The default value of the `PATH` variable contains only `/usr/bin`, `/usr/ccs/bin` and `/usr/contrib/bin`. You may wish to include other directories in the search path for yourself and the users on your system. Some commonly used directories you might wish to add to the `PATH` are:

Directory	What It Contains
`/sbin`	Basic commands the system administrator uses.
`/usr/sbin`	Other commands the system administrator uses.
`/usr/bin`	Frequently used HP-UX commands.
`/usr/contrib/bin`	Contributed programs not supported by Hewlett-Packard.
`/usr/local/bin`	Programs and commands written locally (at your location).
`/usr/bin/X11`	X Window System programs.
`$HOME/bin`	A directory you might create for your own shell scripts and programs.
`.`	Your current directory. If `.` ("dot") is in your `PATH`, it should be the last entry.

Note Adding `.` ("dot," denoting the *current directory*) to the `PATH` variable can be a security risk, especially for the `root` user.

TERM

To communicate effectively with your terminal, HP-UX must know the type of terminal or graphics display you're using.

To find out the type of your terminal, use `ttytype` to "ask" your terminal for its ID string.

When a user logs on, he is prompted to enter the terminal type. The default value is `hp`.

This value works with Hewlett-Packard terminals, but it may not let you take full advantage of your terminal or graphics display features. If your user will always be using the same kind of terminal, you can set up the `TERM` variable to have the appropriate value for their terminal. The value you assign to `TERM` must equal a value in the `terminfo` database.

The X Window System as a User Environment

A new user can be set up so that the X Window System is automatically started when that user logs in to the system. Then, when the user exits the window system, she or he is logged out automatically.

If you want the user's login environment to be the X Window System, make sure that the necessary file sets are installed on your system. (If `/usr/bin/x11start` exists, it is likely that the other files are in place as well). If not, use `swinstall` to add it to your system.

To automatically start the X Window System when the user logs in, you must add these lines to the user's local environment file.

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

The TSM Environment

Terminal Session Manager (TSM) is a multi-session manager (windows) for ASCII terminals. A new user can be set up so that TSM is automatically started when that user logs in to the system. Then, when the user exits TSM, she or he is logged out automatically.

Make sure that TSM is installed on your system (check that `/usr/bin/tsm` exists). If not, use `swinstall` to add it to your system. Depending on the software configuration purchased, TSM may be an optional product. To automatically start the TSM system when the user logs in, you must add these lines to the user's local environment file.

```
#Start TSM at login:
if [ -x /usr/bin/tsm ]
     then
          exec /usr/bin/tsm
fi
```

3-11. SLIDE: Communicating with System Users

Communicating with System Users

- The `news` command
  - Displays files in `/var/news` directory
- The `mail`, `mailx`, and `elm` commands
  - Mailer programs used to send messages to specific users
- The `wall` command
  - Sends a message to all users logged in
  - Immediately interrupts whatever the user is doing
- The `/etc/motd` file
  - Contents are displayed when a user logs in

H6296 3-11 32

Student Notes

There are a number of ways that you can communicate with the users of your HP-UX system. The communication method used generally depends on the importance of the message that needs to be sent.

The `news` Command

For messages that are not of great importance ("nice to know" as opposed to "need to know"), create a file containing the news item and place it in the `/var/news` directory. When a user logs in, if there is a new entry in `/var/news`, the user sees the following message:

`news:` *news_filename*

The user can then read the message with the `news` command.

Note that in order for the above to work, the following entry must be in /etc/profile for Bourne and Korn shell users:

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

For C shell users, the following entry must be in /etc/csh.login:

```
if ( -f /usr/bin/news ) then
   news -n
endif
```

Mailers

If you need to send a long message to an individual user, then use one of the mailers.

The wall Command

To simultaneously send a message to all users logged in, use the wall (write all) command. This command is typically used to generate a message that is of immediate concern to the users. For example, if, for some reason, the system must be shut down immediately, wall can be used to send a message to all the users currently logged in warning them of the impending shutdown. When invoked, wall reads standard input until an end-of-file is received and then sends the message to all logged in terminal lines.

```
# wall
The system will be shut down in 5 minutes.  Please logoff.
Ctrl+d

Broadcast Message from root (console) Sat Mar 18 11:22:43...
The system will be shut down in 5 minutes.  Please
log off.

#
```

wall can only be invoked by the superuser. When invoked, any permissions a user may have set to prevent someone from writing to their terminal (for example, mesg n) are overridden.

The /etc/motd File

For messages that every user should be aware of, place an entry in /etc/motd (the Message Of The Day file). For example, if you decide to shut the system down for an evening for preventive maintenance work, place a message in /etc/motd letting the users know which evening the system will be down so they can plan their work time accordingly.

The file /etc/motd is displayed every time a user logs in. The /etc/profile and /etc/csh.login, which are executed automatically during system startup, contain the command cat /etc/motd.

3-12. SLIDE: Security Considerations

Security Considerations

- Log off unattended terminals
  - Set time out (TMOUT or AUTOLOGOUT) variable if available
- Monitor set user ID programs
  - Use ncheck -s to produce a report of files with set user ID mode on
- Keep current directory (.) out of root's PATH variable
- Insure your terminal protections deny access to others
- Monitor security logging files

/var/adm/wtmp	logs successful login attempts
/var/adm/btmp	logs unsuccessful login attempts
/var/adm/sulog	logs use of su command
/etc/securetty	specifies the tty files on which the root can log in

H6296 3-12 33

Student Notes

As an administrator of an HP-UX system, maintaining the overall security of the system is a primary responsibility. There are many aspects of system security, some of which we'll touch upon here; however, some preliminary words may be in order. The UNIX operating system, upon which HP-UX is based, was originally designed to be a research and development system which was no more security conscious than any other available operating system. Yet as UNIX and UNIX- derivative systems become more prevalent and their uses expanded beyond closed-shop research and development installations, there became a heightened interest in the security aspects of the operating system. Since the system administrator is responsible for the overall maintenance of the system, it became part of the administrator's job to employ and monitor security procedures.

The User's Responsibility

The individual user should assume some responsibility for security. Users should:

- Set proper permissions on their files
- Protect their passwords
- Log out when they leave their terminal

Each individual user must be responsible for maintaining the correct permissions on their files and directories. Correct permissions are the only way to prevent someone from accessing unauthorized material. A prudent administrator may, on occasion, look at the permissions set on various important files and directories (for example, home directories) and advise their owners if something appears awry.

Each individual user must also be responsible for protecting his or her password. Users should be encouraged not to divulge their password to anyone, and should clearly understand the implications of supplying their password to another person.

Users should also be responsible for logging out of the system while their terminals are unattended. One way you, as an administrator, can enforce this is to use a time-out value such as the `TMOUT` (Korn shell) or `AUTOLOGOUT` (C shell) variable. With `TMOUT` or `AUTOLOGOUT` set, if a login shell remains idle for a specified number of seconds, the shell is killed and the user is logged off.

Tips for the Administrator

You should make sure that permissions on system files and directories are correct. The following list shows you what the permissions should be on some specific files and directories:

`/etc/`	755 (`rwxr-xr-x`), owned by `root`.
`/usr/bin/*`	555 (`r-xr-xr-x`), owned by `bin`.
`/etc/passwd`	444 (`r--r--r--`), owned by `root`.

Set-User ID Programs

Indiscriminate use of the set user ID (or set group ID) permission setting can result in some security difficulties. When a set user ID program is executed, the effective user ID of the person executing the program is changed to that of the owner of the program for the duration of the program's execution. For example, if the user `john` executes a program owned by bill which has the set user ID bit set, then for the duration of the program's execution `john`'s effective user ID is the same as bill's. Though this is an advantageous feature of the operating system, it does raise some concerns. For example, if a set user ID program is writable, another program can be copied on top of it while retaining the set user ID permission setting. Then, whenever the new program is executed, the effective user ID of the person executing the program is changed. Extreme caution must be taken with set user ID programs that allow escaping to a shell. Since the effective user ID of a user is changed when a set user ID program is run, if a shell can be invoked from within the program, the spawned shell is established with the ownership being the same as the owner of the set user ID program.

There is a command that can be used to monitor set user ID programs. The `ncheck -s` option will list all the files in a file system that have the set user ID bit on.

Tip

Redirect the output of the initial invocation of `ncheck -s` to a file. Use this file afterwards to compare it with the output of following invocations. `ncheck -s` reports about a hundred files (with standard HP-UX installed, not including any optional software), and it's not easy to compare every single line.

Root PATH variable

If the root `PATH` variable contains an entry for the current directory, you have a potential security problem. The following example explains why.

Assume this is root's `PATH`:

```
PATH=.:/usr/sbin:/usr/bin:/usr/contrib/bin:/usr/local/bin:/sbin
```

A user could create a script in his or her home directory (or any other directory) with the same name as a common command (for example, `$HOME/ls`). Assume the script is executable and contains the following commands:

```
echo bandit::0:1:I am Superuser Now:/:/bin/ksh >> /etc/passwd
rm $0
/usr/bin/ls
```

If the user can convince the superuser to log in as root, change to his or her home directory, and run the `ls` command, then the `bandit` entry will get added to the `/etc/passwd` file. This happens because the entry for current directory comes before `/usr/bin` in root's `PATH` variable. (HP-UX searches for commands in directories in the order that they appear in the `PATH` variable.) So, the `ls` script in the user's home directory will execute instead of the real `ls` command in the `/usr/bin` directory. Since root is executing this script, the `echo` command which appends a line to `/etc/passwd` will work. Then, since the *$HOME/*`ls` script invokes the real `/usr/bin/ls` command, it will execute and everything will seem normal. Root wouldn't even know anything happened. Therefore, *never* put the current directory at first in root's `PATH`.

Terminal Protections

Insure that all terminal file protections are `crw--w----` and *not* `crw--w--w-`. If your terminal has write access by other terminals and you are logged in as root, it is possible to program the softkeys from the remote terminal by writing a series of escape sequences to it. Then someone could also send escape sequences to execute the command programmed into the remote softkey with the capability of the user logged in at that remote terminal - that being root in this case. It's complicated, but the moral is: Keep the protections of your terminal assigned only to you if security is an issue. Also, while logged in as root, `mesg n` takes away write permissions to your terminal.

Security Logging Files

Monitor the security logging files. This will help tell you if anyone is trying to break system security. The files are:

`/var/adm/wtmp` The system uses this binary file to keep a history of logins, logouts, and date changes. The system automatically creates this file, which grows without bound. Check the file regularly and empty it. Use `/usr/bin/last` to access the contents of the file.

`/var/adm/btmp` If this binary file exists, the system uses it to keep track of *failed* login attempts. The file grows without bound, so check and empty it regularly. Use `/usr/bin/lastb` to access the contents of the file.

Note

The permissions on `/var/adm/btmp` should be 644.

`/var/adm/sulog` If this text file exists, the system uses it to keep track of information on the use of the su command. It contains old and new user names, the corresponding terminal name, the date and time, and whether the attempt was successful. To enable logging of this information, create a zero-length file called `/var/adm/sulog` with the `touch` command. To look at the information, use the `more` or `cat` command.

`/etc/securetty` If it exists, this text file specifies the `tty` files on which the root user can log in. You must explicitly create this file and place the `tty` device file names in it. To look at the information, use the `more` or `cat` command.

Table 3-1.

	enable logging	display log file
/var/adm/wtmp	(automatic)	last
/var/adm/btmp		lastb
/var/adm/sulog		more /var/adm/sulog
/etc/securetty	vi /etc/securetty	more /etc/securetty

Note

`/var/adm/wtmp` and `/var/adm/btmp` grow without bound. Empty them periodically. You can do this by emptying the file with `> /var/adm/btmp`. This is better than removing the file with `rm` and re-create it with `touch`.

3-13. LAB: Hands-On Adding Users

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. Invoke `sam` and add a user to your system. (You must be superuser to invoke SAM.) Use your name as a user name. Assign the user to a group called `class` and give him or her the Posix shell.

Now, exit SAM and look at the `/etc/passwd` and `/etc/group` files. Do you see the user you added?

2. Add a user to the system using HP-UX commands. This time, use your partner's name as the user name. (If you don't have a partner, pick any name.) Use a group called `class` and give the new user the C shell. The steps are provided below.

- Add entries to `/etc/passwd` and `/etc/group`.
- Create a home directory for the user and set the permissions and ownership correctly.
- Copy default set-up files to the home directory and set correct permissions and ownership.

Now, use the `login` command to log in as the user you created. Type `id` to verify your user ID.

3. Run the commands to check the integrity of the `/etc/passwd` and the `/etc/group` files. Discuss your findings with the instructor.

4. Add a user called `date` that executes the `date` command. What would happen if you tried to log in using the user name `date`?

5. Try changing the password on your own student account (not the one you just created) to your first name. Does this work? Why or why not?

6. Create a news file that everyone on the system will be able to see, which contains your name and the name of your company. Use your name as the name of the file.

3-14. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. What steps is SAM taking when performing the following tasks.

   Adding a new user

   De-activating a user

   Modifying a user's information

   Adding a new group

2. Describe the 7 fields of the `/etc/passwd` file

3. Describe the fields of the `/etc/group` file.

4. What does it mean to set up groups? Explain.

5. List the steps to manually add a user.

6. Describe two ways to create a user and send him straight into an application program.

7. What is the difference between `/etc/profile` and `.profile`?

8. What is the `wall` command and how could it be useful?

9. If you were going to shutdown the system tonight at midnight, how might you communicate this to all of your users?

10. Explain why it is dangerous to put `.` ("dot") in root's `PATH` variable.

Module 4 — HP 9000 Hardware Overview

Objectives

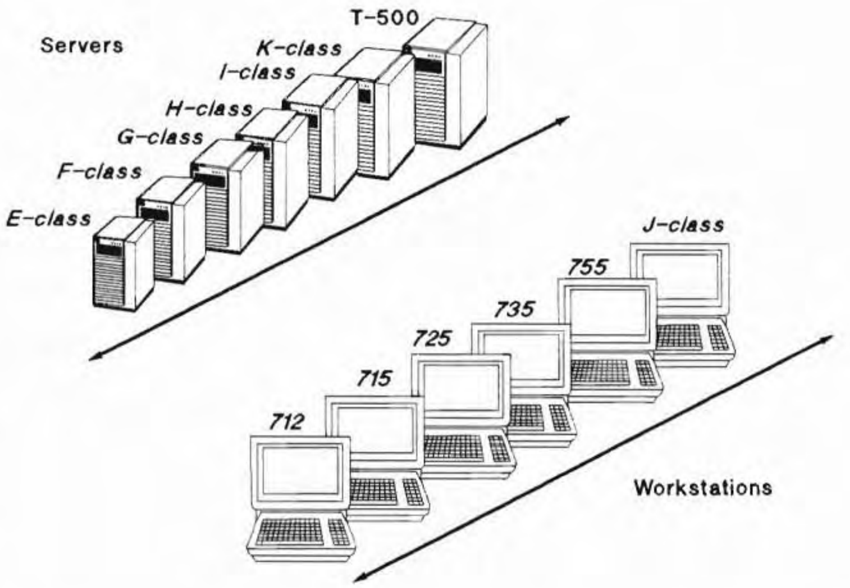
Upon completion of this module, you will be able to:

- List features of PA-RISC Architecture.
- Describe the components of an HP 9000 PA-RISC system.
- Describe types of I/O interfaces and their functions.
- Determine the hardware configuration of the local system.

Module 4 — HP 9000 Hardware Overview

4-1. SLIDE: The HP 9000 Product Family

The HP 9000 Product Family



H6296 4-1 34 © 1995 Hewlett-Packard Co.

Student Notes

The HP 9000 product family encompasses both server and workstation platforms.

HP 9000 Servers

The HP 9000 server product line contains the broadest range of RISC systems in the UNIX market. The servers are designed to meet diverse business needs and to provide customers with a long-term growth path. All servers use the same HP-UX operating system and provide object-code compatibility across the entire product line.

Module 4 — HP 9000 Hardware Overview

HP 9000 Workstations

The HP 9000 workstation family includes a full range of workstations—from entry-level price/performance leaders, to mid-range workhorses, to high-end performance superstars and a complete family of X stations and terminals, as well as a full range of graphics systems.

HP 9000 workstations are built on open systems standards, and they are fully upward- and downward-compatible—within the workstation family as well as with HP's servers. That's because both the families are based on HP's powerful PA-RISC processor architecture and the UNIX-based HP-UX operating system.

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.

PA-RISC

All HP 9000 systems use HP's Precision Architecture RISC (PA-RISC) technology to provide high performance and reliability. PA-RISC is built upon Reduced Instruction Set Computing (RISC) principles, a design approach that delivers greatly simplified computers that are optimized to provide the highest performance for a given integrated circuit technology. The inherent simplicity of PA-RISC implies that computer systems can be implemented with fewer components to achieve superior reliability when compared to older Complex Instruction Set Computer (CISC) systems.

At the core of PA-RISC is an instruction set containing 140 carefully selected, fixed format instructions. Because the instruction set is simplified, instructions can be hardwired directly into the CPU. Hardwiring eliminates microcode and the necessity to decode complex instructions, which leads to maximum performance.

PA-RISC utilizes a load/store design and register-to-register operations to reduce the number of memory accesses. To further enhance performance, optimizing compilers schedule instructions and manage the instruction pipeline. Instruction pipelining is a technique that overlaps instruction processing so that one instruction can begin to execute before the previous one has finished. With hardwired control, a load/store design, and optimizing compilers, instructions can be executed on virtually every clock cycle. Single-cycle instruction accounts for much of the superior performance of PA-RISC.

All instructions are 32 bits in length. This simplifies the instruction fetch mechanism since the location of instruction boundaries is not a function of the instruction type. Full 48-bit or 64-bit virtual memory addresses are supported, representing a significant expansion over the addressability of typical 32-bit systems.

4-2. SLIDE: I/O Architecture and the SPU

I/O Architecture and the SPU

HP 9000 Workstations

- Expansion Card Bus: EISA (Models 715, 725, 74x, 755), VME (Models 74x)

HP 9000 Servers

- Expansion Card Bus: HP-PB (Integrated Business Servers and Models T500, 890, 8x2S, 815, 808
- Expansion Card Bus: CIO (Models 8x5S/SE, 840, 850, 860, 870)

H6296 4-2 35

Student Notes

The HP 9000 workstation and server families utilize several different hardware architectures and I/O subsystems. All of these systems are designed to accomplish the same goal, to transfer data between the central processing unit, the physical system memory, the virtual system memory (swap space), and the peripheral devices.

These transfers are accomplished via one or more internal and external buses. The hardware addressing varies in complexity and length as we move from the simple architectures to the larger, more advanced architectures.

We will start out by reviewing the hardware components of the workstation, followed by the server system.

Terminology

The following list describes some of the components that are involved with system I/O:

SPU System Processing Unit. The SPU cabinet contains the CPU, system memory, and the I/O system. Many models will also have internal peripheral devices such as disk tape drives. The SPU will also have one or more internal interface cards providing the mechanism to connect many types of external devices.

CPU Central Processing Unit. In this slide, the term refers to the instruction-processing module inside the computer, not to the computer itself. The CPU processes data supplied to it by the I/O system.

Memory Physical Random Access Memory (RAM) located in the SPU and available for use by the CPU and I/O system. This is where all data is operated on.

Internal Bus The electronic path that connects the various areas of the SPU and allows data to flow throughout. Some models have several different internal buses joined by bus converters to facilitate efficient data flow between areas that operate at different speeds.

I/O System The physical hardware in the SPU that allows both built-in and add-on interface cards to be connected.

Interface Card An accessory card that is either built into the SPU or plugs into slots in the I/O system (typically on the back of the SPU case). This card will provide connections for and access to peripheral devices.

External Bus A mechanism (like a cable) that can connect many like devices to one interface card. This is how devices like disks and tape drives are attached to the system.

4-3. SLIDE: HP 9000 Workstations

HP 9000 Workstations

- First generation workstations used PA-RISC 1.1 for demanding graphics and computation-intensive applications
- Second generation workstations use PA-RISC 7100 superscalar architecture. Subsequent systems use the 7100LC chip
- PA-RISC 1.1 upgradable to PA-RISC 7100
- J-Class workstation, based on the PA-RISC 7200 chip, offers 2-way Symmetric Multiprocessing

H6296 4-3 36

Student Notes

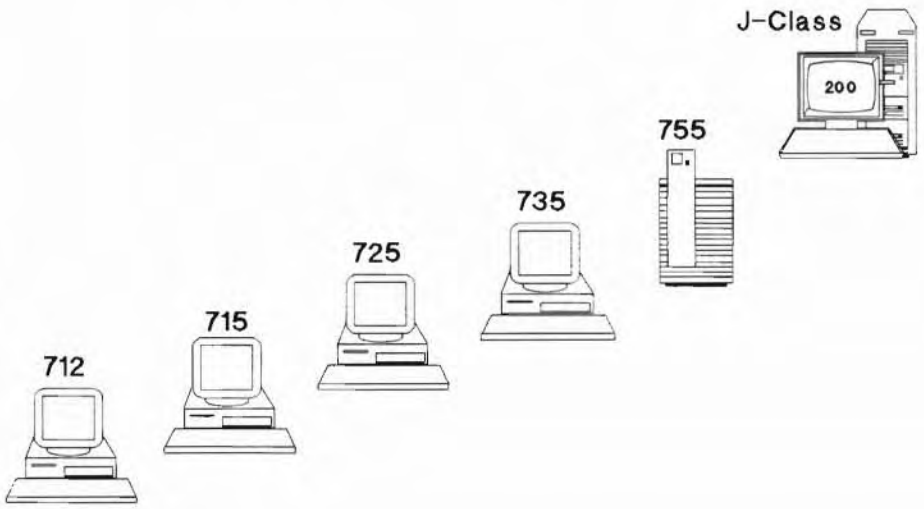
HP 9000 first-generation workstations are designed around the PA-RISC 1.1 chip set for demanding graphics and computation-intensive applications. Second-generation HP 9000 workstations are based on the PA-RISC 7100 superscalar architecture.

The PA-7100LC is a low-power, low-cost RISC design with floating-point and integer processors integrated in the same chip.

The J-class is the first HP workstation to offer two-way Symmetric Multiprocessing (SMP). By adding a second processor, users can improve not only the overall throughput of the system, but also the opportunity to reduce the time to completion for critical simulations.

4-4. SLIDE: HP 9000 Workstation Family

HP 9000 Workstation Family



H6296 4-4 37

Student Notes

The HP 9000 workstations are based on either the PA-RISC 1.1, the PA-RISC 7100, or the PA-RISC 7200 chip sets.

Module 4 — HP 9000 Hardware Overview

Table 4-1.

PA-RISC 1.1	PA-RISC 7100	PA-RISC 7200
705	712	J200
710	715	J210
720	725	
730	735	
750	755	

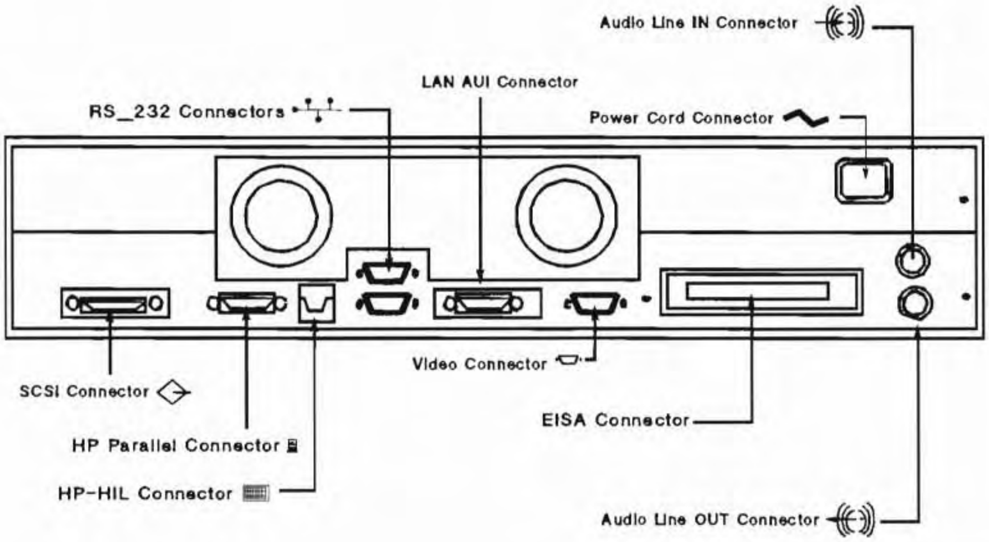
The HP 9000 workstations include these fundamental hardware components:

- System Processing Unit (SPU)
- Memory
- Monitor
- Disk (optional)

Workstation systems are differentiated by processor speed, memory and cache size and speed, mass storage capacity, graphics hardware and monitors available, and configurability/expandability.

4-5. SLIDE: Example of HP 9000 Workstation Hardware

Example of HP 9000 Workstation Hardware



H6296 4-5 38

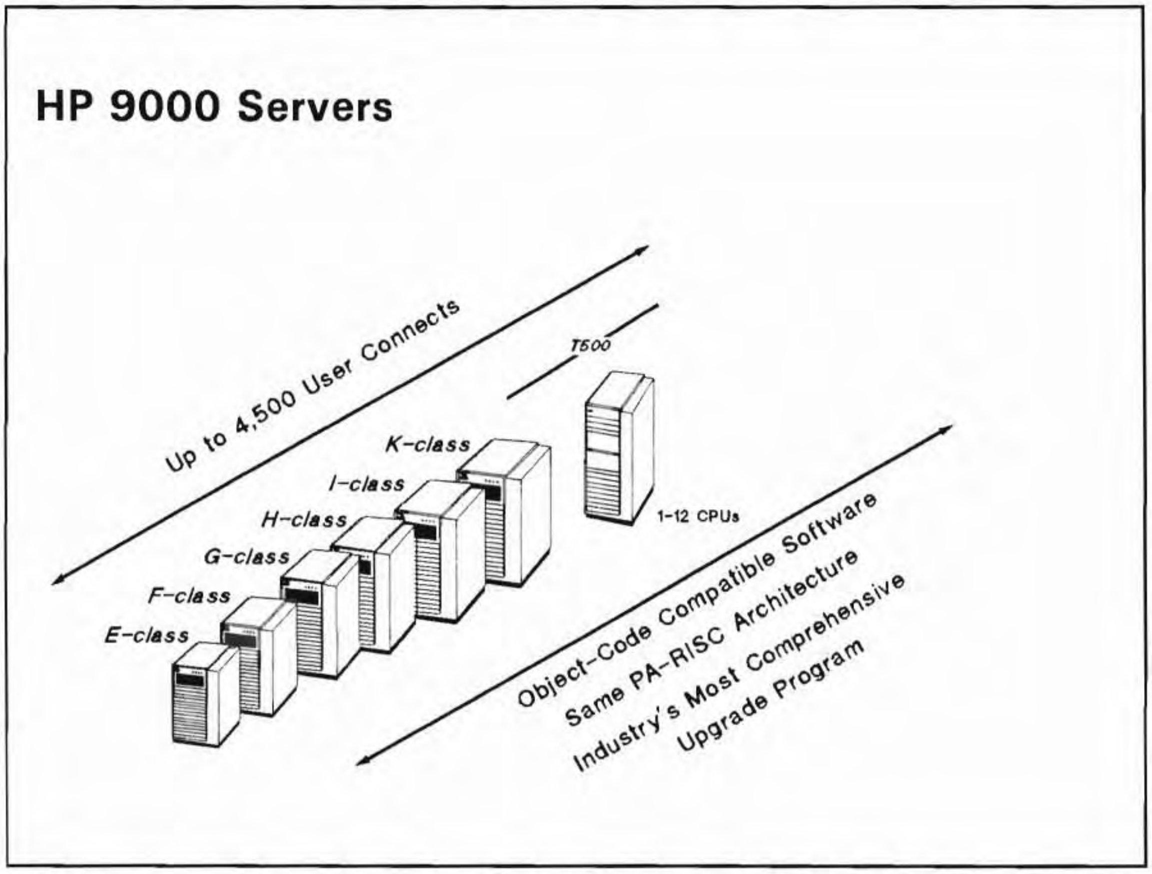
Student Notes

This is a view of a Series 715. The backplane is representative of the workstation family. Although all models will have the same core set of interfaces, they may be located in different areas on the backplane. Depending upon the model, additional I/O capabilities may be supported. For example, the 735 and 755 models come standard with Fast and Wide SCSI-2 interface, and can optionally have the FDDI network interface.

The list below describes some of the differences between the systems in terms of processing speed and I/O connectivity.

Model	Processor Speed	Number EISA Slots
712/60	60MHz	0
712/80	80MHz	0
715/33	33MHz	1 optional
715/50	50MHz	1 standard
715/64	64MHz	1 standard
715/75	75MHz	1 standard
715/80	80MHz	1 standard
715/100	100MHz	1 standard
725/50	50MHz	4 standard
725/75	75MHz	4 standard
735	99MHz	1 standard
735/125	125MHz	1 standard
755	99MHz	4 standard
755/125	125MHz	4 standard
J200	100MHz	4 standard
J210	210MHz	4 standard

4-6. SLIDE: HP 9000 Servers



H6296 4-6 39

Student Notes

The HP 9000 Server platform is unique in its ability to support environments of all sizes, ranging from workgroups and replicated sites to the departments and data centers of large enterprises. The HP 9000 Servers are the broadest family of binary compatible servers in the industry, encompassing one user at a desktop to thousands of users in a data center. Utilizing PA-RISC technology, the HP 9000 Servers have provided customers with proven performance leadership and outstanding price/performance. Now with the HP/Intel alliance, HP customers can be assured of application compatibility with the volume chip architecture of the next decade. Instead of obsoleting equipment through disruptive changes that other vendors require, HP innovates on the proven HP 9000 design, providing advanced technology such as symmetrical multiprocessing and a unique system design that is built for the future, yet provides leadership performance today.

E-Class

E-Class Servers are ideally suited to meet customer needs for small businesses, replicated sites, and branch or workgroup servers. With plenty of performance headroom to add new applications and additional users, these entry-level systems are exceptionally cost-effective. Yet in comparison to PC servers, the E-Class Servers go much further and provide enterprise-class performance, reliability, integration, and systems management functionality required for running business-critical applications. E-Class Servers feature PA-RISC price/performance, integrated I/O devices to minimize complexity, and a compact chassis footprint. At similar prices to less functional PC LAN servers, E-Class Servers offer a tremendous value, providing users with an entry-level server with enterprise-class functionality.

G-Class, H-Class, I-Class

Designed to be used as departmental or division servers, these midrange business servers offer a unique combination of highly competitive performance and value. With a choice of systems expandable to a 2-way symmetrical multiprocessor architecture, a floating-point coprocessor, and extensive configurability in memory and I/O slots, these systems deliver maximum flexibility and choice for midrange customers. Designed for easy upgradability to protect customer investments, G, H, and I- Class Servers provide 5X performance scalability within the same system cabinet.

K-Class

As the newest line of HP 9000 Servers, the K-Class addresses requirements that extend from the midrange to the data center. Using HP's new PA-7200 processor, the K-Class initially offers twice the performance of the G, H, and I-Class Servers but maintains full binary compatibility for applications. Memory and disk capacity are also expanded. In addition, for networking and clustered environments the K- Class supports next-generation ATM and Fibre Channel communications. Similar to the G, H, and I-Class Server design, the K-Class Servers all contain a single system cabinet which provides for unmatched investment protection. Lastly, K-Class Servers will provide midrange performance leadership through simple board upgrades to multiple generations of PA-RISC technology.

T500 Corporate Business Servers

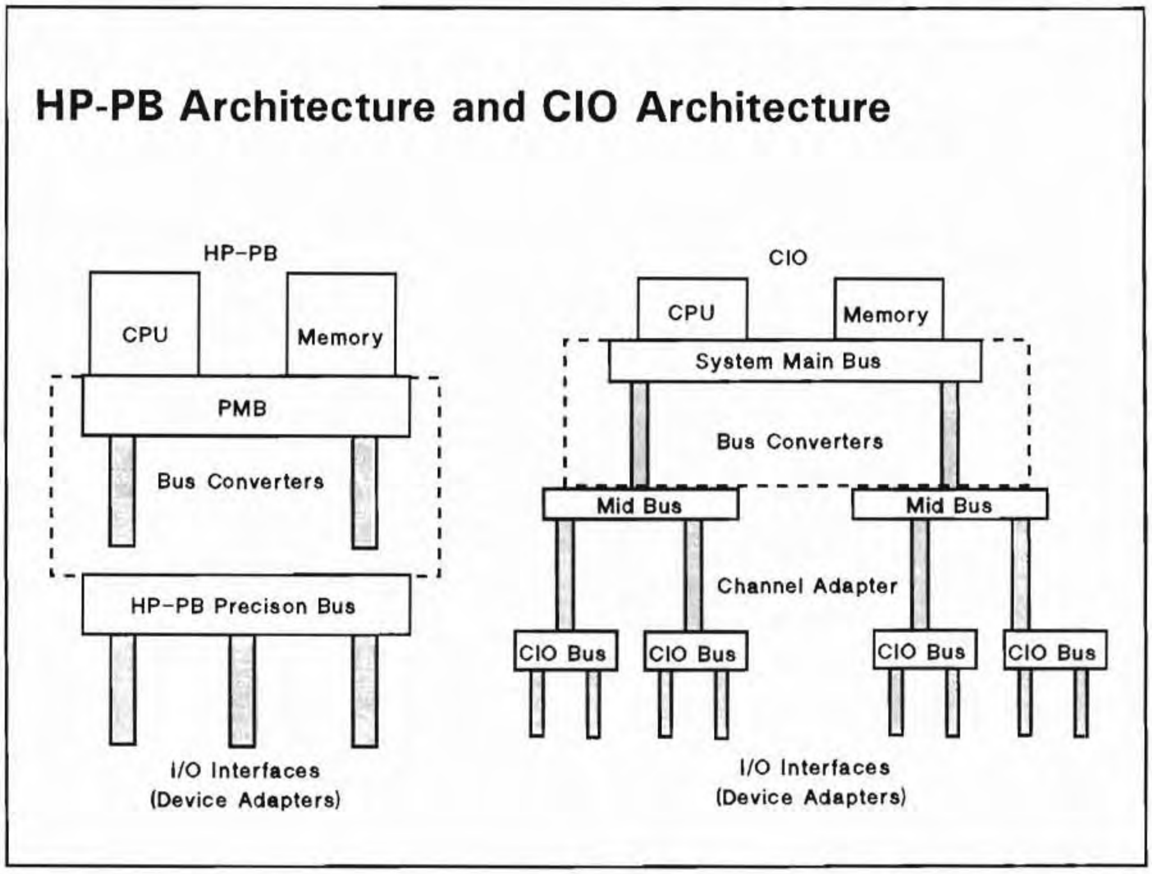
The HP T500 Servers are the most capable and expandable high-end UNIX- based servers available for commercial applications. These servers were designed to be modern, viable open systems alternatives to older, inflexible, and expensive mainframe systems. With up to 12-way symmetrical multiprocessing and balanced high-end system design, these systems deliver maximum performance, growth capabilities, and value in the data center. In addition, the T500 Servers contain a remote service processor which monitors the health of the server to proactively assure maximum uptime for corporate data center applications. Designed for an unprecedented combination of performance scalability, high availability, and investment protection, the T500 Servers provide an unmatched value in enterprise-class data center computing.

Table 4-2. HP 9000 Servers

Server Class	Memory	Maximum Disk	Available I/O Slots
E-Class	16-512MB	up to 1,000GB	2-4 HP-PB slots
G-Class, H-Class, I-Class	32-768MB	up to 3,000GB	4-12 HP-PB slots
K-Class	32-2048MB	up to 5,000GB	5 new HP-HSC slots and 8 HP-PB slots
T500 Corporate Business Servers	256-2048MB	up to 10,000GB	14-112 HP-PB slots

Other models in the HP 9000 server line include the Model 8x2 Family, the Model 8x5 Family, and a group of systems once referred to as the "High End" family, consisting of Models 850, 855, 860, and 870/100 and 870/200. You may have some of these systems at your site. These systems are quite old and are not specifically covered in this section.

4-7. SLIDE: HP-PB Architecture and CIO Architecture



H6296 4-7 40

Student Notes

There are two types of HP-PA machines—Precision Bus architecture (HP-PB), used with current systems, and Channel I/O architecture (CIO), used in older systems.

Systems with CIO architecture include the Model 8x5S/SE family and Models 840, 850, 860, and 870.

The CIO machines have a multi-tiered bus structure. As you move up the hierarchy, the buses get faster. There can be either a two-tier or a three-tier structure. The two-tier structure consists of a CIO and a Mid-Bus. The three-tier structure consists of a CIO, Mid-Bus, and System Main Bus (SMB).

Systems with HP-PB architecture include all current systems as the Integrated Business Server family, models T500 and 890, as well as the older Model 8x2S family, and Models 808, 815.

Module 4 — HP 9000 Hardware Overview

The Integrated Business Servers /F/G/H/I systems have only one bus, the Precision Bus. Models Exx, T500 and 890 uses a two-tier bus structure: PMB for processor(s) and memory, and HP-PB for I/O.

Terminology

SPU	The System Processing Unit consists of all hardware components enclosed within the computer's "box", excluding peripherals like the disk and tape drive that may be housed within the SPU box, as in the case of systems in the Integrated Business Server family.
CPU	Central Processing Unit. In this case, the term refers to the instruction-processing module inside the computer, not to the computer itself.
CIO	Channel Input/Output. Refers to the hardware I/O architecture of systems in the Model 8x5S/SE, 6x5SV, 850, 860, 870 families.
HP-PB	HP-Precision Bus. Refers to the hardware I/O architecture of the Integrated Business Server and 8x2S families, as well as Models 808, 815, 890 and all E, F, G, H and I Models.
Mid Bus	The Mid Bus (also known as Central Bus) exists only on CIO machines. The Mid bus is a high speed bus over which the memory and the CPU communicate.
CIO Bus	The CIO Bus exists on the CIO machines. The CIO Bus is the general-purpose I/O bus (low speed compared to the Mid Bus) over which the I/O devices communicate with each other and the Channel Adapter.
Precision Bus	The Precision Bus is found on HP-PB machines. The Precision Bus is the only bus on Integrated Business Servers, other than Model Exx. The CPU, memory and I/O devices all communicate with each other using the Precision Bus.
SMB	The System Main Bus is found only on "High End" systems. It is a high speed bus over which the CPU and memory communicate.
PMB	The Processor Main Bus is found only on Corporate Business Server 890.
Slot	The physical place in the back of the computer where a card plugs in. Each slot has a number/letter. On a CIO machine, memory, CPU, channel adapters and, in some cases, graphics cards plug into Mid Bus slots; and CIO cards plug into the CIO Bus. On an HP-PB machine, all devices plug into the Precision Bus.

Hardware Modules that Connect to Buses

Hardware modules (bus converters, channel adapters, and device adapters), are cards or boards, inserted into slots in the system's card cage to connect to the bus architecture. Once configured, transfer of data among bus structures is transparent. Each slot in the card cage is identified by a number or letter for hardware addressing.

The following hardware modules connect buses to buses, or buses to devices.

Bus Converter	A board that serves as an internal interface between higher- and lower-speed buses. Bus converters can increase the number of modules in a system, thus enabling more devices to be connected.
LAN Converter	A board that serves as an internal interface between the system bus and the 802.3 LAN interface.

Channel I/O Adapter

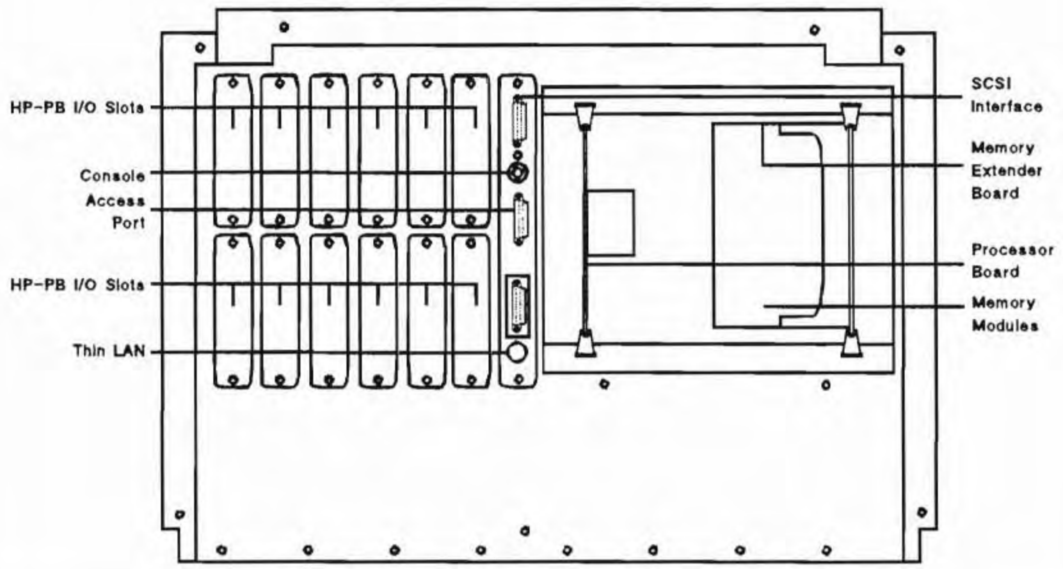
A module that serves as an internal interface between the mid-bus and the lower-speed CIO bus. More than one channel adapter can be installed on most systems, thus expanding the number of possible modules.

Device Adapter

Interface cards such as HP-PB, HP-FL, SCSI, LAN, and MUX cards, installed in the I/O slots of the CIO or HP-PB bus. Device adapters make the link that enables the system to communicate with external peripherals. Device adapters are also called I/O cards and host adapter cards. CIO device adapters are also called CIO cards. SCSI, HP-FL, and HP-IB interfaces accommodate multiple devices per interface card; the devices are cabled together.

4-8. SLIDE: The Integrated Business Servers (G H I K Series)

The Integrated Business Servers (G H I K Series)



H6296 4-8 41

Student Notes

The HP 9000 Models Gxx, Hxx, Ixx, and Kxx are integrated mid-range systems. These systems differ in the amount of memory, total disk storage space, number of slots, and processing power. In addition, the Kxx series can be configured with up to 4 100MHz processors.

All of these systems use the HP Precision Bus architecture. Integrated packages include a processor, an embedded disk drive and an embedded Digital Data Storage (DDS) tape drive. The package for these components was designed to minimize the amount of space taken by a computer in an office or computer room. In compliance with the Eurocard standard, the HP-PB is designed to accommodate single-high and double-high I/O cards.

As previously mentioned, each model supports a different number of interface slots. If the slots are not available, they will be covered in plastic.

I/O Cards for Integrated Business Server Systems

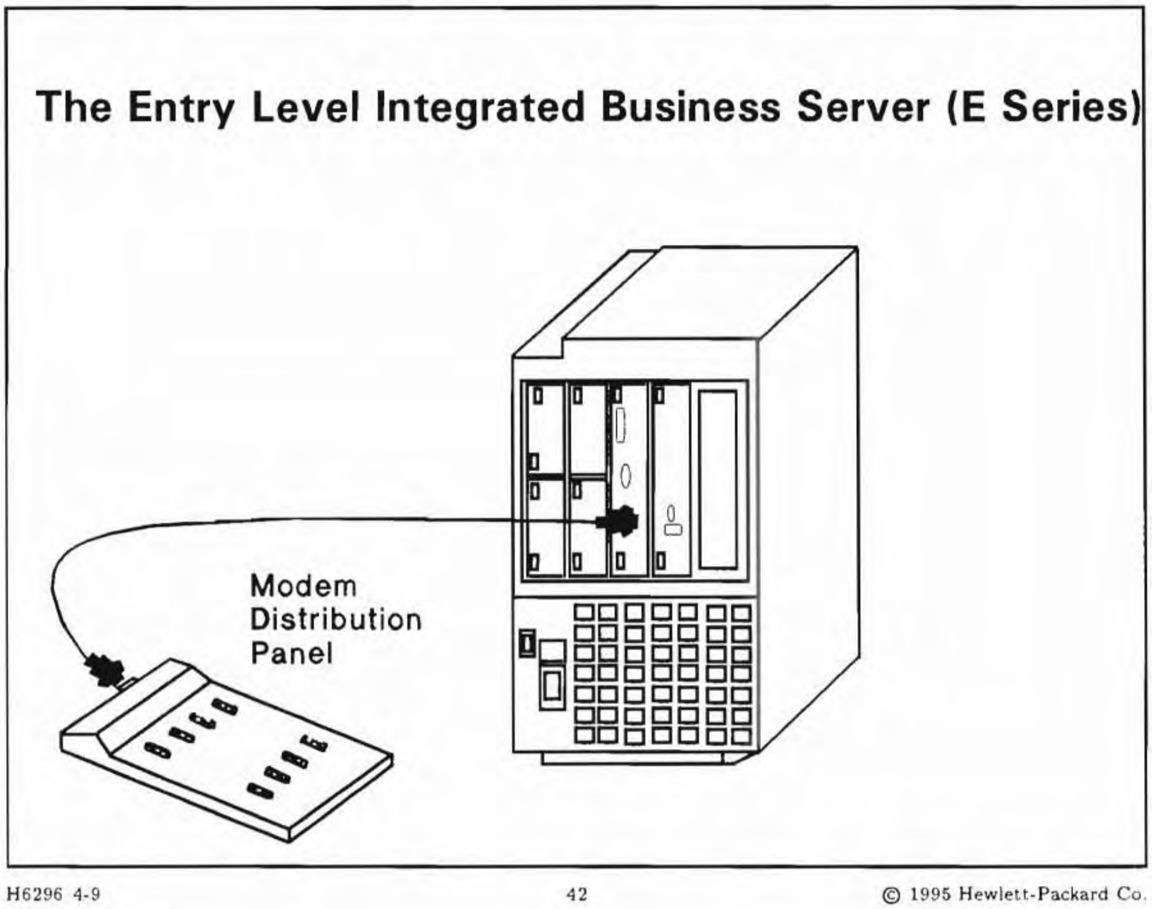
These I/O cards can be placed in single-high HP-PB slots:

- 8-channel or 16-channel MUX card
- SCSI/Centronics card
- HP-IB card
- LAN card
- Token ring
- X.25 card
- SNA link card

These I/O cards can be placed in a double-high HP-PB slot:

- HP-FL card
- FDDI Interface card

4-9. SLIDE: The Entry Level Integrated Business Server (E Series)

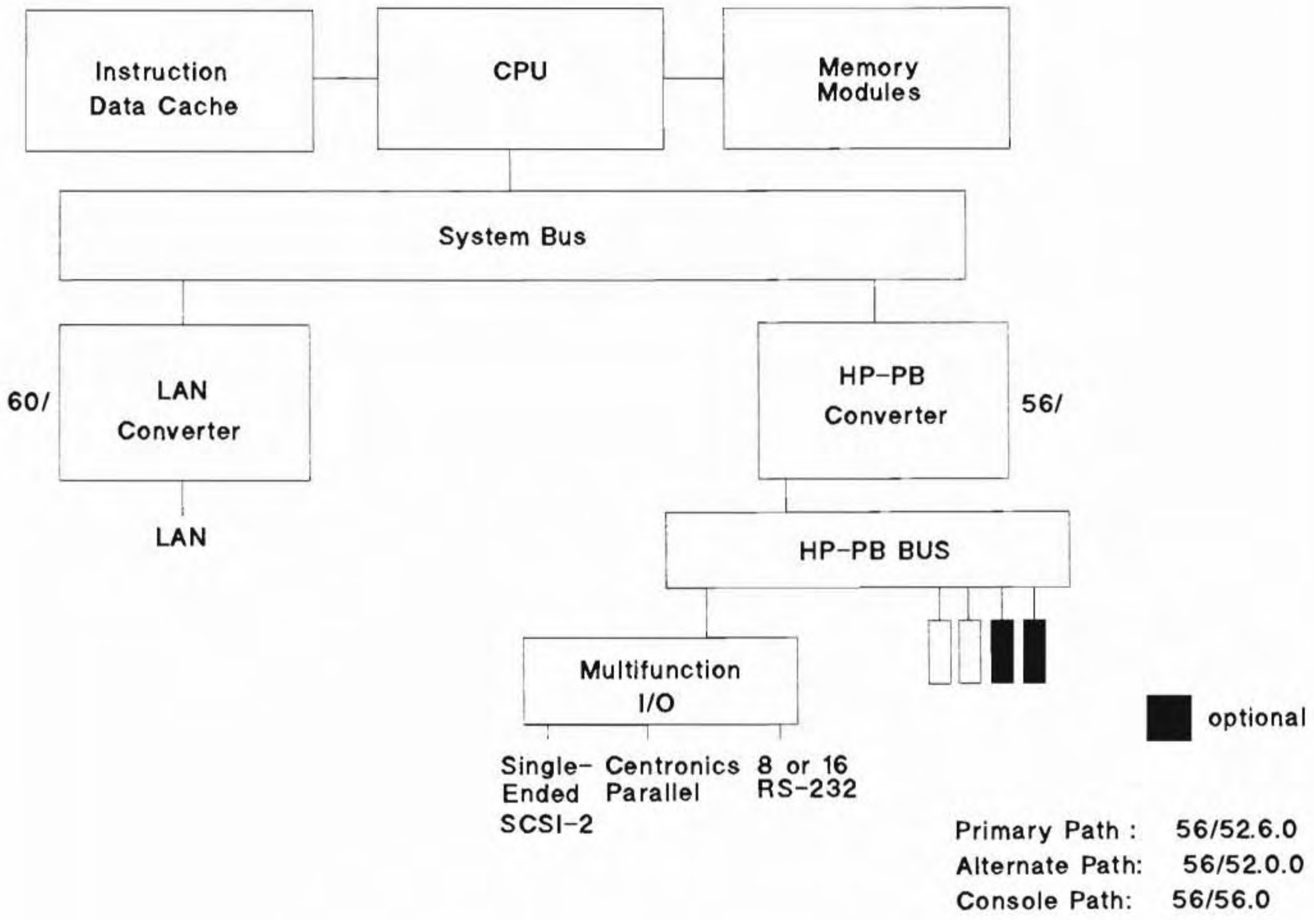


Student Notes

Although the E Series is also part of the Integrated Business Server Family, its backplane is slightly different than the F G H or I Series. The multifunction I/O card is connected to the HP-PB bus, which in turn is connected to a Bus-Converter, which has an address of 56.

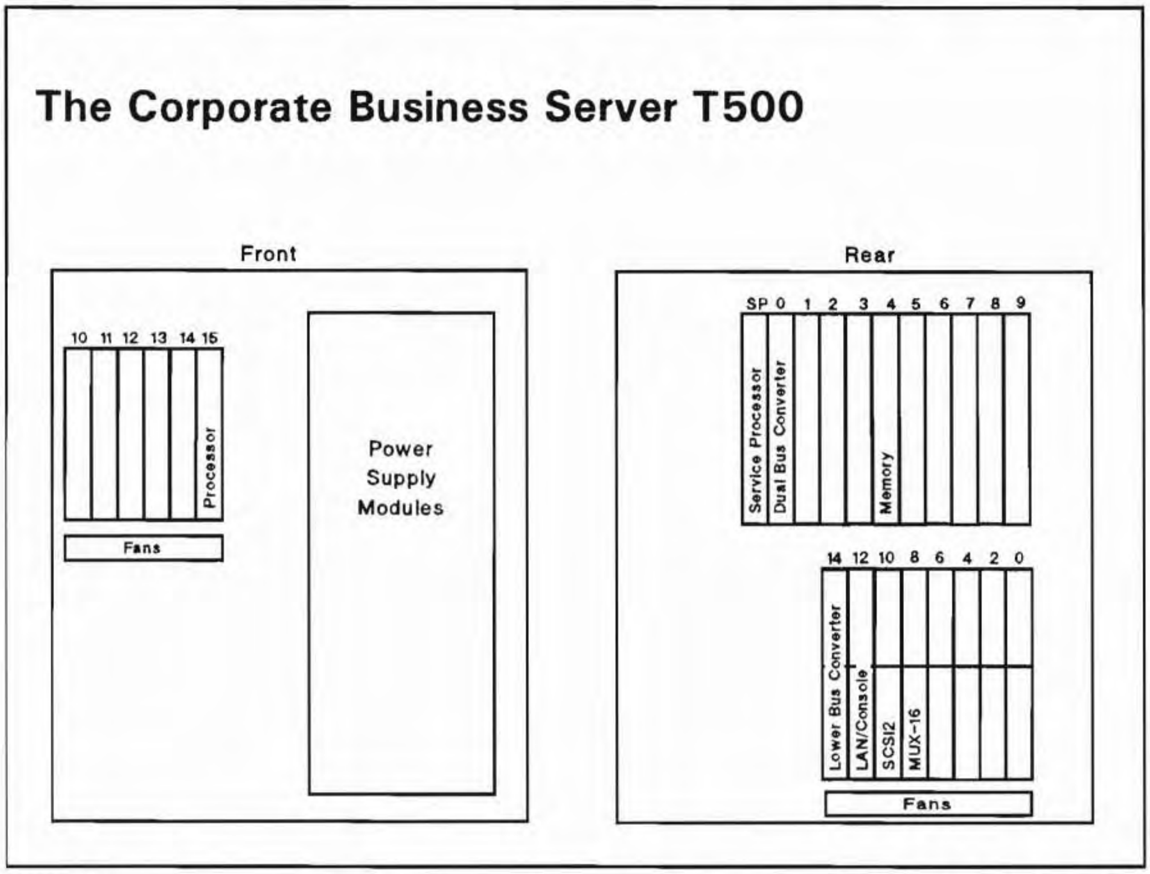
The LAN adapter is not on the multifunction I/O card, and is connected to a LAN Converter, which has an address of 60.

Below is a graphical representation of the E Series bus structure:



The E Series Bus Structure

4-10. SLIDE: The Corporate Business Server T500



H6296 4-10 43

Student Notes

The HP 9000 Corporate Business Server T500 is a high performance server designed for use in computing environments where capacity, performance, and growth potential are of prime importance. It is a technology upgrade to the 890. It can be configured with up to twelve 90 MHz HP-PA RISC processors, 2 Gigabytes of main memory, and 1900 Gigabytes of primary on-line disk storage using RAID technology and supports up to 4500 connected users.

The Model T500 uses a two-tiered architecture consisting of a Processor/Memory Bus and up to eight HP-PB buses. Dual Bus Converters (up to four) connect the PMB and the HP-PB buses. Each Dual Bus Converter supports two HP-PB buses. Each HP-PB has 14 available I/O slots, for a total of 112 I/O slots possible.

4-11. SLIDE: Device Adapters

Device Adapters

SCSI	Small Computer System Interface. There are two SCSI-2 interfaces available: Single-ended and Fast and Wide. Supports SCSI disks, DDS tape drives, CD-ROM drives, MO drives, quarter inch cartridges (QIC), 8mm tapes and IBM 3480-compatible drives
HP-IB	HP Interface Bus. Supports HP-IB disks, tape drives, CD-ROM drives, printers and plotters
HP-FL or PBA-FL	HP, or Precision Bus, Fiber Link (server only) Supports fiber optic hard disks only.
MUX	Multiplexer. Supports serial terminals, printers, plotters, modems, and Access Port.
LAN	Local Area Network. For IEEE 802.3 and Ethernet networks.
GPIO	General Purpose I/O. Supports general purpose parallel I/O.

H6296 4-11 44

Student Notes

I/O interfaces (or device adapters) connect external devices to the CIO or Precision Bus. This slide lists I/O interfaces (or device adapters) commonly found on servers and identifies the external devices that each support.

SCSI, HP-FL, and HP-IB interface accommodate multiple devices per interface card; the devices are cabled together.

Cabling guidelines for these I/O interfaces are summarized in the following table.

Table 4-3. Cabling Guidelines

I/O Interface	Maximum Devices on an Interface	Maximum Total Cable Length	Mix Devices on an Interface	Terminator Required?
Single-ended SCSI	7 SCSI devices	6 meters	yes	on last device
Fast and Wide SCSI	15 SCSI devices	25 meters	yes	on last device
HP-IB	4 HP-IB disks 4 HP-IB plotters 4 HP-IB printers 8 non-disk devices	1.25 kilometers (with extender)	not with disk	no
HP-FL or PBA-FL	8 FL disks	500 meters	disks only	no
16-channel MUX	16 serial devices	5 or 15 meters	yes	no
8-channel MUX	8 serial devices	5 or 15 meters	yes	no
6-channel MUX	6 serial devices	5 or 15 meters	yes	no
Centronics	1 centronics printer	2 meters	no	no

Cabling guidelines for these I/O interfaces are described in detail below:

SCSI Guidelines

- Up to seven SCSI devices can be on the same Single-ended SCSI bus, while fifteen SCSI devices can be on the same Fast and Wide SCSI bus.
- There are 7 SCSI device addresses (0-6) available for each Single-ended SCSI card, and 15 devices addresses (0-6,8-15) available for each Fast and Wide SCSI card. (SCSI address 7 is reserved for the SCSI controller) Most SCSI devices use one SCSI address, but the Optical Disk Library System (HP C1700A) uses 3 addresses (two for the MO drives and one for the AutoChanger). The HP5000 printer requires its own SCSI card, with one device per card.
- Different SCSI devices (disks and tapes) can be on the same SCSI bus, but you cannot mix single-ended and Fast and Wide SCSI devices on the same interface.
- The last SCSI device on a SCSI bus must have a terminator. This terminator provides matching impedance on the bus circuits. Without a terminator, the devices on the bus will not work.
- The total cable length of a SCSI bus (including external and internal cables) cannot exceed 6 meters for single-ended and 25 meters for Fast and Wide.
- Do not connect, disconnect, or power off any SCSI devices while the system is running. Doing so may corrupt data being transmitted on the SCSI bus.

HP-IB Guidelines

- Up to 4 HP-IB disks drives can be on the same HP-IB bus.
- HP-IB devices have a priority scheme where 0 is the highest priority.

- Generally speaking, do not connect disks with slower devices (for example, tape drives, printers) on an HP-IB bus. The performance of disks can degrade if they are on the same bus with slower devices.
- The maximum cable length for the standard speed HP-IB interface is 20 meters total.
- The maximum cable length for the high speed HP-IB interface is 15 meters total.
- An HP-IB extender can be used to extend the transmission distance between the SPU and external devices to as much as 1.25 km.

HP-FL (or PBA-FL) Guidelines

- Up to 8 Fiber Link disks drives can be on the same HP-FL or PBA-FL bus. (Note that the only Fiber Link devices available are disks.)
- HP-FL devices have a priority scheme where 0 is the highest priority.
- The maximum length of a Fiber Link cable connecting Fiber Link disks is 500 meters.

MUX Guidelines

- Any serial (RS-232C) device can be connected to a port of the MUX. Examples of serial devices include terminals, some printers, some plotters, and modems.
- Both 5 meter and 15 meter cables are available for most serial devices that connect to a MUX. Some devices, like terminals only use 5 meter cables.

Note

Power on all external devices before powering on the SPU (computer). Make sure that the devices have time to complete their self test before powering on the SPU. Power on all SCSI peripheral devices that provide termination power first.

4-12. SLIDE: SCSI Cabling Guidelines

H6296 4-12 45

Student Notes

SCSI interface cards support multiple SCSI devices. Here are some details and guidelines on how it's done. The SCSI interface that comes standard on the workstations is a SCSI-2 single-ended interface. (Single-ended SCSI-2 is also known as normal or standard SCSI-2.) This interface supports data rates slower than the optional SCSI-2 Fast and Wide or differential interface that plugs into a workstation's EISA slot. (Fast and Wide also is standard on the 735 and 755 models). SCSI-2 single-ended supports data rates of 5 MB/sec synchronous and up to 5 MB/sec asynchronous. SCSI-2 differential supports data rates of 10 MB/sec synchronous and up to 5 MB/sec asynchronous. Fast and Wide SCSI-2 supports data rates of 20 MB/sec.

Single-ended, differential or Fast and Wide SCSI-2 devices cannot be mixed on the same interface.

Cable lengths of 6 meters on single-ended and 25 meters on differential and Fast and Wide SCSI-2 are supported. No cable length segment can be less than 0.5 meters. Remember that devices internal to the chassis also have cable lengths that must be included in this maximum cable length calculation.

A maximum of seven SCSI devices are supported on a single-ended or differential SCSI interface card. Devices on these cards will each have a unique bus address between 0 and 6. SCSI bus address 7 is reserved for the SCSI interface and not to be used for a SCSI device bus address.

A maximum of 15 devices are supported on the Fast and Wide SCSI interface cards. Devices on the cards will use a unique bus address between 0 and 6, and 8 through 15. SCSI bus address 7 continues to be reserved for the SCSI controller interface and cannot be used for a SCSI device bus address.

Terminators are required on each end of a SCSI chain, namely on the host adapter (SCSI interface card) and last SCSI device in chain. The host adapter card is terminated internally. For the last SCSI device in the chain, there are two types of SCSI terminators: high-density terminators and low-density terminators. These terminators are electrically the same, but mechanically different. In other words, if the high-density terminator won't fit on your device, you are safe if you use a low-density terminator and vice versa.

4-13. SLIDE: SCSI Device Power Up Guidelines

SCSI Device Power Up Guidelines

- DO NOT
  - Connect or disconnect any device while the system is running
  - Turn power on or off to any device while the system is powered-up
- DO
  - Power on and complete self-tests on all peripherals before powering on SPU
  - Change bus addresses with device powered off

H6296 4-13 46

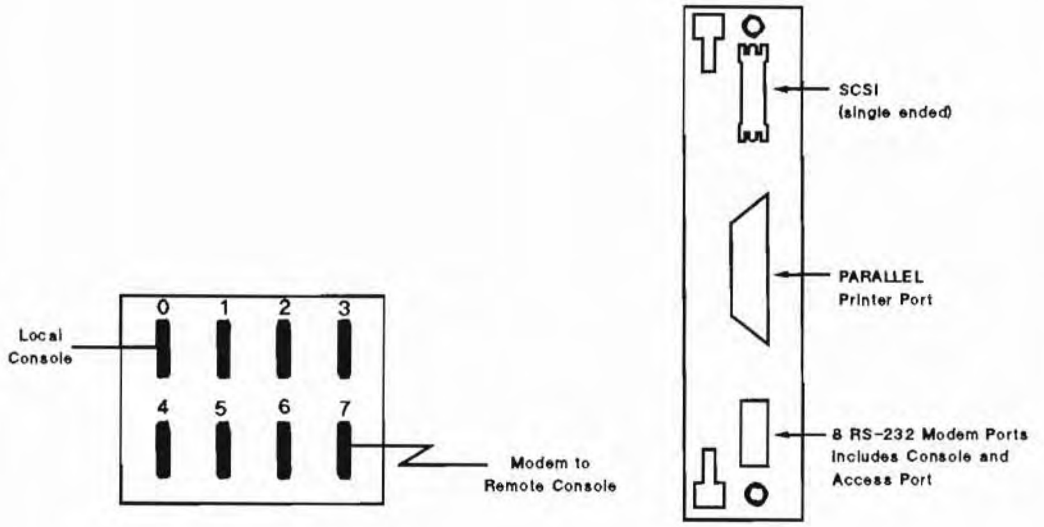
Student Notes

You will need to follow these SCSI device power up guidelines when changing the bus address on a SCSI device:

- *Do not* connect or disconnect any device while the system is running.
- *Do not* turn power on or off to any device while the system is powered-up.
- *Do* power on and complete self-tests on all peripherals before powering on SPU.
- *Do* change bus addresses only with device powered off.

4-14. SLIDE: Access Port on HP-PB Systems (Servers only)

Access Port on HP-PB Systems (Servers only)



H6296 4-14 47

Student Notes

The Access Port (AP) allows a remote system console to be installed, enabling a person at a different geographic location to perform system administration functions on your HP 9000 server. When connected, the access port allows a remote terminal to "take over" as the system console. Most tasks can be performed from the remote console (with the obvious exception of changing tapes). Note that it is possible to have both a local and a remote system console. Both will display the same messages from the HP-UX operating system.

The AP is standard on all E, F, G, H and I systems and on several older Server systems.

The Integrated Business Server systems support two interface cards that allow a remote console (and a local console) to be connected. The Model 807S system supports the SCSI/APMUX card shown on the right. All other 8x7S systems support the SCSI/Parallel/APMUX card shown on the left. A remote and local console can be connected to this card via an ADP.

When the remote console is enabled, it works in parallel with the regular console. Input is echoed to both terminals.

- To enable the remote console:
  1. Type Ctrl+b to enter AP control mode.
  2. In response to the CM> prompt, enter the command ER. This is the "enable remote" command.
  3. The remote console must enter a password.
- To disable the remote console:

```
CM> DR
```

- To get help:

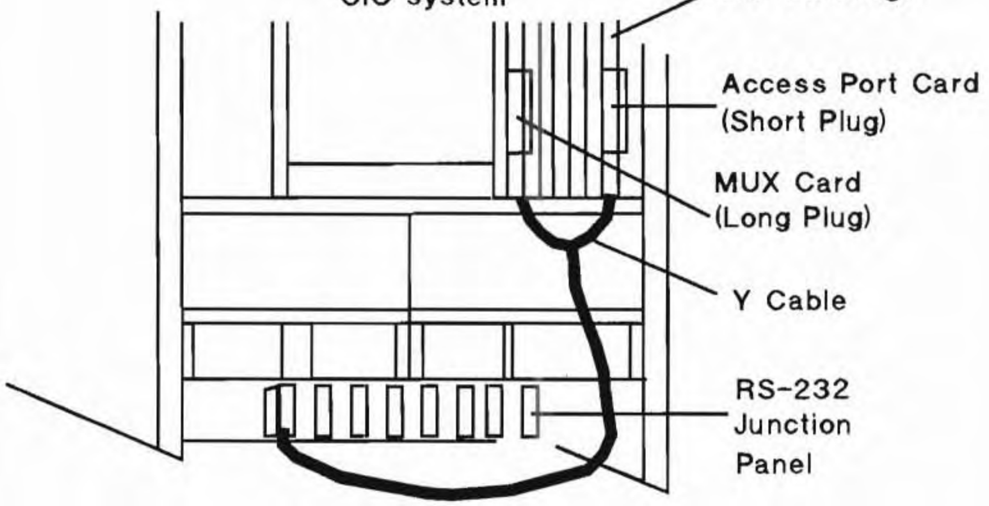
```
CM> HE
```

Commands and explanations are listed below:

CA	allows the data configuration of the AP. It lists the current configuration and requests changes.
CO	switches the session to console mode and is only valid from the local console.
DI	disconnects the modem line cleanly from the modem port.
DR	disables remote modem port by disallowing Ctrl-b from the remote port and turns off the LED on the front panel.
ER	enables remote modem port by allowing Ctrl-b from the remote port and lights the LED on the front panel.
ES	allows the hex display on the front panel to be displayed at the bottom of the console display during console mode.
RS	reset; forces a power ON or hard reset which results in total loss of memory and system reboot. Displays a warning if the system is running.
TA	test AP; invokes the AP self test. You can specify a parameter to initiate a nondestructive (use parameter 0) or destructive (use parameter 1) self test. If you use 1, a warning is issued and the AP state is lost.
TC	transfer of control; performs a soft reset of the system. It transfers control of the operating system to a routine that flushes the memory and dumps the system onto disk for analysis.
TE	tell; allows messages to be passed between the local and remote consoles. It allows messages to be sent without causing the AP command interpreter to generate errors.
SE	connects to session mode from AP command mode. Only valid from the remote modem port.

4-15. SLIDE: Access Port on CIO Systems (Servers only)

Access Port on CIO Systems (Servers only)

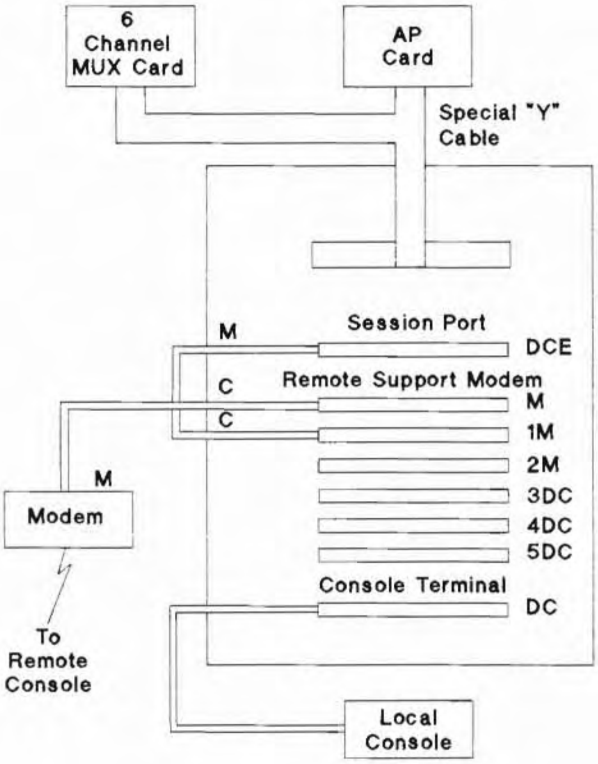


H6296 4-15 48

Student Notes

On CIO systems, the Access Port card resides in a dedicated CIO slot and requires a special RS-232C junction panel (6 port) which is shown on the slide.

The following diagram shows how a remote and local console are connected, via a junction panel, to a CIO system that has an Access Port card installed.



Access Port, 6-channel MUX and Junction Panel.

In the figure above, ports 1M and 2M accept terminals or modems. Ports 3DC, 4DC, and 5DC accept terminals only. Port DC must be hooked to the system console.

4-16. SLIDE: E/ISA Configuration

E/ISA Configuration

ISA	Industry Standard Architecture
EISA	Extended Industry Standard Architecture

EISA board configuration:

- Have cfg files to describe resources and addresses
- Use automatic mode of eisa_config

ISA board configuration:

- Have physical jumpers to describe resources and addresses.
- Use interactive mode of eisa_config

H6296 4-16 49

Student Notes

E/ISA boards cannot be used until they have been configured with the eisa_config program. (ISA stands for Industry Standard Architecture. EISA stands for Extended Industry Standard Architecture. E/ISA stands for both architectures.)

- If you are adding, moving, or removing EISA boards, the automatic mode of eisa_config can usually configure the boards without any user intervention.
- If you are adding, moving, or removing ISA boards, you must run eisa_config in interactive mode. ISA boards do not have readable ID registers and thus cannot be automatically detected by eisa_config.

Each E/ISA card in a backplane can use one or more system resources, such as direct-memory-access channels, interrupt lines, and memory. However, a given card may be able to use only a subset of the choices available for a resource. For example, 11 interrupt lines might be available, but some cards might only be able to use lines 3, 5, and 6. You need a way to tell the card which resources to use. The `eisa_config` program helps you create a conflict-free configuration for E/ISA cards.

ISA Boards

For ISA boards, a set of physical switches or jumpers on the board determines which resources the board will use, to what address range the board will respond, and other board-specific options and modes. The board manufacturer tells you how to set these switches and jumpers. There are few conventions for switch and jumper usage among ISA boards, and it is easy to inadvertently assign the same resource to two different boards by setting a switch incorrectly. Symptoms of such conflicts vary and can be difficult to diagnose.

EISA Boards

EISA boards usually do not have physical switches or jumpers for resource assignment. Instead, each EISA board has a configuration file (`cfg` file) that describes how the board can be used and which resources it needs. The `eisa_config` program then uses these `cfg` files to develop a conflict-free configuration.

All EISA boards have corresponding `cfg` files. ISA boards used in HP-UX workstations must also have corresponding `cfg` files. The `cfg`files are located in the `/sbin/lib/eisa` directory, and have file names in the following format:

```
rXXXnnnn.cfg
```

where:

r	is usually ! or a hex digit
XXX	is a three-letter abbreviation for the board manufacturer's name
nnnn	is four digits. The first three digits represent the product ID and the fourth is the revision level.

How `eisa_config` Works

The `eisa_config` program has two modes, automatic and interactive. In either mode, after a conflict-free board configuration has been built, the configuration is saved in EISA nonvolatile memory (NVM). When `eisa_config` saves a configuration in NVM, init also writes that configuration to the system configuration information (sci) file, `/sbin/lib/eisa/system.sci`, so that the `sci` file and NVM are identical.

Automatic Mode

Executes automatically from `/sbin/bcheckrc` after the root file system has been mounted, but before any other file systems or swap areas are in use.

Interactive Mode

The interactive mode of `eisa_config` allows you to add, remove, or move ISA boards, as well as to view information about E/ISA boards in the configuration and to change currently selected choices for functions.

You must use the interactive mode of `eisa_config` in three cases:

- You need to add, move, or remove an ISA board.
- Automatic mode was unable to generate an error-free configuration.
- You want to change the choice that `eisa_config` automatically selected for a given board function.

4-17. TEXT PAGE: Using eisa_config

The E/ISA board device driver must be part of the kernel before you can run `eisa_config`. Make sure the `eisa` driver and the appropriate peripheral device driver are configured in the kernel.

Using Automatic Mode

HP EISA boards can be configured using SAM. If you are adding a non-HP EISA board, you must use HP-UX commands to load the `cfg` files and drivers shipped with the board, and configure the drivers in the kernel. After the dr ivers have been added to the kernel and the boards are installed,you can reboot your system and `eisa-config` will complete the E/ISA configuration for you.

Using Interactive Mode

`eisa_config` must be run interactively when you need to change the ISA board configuration or when you need to change the choice for a board function.

To start the `eisa_config` program, type:

```
/sbin/eisa_config [ -c cfgfile ] [ -n scifile ]
```

The `-c` option allows you to check the specified `cfg` file for correctness. The `-n` option allows you to initialize the E/ISA configuration from a particular `sci` file instead of from NVM.

`sci` files are useful is your site has several identically configured workstations. You can run `eisa_config` on one workstation to build an optimal configuration, use the save command to save it to a new `sci` file, and copy the file to the other workstations.

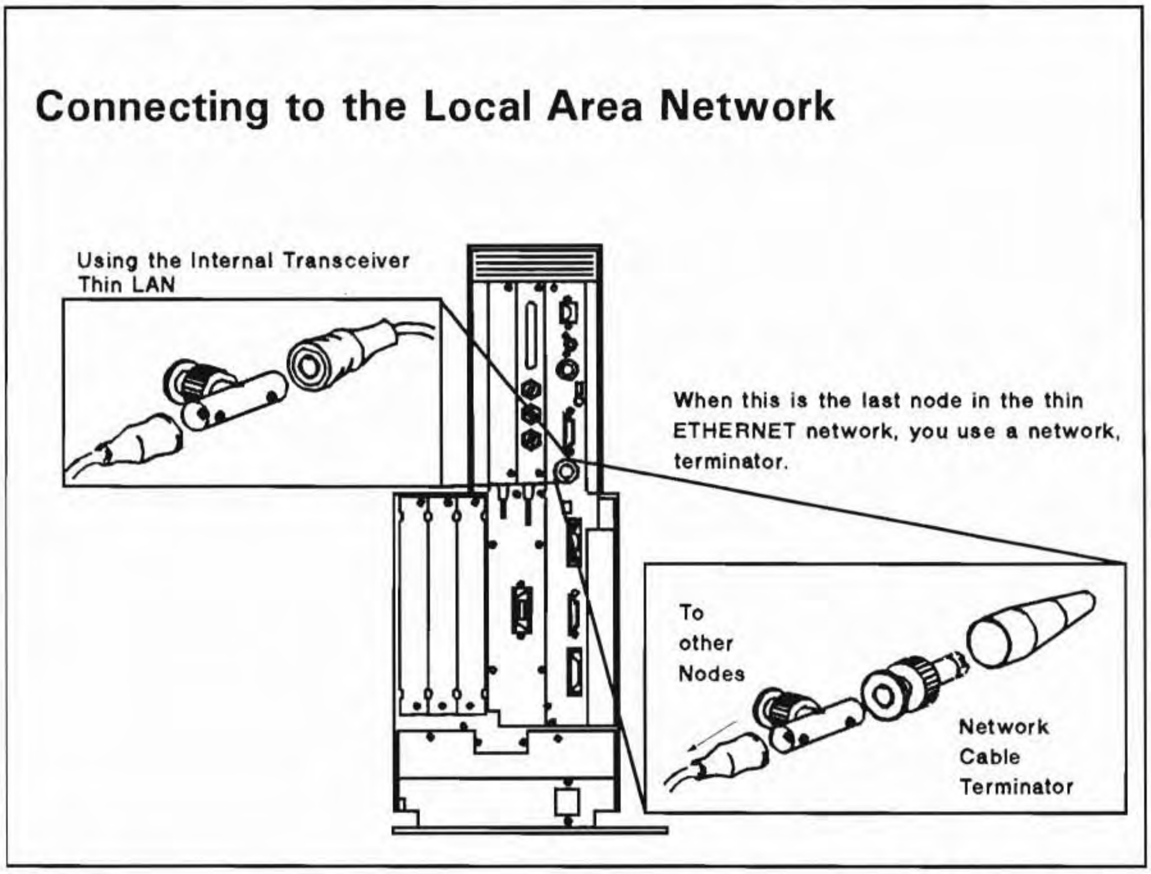
When you start `eisa_config` you will see a display similar to the following :

```
HP-UX E/ISA CONFIGURATION UTILITY

    Type q or quit to leave eisa_config
Type ? or help for help.

Slot      CFG File     Contents

  0     !HWPC010    HP Series 700 EISA System Board
  1     !XYZ1401    XYZ SCSI Controller
  2           ** EMPTY**
  3     !XYZ1702    XYZ Centronics Interface
  4           **EMPTY **
```

Table 4-4. Basic `eisa_config` Commands

Command	Description
`add` *cfgfile slotnum*	Adds the specified board to the specified slot.
`cfgfiles`	Lists the `cfg` files currently in `/sbin/lib/eisa`.
`cfgtypes`	Lists and explains the types of boards that have `cfg` files in `/sbin/lib/eisa`. For example, `NET` is a network board.
`change` *slotnum functionnum choicenum*	Changes the choices used for a given board function.
`comment`	Displays any comments or help supplied by a board manufacturer in the `cfg` file.
`help` or `?`	Lists and explains `eisa_config` commands.
`init [` *filename* `]`	Deletes all changes in this session of `eisa_config` by initializing the configuration.
`move` *curslotnum newslotnum*	Moves a board from one slot to another.
`quit` or `q`	Exits `eisa_config`.
`remove` *slotnum*	Removes a board from the specified slot.
`save [` *filename* `]`	Saves the current configuration.
`show`	Displays working configuration. Shows a list of all slots and whether they are empty or occupied by a particular board.

4-18. SLIDE: Connecting to the Local Area Network



H6296 4-18 50 © 1995 Hewlett-Packard Co.

Student Notes

A LAN connection is used for communication with other systems on a Local Area Network. On the workstation, you may have one of the following two LAN connections:

A 15 pin AUI connection

A BNC to access ThinLAN.

Both LAN connections come standard with the servers.

If your system is ordered with the BNC connection, the Media Access Unit (MAU) or transceiver is integrated onto the LAN interface card, and this will be the default configuration (as shipped).

You may choose to use the 15-pin AUI cable after you have already taken delivery of your hardware. To make this choice, move the internal jumper to select the AUI LAN port. See the installation manual for your SPU to perform this step. Unless you ordered a factory option to change the jumper, your system is shipped with a default setting for use of the internal ThinLAN MAU (if exist).

To connect to the internal transceiver (ThinLAN MAU), connect the BNC cables as indicated in the slide. The terminator may have a slightly different appearance than indicated in the slide. The slide depicts an insulated terminator.

Note

LAN cables require termination at each extreme end of the cable.

Note

Techniques for verifying and troubleshooting the network interface will be discussed in the Local Area Network module.

4-19. SLIDE: Finding the Current Configuration

Finding the Current Configuration

Tools to use:

/usr/sbin/dmesg	To see system configuration by hardware path and device driver.
/usr/sbin/ioscan	To see system configuration by class of module, hardware path, and hardware status.

Syntax:

```
/usr/sbin/ioscan [-d driver|-C class][-f]
/usr/sbin/dmesg
```

H6296 4-19 51

Student Notes

Root users can use the command `/usr/sbin/dmesg` to see what is configured on the system.

During boot-up, the system error message buffer is written with information about the hardware that is found. During loading, the kernel checks this buffer for existing hardware. The `/usr/sbin/dmesg` command reads the contents of this buffer. The buffer is written "FIFO" (first in first out), so it is possible that later messages can overwrite earlier ones.

`/usr/sbin/dmesg` shows system configuration by hardware path and device driver. It also shows what subsystems and which type of file systems are initialized.

Sample dmesg Output

```
    May 19 15:54

Block TLB graphics mapping enabled
    Block TLB entry #8 from 0xf5000000 to 0xf5ffffff allocated.
HPA1991AC19 Bit-Mapped Display (revision 8.02/10) in SGC slot 0
0 (graph3) claimed by wsio (graph3)
2 (bus_adapter) claimed by core (core_adapter)
2/0.1 (c700) claimed by wsio (c700)
2/0.1.2 (target) claimed by wsio (unknown)
2/0.1.2.0 (stape) claimed by wsio (stape)
2/0.1.5 (target) claimed by wsio (unknown)
2/0.1.5.0 (sdisk) claimed by wsio (sdisk)
2/0.1.6 (target) claimed by wsio (unknown)
2/0.1.6.0 (sdisk) claimed by wsio (sdisk)
2/0.2 (lan2) claimed by wsio (lan2)
2/0.3 (hil) claimed by wsio (hil)
2/0.4 (asio0) claimed by wsio (asio0)
2/0.5 (asio0) claimed by wsio (asio0)
2/0.8 (audio) claimed by wsio (audio)
4 (eisa) claimed by eisa (eisa_adapter)
62 (processor) claimed by pa (processor)
63 (memory) claimed by pa (memory)
    System Console is on the Built-In Serial Interface
    End of 2nd Level I/O Configuration
    Networking memory for fragment reassembly is restricted to
2785280 bytes
    Entering lvmconf
    Logical volume 64, 0x1 configured as ROOT
    Logical volume 64, 0x2 configured as SWAP
Swap device table:  (start & size given in 512-byte blocks)
entry 0 - major is 64, minor is
0x2; start = 0, size = 204800
Configuring Dump Devices
    Dump device table:  (start &  size given in 1-Kbyte blocks)
    entry 0 - major is 31, minor is 0x5000; start= 121696, size =
32768
    Starting the STREAMs  daemons.
    Memory Information:
Physical: 32768 Kbytes,
lockable: 20592 Kbytes,
available: 26112 Kbytes
```

Interpreting dmesg output

For each step in the hardware path, there is a corresponding software module. The software module path leads to a device driver that controls the external device. The software module names are indicative of what kind of hardware is at the same place on the corresponding hardware path.

In the example above, for instance, the **dmesg** recognizes a SCSI, a LAN, a HIL, two RS232 and an **Audio** interfaces.

The system also contains an EISA Expander without any cards plugged in.

The `ioscan` Command

The command `/usr/sbin/ioscan` displays devices on your system by the following major characteristics:

- Hardware path
- Device driver class
- Class of device

`/usr/sbin/ioscan` scans system hardware, usable I/O system devices, or kernel I/O system data structures. These are the options for `ioscan`.

-d *driver*	Only list information about devices controlled by the specified driver.
-C *class*	Only list information about devices belonging to the specified c lass.
-H *hw_path*	Restrict the scanning and listing to those devices connected at the specified hardware subset specified.
-f	Produce a full listing.
-n	Add device file names to the output listing.

Sample ioscan -f Output

This sample is from a Model 715.

```
Class        I  H/W Path    Driver       S/W State   H/W Type        Description
===============================================================================
bc           0              root         CLAIMED     BUS_NEXUS
graphics     0  0           graph3       CLAIMED     INTERFACE   Graphics
ba           0  2           bus_adapter  CLAIMED     BUS_NEXUS   Core I/O Adapter
ext_bus      0  2/0.1       c700         CLAIMED     INTERFACE   Built-in SCSI
target       0  2/0.1.2     target       CLAIMED     DEVICE
tape         0  2/0.1.2.0   stape        CLAIMED     DEVICE      HP HP35470A
target       1  2/0.1.5     target       CLAIMED     DEVICE
disk         0  2/0.1.5.0   sdisk        CLAIMED     DEVICE      QUANTUM LPS525S
target       3  2/0.1.6     target       CLAIMED     DEVICE
disk         2  2/0.1.6.0   sdisk        CLAIMED     DEVICE      HP C3010
lan          0  2/0.2       lan2         CLAIMED     INTERFACE   Built-in LAN
hil          0  2/0.3       hil          CLAIMED     INTERFACE   Built-in HIL
tty          0  2/0.4       asio0        CLAIMED     INTERFACE   Built-in RS-232C
tty          1  2/0.5       asio0        CLAIMED     INTERFACE   Built-in RS-232C
unknown     -1  2/0.6                    UNCLAIMED   UNKNOWN     Built-in Centronics Interface
audio        0  2/0.8       audio        CLAIMED     INTERFACE
ba           1  4           eisa         CLAIMED     BUS_NEXUS   EISA Adapter
unknown     -1  4/1                      UNCLAIMED   UNKNOWN     EISA card HWP0C70
processor    0  62          processor    CLAIMED     PROCESSOR   Processor
```

```
memory        0  63          memory       CLAIMED     MEMORY       Memory
```

Sample ioscan -f Output for Servers

```
Class       I  H/W Path  Driver    S/W State H/W Type  Description
===================================================================
bc          0            root      CLAIMED   BUS_NEXUS
ext_bus     0  52        scsi1     CLAIMED   INTERFACE HP 28655A- SCSI Interface
target      0  52.0      target    CLAIMED   DEVICE
tape        0  52.0.0    tape2     CLAIMED   DEVICE    HP      HP35470A
target      1  52.2      target    CLAIMED   DEVICE
disk        0  52.2.0    disc3     CLAIMED   DEVICE    TOSHIBA CD-ROM XM-3301TA
target      2  52.4      target    CLAIMED   DEVICE
disk        1  52.4.0    disc3     CLAIMED   DEVICE    HP      C2474S
target      3  52.5      target    CLAIMED   DEVICE
disk        2  52.5.0    disc3     CLAIMED   DEVICE    HP      C2247M1
target      4  52.6      target    CLAIMED   DEVICE
disk        3  52.6.0    disc3     CLAIMED   DEVICE    HP      C2244M1
lanmux      0  56        lanmux0   CLAIMED   INTERFACE  LAN/Console
tty         0  56.0      mux4      CLAIMED   INTERFACE
lan         0  56.1      lan3      CLAIMED   INTERFACE
lantty      0  56.2      lantty0   CLAIMED   INTERFACE
processor   0  62        processor CLAIMED   PROCESSOR Processor
memory      0  63        memory    CLAIMED   MEMORY    Memory
```

Note

Some devices may show up with a S/W State UNCLAIMED; these devices are right now not usable, although they are on the system. There are no device drivers for these devices in the current kernel. The System Administrator should reconfigure the kernel /stand/vmunix to include these drivers so that the system will be able to use all of its devices.

4-20. SLIDE: Installing I/O Interface Cards

Installing I/O Interface Cards

- Read the card installation guide before you start
- Halt the computer and power it off
- Stand on a static-free mat
- Wear a static strap
- Connect all equipment together
- Keep uninstalled printed circuit boards in their protective antistatic bags
- Handle printed circuit boards by their edges

H6296 4-20 52

Student Notes

Electrostatic charges can damage the integrated circuits on printed circuit boards. To prevent such damage from occurring, observe the following precautions during board unpacking and installation:

- Read the card installation guide before you start. You may also find it helpful to refer to *Configuring HP-UX for Peripherals.*
- Halt the computer by using the `shutdown -h` command, power it off, and unplug all power cords connected to computer.
- Stand on a static-free mat.
- Wear a static strap. This strap is typically attached to your wrist at one end and the chassis of the computer on the other end. This ensures that any accumulated electrostatic charge will be discharged from your body to ground.

- Connect all equipment together, including the static-free mat, static strap, and peripheral units to ensure the same ground reference point.
- Keep uninstalled printed circuit boards in their protective antistatic bags.
- Handle printed circuit boards by their edges, once you have removed them from their protective antistatic bags.

Note If you do not follow these precautions, you may cause serious damage to your system.

4-21. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. What is the function of an I/O interface (or device adapter)?

2. What is an Access Port used for (Servers)?

3. How does a workstation extend I/O connectivity?

4. Use `ioscan` and/or `dmesg` to draw a diagram of your class system. Draw a line diagram similar to those you saw on the slides. Identify all interface cards, peripheral devices, and all addresses.

5. Draw a diagram showing the I/O interfaces and peripheral devices on the system that you administer or use at your own work site.

Module 5 — Device Addresses and Special Files

Objectives

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

- Use the appropriate device file to access a peripheral.
- Describe the autoconfiguration process.
- Describe the function of a device file.
- Differentiate between block and character I/O systems.
- Determine the conventional name for a specific device.
- Describe major and minor numbers.
- Create device files with `mksf` and `insf`.
- Decide when to use the `insf` versus the `mksf` command.
- List device files with `lssf`.
- Remove device files with `rmsf`.

Module 5 — Device Addresses and Special Files

5-1. SLIDE: Introduction To Device Files

Introduction To Device Files

- Device files, or special files, link the kernel with the rest of HP-UX
- Device files are automatically created after reconfiguration.
- Each I/O device has one or more device file(s).
- To communicate with a device you redirect input from, or output to, the device file.
- Device files can be created with `insf`, `mksf`, or with SAM.
- Remove device files with `rmsf`.

H6296 5-1 53

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. Therefore device files do not occupy disk space other than the few bytes used to hold their information (the inode).

Instead, a device file specifies how HP-UX is to communicate with a device. Device files are created with the `insf` or `mksf` commands, or with SAM. Created devices files are stored in the `/dev` directory.

Module 5 — Device Addresses and Special Files

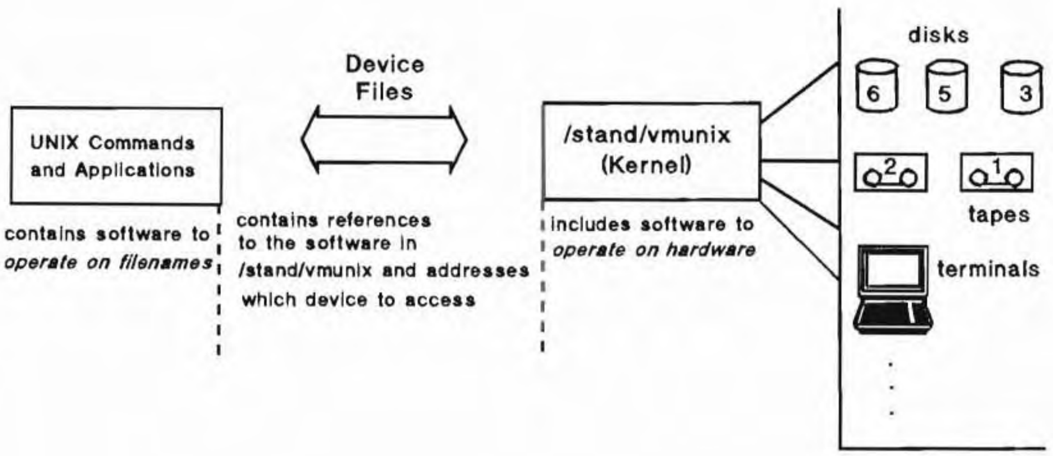
Note

Device file is synonymous with **special file**. You will see these terms used interchangeably.

Since all device files you will need are created for you by the HP-UX system, you need to know which device files to use when you access a peripheral device. Occasionally you will need to create device files. A device file should be removed if you permanently disconnect a peripheral device.

5-2. SLIDE: Autoconfiguration and Device Files

Autoconfiguration and Device Files



H6296 5-2 54

Student Notes

For a peripheral device to work with your system, the following must be true:

- The device must be connected to the computer and turned on.
- The appropriate drivers must be part of the HP-UX kernel.
- The drivers must be connected (bound) in the proper order.
- At least one device file must exist for the device.

This information also needs to be mapped in a way that allows the kernel to associate a device file with the appropriate hardware address and driver. When system hardware is configured, you are informing the operating system what hardware is present. Much configuration is done automatically when you boot the system.

How much you actually have to do depends on whether or not the device is autoconfigurable and whether or not the driver is present in the currently running kernel. If the device is not autoconfigurable or if the driver is not present in the currently running kernel, use SAM or HP-UX commands to add the device.

What Is Autoconfiguration?

At system boot, the kernel performs several system initialization tasks, including probing all hardware installed on the system. During the hardware probe, the kernel identifies all devices—buses, channel adapters, device adapters, and external devices—that can be autoconfigured. The kernel binds (matches) an appropriate driver to each device detected at a specific hardware address. This only happens for autoconfigurable devices.

After completing system initialization tasks, including hardware probing, 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 kernel data structures, `io_tree`. Next, `ioinit` executes `insf`. (Because `ioinit` is executed with the `-i` option in `/etc/inittab`, `insf` is invoked.)

All Hewlett-Packard peripheral devices supported by HP-UX Release 10.0 are automatically configurable. Device files are automatically created during the reboot process for devices or I/O cards.

5-3. SLIDE: Characteristics of Device Files

Characteristics of Device Files

- I/O type (block or character)
- Special Location (dev directory)
- Naming Conventions
- Major numbers point to device drivers in the kernel
- Minor numbers point to device location and other characteristics

H6296 5-3 55

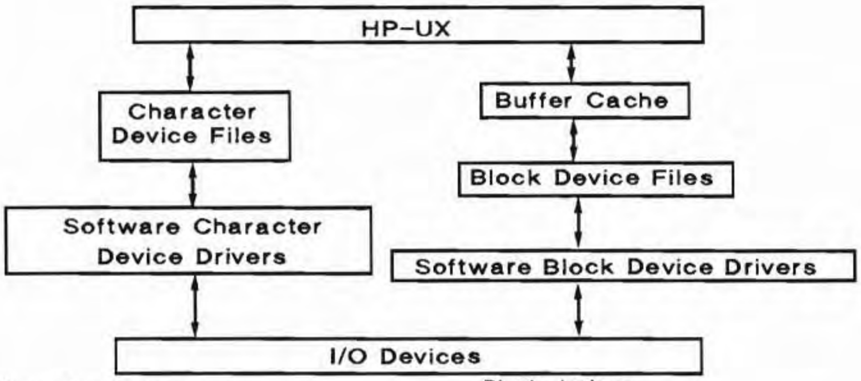
Student Notes

In order to configure devices on a system, or to interpret output from system configuration, you need to understand the basic characteristics of device files.

Device files should be located in the /dev directory. They specify whether I/O is to be handled in block or character mode (this is referred to as I/O type). Device files also have major and minor numbers that indicate device drivers, hardware addresses, and other characteristics.

5-4. SLIDE: Block and Character Devices

Block and Character Devices



Character devices

- I/O in character data streams
- Disks or tapes not holding a mountable file system
- Terminals
- Printers
- Modems
- All other devices

Block devices

- I/O in block units
- Disks or tape drives holding a mountable file system

H6296 5-4 56

Student Notes

I/O devices can be classified as *block* or *character* devices.

Block device files Block device files transfer data using the system buffers. This speeds up I/O transfer. Storage devices can use block device files. These devices include tape drives, hard and floppy disks, and magneto-optical drives.

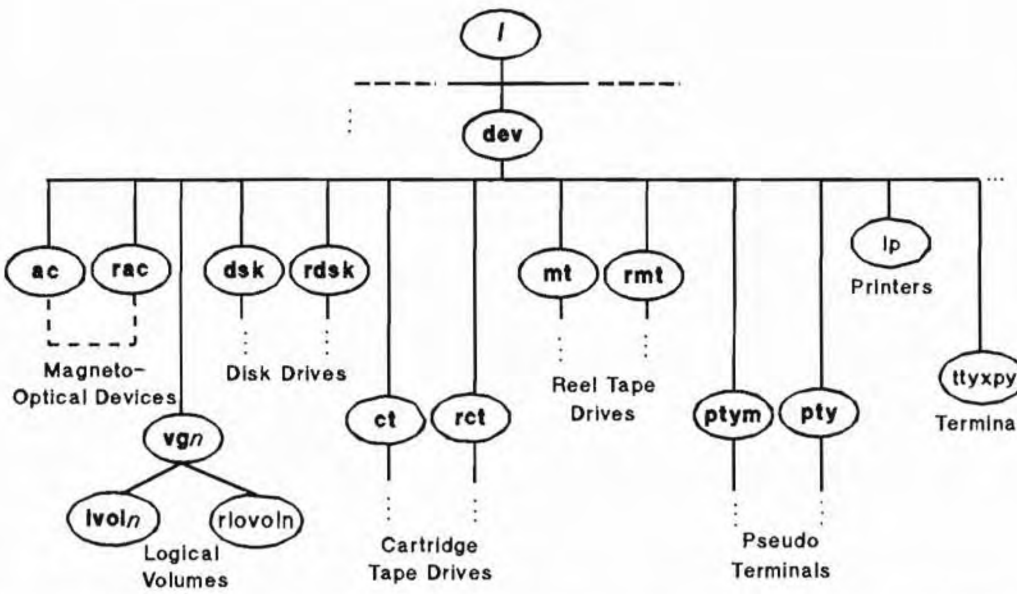
Character device files Character device files transfer data one character at a time. They do not use the system's I/O buffers. The buffering must be controlled by the application program. Devices such as terminals, printers, plotters, as well as storage devices, use character I/O. Character I/O is also called **raw I/O**, and character devices are often referred to as **raw devices**.

Some devices are capable of I/O in both block and character mode. Such devices have two device files: one for block and one for character mode. A tape drive is an example of device which uses both character and block device files. The block device file is used when you want to treat the tape as a file system (usually done in emergency cases only). The character device file is used when you want I/O to occur in a stream. This is the normal mode for creating or restoring tape backups.

In most cases, disks should have both block and character device file entries since disks usually hold mountable file systems. All other devices typically have only character device file entries.

5-5. SLIDE: The Layout of /dev

The Layout of /dev



H6296 5-5 57

Student Notes

By convention, device files are maintained in the directory `/dev` (for devices). Some device files are defined in the `/dev` directory itself while others are grouped in sub-directories under `/dev`. Device files that are defined in sub-directories are grouped by device type (reel tape, cartridge tape, etc.) and by device file class (block or character).

The slide shows examples of some of the device files and sub-directories in `/dev`. These sub-directories are standard and are used to make it easier to locate and name device files.

`/dev/ac`	all the block device files for magneto-optical devices
`/dev/rac`	all the character device files for magneto-optical devices
`/dev/dsk`	all the block device files for disk sections and LVM disks
`/dev/rdsk`	all the character device files for disk sections and LVM disks

/dev/vg*n*	a directory for a volume group.
/dev/vg*n*/lvol*n*	all the block device files for logical volumes in a volume group
/dev/vg*n*/rlvol*n*	all the character device files for logical volumes in a volume group
/dev/ct	all the block device files for cartridge tape drives
/dev/rct	all the character device files for cartridge tape drives
/dev/mt	all the block device files for 1/2-inch reel and DDS tape
/dev/rmt	all the character device files for 1/2-inch reel and DDS tape
/dev/ptym	all the master pseudo terminal device files
/dev/pty	all the slave pseudo terminal device files
/dev	all the terminal, modem, and printer device files

5-6. SLIDE: Device File Naming Convention

Device File Naming Convention

Type of Device	Type of Device File	Naming Conventions
Disk	Character Block	/dev/rdsk/c*X*t*Y*d*Z* /dev/dsk/c*X*t*Y*d*Z*
Magneto Optical Disk	Character Block	/dev/rac/c*X*t*Y*_pF /dev/ac/c*X*t*Y*_pF
Cartridge Tape Drive	Character Block	/dev/rct/c*X*t*Y*d0 /dev/ct/c*X*t*Y*d0
Reel & DDS Tape Drive	Character Block	/dev/rmt/*X*[l|m|h][n][c] /dev/mt/*X*[l|m|h][n][c]
Printer	Character	/dev/lp
Terminal	Character	/dev/tty*X*p*P*

X = Card Instance
Y = Target Number
Z = Drive/Unit Number
p = Platter and Surface
P = Port Number

H6296 5-6 58

Student Notes

When the HP-UX system creates device files for peripherals at system boot time it uses naming conventions for each type of peripheral. You should be familiar with the naming conventions so that you use the appropriate device file name assigned to a peripheral.

When you create a device file, you can arbitrarily choose the device file name to use. However, we recommend that you follow the HP-UX naming conventions. The naming conventions help make it easier to locate and identify a particular device file for a device. We have already seen that all device files are kept in the `/dev` directory, and that some device files are grouped in sub-directories under `/dev` by device type and by device file class (block or character). The following section examines the naming convention used for several devices.

Module 5 — Device Addresses and Special Files

Disk Partitioning

On an HP 9000 server HFS implementation, a disk can be divided into areas (**sections**, or **partitions**) that can accommodate file systems, raw I/O, and swap. Disk partitions, whether traditional disk sections or Logical Volumes, can be addressed like separate disk drives. Each disk partition requires its own device files.

5-7. TEXT PAGE:Device File Naming Conventions

Naming Conventions for Device Files

Disk Naming

Within the `/dev/dsk` and `/dev/rdsk` directories, the following naming conventions are used to differentiate between the devices and disk sections.

Disks can be apportioned into sections or logical volumes to hold

multiple file systems. Disk sections are portions of the disk which can be addressed as though they were separate disk drives.

`/dev/[r]dsk/c`*C*`t`*T*`d`*D*`[s`*S*`]`

C is the *controller*, referencing the controller on the system to which the disk drive is connected. This number will be the same for all disks connected to that controller. Each separate physical disk controller will have a unique controller number.

T is the *target* number. Each disk, for example on a SCSI bus, has its unique target number. It can be modified on the disk device itself.

D is the hardware *device* unit number. This is only important for disk/tape products that have two or more devices with a shared controller (for example, HP 7946, HP 7914CT or disk arrays). For products that are not on a shared controller, the *device* unit number will always be 0. For products that are on a shared controller, the numbers reference the internal number of the device units.

S is the *section number* of the disk. By default, the `insf` command will *not* create special files for all disk sections of a disk. If you don't want to use the logical volume manager you must create the device files for the different disk sections manually with `mksf`.

Examples:

`/dev/dsk/c0t6d0`	*The block special file for disk 6 at controller 0*
`/dev/dsk/c0t6d0s1`	*The block special file for section 1 of disk 6.*
`/dev/rdsk/c1t5d0`	*The character special file for disk 5 on the second disk controller in the system.*

Logical Volume Naming

The Logical Volume Manager allows you to partition disks in a manner similar to, but more flexible than, disk partitioning. Using Logical Volumes, you can combine one or more disks (physical volumes) into a volume group, which is then subdivided into logical volumes.

LVM disks (physical volumes) are named for section zero, the entire disk. Device files automatically created by `insf` during system boot always reference section 0, the entire disk.

Module 5 — Device Addresses and Special Files

Volume Groups

Directories for volume groups are numbered vg00, vg01, ... vg*nn* in the order they are created. These are regular directories, not special files.

Logical Volumes

Device files for logical volumes are listed in the subdirectory for the volume group to which they belong.

/dev/vg*g*/lvol*n*

g	is the number of the volume group.
n	is the number of the logical volume.

Example:

/dev/vg01/lvol4	*The block special file for logical volume 4 of volume group 01*
/dev/vg01/rlvol4	*The character special files for logical volume 4 of volume group 01*

9-Track Reel Tape and DDS /DAT Naming

Tape device files use the same mechanism to select the target as disk naming does. After selecting the tape drive in the name, the options for this device are named. To simplify the use of tape device filenames, the insf command automatically creates more than one device file.

DDS /DAT tape drives do support only two different densities, Compressed or Non-Compressed.

9-Track Reel Tapes can be written with four densities, 800, 1600, 6250 bpi and compressed.

/dev/[r]mt/c*C*t*T*d*D*[options]
/dev/[r]mt/*T*[options]

T	is the target number of the tape drive.
BEST	sets the best known options for this device, including hardware compression on all devices which support compression.
h|m|l	specifies the density at which the tape is to be written or read. The l indicates low density (800 bpi), the m indicates medium density (1600 bpi), and the h indicates high density (6250 bpi).
n	A reel tape device file ending in an n indicates that the tape will not be rewound or repositioned in any way when the device file is closed.
c	indicates data compression.
b	indicates Berkeley style access format when reading.

Examples:

/dev/rmt/c1t0d0BEST	*The character special file for reading or writing a tape at best density.*

/dev/rmt/c1t0d0BESTnb	*The character special file for reading or writing a tape at high density with no rewind on close, Berkeley style.*
/dev/rmt/0mn	*The character special file for reading or writing a tape at medium density with no rewind on close.*

Terminal, Modem, and Printer Naming

Terminal, modem, and printer special files have no sub- directory of their own and are kept in the /dev directory. They follow the naming conventions listed below.

/dev/tty*C*p*P*

/dev/ttyd*C*p*P*

/dev/cul*C*p*P*

/dev/cua*C*p*P*

/dev/lp*C*

C	is the *controller* unit number of the MUX card.
P	is the port number on the MUX card. Ports are numbered starting with 0.

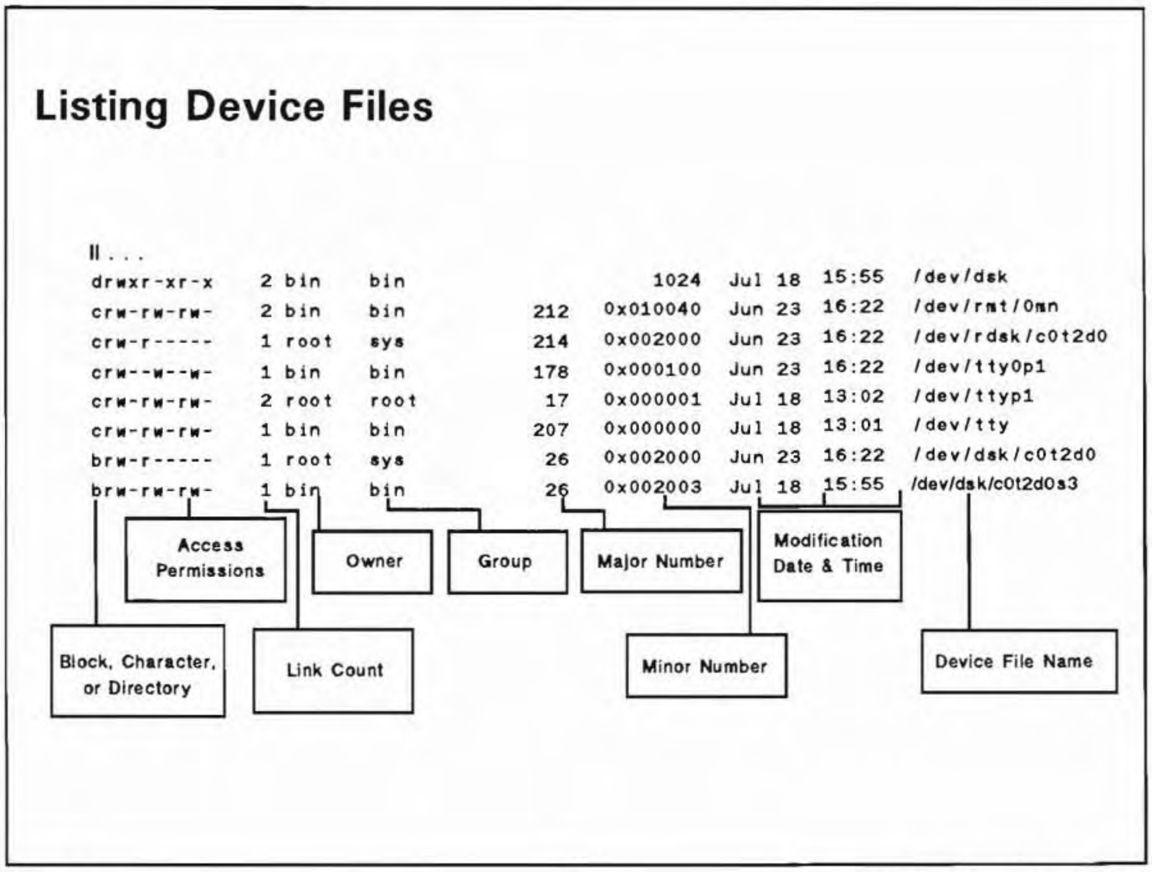
Printers are usually numbered starting at 0 or 1.

Examples:

/dev/tty0p3	*The character special file for a terminal port on the first MUX at port* 3.
/dev/lp0	*The character special file for the first printer (using a special printer card) added to the system.*

Note Do not confuse printers connected to a parallel or serial/MUX line with printers using their own controllers. For example, some electrostatic printer require a HP-PB card to be installed; this card uses its own driver.

5-8. SLIDE: Listing Device Files



H6296 5-8 59

Student Notes

Special files are identified by their name and major and minor numbers. You can look at this information when you do a long listing of the `/dev` directory as shown on the slide.

The first character of each line identifies the type of device file. A `b` denotes a block device whereas a `c` denotes a character device.

The major and minor number appear immediately before the date. For example, the device file `/dev/tty0p1` has a major number of `178` and a minor number of `0x000100`.

Device File Major Number

The **major number** is a pointer to the kernel driver HP-UX is to use when communicating with the peripheral.

The **kernel driver** is a software program that controls I/O for a particular device (or class of devices). The value chosen for the major number is based on both the device driver and on the access method (block or character). For devices needing both a character and block device file, there are different character major numbers and block major numbers.

Device File Minor Number

The **minor number** is a compact representation of the device address and some device-specific information. It typically defines one or both of the following:

- The device's address in terms of information such as physical location and switch settings. This information describes how to "get to" the peripheral.
- Behavioral information. The minor number's content depends on the type of device, for example, for a tape drive it might reflect tape density and other behavioral information.
- 24 bit field contains:
  - Card Instance number (all physical devices)
  - target number (disk)
  - Port number (terminal)
  - HP-IB address (plotter)
  - Device options (most devices)

The interpretation of the minor number is different for different devices.

The table below lists device files that are required by HP-UX.

Table 5-1.

Filename	Recreated by	Use
`syscon, systty, console`	`insf -d cn`	HP-UX uses these files to access the system console.
`diag0`	`insf -d diag`	HP-UX uses this file for diagnostic purposes.
`kmem`	`insf -d mm`	Virtual memory uses this file to access the kernel.
`mem`	`insf -d mm`	Physical memory uses this file to access the kernel
`null`	`insf -d mm`	This is a "bit bucket"—an empty destination for unused output.
`tty`	`insf -d sy`	HP-UX uses this file to access a user terminal

Note

The permissions on the `/dev` directory should be 755. Owner should be `root`. Do not remove any of these files from the HP-UX system.

5-9. SLIDE: Listing Drivers and Classes with `lsdev`

Listing Drivers and Classes with `lsdev`

```
# lsdev
    Character      Block        Driver          Class
        0           -1          cn              pseudo
        1           -1          mux0            tty
       16           -1          ptym            ptym
       17           -1          ptys            ptys
       64           64          lv              lvm
       69           -1          dev_config      pseudo
      119           -1          dlpi            pseudo
      175           27          disc2           disk
      177           28          disc3           disk
      178           -1          mux4            tty
      185           -1          lan3            lan
      188           31          sdisk           disk
      193           -1          mux2            tty
      203           -1          sctl            pseudo
      212           -1          tape0           tape
      214           26          disc1           disk
```

H6296 5-9 60

Student Notes

With no arguments, `lsdev` lists, one set per line, the major device numbers (both character and block), the driver names and the class names of all device drivers configured into the system and available for invocation via special files. A "-1" in either the block or character column means that a major number does not exist for that type.

The command `lsdev` is simply a quick-reference aid.

The syntax of `lsdev` allows you to optionally specify one or more major numbers. Typing `lsdev 7` would produce a list of I/O drivers in the kernel that have either a block or character major number of 7.

5-10. SLIDE: Listing Device Information with lssf

Listing Device Information with lssf

- Lists characteristics of device files
- Uses the device file name and the kernel (/stand/vmunix) to obtain information

Syntax:

/usr/sbin/lssf *path* [*path* ...]

Examples:

```
# lssf /dev/rdsk/c1t6d0
disc3 card instance 1 SCSI target 6 SCSI LUN 0
    section 0 at address 52.6.0 /dev/rdsk/c1t6d0
# lssf /dev/rmt/0mn
tape2 card instance 1 SCSI target 0 SCSI LUN 0 at&t no
rewind best density available at address 52.0.0 /dev/rmt/0mn
```

H6296 5-10 61

Student Notes

Given a device file name, the lssf command will list the characteristics of the device file, such as driver name, card instance number and options specific to that device. Options specific to a device include such things as section number for disks, density for reel tapes and port number for terminals.

The lssf command extracts the device file type (block or character), the major number and minor number from the inode of the device file. It then gets the driver name and hardware address from the kernel.

5-11. SLIDE: Why Create or Modify Device Files?

Why Create or Modify Device Files?

- To restore device files accidentally deleted
- To override standard naming conventions
- To create device files HP-UX system cannot create

H6296 5-11 62

Student Notes

This slide lists the primary reasons you would need to create or modify device files.

In most cases you do not need to create device files. When the HP-UX operating system is first installed, the `insf` command creates device files for all devices found by the system during its hardware probe. Then each time the system is rebooted, `insf` creates device files for any new devices that have been connected to the system. Hence, most device files you use will have a device file automatically created for them at boot time.

In some special cases, such as those listed in the slide, you will need to manually create or modify device files. Two commands, `insf` and `mksf` are commonly used to create or modify device files. The syntax for these two commands is described on the following pages.

5-12. SLIDE: Creating a Device File with `mksf`

Creating a Device File with `mksf`

Syntax:

`/sbin/mksf` [`-d` *driver* | `-C` *class*][`-I` *card instance*][`-H` *HW path*]
[`-D` *directory*][*-other driver opts*][*special file*]

Examples:

1. `# mksf -d tape2 -I 1 -b 6250 -n -r /dev/rmt/1hn`
2. `# mksf -d lpr2 -I 1 /dev/pr600lpm`
3. `# mksf -d mux0 -I 0 -p 5 -c -i`
4. `# mksf -C printer -I 2 /dev/printer`
5. `# mksf -H 56/52.0.0 -r -b 1600 -n`

H6296 5-12 63

Student Notes

The `mksf` command is used to create a device file *if the device is already known to the system.*

The syntax for the `mksf` command is shown on the slide. Options to `mksf` are:

`-d`	selects a particular device by its driver name
`-I`	selects *card instance*
`-C`	matches devices that belong to a given class, like *disks*
`-H`	matches a device at a given Hardware Path
`-D`	override the default device installation directory and install special file in *directory*. Note that *directory* must exist.
other driver opts	depend on the driver name

special file selects an alternate device file name (default: naming conventions). The selected alternative name must be an absolute filename. Relative path names will be used to create files and subdirectories below the /dev directory.

Before creating a customized device file for an existing device, the card instance number can be found using the ioscan command.

Options for each driver vary widely. Options that are meaningful to one device driver are meaningless to another. When using this command, use the man pages for an explanation of options.

Explanation of Examples:

1. `# mksf -d tape2 -I 1 -b 6250 -n -r /dev/rmt/1hn`

   creates a character device file /dev/rmt/1hn for a 6250 bpi mag tape drive with a "norewind" flag on card instance 1.

   ```
   crw-rw-rw-   1 root   other    212 0x000400 Feb 22 14:59  1hn
   ```

2. `# mksf -d lpr2 -I 1 /dev/pr600lpm`

   creates a device file named /dev/pr600lpm for an HP 2564B printer. This printer operating with 600 lpm (lines per minute) is assumed to be connected to card instance 1. The standard naming convention is not used in this example.

   ```
   crw-rw-rw-  1 root  other  26 0x000100 Feb 22 14:07 pr600lpm
   ```

3. `# mksf -d mux0 -I 0 -p 5 -c -i`

   creates a device file for a dial-in terminal with CCITT (European) protocol on port 5 of the first MUX. The device file created will be named /dev/ttyd0p5.

4. `# mksf -C printer -I 2 /dev/printer`

   creates a device file named /dev/printer and maps it to the line printer with card instance #2.

5. `# mksf -H 56/52.0.0 -r -b 1600 -n`

   creates a device file, using the default naming conventions, for the tape device at hardware path 56/52.0.0. The driver specific options specify raw mode, 1600 bits per inch and no rewind on close.

6. (not shown on slide):

   ```
   # mksf -d disc3 -I 4 -u 1 -s 2
   # mksf -d disc3 -I 4 -u 1 -s 2 -r
   ```

   generates the character and block device files for the second, right disk in an HP 9122D micro-floppy disk drive on the default address 4.6.0. The device file names are created according to the naming conventions and the directories /dev/dsk and /dev/rdsk must exist.

   ```
   brw-rw-rw-  1 root    other     0 0x000422 Feb 22 14:16 /dev/dsk/c4d1s2
   crw-rw-rw-  1 root    other     4 0x000422 Feb 22 14:17 /dev/rdsk/c4d1s2
   ```

For further information, refer to `mksf(1M)`.

5-13. SLIDE: Creating Device Files with `insf`

Creating Device Files with `insf`

Syntax:

/sbin/insf [-C *Class*][-H *HW Path*][-d *driver*][-I *card instance*]
[-D *directory*][-k|-e]

Examples:

1. # insf
2. # insf -d disc3 -I 2
3. # insf -d mux2 -I 1 -e
4. # insf -C tty
5. # insf -H 4.2.0

H6296 5-13 64

Student Notes

The `insf` command is used to create a device file *if the device has not been assigned yet*. It creates the device file and also obtains a card instance number for the device.

The syntax for the `insf` command is shown on the slide. Options to `insf` are:

`-d`	selects particular devices by driver name
`-C`	matches devices that belong to a given class, like disks
`-H`	matches a device at a given Hardware Path
`-I`	selects *card instance*
`-k`	assign card instance numbers but do not create device file names
`-e`	create/re-installs device files for existing devices

`-f` force creation of device files

`-D` override the default device installation directory and install special file in *directory*. Note that *directory* must exist.

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 kernel (`/stand/vmunix`). Use the `ioscan` command to list the hardware paths in the kernel.

The `-k` option assigns card instance numbers only and doesn't create device files, which will be discussed in detail on a following slide.

```
# insf -k -d disc1
```

The `insf` command can not create device files for existing devices unless you explicitly say to recreate device files using the `-e` option. This might be required if the device files are accidentally deleted.

```
#insf -e
#insf -e -C printer
```

Device files can be made for all devices on your system. In addition, device files can be made for just one particular device type (driver name) or just an individual device within a device type.

You cannot specify special device options with `insf`. If you have some device that requires special options, you need to use `mksf` after running `insf`.

Explanation of Examples

1. To install all device files needed for the devices on your system, simply issue the following command:

```
# insf
```

2. To add an additional disk with the card instance number 2, issue the following command:

```
# insf -evd disc1 -I 2
```

The following device files and directories will be created with the appropriate permissions and ownerships. Notice the standard naming conventions are used.

```
insf: Installing special files for disc1 instance 0 address 48.2
        making dsk/c0t2d0
        making rdsk/c0t2d0
        making diag/dsk/c0t2d0
        making diag/rdsk/c0t2d0
        making ct/c0t2d0
        making ct/c0t2d1
        making rct/c0t2d0
        making rct/c0t2d1
        making diag/rct/c0t2d0
        making diag/rct/c0t2d1
```

3. If you want to extend your system by adding a second multiplexer interface board (card instance 1), issue the following command:

```
# insf -d mux0 -I 1
```

The following device files and directories will be created:

```
insf: Installing special files for mux0 instance 1 address 44
        making diag/mux1
        making tty1p0 tty1p1 tty1p2 tty1p3 tty1p4 tty1p5
        making diag/ tty1p0 tty1p1 tty1p2 tty1p3 tty1p4 tty1p5
```

4. To create device files for all devices of class `tty`, you would use:

```
insf -C tty
```

5. To create device files for the device at address 4.2.0:

```
insf -H 4.2.0
```

5-14. SLIDE: Using `insf` versus `mksf`

Using `insf` versus `mksf`

Use `insf` to:

- Create multiple device files at one time, using standard naming conventions.

Use `mksf` to:

- Create a single device file that does not use standard naming conventions.
- Create device files that `insf` could not create.

H6296 5-14 65

Student Notes

The `insf` command automatically creates default device files for all "new" devices and also assigns card instance numbers at boot time. You can also manually invoke `insf` to create multiple device files.

The `mksf` command creates only one device file for a device. The following section describes several scenarios that illustrate when you would use `insf` or `mksf`.

Example 1:

Almost all device files are automatically made by `insf` except for some which require additional minor number information. This additional information can only be specified with the `mksf` command's driver options. `mksf` also allows creation of a device file with an arbitrary file name.

For example, modem device files are created with `mksf` command because the `insf` command only creates a "plain" `tty` terminal line. With `mksf`, you can specify a `tty` device file with dialin and dialout line options.

```
# cd /dev
# mksf -d mux2 -I 2 -p 2 -i ttyd2p2
```

This makes a dial-in modem device file (called `ttyd2p2`) for the card instance 2 MUX at port 2.

Example 2:

`insf` may not create all necessary device files. For example, if the second drive of an HP 9122D flexible disk drive with unit number 1 is to be accessed, `mksf` must be used to create the necessary device files: `/dev/dsk/c?t?d1` and `/dev/rdsk/c?t?d1`.

Example 3:

`mksf` and `insf` can be used to restore device files that have been removed accidentally or otherwise. For example, an existing disk with section 0 is re-sectioned into 2 sections a year later. Only the one character and block device file for section 0 of the disk exist.

`# mksf -d disc3 -I 1 -s # [-r]`	*Creates one at a time*
`# insf -d disc3`	*Creates two for each disk (raw and block device file)*
`# insf -e`	*Creates for all devices*

Note

Be careful with `insf -e`! This recreates all device files with the standard naming conventions. If any customized device files (with standard names, but nonstandard options) exist on the system in the standard naming convention, they will be overwritten with the new uncustomized *generic* device files created by `insf`.

5-15. SLIDE: Making Device Files User Friendly

Making Device Files User Friendly

Use the `ln` command to create "User Friendly" device file names.

Syntax:

```
ln [-f] [-i] [-s] file1 new_file
ln [-f] [-i] [-s] file1 [file2...] dest_directory
ln [-f] [-i] -s directory1 [directory2...] dest_directory
```

Examples:

```
# ln  /dev/c0t15_lp  /dev/laser       servers
# ln  /dev/c1t0d0_lp  /dev/deskjet    workstations
# ln  /dev/rmt/3m  /dev/magtape
# ln  /dev/rct/c0t3d0  /dev/carttape
```

H6296 5-15 66

Student Notes

It is recommended that you create device file names that are easy to use, such as `magtape` and `deskjet`. Using the `ln` command allows you to create a second name for the same physical file, so the file can then be referred to by either name. This makes it easier for your users, who may not understand the device file naming conventions.

5-16. SLIDE: Removing Device Files with `rmsf`

Removing Device Files with `rmsf`

`rmsf` removes device files and card instance number(s) assigned to hardware path(s) in the kernel.

Syntax:

rmsf [-a|-k] *devfile*
rmsf [-k] [-d *driver*| -C *class*] -H *hdw_path*

Examples:

`# rmsf tty2p0`	*remove specified device file*
`# rmsf -a dsk/c0t1d0`	*remove device files in /dev*
`# rmsf -H 56/52.3.0`	*remove device files at 56/52.3.0*

H6296 5-16 67

Student Notes

Sometimes more special files are created than you need. You should use `rmsf` to remove the device file names not needed.

When using relative path names such as `dsk/c0t1d0`, the `rmsf` command always looks in `/dev` for the device files that you specify for removal. To remove device files from other locations, always specify the absolute path name (beginning with "/ . . . ").

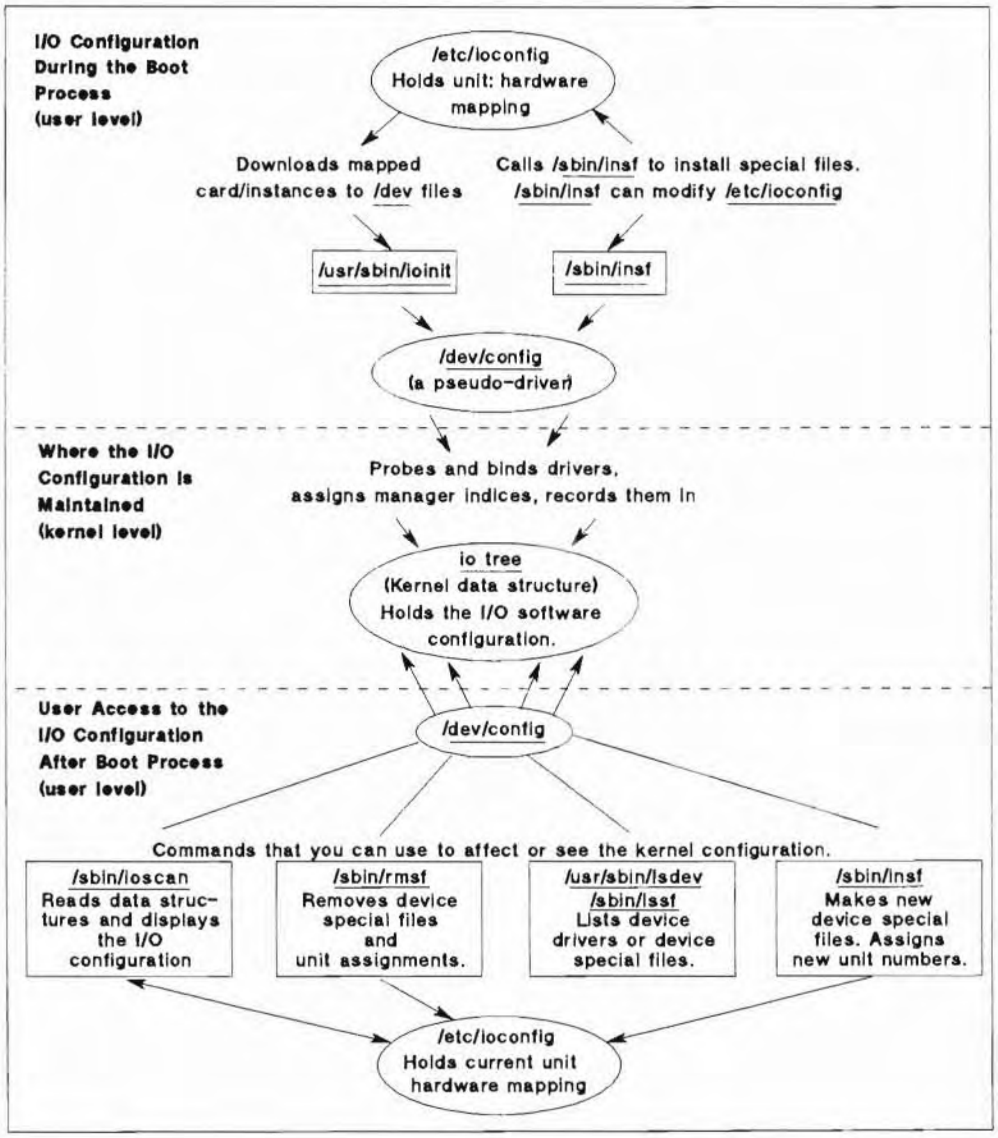
Devices may have been removed from the system without removing their device files and kernel references. The kernel data structure should reflect the real system layout, therefore remove references to nonexistent devices with these commands:

`# ioscan -fk`	*This shows the actual kernel data structure. Compare the output with the output of* ioscan -f. *If they are not equal, enter:*

```
# rmsf -k -H hardware path
```

5-17. TEXT PAGE: Review of Autoconfiguration

The following diagram shows the commands and files involved in the device configuration process.



5-18. LAB: Hands-On Creating Device Files

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. Use `lssf` to determine the options specified for all devices in the `/dev/rmt` directory.

2. Use the `ioscan` command to determine the current configuration of the system.

3. Run the `ioscan -f` command to determine the current system configuration.

4. Run the `lssf` command on all device files found in `/dev/rmt`. The `lssf` command can help you interpret the meaning of device file names.

5. Use `ioscan` to help determine the device file name corresponding to the first tape drive on the system. (If the first tape drive is a 9-track, assume 1600 bpi. Assume no-rewind on close.)

6. Create the device files for a dial-in modem for an available port on your system. (Assume U.S. modem protocol.)

7. Remove all device files for a non-root disk drive. Recreate the device files without rebooting the system.

5-19. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. What is a device file?

2. What is the difference between a block and a character device file?

3. Why do you need both block and character device files for disks?

4. Describe the following device files:

a. `/dev/rdsk/c1t3d0s8`

b. `/dev/lp0`

c. `/dev/mt/3mn`

d. `/dev/tty1p3`

5. What are major and minor numbers?

6. What do the `insf` and `mksf` commands do?

7. What is the difference between the `ll` and the `lssf` commands related to device files?

8. What does the `lsdev` command do? What does the `dmesg` command do?

9. How can the `ln` command be useful in terms of device files?

Module 6 — Booting Your HP-UX System

Objectives

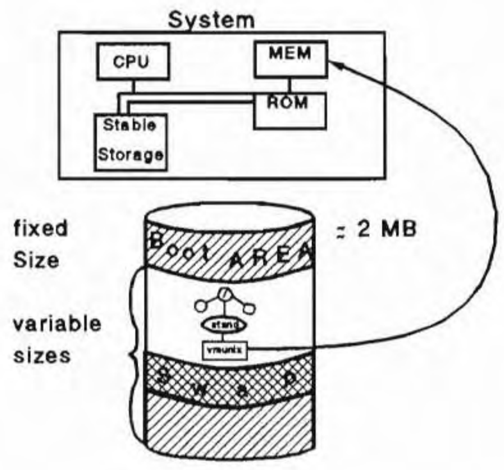
Upon completion of this module, you will be able to:

- Describe the LIF files and utilities found in the boot area.
- Describe the system boot ROM startup sequence.
- Explain the function of the ISL.
- Differentiate between autoboot and manual boot.
- Boot HP-UX in Manual mode.
- Change primary and alternate boot paths.

Module 6 — Booting Your HP-UX System

6-1. SLIDE: How Can We Load the Kernel?

How Can We Load the Kernel?



H6296 6-1 68

Student Notes

System startup begins with either a Reset (RS), Transfer of Control (TC), or cycling the power, and has two main phases:

1. The boot ROM startup sequence
2. The HP-UX startup sequence

When the system is powered up, the boot program initializes and tests hardware to bring the system to a state that is usable by the operating system.

On a Server, if the system is already powered on, then a boot sequence can be initiated by executing either a TC or an RS at the `CM>` prompt, or at the `SP>` prompt on a Model T500 or 890. *Note that this feature is only supported on the HP 9000 servers.*

Module 6 — Booting Your HP-UX System

The CM> prompt is obtained at the console by typing Control+B. On a Model T500 or 890, the SP> prompt is obtained by typing SP at the CM> prompt.

Table 6-1. Executing a Transfer of Control or Hard Reset

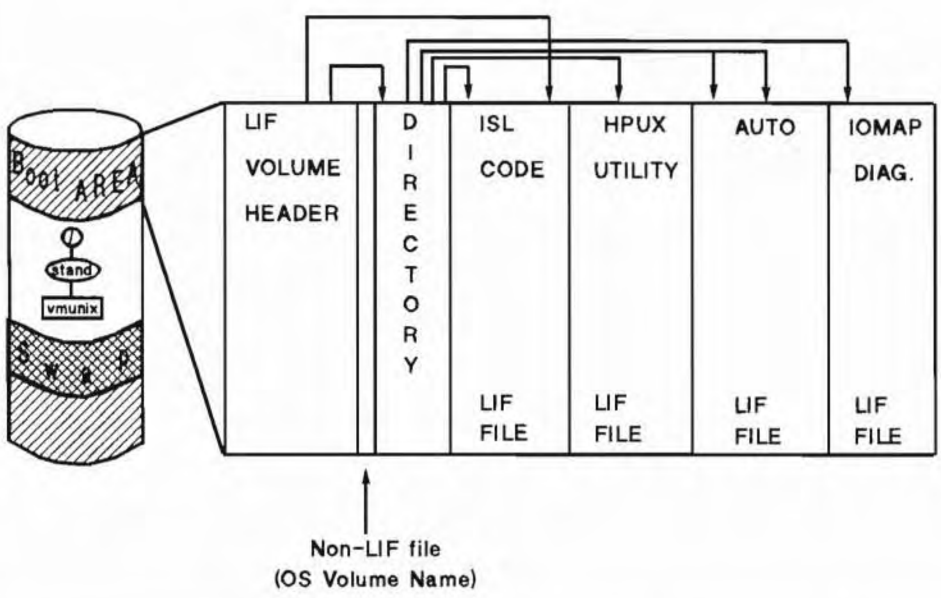
	Models T500 and 890	**Other PA-RISC Systems**
Step ONE: At the system console	Enter Control+B	Enter Control+B
Step TWO: At the AP Command Mode (CM) prompt	Enter SP mode: CM > SP	Enter the desired command: CM > TC (or RS)
Step THREE: At the Service Processor (SP) prompt	Enter the desired command: SP> TC (or RS)	

The difference between a TC and an RS is that the TC command preserves the data that is in the memory. This data is potentially useful for troubleshooting in situations where there has been a problem. A TC is also known as a "soft reset." The RS command clears the memory, removing potentially useful troubleshooting data. In routine situations, a TC is adequate for restarting the system. When updating the operating system, or when recovering from system interruptions, an RS may be necessary.

After finding and checking the hardware, the boot program searches for a copy of the operating system (/stand/vmunix). It searches a list of potential sources, and from the first available source it finds, loads the **Initial System Loader**(ISL)> into memory. The secondary loader then loads /stand/vmunix into memory, and starts it up to enable you to use your system.

The boot program contains code to work with a specific hardware architecture. It finds the ISL on the boot media (typically in the LIF volume of a mass storage media), loads it into memory, and starts it running. The ISL is written in LIF (Logical Interchange Format) and allows it to deal with changes to the booting process from one operating-system release to the next.

6-2. SLIDE: Boot Area on Root Disk



H6296 6-2 69

Student Notes

The media on which the bootable system resides has a boot area and a root partition. It may also contain a swap area, and sometimes other file systems. The boot area contains the bootstrap program and other files needed for bringing up the system.

The boot area contains:

- a LIF volume header that identifies the volume format as LIF (Logical Interchange Format)
- a non-LIF file that contains the HP-UX operating system volume name
- a LIF volume directory that identifies the LIF files in the boot area
- several utilities including ISL HPUX, RDB, ODE and
- a LIF file called AUTO

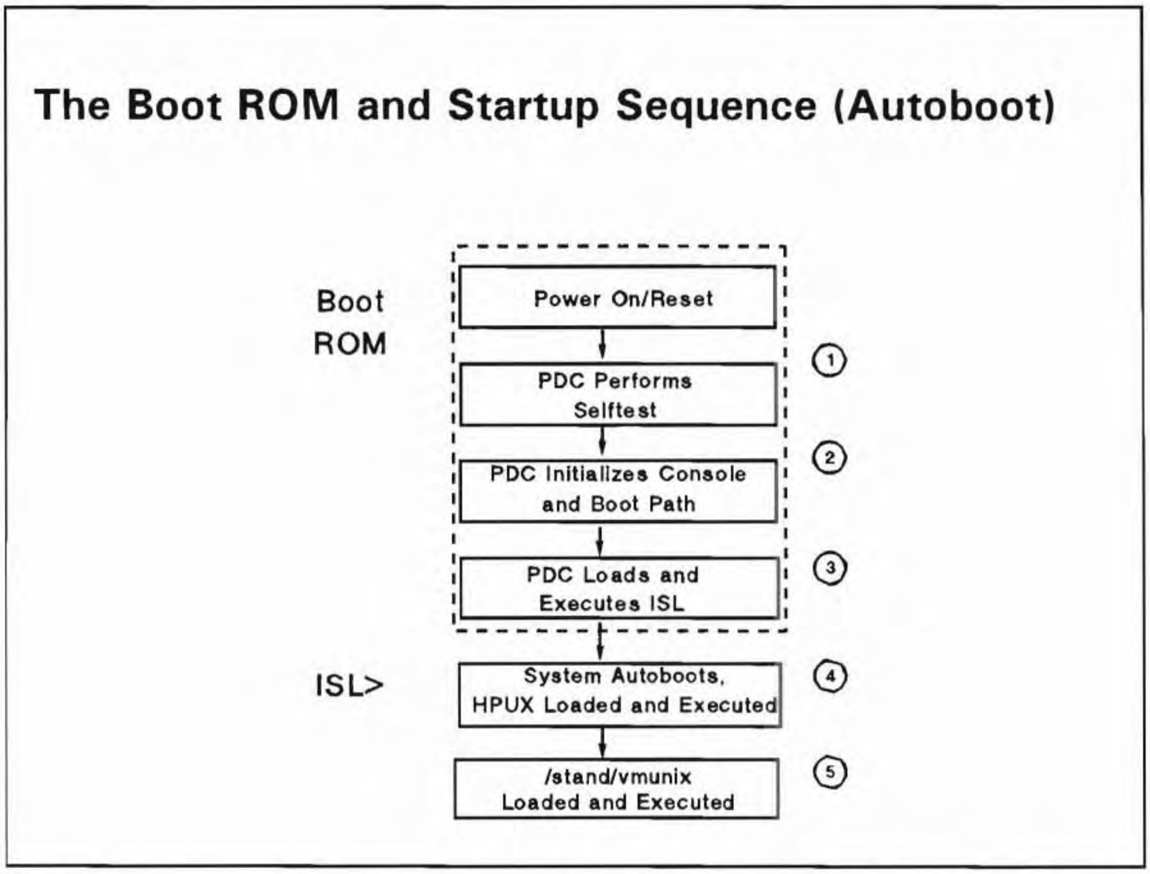
Module 6 — Booting Your HP-UX System

At boot time the ISL code is loaded into memory and begins executing. The following section describes how the LIF files and utilities are utilized during system boot:

- ISL accesses the AUTO file and parses the run-string found there. The AUTO file contains a run string, the first field of which is the HPUX utility. The HPUX utility gets loaded into memory. ISL passes HPUX the remaining arguments in the AUTO file's run string.
- HPUX uses the rest of the run-string arguments to locate the kernel (`/stand/vmunix`) on the root disk and load the kernel into memory. HPUX then passes control over to the kernel.

The hpux command in the AUTO file controls the Autoboot sequence on HP-UX systems. This command specifies the location of the HP-UX kernel to boot from.

6-3. SLIDE: The Boot ROM and Startup Sequence (Autoboot)



H6296 6-3 70

Student Notes

Whenever the computer is powered on:

- The hardware is tested
- The kernel is loaded into memory
- The kernel probes the hardware to determine the hardware addresses and the appropriate drivers to match to each device
- Various system resources are initialized.

This sequence of events is known as system **boot**.

The Boot ROM initializes the primary boot path, loads ISL, and allows you to select either the manual or autoboot mode.

Module 6 — Booting Your HP-UX System

The system boot procedure of an HP 9000 computer is controlled by a program residing in ROM (read-only memory). This program is called Processor Dependent Code (PDC). Each computer model has a different PDC written for it. When the system starts up, it initially follows the instructions in PDC. The PDC tests the hardware, copies the console path into memory and finds and initializes the system console, and then copies the boot path and autoboot flag into memory and initializes the boot path. If the Autoboot flag is set, you will get the following message depending upon the system type:

Servers

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

Workstations

```
Selecting a system to boot.
To stop selection process, press and hold the ESCAPE key.
```

If the autoboot is interrupted as indicated above, a manual boot is performed. The following slides will cover the process of manually booting the systems.

The hardware path the system chooses to boot from is set in Stable Storage (a non-volatile are of memory).

The following table shows the hardware paths in Stable Storage:

Table 6-2.

	E Series	G H or I Series	T500 890	Workstations
Primary boot path (system disk)	56/52.6.0	52.6.0	0/52.6.0	2.0.1.6.0
Alternate boot path (tape or CD-ROM drive)	56/52.0.0	52.0.0	0/52.0.0	2.0.1.3.0
Console path	56/56.0	56.0	0/40.0	graphics monitor

The hardware paths listed in this table were set at the factory, but can be changed by the administrator.

Note

The address of the system disk for the T500 may be different depending upon which type of interface was chosen.

Unlike servers, workstations do not come with a DDS tape drive standard.

6-4. SLIDE: Autoboot Versus Manual Boot

Autoboot Versus Manual Boot

Autoboot:

- Enabled by command: `ISL> autoboot on`.
- Selected if you let the 10-second override period expire.
- Chooses the boot path from Stable Storage.
- No manual intervention is required.
- Should be the normal mode of operation.

Manual boot:

- Entered by pressing appropriate key during 10-second override period.
- Select the boot path to use.
- Type `hpux` command at `ISL>` prompt.

H6296 6-4 71

Student Notes

You enable autoboot by typing:

```
ISL> autoboot on
```

If you do not want to automatically boot during the boot process, you could disable this flag with `autoboot off`. If you do this, then every time you boot the system it will require manual intervention. It is not really necessary to disable autoboot since you can interrupt the autoboot sequence by pressing any key within the 10-second override period.

Your normal mode of operation should be autoboot. You would manually boot your system if you are booting from a backup kernel, or from a kernel that resides on a disk that is not in your primary or alternate boot path. You also need to manually boot your system the first time you install the HP-UX operating system.

6-5. SLIDE: Manually Boot HP-UX Servers

Manually Boot HP-UX Servers

```
Boot from primary boot path (Y or N?)> n
Boot from alternate boot path (Y or N?)> n
Enter boot path > 52.6.0

Booting.

Console IO Dependent Code (IODC) revision 4
Boot  IO Dependent Code (IODC) revision 4

Interact with IPL (Y or N?)> y

Hard Booted.

ISL Revision A.00.12 February 11, 1991

ISL> hpux (;0)/stand/vmunix
Boot
```

H6296 6-5 72

Student Notes

This slide shows the hpux command you would execute to manually boot HP-UX. To boot manually, you must interrupt the autoboot sequence by pressing any key before the 10 second override period expires.

Depending upon the type of system, the dialogue may look different from what is seen on the slide.

For example, on an E series, the administrator will see the following output at the console:

```
MAIN MENU
-----------------------------------------------------------
Command                                    Description
-------                                    -----------
BOot [PRI|ALT]|[path]                      Boot from specific path
PAth [PRI|ALT][path]                       Display or modify a path
```

```
SEArch [DIsplay|IPL][IPL][path]           Search for boot devices
COnfiguration menu                        Displays or sets boot
value
INformation menu                          Displays hardware
information
SERvice menu                              Displays service
commands
DIsplay                                   Redisplays current menu
HElp [menu|command]                       Display help for menu or
command
RESET                                     Restarts the system
-----------
Main Menu: Enter Command or Menu :
```

The administrator will always be prompted whether or not to interact with IPL. This will get the ISL> prompt.

Ordinarily you would use autoboot to boot from a kernel file that resides on the root disk (typically the assigned primary path). A manual boot is sometimes necessary when you boot from a backup kernel, or when you boot from a kernel that resides on a disk other than the root disk and is not specified in the AUTO file.

The syntax of the hpux boot command is:

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

where the variables are:

disc# — disc1, disc2, or disc3 depending on the type of manager required for the disk that contains root.

path — the disk drive from which to boot. Has the following format:

```
bus_converter/hp-pb_slot.bus_address

   OR

hp-pb_slot.bus_address
```

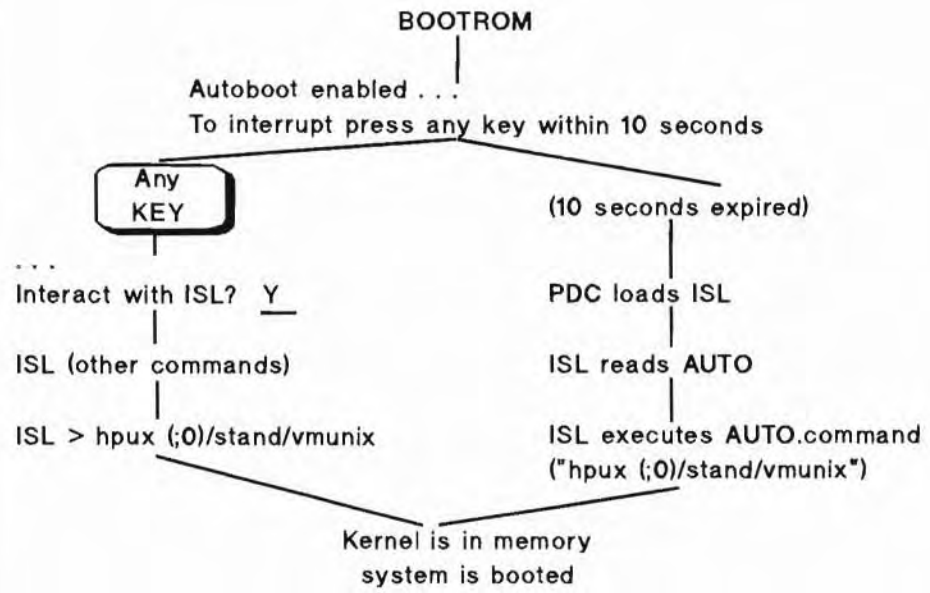
section — the number of the section containing the root partition. If booting from a root logical volume, the section number is always 0.

kernel — the name of the kernel you want to boot (the default is /stand/vmunix but you may need to boot a backup kernel or some other customized kernel).

A backup kernel is typically created before you reconfigure a new kernel. It allows you to boot from the old kernel if you have problems with the new kernel.

6-6. SLIDE: Summary of the Boot Process (Servers)

Summary of the Boot Process (Servers)



H6296 6-6 73

Student Notes

To summarize, the boot process is how the kernel gets loaded from the boot media into the system memory.

When the system is powered up, the boot ROM initializes and tests the hardware, and presents a prompt that gives you ten seconds in which to interrupt the boot sequence and interact with it by entering Attended Mode.

If you interrupt the boot process, you can boot to ISL. From ISL, you specify a kernel to boot.

If you do not interrupt the boot process, the system automatically boots to ISL and boots the default kernel.

6-7. SLIDE: Accessing the Boot Console User Interface Models 712, 715/64, 715/80, 715/100

Accessing the Boot Console User Interface Models 712, 715/64, 715/80, 715/100

Interrupt the Boot Sequence

1. Turn on the power to your workstation.
2. When the boot screen appears, press and hold the (ESC) key.
3. Release (ESC) when the screen displays the boot administration command menu followed by the BOOT_ADMIN> prompt.

H6296 6-7 74

Student Notes

To interrupt the boot sequence and use the Boot Console User Interface on these models:

1. Turn on the power to your workstation.
2. When the message to interrupt the boot process appears, press and hold the (ESC) key.
3. Release (ESC) when the screen displays the boot administration which is illustrated on the slide

Searching for Bootable Devices

To search for bootable devices that have a LIF volume and initial program loader (IPL), type:

```
BOOT_ADMIN> search
```

The system will then display a list of devices containing bootable media. The list may include SCSI devices (disks, CD-ROMs, and tape drives) and/or LAN devices (HP 9000 computers configured to boot your system over the LAN).

To halt the search, press (ESC).

Booting from a Selected Device

Usually, you turn on your system and wait for HP-UX to boot automatically. However, you may wish to boot your system from a different operating system stored on another device.

If your search for bootable media identifies more than one bootable device, you can boot your system from any given device by typing:

BOOT_ADMIN> boot *device*

where *device* is the hardware path to the selected device. The hardware path is specified in this form:

scsi.*scsi_address*.*logical_unit_number*

For example, to boot an operating system from a DDS-format tape drive located at scsi.1.0, type:

BOOT_ADMIN> boot scsi.1.0

The primary or alternate paths can also be abbreviated with keywords. For example, to boot from the primary path simply type:

BOOT_ADMIN> boot pri

Boot Administration

From the BOOT_ADMIN> prompt, you can alter default behaviors exhibited at boot-up, or obtain useful information about the hardware as configured.

6-8. SLIDE: Accessing the Boot Console User Interface Models 715/33, 715/50, 725, 735, 755

Accessing Boot Console User Interface

Models 715/33, 715/50, 725, 735, 755

```
Selecting a system to boot.
To stop selection process, press and hold the ESCAPE key.

Selection process stopped.
Searching for Potential Boot Devices.
To terminate search, press and hold the ESCAPE key.
Device Selection     Device Path              Device Type
-------------------------------------------------------------

P0                   scsi.6.0                 QUANTUM PD210S
P1                   scsi.5.0                 QUANTUM PD210S

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 ipl
```

H6296 6-8 75

Student Notes

This is how the attended mode boot looks on Models 715/33, 715/50, 725, 735, 755 .

Each time the machine is powered on, you have the opportunity to interact with it by entering Attended Mode. For example, you might need to interrupt the boot sequence to:

- Redirect the boot sequence
- Perform a boot administration function provided by the Boot Console User Interface.

If you wish to interrupt the boot sequence, press and hold the (ESC) key until the message,

```
Terminating selection process.
```

is displayed.

The automatic boot sequence has now been halted and the system is in fully "attended" or interactive mode.

At this point, the Boot Console User Interface is invoked and the first message displayed is the following:

```
Searching for potential Boot devices.
```

The system searches the SCSI, LAN, and EISA interfaces for all potential boot devices.

Redirect the Boot Sequence

If you wish to boot from a device other than the one specified, you choose b. You can boot either automatically or interactively. You can type the arguments to the b(oot) command on the command line.

For example to boot from the disk at bus address 6, you could type any of the following:

```
Select from menu:  b p0

Select from menu:  b p0 isl

Select from menu:  b p0 ipl
```

The first example invokes an automatic boot from the specified device. The second and third examples are equivalent. They invoke an interactive boot.

Searching for Bootable Devices

If you choose s, the system will search a list of potential bootable devices for only those that have a LIF volume and initial program loader (IPL). The system will then display a table listing only those devices from which you can boot.

Boot Administration

If you choose a, you will enter Boot Administration mode, from which you can alter default behaviors exhibited at boot-up, or obtain useful information about the hardware as configured.

6-9. SLIDE: Interactive Boot Using ISL

Interactive Boot Using ISL

From the BOOT_ADMIN> **prompt, type:**

```
BOOT_ADMIN> boot scsi.6.0 isl
Trying scsi.6.0
Boot path initialized.
Attempting to load IPL.
Hard booted.

ISL Revision A.00.18
ISL>
```

Examples:

```
ISL>  hpux boot disk(scsi.6;0)/stand/vmunix
ISL>  hpux -is boot disk(scsi.6;0)/stand/vmunix
```

H6296 6-9 76

Student Notes

If you wish to interact with ISL, you must boot to ISL from the Boot Console User Interface, which allows you to perform an interactive boot.

ISL> is the interactive boot prompt. To access this interface ask for isl in your boot request. For example:

```
BOOT_ADMIN> boot scsi.6.0 isl
```

From ISL you can invoke the hpux utility for bootstrap and initial system installation.

hpux *[-istring]* boot *[devicefile] [arguments]*

The hpux command supports the following operations:

`hpux boot`	Loads an object file from an HP-UX file system, LIF, or raw device, then transfers control to the loaded image. This is typically `/stand/vmunix`.
`hpux restore`	Used for system installation and recovery.
`hpux ls`	Lists the contents of HP-UX directories, in a format similar to `ls`.

These operations accept *devicefile* specifications, which have the following format:

manager(path;n)filename

where

manager	The class of device manager to be used. Valid *managers* are `disk`, `tape`, and `lan` managers. `disk` manages all disks. `lan` manages remote boot through the LAN connection. `tape` manages the DAT tape drive.
path	The physical hardware path to the device. The path can be specified explicitly, as in `2/0/1.1`, or mnemonically, as in `scsi.1`. For the core I/O board, valid managers are `scsi`, `EISA` and `lan`.
n	The minor number. Currently all minor numbers are 0. They are reserved for future enhancements.
filename	Specified file names can be standard HP-UX path names. This is typically `/stand/vmunix`.

Boot

The `boot` command is the default command for `hpux`. Therefore the commands `hpux` and `hpux disk(;0)/stand/vmunix` are equivalent when booting from disk. However, if the boot command is specified, it must be given a device file, as in `hpux boot disk(scsi6;0)/stand/vmunix`.

The `boot` command also supports an HP-UX argument parameter. `-i`*string* accepts a string that is passed to `init`. `init` uses this string to determine the run-level of the machine if the default run-level is not used. The run-level will determine what software will be run at boot up. Typical run-levels will allow multi-user or single user access to the system. The default run-level is set to multi-user or run-level 4 in the `/etc/inittab` file.

For example, to boot to single-user mode, the command would be:

```
ISL>hpux -is boot disk(scsi.6;0)/stand/vmunix
```

6-10. SLIDE: Boot Console User Interface: BOOT_ADMIN

Boot Console User Interface: BOOT_ADMIN

```
BOOT_ADMIN> info

----------------------------- Hardware Configuration -------
Machine model: 9000/720

Processor Frequency          = 50000000 Hz
I/O Subsystem Frequency      = 25000000 Hz

SCSI Jumper Frequency setting = 50000000 Hz

LAN Jumper Status: Internal ThinLAN Port selected

Processor Revision                     = 2
System Controller Revision             = 1
Floating Point Coprocessor Revision = 3


Hardware Version               8192      (0x00002000)
Software Version               1153      (0x00000481)
```

H6296 6-10 77

Student Notes

By pressing (ESC) to interrupt normal boot, we enter the Boot Console User Interface main menu. It provides access to the BOOT_ADMIN prompt. (On some systems, you will automatically enter BOOT_ADMIN as soon as (ESC) is pressed). This interface will allow you to alter default behaviors exhibited at boot up and also obtain useful information about the hardware as configured.

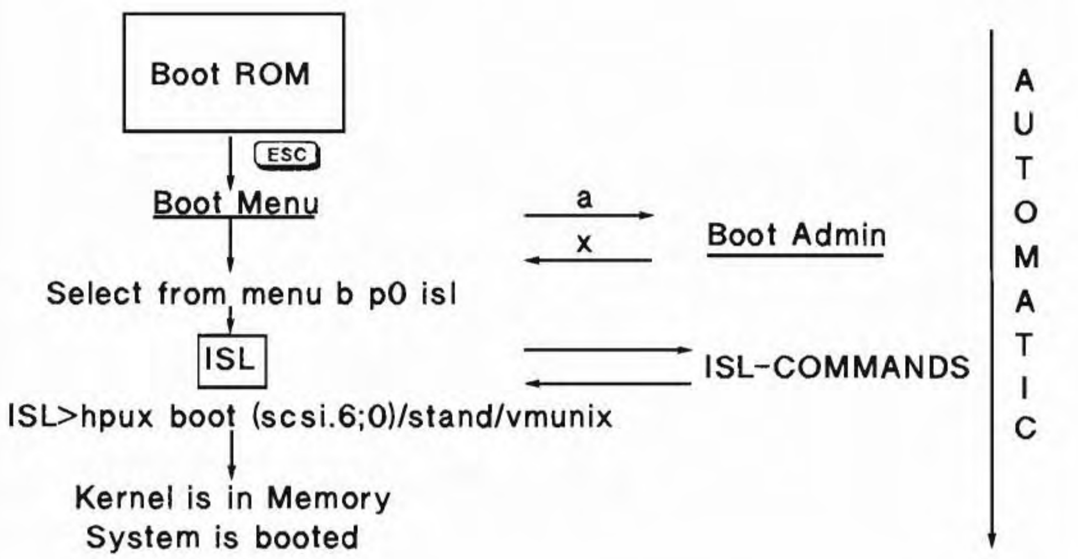
The commands in Table 6-3 may be executed from the BOOT_ADMIN> prompt:

Table 6-3. Boot Administration Commands

Command	Action	Available on Models
auto	Display state of Autoboot/Autosearch flags	all
autosearch	Set state of Autosearch flag	all
autoboot	Set state of Autoboot flag	all
boot	Boot from primary/alternate path or specified device	all
date	Read/Set the Real-Time Clock	715/33, 715/50, 725, 735, 755
diagnostic	Show (on)/Conceal (off) diagnostic messages during boot	712, 715/64, 715/80, 715/100
exit	Return to previous menu	715/33, 715/50, 725, 735, 755
fastboot	Display/Set quick (on) or thorough (off) testing of memory and I/ O interface	712, 715/64, 715/80, 715/100
fastsize	Display/Set FASTSIZE memory parameter	715/33, 715/50, 725, 735, 755
help *item*	Display Help information for *item*	715/33, 715/50, 725, 735, 755
info	Display boot/revision information	715/33, 715/50, 725, 735, 755
information	Display boot/revision information	712, 715/64, 715/80, 715/100
lan_addr	Display LAN station address	715/33, 715/50, 725, 735, 755
lanaddress	Display LAN station address	712, 715/64, 715/80, 715/100
monitor	Display/Set monitor type	712, 715/64, 715/80, 715/100
os	Display/Select operating system	715/33, 715/50, 725, 735, 755
path	Display/Modify path information	all
pim *fault_type*	Display Processor Internal Memory information on a particular type of fau lt (hpmc, toc, lpmc)	712, 715/64, 715/80, 715/100
pim_info	Display Processor Internal Memory information	715/33, 715/50, 725, 735, 755
reset	Reset the system	all
search	Search for boot device	all
secure	Display/set secure boot mode	all
show	Display the results of the previous search	715/33, 715/50, 725, 735, 755

6-11. SLIDE: Summary of the Boot Process (Workstations)

Summary of the Boot Process (Workstations)



H6296 6-11 78

Student Notes

To summarize, the boot process is how the kernel gets loaded from the boot media into the system memory.

When the system is powered up, the boot ROM initializes and tests the hardware, and presents you with the opportunity to interrupt the boot sequence and interact with the Boot Console User Interface.

If you interrupt the boot process, you can boot to ISL. From ISL, you specify a kernel to boot.

If you do not interrupt the boot process, the system automatically boots to ISL and boots the default kernel.

6-12. SLIDE: ISL Commands

ISL Commands

```
ISL> primpath
Enter primary boot path: 52.6.0 (Server)
Enter primary boot path: 2/0/1.6 (Workstation)

ISL> altpath
Enter alternate boot path: 52.5.0 (Server)
Enter alternate boot path: 2/0/1.5 (Workstation)

ISL> conspath
Enter system console path: 56.0 (Server)

ISL> autoboot
Enter ON or OFF: on

ISL> display

ISL> hpux disc3(52.6.0;0)/stand/vmunix (Server)
ISL> hpux boot disk(scsi.6;0)/stand/vmunix.backup(Workstation)
```

H6296 6-12 79

Student Notes

When you manually boot the system you will need to interact with the ISL. (An example of an instance when you would perform a manual boot is the installation of the HP-UX operating system.) The ISL allows you to perform tasks such as loading the HP-UX kernel into memory, changing the primary or alternate boot paths, changing the console path, and turning autoboot on or off.

The follow list describes several commonly used ISL commands:

`? or help`	Help—a list of commands and available utilities.
`listf` or `ls`	List available utilities.
`autoboot`	Enable or disable the autoboot sequence.
`autosearch`	Enable or disable the autosearch sequence.
`primpath`	Modify the Primary Boot Path.

`altpath`	Modify the Alternate Boot Path.
`conspath`	Modify the Console Path.
`lsautofl`	List contents of the autoexecute file.
`display`	Display the Primary Boot, Alternate Boot, and Console paths.
`secure`	Show or set secure status.
`hpux`	This is the `isl` utility for bootstrap and first-time installation. It has many operations, such as `boot`, and `install`, The `install` operation accepts several options, such as `-i`*string*, which allows you to specify the initial run-level for `init`. For more information, see `hpux_800 (1M)` or `hpux_700 (1M)`.

Remember, the primary boot path is generally set to the system root disk. The alternate boot path is generally set to the 9-track reel tape device or DDS tape device used for installations and updates (though you could set it to the CD-ROM device if you will be installing from this devices). The console path is set to the hardware address of the system console.

For more information, see `isl(1M)`.

Booting to Single User Mode

As the HP-UX-specific initial system loader utility for bootstrap and first-time installation, the `hpux` command has many options.

One option you may use fairly frequently is the `-i` option, which allows you to boot to a run-level other than that specified as the default run-level in `/etc/inittab`. Typically you will use this option to boot to single-user mode.

In order to perform some tasks you may need to boot the system to single-user mode. The command to do this is:

```
hpux -is <boot-path>
```

There are many additional options and parameters that may be used with the `hpux` command, for booting from other disk locations, other kernel files, other devices, etc. These are typically used in special circumstances only. For more information, refer to `hpux_800(1M)` or `hpux_700(1M)`.

Booting in single-user mode is especially useful when the system is not able to correctly complete the boot process (e.g. hangs at some point during normal boot). In single-user mode, the system does not execute any of the commands in `/etc/inittab` but allows the system administrator to repair the faulty files/commands in a shell.

If a system using the Logical Volume Manager (LVM) is booted in single-user mode, it sometimes happens that the system is unable to make the root volume group available to the system. Executing the command

```
# vgchange -a y vg00
```

(change the *a*vailability of volume group `vg00` (which is usually the root volume group) to *y*es) makes the root volume group available to the system, so that other file systems within this volume group can be used. This command is normally executed during system boot by the `lvmrc` script while in run-level 1.

6-13. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. What is Stable Storage and what is stored there?

2. Describe two commands that can be executed at the ISL prompt, besides `hpux`.

3. How would you boot your system if your regular `/stand/vmunix` kernel will not boot? Write the command to boot your system.

Server: Assume your regular kernel is located on the root disk with hardware path 52.6.0, which is also the primary boot path.

Workstation: Assume your regular kernel is located on the root disk on the scsi disk at address 6, which is also the primary boot path.

4. What would you do if you forgot the root password?

Module 7 — System Startup

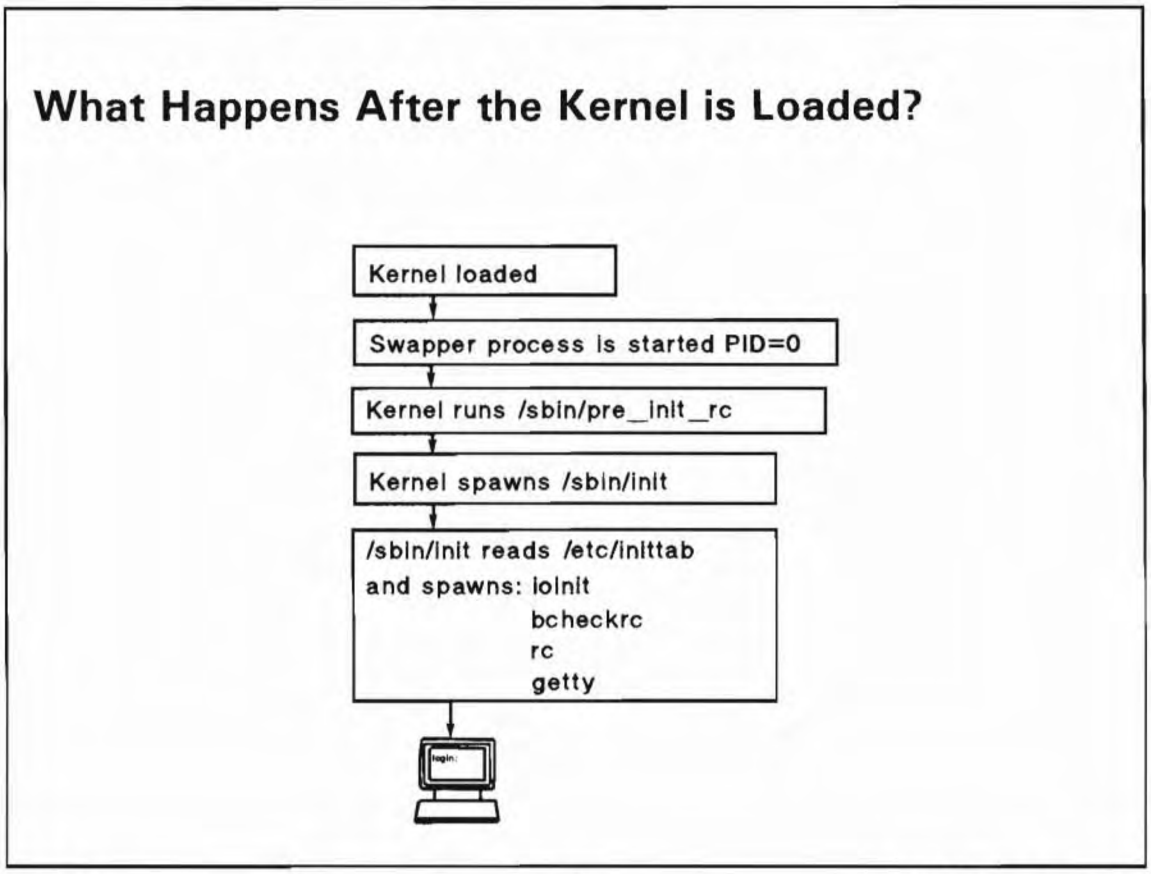
Objectives

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

- Explain the purpose and syntax of the `/etc/inittab` file.
- Add an entry to `/etc/inittab`.
- Explain how `init` reads the `inittab` file.
- Identify the default run-levels supplied with an HP-UX system.
- Change the default system run-level.
- List and describe the functions of these system startup files:

```
/sbin/bcheckrc
/sbin/rc
/sbin/lvmrc
```

7-1. SLIDE: What Happens After the Kernel is Loaded?



H6296 7-1 80

Student Notes

System startup begins with either a Reset (RS), Transfer of Control (TC), or cycling the power, and has two main phases:

1. The boot ROM startup sequence
2. The HP-UX startup sequence

When the system is powered up, the boot program initializes and tests hardware to bring the system to a state that is usable by the operating system.

After finding and checking the hardware, the boot program searches for a copy of the operating system (`/stand/vmunix`). It searches a list of potential sources, and from the first available source it finds, loads the **secondary loader** into memory. The secondary loader then loads `/stand/vmunix` into memory, and starts it up to enable you to use your system.

Module 7 — System Startup

/stand/vmunix then locates and configures the hardware devices, locates the root file system, and starts a shell to read commands from /sbin/pre_init_rc. It then starts the first process, /sbin/init, which reads from the /etc/inittab initialization file to define the environment for normal working conditions.

7-2. SLIDE: HP-UX Startup Sequence and pre_init_rc

HP-UX Startup Sequence and pre_init_rc

Once HP-UX starts, it starts a shell to read the script:

/sbin/pre_init_rc

which performs an fsck on the root disk.

H6296 7-2 81

Student Notes

Once the HP-UX kernel takes control, it runs the pre_init_rc script. Among other things, this script does an fsck of the root disk before any processes have the opportunity to write to it (and possibly corrupt it).

Caution Do not modify the /sbin/pre_init_rc script. Doing so could render your system unbootable.

7-3. SLIDE: Run-Levels

Run-Levels

- At all times, HP-UX is in a particular run-level
- Pre-defined run-levels are shipped with the system
- The run-level in which your system boots is defined by the `initdefault` entry in `/etc/inittab` (typically 3)

  `init:3:initdefault`
- The run-level can be changed with the `init` command
- Use `who -r` to determine the current run-level

H6296 7-3 82

Student Notes

At all times, the HP-UX system is in a particular run-level. A **run-level** is a system state in which a specific set of processes is allowed to run. This set of processes is defined in the `/etc/inittab` file for each run-level. Run-levels correspond to the following numbers/letter: 0, 1, 2, 3, 4, 5, 6, s, and S. You can define (or change) the run-levels 1-6. You cannot change run-levels s, or S.

Your system comes with several pre-defined run-levels: run-levels 0, 1, 2, 3, and run-levels s and S.

Run-level 0	is reserved for system shutdown. When running in run-level 0, the system performs the normal shutdown procedure, thereby stopping all processes and halting the system.
Run-level s	is a special run-level reserved for system administration tasks. It is also referred to as single-user run-level meaning it is reserved for a single user, typically, the

	system administrator. For example, shutting down the system (`/sbin/shutdown`) brings you to run-level s.
Run-level S	is similar to run-level s. With `init s` only the physical system console has access to the operating system, whereas `init S` (capital S) switches the capabilities of the system console to the terminal where you are logged in, thus making it the virtual system console.
Run-levels 1 to 3	are used during system startup. We will talk about them when discussing `/sbin/rc` in this module.
Run-level 4	For HP VUE users. In this mode, HP VUE is active.

The remaining run-levels can be defined by the system administrator. For example, the administrator may wish to define a run-level where only certain processes are allowed to run. We will talk more about designing your own run-level after we discuss `/etc/inittab`.

You can change the run-level of the system with the `init` command. Invoking `init` with an argument causes `init` to change the run-level of the system to the level specified by the argument. `init` scans `/etc/inittab` for all entries matching the new run-level (including those entries that are valid for all run-levels) and executes the commands associated with the entries. For example, the convention on most HP-UX systems is that run-level 3 or 4 is used for multi-user operation. So, to change the system from single-user to multi-user mode, the administrator would enter:

```
init 3
```

Whenever the run-level of the system is changed, any process started by `init` at the previous run level, and which does not have an entry for the new run-level, is sent a warning signal and then, after a 20 second grace period, is killed.

We have seen that the run-level of an HP-UX system is controlled by `init`. The actions of `init` are in turn controlled by a configuration file called `/etc/inittab`. Next, we will look at the contents and format of this file and how `init` utilizes this information to control the run-level of the machine.

7-4. SLIDE: Fields in the `inittab` File

Fields in the `inittab` File

Format:

```
id:rstate:action:process
```

Examples:

```
init:3:initdefault:
tty4:236:respawn:/usr/sbin/getty tty0p4 9600 #office L12
```

Purpose:

Determines what init will do

H6296 7-4 83

Student Notes

Each of the fields in the `inittab` file is defined as follows:

id	A one-to-four character unique label which identifies an entry.
rstate	Defines run-levels in which the entry will be processed. You can have multiple entries in the *rstate* field. If the *rstate* field is blank, the entry is valid for all run levels.
action	A keyword which defines how to execute the *process* (or program).
process	Shell command to be run if the entry's *rstate* matches the run-level and/or the *action* field indicates such action.

It is recommended that you use a four-character `id`, because many `pty` servers use the last two characters of the `pty` name as an `id`. If an `id` chosen by a `pty` server collides with an `id` used in the

/etc/inittab file, the /etc/utmp file can become corrupted, which can cause commands such as who to report inaccurate information.

Each terminal or RS-232 port used as an incoming terminal device must have an inittab entry similar to the one above.

Examples

In the first example, the initdefault action causes the initial (default) run-level to be the value of the *rstate* field. The initdefault entry does not require a *process*. We looked at this entry briefly on the previous slide.

In the second example, each of the fields can be explained as follows:

tty4	is the id, or label; a unique number that identifies the entry in /etc/inittab.
236	are the run levels. When the current system run level matches the run level in an entry, the program associated with it is executed.
respawn	is the action, that the system should take with the program.
/etc/getty tty0p4 9600	is the process, or program; the part of the entry that will be passed to the shell for execution.
#office L12	is a comment. Comments must be preceded by a #.

The entries shown in the samples are from typical inittab files. The file that comes with your system will probably be a little different. It is your responsibility as system administrator to modify this file as needed.

7-5. SLIDE: Action and Process in /etc/inittab

Action and Process in /etc/inittab

Important *action* keywords

```
initdefault
wait
boot
bootwait
respawn
ondemand
off
sysinit
powerfail
powerwait
once
```

H6296 7-5 84

Student Notes

Actions

The third field, or the *action* field, contains a keyword which tells `init` how to execute the program specified in the fourth field. This allows you the ability to tailor the way in which processes are executed. For example, you can specify whether you want `init` to wait for the process to complete before starting another, or go ahead and start a new process while the first is still running.

There are many keywords allowable in the *action* field. The most important ones are listed below. For more information, see `inittab(4)` in the on-line manual pages:

```
# man inittab
```

`initdefault` Causes the initial (default) run-level to be the value of the *rstate* field. If more than one run- level is specified in *rstate* init uses the highest specified run-level.

`wait` On entering the run-level that matches the *rstate* field of this entry, run *process* and wait for it to die before reading the next entry.

`boot` Run the command specified in the *process* field at boot-time only. Do not wait for *process* to die before reading the next entry.

`bootwait` Run the command specified in the *process* field at boot-time only. Wait for *process* to die before reading the next entry.

`respawn` On entering the run-level that matches the *rstate* field of this entry, run *process* if it is not already running. Do not wait for *process* to die before reading the next entry. If/when the process dies, run it again.

`ondemand` This instruction is really a synonym for the `respawn` action. It is functionally identical to `respawn`.

`off` When the run-level is entered, if the *process* is running, `init` will send *process* a warning signal and then wait 20 seconds before killing it. If *process* is not running, the entry is ignored. Thus `off` is also used to deactivate an entry for some time.

`sysinit` Entries of this type are executed before `init` tries to access the console. It is expected that this entry will be only used to initialize devices on which `init` might attempt to obtain run level information.

`powerfail` Execute the process associated with this entry only when `init` receives a power-fail signal (SIGPWR).

`powerwait` Execute the process associated with this entry only when `init` receives a power-fail signal (SIGPWR) and wait until it terminates before continuing any processing of `inittab`.

`once` When `init` enters the run level that matches the entry's `rstate`, start the process once and do not wait for its termination. If the process dies it is not restarted.

Process

The fourth field of `inittab` is the *process* or *program* field. This is the shell command `init` will execute if the entry's *rstate* matches the run-level and/or the *action* field indicates such action. The command in this field is automatically `exec`'d and passed to a child shell as `sh -c 'exec` *command*'. A comment can be inserted in this field by preceding the comment with a `#`.

For example, the `inittab` line

```
tty2:23:respawn:/usr/sbin/getty -h tty0p2 9600  # office K17
```

creates a `getty` child process, with `init` as the parent process.

7-6. SLIDE: inittab Example — Creating New Run-Levels

inittab Example – Creating New Run-Levels

```
init:4  :initdefault:
            .
            .
            .
cons:   :respawn:/usr/sbin/getty -h console console
tty1:456:respawn:/usr/sbin/getty -h tty0p1 9600 #faculty 1
tty2:456:respawn:/usr/sbin/getty -h tty0p2 9600 #faculty 2
tty5:45 :respawn:/usr/sbin/getty -h tty0p5 9600 #grad 1
tty6:45 :respawn:/usr/sbin/getty -h tty0p6 9600 #grad 2
tty8:4  :respawn:/usr/sbin/getty -h tty0p8 9600 #student 1
tty9:4  :respawn:/usr/sbin/getty -h tty0p9 9600 #student 2
```

H6296 7-6 85

Student Notes

Assume there are terminals for three groups of people: undergraduate students, graduate students, and faculty. At all times, you want the faculty members to have access to the system. But there are times that you want to restrict undergraduate and/or graduate students from having access.

Case 1 — Undergraduate students are restricted from access. Both graduate and faculty have access.

Case 2 — Undergraduate and graduate students are restricted from access. Only the faculty has access to the system.

To do this, you would create three run-levels as shown on the slide. Run-level 4 allows everyone to access the system. Run-level 5 allows only graduate and faculty to access the system. Run-level 6 allows only the faculty members to access the system. The system comes up in run-level 6. To allow everyone access you would simply change the run-level of the system to 4.

7-7. TEXT PAGE: Additional inittab Example

A Sample Inittab file for HP 9000 Servers

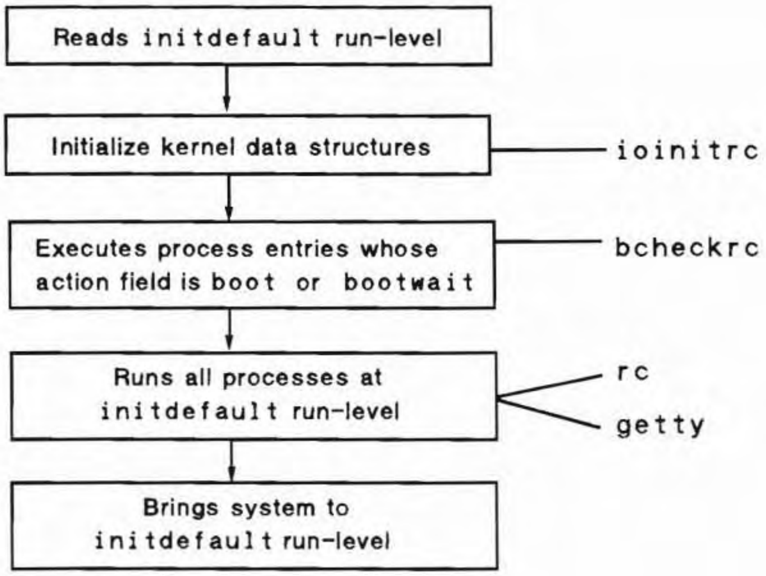
```
init:3:initdefault:
ioin::sysinit:/sbin/ioinitrc >/dev/console 2>&1
tape::sysinit:/sbin/mtinit > /dev/console 2>&1
muxi::sysinit:/sbin/dasetup   </dev/console >/dev/console 2>&1 #mux init
stty::sysinit:/sbin/stty 9600 clocal icanon echo opost onlcr ixon icrnl ignpar </dev/systty
brc1::bootwait:/sbin/bcheckrc </dev/console >/dev/console 2>&1 #fsck, etc.
link::wait:/sbin/sh -c "/sbin/rm -f /dev/syscon; \
              /sbin/ln /dev/systty /dev/syscon" >/dev/console 2>&1
cprt::bootwait:/sbin/cat /etc/copyright >/dev/syscon      #legal req
sqnc::wait:/sbin/rc </dev/console >/dev/console 2>&1      #system init
powf::powerwait:/sbin/powerfail >/dev/console 2>&1         #powerfail
cons:1234:respawn:/usr/sbin/getty console console          #system console
ttp1:2:respawn:/usr/sbin/getty -h tty0p1 9600
ttp2:2:respawn:/usr/sbin/getty -h tty0p2 9600
ttp3:2:respawn:/usr/sbin/getty -h tty0p3 9600
ttp4:2:respawn:/usr/sbin/getty -h tty0p4 9600
ttp5:2:respawn:/usr/sbin/getty -h tty0p5 9600
```

A Sample inittab file for HP 9000 Workstations

```
init:4:initdefault:
ioin::sysinit:/sbin/ioinitrc >/dev/console 2>&1
tape::sysinit:/sbin/mtinit > /dev/console 2>&1
stty::sysinit:/sbin/stty 9600 clocal icanon echo opost onlcr ixon
icrnl ignpar </dev/systty
brc1::bootwait:/sbin/bcheckrc </dev/console >/dev/console 2>&1 #fsck, etc.
link::wait:/sbin/sh -c "/sbin/rm -f /dev/syscon; \
              /sbin/ln /dev/systty /dev/syscon" >/dev/console 2>&1
cprt::bootwait:/sbin/cat /etc/copyright >/dev/syscon            #legal req
sqnc::wait:/sbin/rc </dev/console >/dev/console 2>&1            #system init
cons:1234:respawn:/usr/sbin/getty console console               #system console
vue :4:respawn:/usr/vue/bin/vuerc                               #VUE invocation
```

7-8. SLIDE: init Actions After Booting

init Actions After Booting



H6296 7-8 86

Student Notes

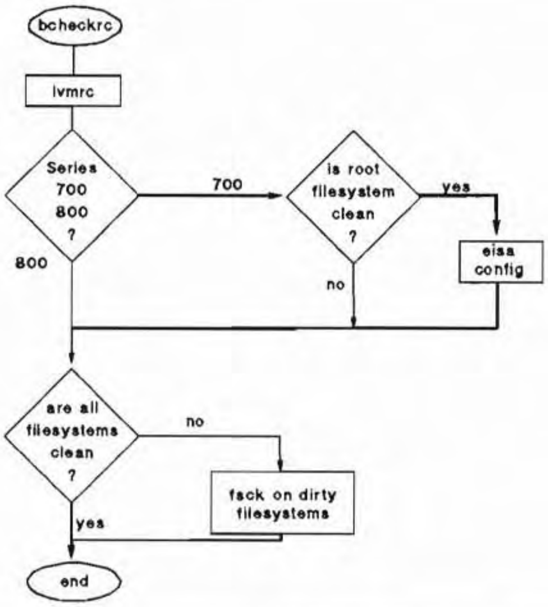
The first thing init does is scan inittab for an initdefault entry in the *action* field. The run-level associated with this entry is the initial run-level init will enter. If there is no entry for initdefault, init will prompt the administrator to specify a run-level to enter.

Then, init initializes the kernel data structures. It calls ioinitrc, which uses information from /etc/ioconfig and then calls insf to assign logical unit numbers and create special files for all new devices on the system.

Next init will scan inittab for all entries marked boot or bootwait in the *action* field. Any commands associated with these entries are executed. In our sample inittab file, the program /sbin/bcheckrc is run.

7-9. SLIDE: The /sbin/bcheckrc Program

The /sbin/bcheckrc Program



H6296 7-9 87

Student Notes

```
brc1::bootwait:/sbin/bcheckrc </dev/console >/dev/console 2>&1
```

The /sbin/bcheckrc (Boot Check Run Command) program checks to see if the system was properly shutdown.

If your system is using disk mirroring, /sbin/bcheckrc calls /sbin/mirrorrc. Tasks performed by mirrorrc include: configuring mirrors, running fsck, invoking mirrorlog and then re-imaging mirrors.

If you have implemented the Logical Volume Manager (LVM), bcheckrc then calls /sbin/lvmrc to activate LVM volume groups.

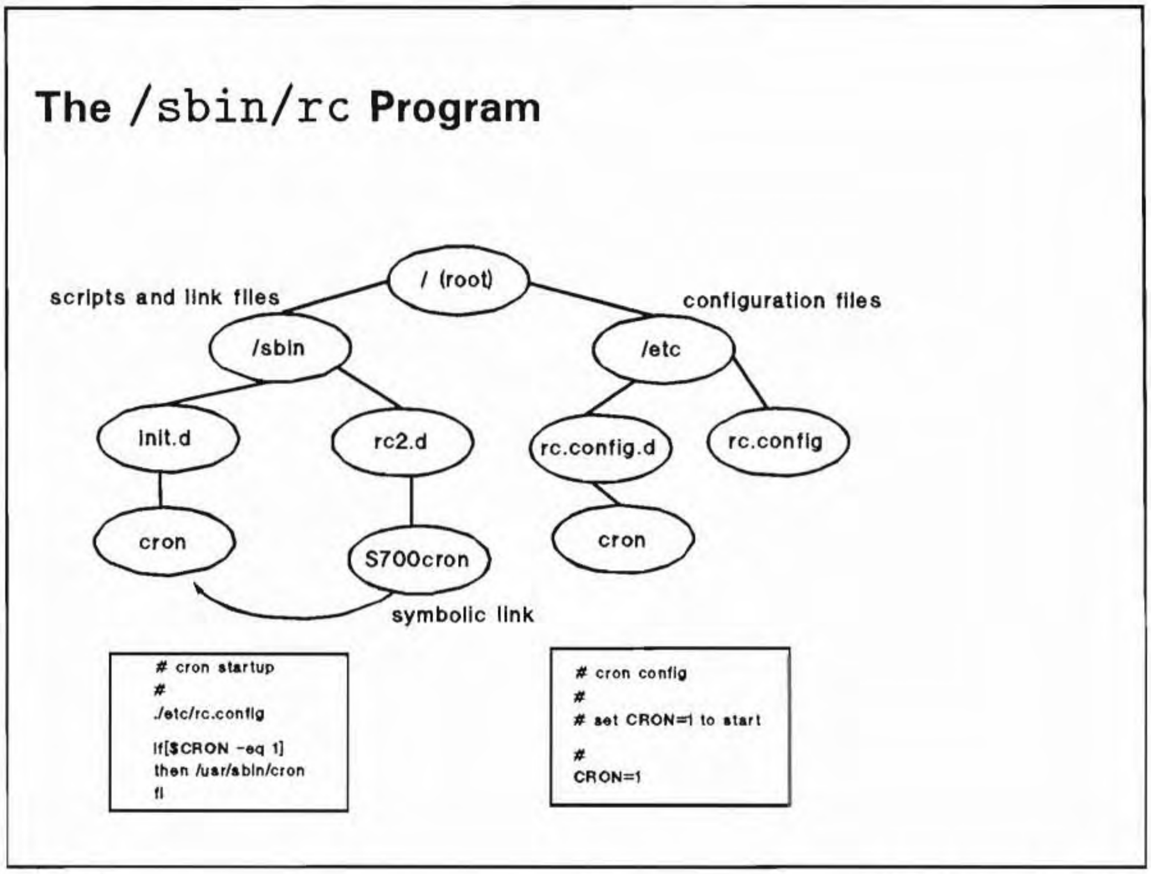
Next, to determine if the system was properly shutdown, bcheckrc calls the fsclean program. fsclean checks each file system of type hfs in /etc/fstab to see if there might be a consistency problem. To do this, `fsclean` looks at a flag called the **clean byte** in the primary super block of each file system. When a file system is created, the clean byte flag is set to FS_CLEAN. When the file system is mounted (using the `mount` command), the clean byte flag is set to FS_OK. During normal shutdown (that is, during the execution of the `reboot` or `shutdown` command), the clean byte is reset to FS_CLEAN. So, under normal conditions, the file system can be unmounted and set to FS_CLEAN, or mounted and set to FS_OK.

If, when `fsclean` checks the clean byte, it finds the file system is unmounted and set to FS_OK, then the file system might be in an inconsistent state (due to a crash or other incorrect shutdown).

In this case, `bcheckrc` will run `fsck` automatically, using the `preen` mode. This will correct most errors found.

The root file system, or the primary file system, is checked before `init` runs. So `bcheckrc` will check all file systems, except the root file system, which has already been checked.

7-10. SLIDE: The /sbin/rc Program



H6296 7-10 88

Student Notes

```
sqnc::wait:/sbin/rc </dev/console >/dev/console 2>&1
```

The above line means:

- Start `/sbin/rc` and `wait` for its end, before other actions are taken
- Read input from `/dev/console` (the system console's keyboard)
- Output to `/dev/console` (the system console's screen)
- Redirect error output also to `/dev/console`

What does `/sbin/rc` do ?

- Executes whenever a new run-level is invoked

- When moving from a lower to a higher level:
  - Runs the configuration files of all startup scripts for all levels between the old and the new level
  - Starts all servers and functions defined for all levels between the old and the new level, depending on the configuration files read earlier
- When moving from a higher to a lower level:
  - Stops all servers and functions defined for all levels between the old and the new level
  - Halts system when moving to run-level 0

As we have seen, whenever the run-level of the machine is changed with `init` the file `/etc/inittab` is read for entries matching the new run-level. One of the entries in `inittab` invokes `/sbin/rc`. As shipped, this file is invoked every time the run-level of the HP-UX system is changed.

The run-levels `0`, `S`, `1`, `2` and `3` are predefined. For each of these run-levels (except `S`, which uses the same directory as run-level `0`) there is a directory in `/sbin`. They are named `/sbin/rc`*R*`.d`, where *R* is the run-level. For example: `/sbin/rc2.d`.

The predefined run-levels are defined as:

`0`	Halted or reboot level. All kill (`K`) and start (`S`) scripts in `/sbin/rc0.d` are executed.
`S`	Single user state. All kill/start scripts in `/sbin/rc0.d` are executed.
`1`	Boot time system configuration. Necessary system configuration is done at this state. (For example: assignment of the hostname, mount of disk file systems, performing software configuration.)
`2`	Typically called multi-user state. Allows users to access the system.
`3`	Networked multi-user state. NFS file systems can be exported.
`4`	For HP VUE users.
`5,6`	Not defined.

How `/sbin/rc` works

The `/sbin/rc` script performs the following:

- Runs `/sbin/rc.utils`. This script shows only short startup messages on the console instead of the complete output of all startup scripts. The complete output is logged to `/etc/rc.log`.
- Runs `/etc/rc.config`. This performs only one function: run all configuration files in `/etc/rc.config.d`. See next step.
- Runs all files in `/etc/rc.config.d`. These scripts only set variables which are evaluated by `/sbin/rc`. The files in `/etc/rc.config.d` are the only files where you can configure the system to adapt it to your needs.
- Checks the new level. When moving to a level *not* adjacent to the current level, `/sbin/rc` performs all tasks of the run-levels between current and new level.
  - When the new level is higher than the current level, then all scripts in `/sbin/rc`*R*`.d` (where *R* is the current run-level plus 1) with names beginning with `S` (for "Start") are run. For example:

/sbin/rc2.d/S700cron. Then the system moves to the next higher level; if this is not the target level, this step is repeated.

- When the new level is equal to the current level, no action is taken.
- When the new level is lower than the current level, then all scripts in /sbin/rc*R*.d (where *R* is the current level minus 1) with names beginning with an uppercase K (for "Kill") are run. For example: /sbin/rc1.d/K012cron. Then the system moves to the next lower level; if this is not the target level, this step is repeated.

Example

Your system is in run-level 1. You command your system to move to run-level 3 by entering init 3. /sbin/rc moves from run-level 1 to 2 and runs all scripts belonging to level 2. Then it moves from run-level 2 to 3 and runs all scripts belonging to level 3.

These steps are:

- init sets the system from run-level 1 to run-level 3.
- /sbin/rc is started.
- /sbin/rc.utils checks your console and prepares to display startup messages.
- /etc/rc.config runs all configuration scripts in /etc/rc.config.d, thereby setting options to all startup scripts, for example /etc/rc.config.d/cron.
- /sbin/rc searches /sbin/rc2.d for filenames beginning with an S. Actually, these files are symbolic links to files in /sbin/init.d. For example the script /sbin/rc2.d/S700cron is a symbolic link to /sbin/init.d/cron.
- The referenced files in /sbin/init.d are started with the option start. They perform various operations, depending on what was found in the /etc/rc.config.d files.
- Now, since the /sbin/rc has performed all scripts for run-level 2, the last two steps repeated with run-level 3.

This works similarly when moving from a higher run-level to a lower one, except that the run-levels are decremented by 1 on each pass, and the "kill" scripts are run.

Caution

The only files where you can configure the system are in /etc/rc.config.d. All other files, like the links/files in /sbin/rc2.d or /sbin/init.d, should not be modified because they may be overwritten without warning during system updates.

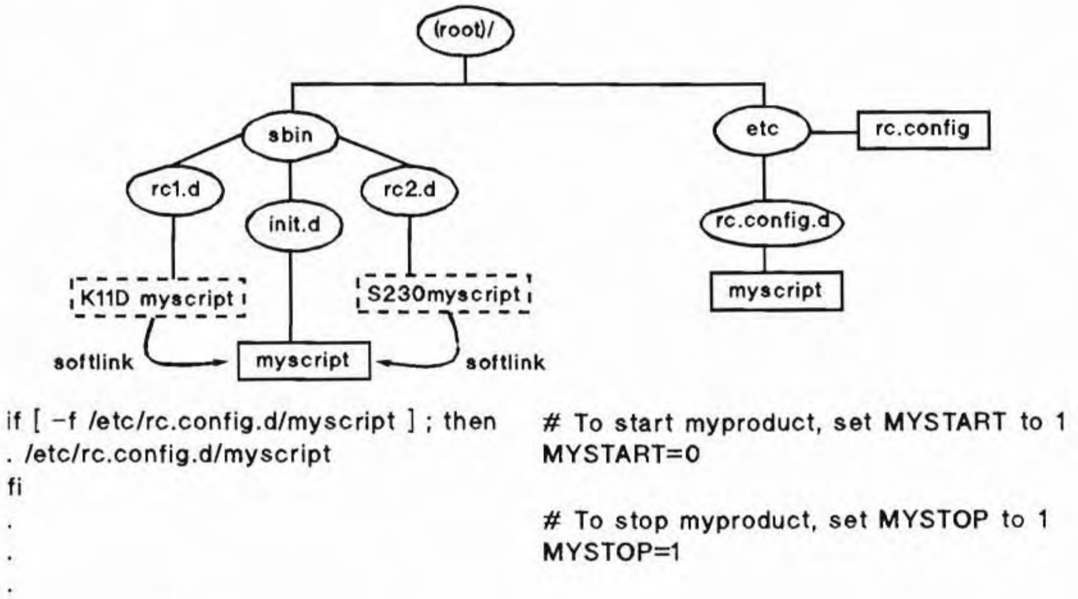
The numbers in the link names of /sbin/rc2.d (and others) guarantee the correct sequence of system startup. Changing these numbers may result in an undefined system state.

Module 7 — System Startup

Example of customizing system startup

Here is an example of a (very basic) set of files for startup and shutdown of a user function:

File system view



File Contents

/etc/rc.config.d/myscript

```
# My own startup configuration script
MYSTART=1
MYSHUT=1
```

/sbin/init.d/myscript

```
##########
# My own startup and shutdown script
#
##########
# Set variables configuring this script
##########

if [ -f /etc/rc.config.d/myscript ]; then
. /etc/rc.config.d/myscript
fi

##########
```

```
# The heart of myscript
##########

case $1 in
start_msg)              # for rc.util (formatted output)
   echo "Execute my start script"
   retval=0
   ;;
stop_msg)               # for rc.util (formatted output)
   echo "Execute my stop script"
   retval=0
   ;;

'start')                # for rc (execution)
   if [ "$MYSTART" -eq 1 ]; then
                        # MYSTART was set in
/etc/rc.config/myscript
        if /opt/myproduct/bin/start_myproduct; then
           retval=0     # myproduct is successfully started
        else
           retval=1     # myproduct is NOT successfully started
        fi
   fi
   ;;
'stop')                 # for rc (execution)
   if [ "$MYSTOP" -eq 1 ]; then
                        # MYSTOP was set in /etc/rc.config/myscript
        if /opt/myproduct/bin/stop_myproduct; then
           retval=0     # myproduct is successfully stopped
        else
           retval=1     # myproduct is NOT successfully stopped
        fi
   fi
   ;;
*)
   retval=1             # all other cases are errors
   ;;
esac

exit $retval            # return success/error to rc script
```

Startup/shutdown commands

The commands executed during startup to run-level 2 are:

```
# . /sbin/rc.config.d/myscript
        This sets the variables in the current shell.
# /sbin/rc2.d/S230myscript start_msg
        The startup message is displayed on the screen.
# /sbin/rc2.d/S230myscript start
```

```
        /opt/myproduct/bin/start_myproduct is run.
```

For system shutdown, i.e. when moving from run-level 2 to run-level 1, myproduct is stopped with the command:

```
# . /sbin/rc.config.d/myscript
        This sets the variables in the current shell.
# /sbin/rc2.d/K770myscript stop_msg
        The shutdown message is displayed on the screen.
# /sbin/rc2.d/K770myscript stop
        /opt/myproduct/bin/stop_myproduct is run.
```

7-11. SLIDE: /usr/sbin/getty at the console

Student Notes

Once /sbin/rc has completed the change to run-level 3, control returns to init, which runs the commands from the *process* field of all run-level 3 entries in /etc/inittab. Run-level 3 commands usually consist of /usr/sbin/getty commands, one for each terminal on which users log in.

The following entry runs a getty at the system console:

```
cons:123456:respawn:/usr/sbin/getty -h console console # system
console
```

The respawn action tells init to restart the getty process after it dies. This is why each time you log off the system console, a new login: prompt is displayed.

7-12. SLIDE: The System Startup Checklist

The System Startup Checklist

```
                HP-UX Start-up in progress
        ---------------------------------------
Mount file systems ........................................ OK
Setting hostname .......................................... OK
Set privilege group ....................................... N/A
Display date .............................................. N/A
Save system core image if needed .......................... N/A
Enable auxiliary swap space ............................... OK
Start syncer daemon ....................................... OK
Configure LAN interfaces .................................. OK
Start Software Distributor agent daemon ................... OK
Configuring all unconfigured software file sets ........... OK
Recover editor crash files ................................ OK
Clean UUCP ................................................ OK
List and/or clear temporary files ......................... OK
Clean up old log files .................................... OK
Start system message logging daemon ....................... OK
Start pty allocator daemon ................................ OK
Start network tracing and logging daemon .................. OK
 ...
```

H6296 7-12 90

Student Notes

During system startup and shutdown, you will see a display similar to this on your system console. There are three possible outcomes for each task:

`OK` The execution script started up (or shut down) the subsystem.

`FAIL` A problem occurred while trying to start up or shut down the subsystem. The asterisk following this message refers to a note at the bottom of the console screen, such as:

```
* - An error has occurred!
* - Refer to the file /etc/rc.log for more information.
```

`N/A` The subsystem was not configured to start.

7-13. SLIDE: Sequencer Directories

Sequencer Directories

```
/sbin/rc3.d/S040diagnostic
        1   2 3   4
```

1. Run Level Numbers
2. Sequencing Type
3. Sequence Number
4. Script name

H6296 7-13 91

Student Notes

These directories are also referred to as the run level directories. There is one `rc`*n*`.d` directory per run level, and they contain only symbolic links (the "link files") to the `/sbin/init.d` execution scripts.

The directory number refers to the run level. Run level 0 and run level S are considered the same. The naming of the links within the directories determines the order of execution of the scripts.

`/sbin/rc` is invoked by `init`, and runs the execution scripts when transitioning from one run level to another. `/sbin/rc` goes through all levels between the start and target level in turn. For example, at boot it goes through `rc1.d`, `rc2.d`, and `rc3.d`. When moving up to a run level all scripts beginning with an S will be run with an argument to start the subsystem. When moving down to a run level all scripts beginning with a K will be run with an argument to stop the subsystem.

Subsystems are killed in the opposite order from which they were started, so kill scripts do not have the same sequence numbers as their start counterparts. Also, the start script will be located in `/sbin/rcN.d`, while its kill script counterpart will be located in `/sbin/rcN-1.d`. For example:

`/sbin/rc3.d/S123subsys`	*Start script*
`/sbin/rc2.d/K654subsys`	*Kill script*

In addition, if two subsystems are started in a given order due to dependencies, the kill script counterparts are numbered so that the subsystems are stopped in the opposite order from which they were started.

Link File Naming

Link files control the execution order and state of the execution scripts. They follow a strict naming convention. The scripts are executed in alphabetical order, and this naming convention enables specific sequencing of scripts.

The name of the script tells `/sbin/rc` the following:

- Whether to run the "start" or the "kill" script. It knows this based on whether it is moving "up" or "down" in the run level sequence.
- The sequence in which to execute the script.

Start and kill links are assigned by HP, and they should not be moved or renamed. You should only use the configuration variable files to control whether or not an execution script runs.

7-14. SLIDE: Sample Configuration File

Sample Configuration File

```
#!/sbin/sh
# @(#) $Revision: 72.4 $
# Savecore configuration
#
# SAVECORE:     Set to 0 to disable saving system core files.
SAVECORE=1

# SAVECORE_DIR:      Directory name for system core files.  Note:
#   the file system in which this directory is located should
#   have as much free space as your system has RAM.
SAVECORE_DIR=/var/adm/crash

# SAVE_PAGES:   Only options are: i, u, k and <NULL>.
#       SAVE_PAGES=     - (DEFAULT) Save complete core file.
#       SAVE_PAGES=i    - Save the complete  ...
#       SAVE_PAGES=u    - Save user and kernel   ...
#       SAVE_PAGES=k    - Save only kernel   ...
SAVE_PAGES=
```

H6296 7-14 92

Student Notes

You modify configuration scripts to control the behavior of subsystems. The configuration scripts are located in `/etc/rc.config.d` and usually have the same name as their corresponding execution script in `/sbin/init.d`.

The control variables in these scripts can be used to selectively start or not start a portion of the subsystem.

Comments in the configuration files provide explanations if the variables and appropriate values for them. In the example above, you set a variable to start the subsystem. An additional variable sets the directory where files are to be saved. Other variables could be set if necessary to configure additional features. Some subsystems may have several variables that you can configure.

The execution scripts in `/sbin/init.d` check the value of variables set in `/etc/rc.config.d` and behave accordingly.

Modifying the Configuration File

You can use any editor to modify the configuration file. In most cases, the first variable is named the same as the subsystem, and you indicate whether to enable or disable the subsystem by modifying the value of the variable:

1	Enabled
0	Disabled

Configuration files are documented with comments which provide additional information about what to modify.

"Original" copies of scripts are kept in `/usr/newconfig/config.d`, so in case of error while editing, you can copy them and begin again.

7-15. SLIDE: Execution Scripts

Execution Scripts

```
case $1 in
'start_msg')
     echo "Start cron daemon" ;;
'stop_msg')
     echo "Stop cron daemon" ;;
'start')
     if [ -f /etc/rc.config.d/cron ] ; then
          . /etc/rc.config.d/cron
     if [ $CRON -eq 1 ] then
          /usr/sbin/cron
     ;;
'stop')
     kill `ps -ef | grep cron | cut -c10-14`
     ;;
*)
     echo "usage: $0 {start|stop}" ;;
esac
```

H6296 7-15 93

Student Notes

Execution scripts are located in /sbin/init.d, and are executed when transitioning from one run level to another. They perform both startup and shutdown functions for the associated subsystem. They respond to standard arguments, issue standard exit codes, and report output and errors to stdout and stderr, respectively. They also display the startup/shutdown checklists on the system console.

Execution scripts are always invoked with one of four arguments, which indicates the "mode":

start	Bring subsystem up
start_msg	Optional message to print upon startup. This is the message that appears in the system startup checklist.
stop	Bring subsystem down
stop_msg	Optional message to print upon shutdown

Caution Do not modify the execution scripts!

Because /sbin/init.d execution scripts must start up and shut down their associated subsystems, they will have two links in the sequencer directories; a start link and a kill link. The start link is referred to as a "start script" and the kill link is referred to as a "kill script."

Start links are named with an S, and are invoked by /sbin/rc with the start argument at system boot, or upon transition to a higher run level. Start mode indicates that /sbin/rc passes the start argument to the script.

Kill links are named with a K, and are invoked by /sbin/rc with the stop argument at system shutdown, or upon transition to a lower run level. Kill mode indicates that /sbin/rc passes the stop argument to the script.

Complete Example

```
#!/sbin/sh
#
# @(#) $Revision: 72.7 $
#
# NOTE:    This script is not configurable!  Any changes made to this
#          script will be overwritten when you upgrade to the next
#          release of HP-UX.
#
# WARNING: Changing this script in any way may lead to a system that
#          is unbootable.  Do not modify this script.
#
# Start cron
#
PATH=/sbin:/usr/sbin:/usr/bin
export PATH

rval=0
set_return() {
     x=$?
     if [ $x -ne 0 ]; then
          echo "ERROR CODE $x"
          rval=1
     fi
}

case $1 in
start_msg)
     echo "Start clock daemon"
     ;;
```

```
stop_msg)
     echo "Stop clock daemon"
     ;;

'start')
     if [ -f /etc/rc.config.d/cron ] ; then
          . /etc/rc.config.d/cron
     else
          echo "ERROR: /etc/rc.config.d/cron defaults file MISSING"
     fi

     if [ "$CRON" -eq 1 -a -x /usr/sbin/cron ]; then
          if [ -f /var/adm/cron/log ]; then
               mv /var/adm/cron/log /var/adm/cron/OLDlog
          fi
          /usr/sbin/cron & echo cron started
          set_return
     else
          rval=2
     fi

     ;;

'stop')
     #
     # Determine PID of process(es) to stop
     #
     pid=`ps -ef | awk '$NF~/cron/ {print $2}'`
     if [ "X$pid" != "X" ]; then
          if kill $pid; then
               echo "cron stopped"
          else
               set_return
               echo "Unable to stop cron"
          fi
     fi
     ;;

*)
     echo "usage: $0 {start|stop}"
     ;;
esac

exit $rval
```

7-16. SLIDE: Adding a Program to Run at Boot Time

Adding a Program to Run at Boot Time

1. Create execution script
2. Create configuration script
3. Create start and kill links

H6296 7-16 94

Student Notes

To create the execution script for a subsystem, you begin with the template file `/sbin/init.d/template`. It is best to name the execution script the same as the subsystem it starts and stops. The execution script must reside in `/sbin/init.d`.

To create the configuration script, use the scripts in `/etc/rc.config.d` as models. The configuration script should be named the same as the execution script. Place the script in `/etc/rc.config.d`.

The sequencer links should be placed in the appropriate sequencer directory, based on startup and shutdown dependencies.

7-17. SLIDE: Writing Execution Scripts

Writing Execution Scripts

Template /sbin/init.d/template

Arguments

```
start_msg
stop_msg
start
stop
```

Exit values

0	Exited without error
1	Encountered errors
2	Skipped
3	Executed normally, must reboot for changes to take effect

H6296 7-17 95

Student Notes

If you are an application developer who must write execution scripts for your product, you should use the template file located in `/sbin/init.d/template`.

Ensure that your script recognizes these four arguments:

`start_msg`	*Display message used by startup checklist*
`stop_msg`	*Display message used by shutdown checklist*
`start`	*Start subsystem*
`stop`	*Stop subsystem*

`start_msg` and `stop_msg` should report the action that will be taken if `start` or `stop` is passed to the script. These arguments should not cause any other action to occur. The message should be a single line, with no more than 30 characters. Do not echo messages indicating entry or exit from the script. `/sbin/rc` redirects the `start_msg` and `stop_msg` output to `/etc/rc.log`.

Do not echo status or error messages to the console. Direct status messages to `stdout`, and error messages to `stderr`. If you use the console to display messages during boot, you will overwrite the startup/shutdown checklist.

Use *only* these standard exit values:

0	Exited without error
1	Encountered errors
2	Skipped
3	Executed normally, must reboot for changes to take effect

If you return any other exit value, the script may appear to have been skipped in the checklist, or you may cause the system to reboot.

Example

```
#!/sbin/sh
#
# @(#) $Revision: 72.5 $
#
# NOTE:    This script is not configurable!  Any changes made to this
#          script will be overwritten when you upgrade to the next
#          release of HP-UX.
#
# WARNING: Changing this script in any way may lead to a system that
#          is unbootable.  Do not modify this script.
#
#
# Syncer helps minimize file system damage in the event
# of a power failure or other system crash.
#

PATH=/sbin:/usr/sbin:/usr/bin
export PATH

rval=0
set_return() {
     x=$?
     if [ $x -ne 0 ]; then
          echo "EXIT CODE: $x"
          rval=1
     fi
}

case $1 in
start_msg)
     echo "Start syncer daemon"
```

```
        ;;

stop_msg)
        echo "Stop syncer daemon"
        ;;

'start')
        if [ -f /etc/rc.config.d/syncer ] ; then
                . /etc/rc.config.d/syncer
        else
                echo "ERROR: /etc/rc.config.d/syncer defaults file MISSING"
        fi

        if [ "$SYNCER" -eq 1 -a -x /usr/sbin/syncer ]; then
                /usr/sbin/syncer & echo syncer started
                set_return
        else
                rval=2
        fi
        ;;

'stop')
          #
        # Determine PID of process(es) to stop
        #
        pid=`ps -ef | awk '$NF~/syncer/ {print $2}'`
        if [ "X$pid" != "X" ]; then
                if kill $pid; then
                        echo "Syncer stopped"
                else
                        set_return
                        echo "Unable to stop syncer"
                fi
        fi
        ;;

*)
        echo "usage: $0 {start|stop}"
        rval=1
        ;;
esac

exit $rval
```

7-18. TEXT PAGE: Template for Execution Scripts

The template for execution scripts is located in `/sbin/init.d/template`.

```
#!/sbin/sh
#
# @(#) $Revision: 72.9 $
#
# NOTE:    This script is not configurable!  Any changes made to this
#          script will be overwritten when you upgrade to the next
#          release of HP-UX.
#
# WARNING: Changing this script in any way may lead to a system that
#          is unbootable.  Do not modify this script.
#
# <Insert comment about your script here>
#
# Allowed exit values:
#    0 = success; causes "OK" to show up in checklist.
#    1 = failure; causes "FAIL" to show up in checklist.
#    2 = skip; causes "N/A" to show up in the checklist.
#           Use this value if execution of this script is overridden
#       by the use of a control variable, or if this script is not
#       appropriate to execute for some other reason.
#      3 = reboot; causes the system to be rebooted after execution.
# Input and output:
#    stdin is redirected from /dev/null
#
#    stdout and stderr are redirected to the /etc/rc.log file
#    during checklist mode, or to the console in raw mode.

PATH=/usr/sbin:/usr/bin:/sbin
export PATH

# NOTE: If your script executes in run state 0 or state 1, then /usr might
#    not be available.  Do not attempt to access commands or files in
#    /usr unless your script executes in run state 2 or greater.  Other
#    file systems typically not mounted until run state 2 include /var
#    and /opt.

rval=0

# Check the exit value of a command run by this script.  If non-zero, the
# exit code is echoed to the log file and the return value of this script
# is set to indicate failure.

set_return() {
     x=$?
     if [ $x -ne 0 ]; then
```

```
        echo "EXIT CODE: $x"
        rval=1    # script FAILed
    fi
}

# Kill the named process(es).
# $1=<search pattern for your process>

killproc() {
    pid=`ps -ef | awk '$NF~/'"$1"'/ {print $2}'`
    if [ "X$pid" != "X" ]; then
        if kill "$pid"; then
            echo "$1 stopped"
        else
            rval=1
            echo "Unable to stop $1"
        fi
    fi
}


case $1 in
'start_msg')
    # Emit a _short_ message relating to running this script with
    # the "start" argument; this message appears as part of the checklist.
    echo "Starting the foobar subsystem"
    ;;

'stop_msg')
    # Emit a _short_ message relating to running this script with
    # the "stop" argument; this message appears as part of the checklist.
    echo "Stopping the foobar subsystem"
    ;;

'start')

    # source the system configuration variables
    if [ -f /etc/rc.config ] ; then
        . /etc/rc.config
    else
        echo "ERROR: /etc/rc.config defaults file MISSING"
    fi

    # Check to see if this script is allowed to run...
    if [ "$CONTROL_VARIABLE" != 1 ]; then
        rval=2
    else

    # Execute the commands to start your subsystem
    :
```

```
     fi
     ;;

'stop')
     # source the system configuration variables
     if [ -f /etc/rc.config ] ; then
          . /etc/rc.config
     else
          echo "ERROR: /etc/rc.config defaults file MISSING"
     fi

     # Check to see if this script is allowed to run...
     if [ "$CONTROL_VARIABLE" != 1 ]; then
          rval=2
     else
     :
     # Execute the commands to stop your subsystem

     fi
     ;;

*)
     echo "usage: $0 {start|stop|start_msg|stop_msg}"
     rval=1
     ;;
esac

exit $rval
```

7-19. SLIDE: Writing Configuration Files

Student Notes

Your configuration file will need to be edited by system administrators who install your product on their system. Therefore it is very important that you document your file very clearly. Use many comments to explain what needs to be done in order to configure your product. Comment lines begin with a # and must be on separate lines from the variable definition lines.

Because the POSIX shell is now the default shell, you should write your scripts for interpretation by the POSIX shell.

The configuration file is sourced by the execution script, to determine whether or not to start the product. Configuration files do not require any sourcing order, as they can be executed either individually, or all at once. Sourcing `/etc/rc.config` will source all the configuration variables below `/etc/rc.config.d`. `/etc/rc.config` will also source `/etc/TIMEZONE` for the definition of the TZ environment variable.

7-20. SLIDE: Creating Link Files

Creating Link Files

"DO NOT CARE" Start Link

/sbin/rc2.d/S900*exec_file*

"DO NOT CARE" Stop Link

/sbin/rc1.d/K100*exec_file*

- "Do Not Care" numbers mean order of startup/shutdown is not important for this process
- HP may also use these sequence numbers

H6296 7-20 97

Student Notes

If you are an application developer, a special link number that will not conflict with any HP numbers is reserved for your use. As long as the specific order of starting your subsystem is not of concern, that is, if the subsystem can be started any time after system boot and initialization, then you can use the following links:

Start link	/sbn/rc2.d/S900*exec_file*
Kill link	/sbn/rc1.d/K100*exec_file*

If you need a specific number assigned, you should contact:

```
PA-RISC Developer's Program
(508) 436-5144
```

```
pard@apollo.hp.com
```

and ask for information regarding "Startup/Shutdown Sequence Number Assignments".

7-21. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. How is the run-level of the system controlled?

2. What is the purpose of the `/etc/inittab` file?

3. What is the difference between `init s` and `init S`?

4. Describe what the `/sbin/rc` file does.

5. Determine the default run-level for the system you are using.

6. Determine the current run-level for the system being used. How many times has the current state been entered previously? What was the previous state of your system?

7. During the normal system startup process, which run-levels does the `/sbin/rc` file run through?

Module 8 — System Shutdown

Objectives

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

- Describe the importance of a proper system shutdown.
- Explain the differences between `shutdown` and `reboot`.
- Use the `shutdown` command.
- Use the `reboot` command.

Module 8 — System Shutdown

8-1. SLIDE: Why Shut Down the System?

Why Shut Down the System?

- To conduct administrative activities without user interference, such as:
  - Checking file systems
  - System updates
  - Backing up file systems
  - Reconfiguring the kernel
- To halt the system so it can be turned off
- The `shutdown` command is most commonly used to shut down the computer system.

H6296 8-1 98

Student Notes

Most of the time the HP-UX system will be in multi-user mode, allowing many user and system processes to run. There are occasions, however, when the administrator must change the run-level of the system from multi-user to single-user. For example, if the administrator wants to backup a file system, the users should not continue to work and possibly change files while the backup is occurring. The administrator should bring the system to single-user state before backing up. Once in single user mode, the administrator might want to halt the system completely or reboot the system.

Halting brings the system to a complete stop; in this state, the only way to restart the system is to cycle the power or reset the hardware. **Rebooting** brings the system to a complete stop, but then restarts the system as if you had booted it.

Whether to halt or reboot your system depends on why you want to shut down in the first place. If you want to shut the computer off for an extended period of time (for example, to add new hardware or to leave the system off for a long weekend), then halting is appropriate. If you want to shut down only to boot the system again (for example, to use a newly configured kernel), then rebooting is appropriate.

There are several different ways to shut down an HP-UX system. The administrator is responsible for making sure the system processes are halted in a consistent and orderly manner. *Pushing the power button is not the proper way to shut down your system.* Typically, the `shutdown` command is used to bring an HP-UX system from multi-user mode to single-user mode.

Note

If an HP-UX system is shut down improperly, you run the risk of corrupting the file system.

8-2. SLIDE: Two Kinds of Shutdown

Two Kinds of Shutdown

- Reboot after reaching run-level s
- Halt completely

Syntax:

```
/sbin/shutdown [ -r | -h ] [-y] [grace]
```

H6296 8-2 99 © 1995 Hewlett-Packard Co

Student Notes

`shutdown` stops system activities in an orderly and consistent manner. After system activities are stopped, administrative activities can be performed, or the machine can be rebooted or turned off.

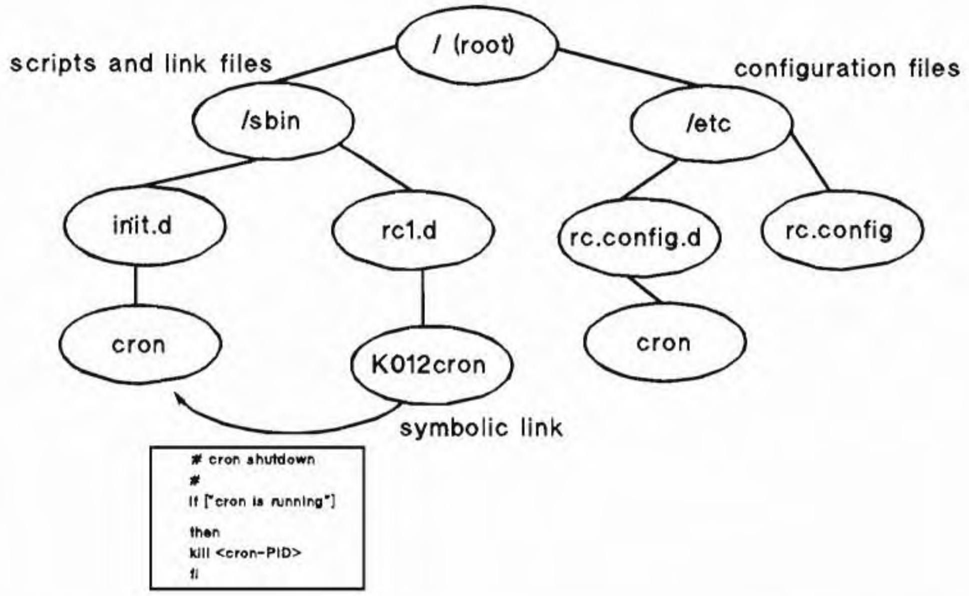
If you invoke the `shutdown` command with no option, it brings the system into single-user mode

`-h`	halts the system after it reaches single-user mode.
`-r`	reboots the system after it reaches single-user mode.
`-y`	turns off user interaction. The shutdown process is normally an interactive one, posing questions to the user and requiring a "y" or "n" answer. The `-y` option disables this interaction, allowing `shutdown` to run continuously.

grace allows the administrator to specify a grace period. The value given is the number of seconds that `shutdown` will wait before terminating all the processes running on the system. The default *grace* value is 60 seconds.

8-3. SLIDE: What Happens When Shutdown Is Executed?

What Happens When Shutdown Is Executed?



H6296 8-3 100

Student Notes

First, `shutdown` checks the user for authorization to execute the command.

Next, if there are other users on the system, `shutdown` prompts to see whether you wish to send the standard broadcast message or enter your own message. If you elect to send your own message, type the message on the terminal when prompted, press (Return) and then (Ctrl)+(d) to signify the end of the message. (If there are no users, `shutdown` will not broadcast a message.)

After the message, `shutdown`:

- Waits the specified (or default) number of seconds.
- Runs all files in `/etc/rc.config.d`. These scripts only set variables evaluated later in `/sbin/rc`.

- Runs all kill-scripts from `/sbin/rcR.d/` (where *R* is the current run-level) through `/sbin/rc0.d/`, decrementing *R* by 1 on each pass. Kill-scripts are those with names beginning with an uppercase `K` (for "Kill"). For example: `/sbin/rc1.d/K012cron`. This should stop all system processes not necessary for single user mode.
- The next step depends on which options are selected:
  - if the system is being halted or rebooted, then `/sbin/reboot` is executed to finish bringing down the system.
  - If the system is being brought down to single-user state, then the following steps occur:
    - All currently executing processes are terminated.
    - All locally mounted file systems are unmounted.

The system is rebooted or halted by executing `/sbin/reboot` if the `-h` or `-r` option was chosen. If the system was being brought down to single-user state, a signal is sent to the `init` process to change states.

Examples of `shutdown`:

This example will shut down and then halt your system with no grace period:

```
# shutdown -h 0
Do you want to send your own message? (y or n):   n
```

This example will shut down and halt the system in 5 minutes with no interactive questions and answers:

```
# shutdown -h -y 300
```

This example will shut down and then reboot your system giving the users 5 minutes grace:

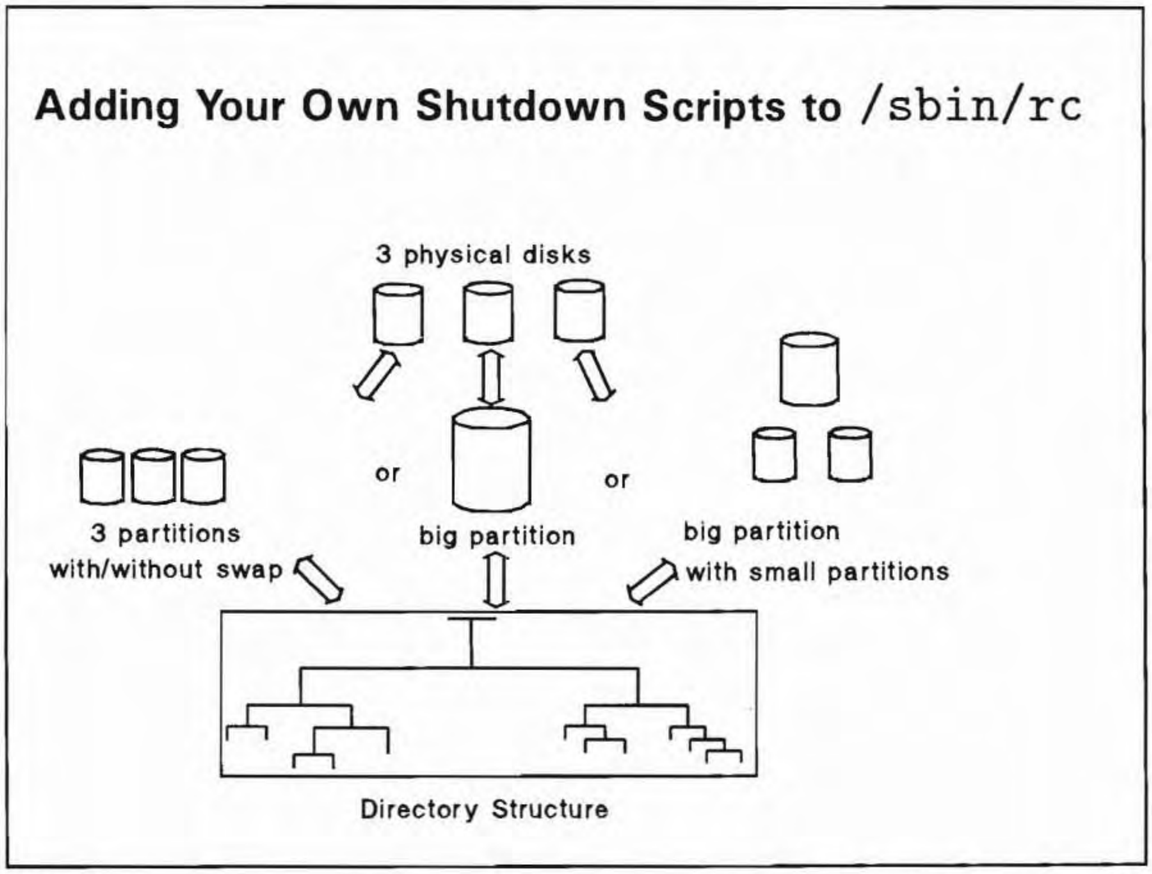
```
# shutdown -r 300                                     sample message
Do you want to send your own message? (y or n):   y
Type your message followed by ctrl d....

Please log off now.
The system will be rebooted in 5 minutes ...         End with CTRL+d
```

(... five minutes elapse ...)

```
Do you want to continue? (y or n):   y
```

(Note: Not all messages generated by `shutdown` are shown.)

8-4. SLIDE: Adding Your Own Shutdown Scripts to /sbin/rc



H6296 8-4 101

Student Notes

Customization of Shutdown

Administrators can customize their system's shutdown through the use of links to user-supplied scripts. The system administrator is free to put the necessary scripts into the directory `/sbin/init.d/` and create links to them in `/sbin/rc`*R*`.d` (where *R* is a run-level).

Scripts run during startup in a specific run-level are usually stopped in the next lower level, e.g. `cron` is started in run-level 2 and stopped in run-level 1.

The naming convention for scripts run during shutdown (and as well when changing from a higher to a lower run-level with `init`) requires that the link names in `/sbin/rc`*R*`.d` must begin with `K` (for "Kill").

```
/sbin/rc1.d/K220AppA -> /sbin/init.d/AppA
/sbin/rc1.d/K260AppD -> /sbin/init.d/AppD
/sbin/rc1.d/K290AppB -> /sbin/init.d/AppB
```

These names cause `K220AppA` to be executed first, followed by `K260AppD`, then `K290AppB`. This naming scheme leaves room for new operations to be added. For example, a file named `K270AppF` would cause the new file to be executed after `K260AppD`, but before `K290AppB`.

During shutdown, the scripts in `/sbin/rc`*R*`.d` (where *R* is a run-level) are called twice:

1. With the option `stop_msg`. (For example: `/sbin/rc1.d/K220AppA stop_msg`.) This usually invokes just an `echo` command for output of a message to console.
2. With the option `stop`. (For example: `/sbin/rc1.d/K220AppA stop`. This invokes the secure shutdown of all processes and functions started by this script during system startup.

Step by step example of adding your own script:

1. Write your script, using `start`, `stop`, `start_message` and `stop_message` in a `case` segment. At the beginning of the script, your configuration file (see below) must be "sourced" in this manner:

   `. /etc/rc.init.d/`*your_name*

2. Put the script in `/sbin/init.d/`*your_name*.
3. Create a symbolic link to this file in `/sbin/rc`*R*`.d/` (where *R* is the appropriate run-level from which the script is to be run during `startup`). Name the link starting with the key letter `S` (for "Start") and include three numbers designating the order in which this new script is to be executed during startup.
4. Create a symbolic link to this file in `/sbin/rc`*R*`.d/` (where *R* is the appropriate run-level from which the script is to be run during `shutdown`). Name the link starting with the key letter `K` (for "Kill") and include three numbers designating the order in which this new script is to be executed during shutdown.
5. If there are any configurable parameters in your script, create a configuration file in `/etc/rc.config.d/` with the same name you used in `/sbin/init.d/`. In this script, shell variables must be set which are evaluated in your script file.

8-5. SLIDE: The `reboot` Command

The `reboot` Command

Usually used to reboot or halt the system once in single-user mode

`/usr/sbin/reboot [-h|-r] [-n|-s] [-m` *mesg*`] [-t` *time*`] [q]`

`-h`	halt the system
`-r`	reboot the system automatically (default)
`-n`	no `sync` before halt or reboot
`-s`	`sync` before halt or reboot (default)
`-m` *mesg*	supply message to users
`-t` *time*	halt at specified *time*
`-q`	quick and silent

`/usr/sbin/reboot` is linked to `/sbin/reboot`

H6296 8-5 102

Student Notes

If you used `shutdown` to bring the system down with no options, then the system remains in single-user run-level allowing you to perform various tasks. Once you have performed these tasks, you might want to reboot or halt the system. The best way to do this is with the `reboot` command.

The default action of the `reboot` command is to sync the disks and reboot the system. `reboot` should always be invoked with the system in a single-user run-level. The format of the command is shown on the slide. The options are explained in more detail below.

See `reboot(1M)` for additional options to the `reboot` command.

`-h`	will shut down the system and halt the CPU.
`-r`	option will reboot the system (default).
`-n`	prevents computer sync from executing before the system is rebooted or halted. You should use this option only if directed by the system.
`-s`	invokes `sync` before rebooting or halting the system (default).
`-t` *time*	allows you to specify what time to bring the system down. *time* can be the word *now* (indicating immediate shutdown) or a future time in one of two formats: +*number* and *hour:min.*
`-m` *mesg*	displays *message* at the terminals of all users on the system at decreasing intervals as reboot *time* approaches.

8-6. SLIDE: `shutdown` and `reboot` Examples

`shutdown` and `reboot` Examples

- Shut down, perform administrative tasks, and reboot system:

  ```
  # shutdown
  ```

  ...perform tasks...

  ```
  # reboot
  ```

- To activate a newly configured kernel, shutdown with no grace period and automatically reboot:

  ```
  # shutdown -r 0
  ```

- To install an interface card, halt system giving users 5 minutes to log off:

  ```
  # shutdown -h 300
  ```

H6296 8-6 103

Student Notes

So, how do you decide whether to use `shutdown` or `reboot`? We have given you some examples on the slide. The command you use generally depends on:

- Whether users are logged in
- How quickly you need to shut down the system

Shutdown

The `shutdown` command shuts down more slowly than `reboot`, but more gracefully. It uses `kill -15` to kill running processes, which lets processes terminate naturally within a grace period. This is the safest way to shut down and it helps ensure file system integrity.

It also displays messages directing users to log off within a specified grace period. You can specify the grace period when you invoke `shutdown`. It is typically used when:

- The system is in a multi-user state.
- The system administrator is not the only person logged in and using the system.

Reboot

The `reboot` command normally shuts down all processes very quickly. It uses `kill -9` to kill any running processes. This can be dangerous (for example, cause loss of data) because `reboot` will shut down processes immediately without letting them terminate normally. It is typically used when:

- The system is in run-level `s`.
- You need to bring the system down very quickly.

8-7. LAB: Hands-On with `shutdown`

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

If You Have Your Own Lab System

1. Shut down your system immediately (0 seconds) to single-user mode using the `shutdown` command. Look at what processes are still running. Then reboot your system with the `reboot` command.

2. After rebooting in the previous exercise, use `init s` to shut down your system. Look at what processes are still running. How does this compare to the processes that were running after using the `shutdown` command in the previous exercise? Reboot your system. (`reboot` or `shutdown -r 0` will work.)

3. Customize the shutdown procedure by modifying the lab scripts and putting them in the appropriate directories. A good idea could be to `touch` a file in /tmp, showing that startup/shutdown worked correctly.

4. Set the system up so that your user name can execute the `shutdown` command.

Module 8 — System Shutdown

If You Do Not Have Your Own Lab System

1. Using the `simulate` command provided by the instructor, run the following simulation:

```
3. Shutting Down Your System
```

8-8. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. While in multi-user mode, why should the `shutdown -h` command be used, rather than `reboot -h` for halting the system?

2. What does `reboot -n` do, and why would you want to do this?

3. Once the system is in an `s` (or `S`) run-level (after execution of `shutdown`), what is the difference between `init 2` and `reboot`? Assume an `initdefault` entry of 2 in `/etc/inittab`.

4. If the System Administrator wants to customize the shutdown process, what should be done?

5. Is the system administrator the only person who can shut down the system? If not how can this capability be given to another user?

Module 9 — File System Concepts

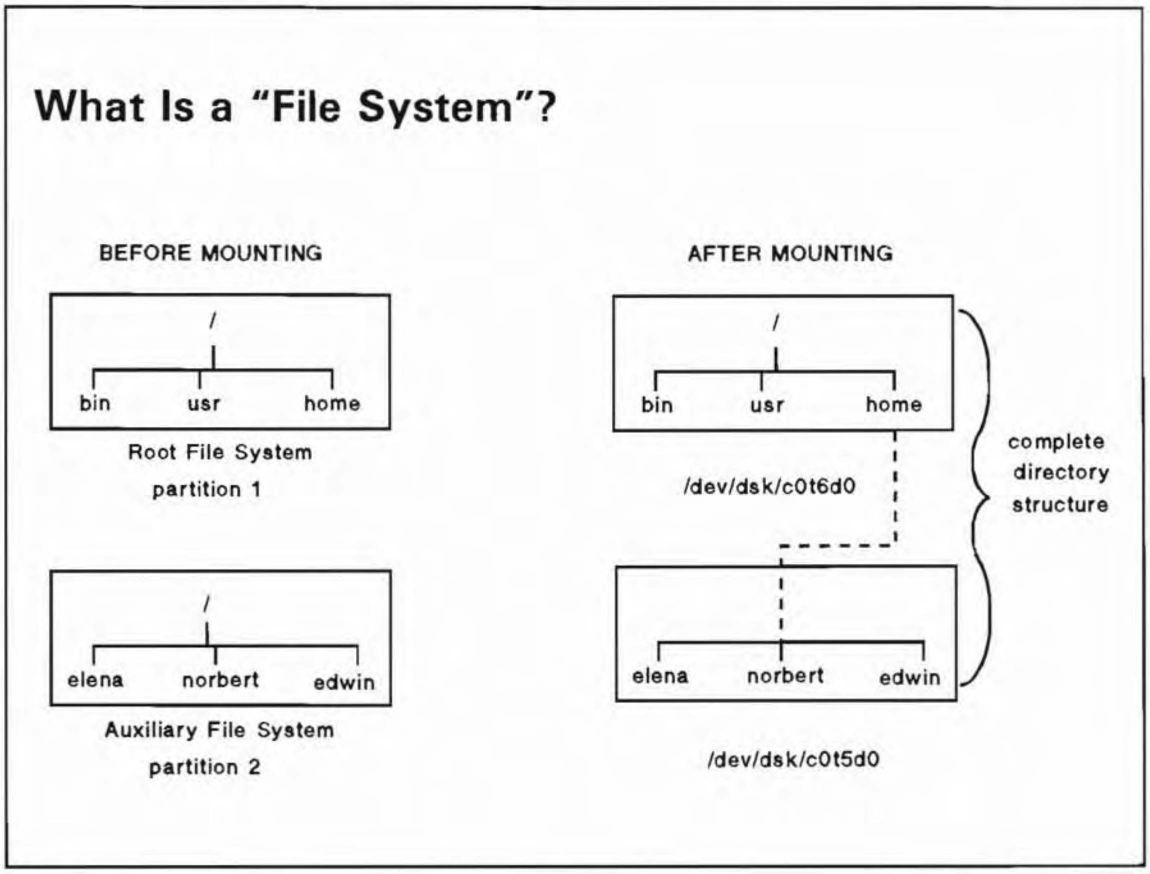
Objectives

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

- List the different types of file system available.
- List the dynamic information in an HFS superblock.
- Describe a cylinder and a cylinder group.
- List the contents of the cylinder group information.
- Describe the inode table.
- List the parts of an inode.
- Explain three different ways that an inode references a data block.
- Explain fragment and block allocation.
- Explain the link process.

Module 9 — File System Concepts

9-1. SLIDE: What Is a "File System"?



H6296 9-1 104 © 1995 Hewlett-Packard Co.

Student Notes

In one sense, the term **file system** refers to the entire hierarchical HP-UX file system tree or directory structure.

In another sense, the term **file system** refers to the specific collection of files on a partition. A partition can be either a whole disk or a logical volume (included in a volume group). This logical volume could be on one disk or on several disks, especially if you use disk striping. In LVM Striping, data is spread across multiple disks to increase performance. See `lvcreate(1m)`.

You create a new file system that consists only of a file system structure. No files are created when you create the file system. Once created, the file system encompasses the whole partition.

Module 9 — File System Concepts

To use or access the file system, you must *mount* that file system to the existing file system tree. Except for the `root` file system on the system disk, you can mount and unmount all file systems on partitions to and from the existing HP-UX file system tree.

These mountable file systems are also referred to as **auxiliary file systems**. They are referred to by the name of the device file associated with the particular partition that contains the file system. You mount the "auxiliary" file system by attaching it to a directory in the root file system. The directory to which you attach the file system is referred to as the **mount point**.

Types of File Systems

There are four types of file systems used by HP-UX:

HFS File System
HFS is an acronym for High-performance File System. HFS file systems physically reside on mass storage devices, usually hard disk drives.

NFS File System
NFS is an acronym for Network File Services. NFS file systems are remote HFS file systems, accessible over a network, that can be used in a local file system.

CD-ROM File System
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. The arrangement of files in a CD-ROM file system is tree-like, as in HFS file systems. You can use HP-UX commands to list, print, or copy files in the CD-ROM file system, but some commands (such as `mv` or `rm`) are not supported because of the read-only nature of a CD-ROM file system.

JFS File System
The HP-UX Journaled File System (JFS) 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. JFS is not available for root or bootable file systems.

Generic Syntax of Commands

Most of the commands on the file system use this syntax:

command [-F *fstype*] [-V] [*generic_options*] [-o *specific_options*] [*special_file_name* | *mount_point*]

The options and arguments to the generic commands are:

-F *fstype*
This option specifies the file system type on which `newfs` should operate.

-V
Echo the completed command line without executing the command.

generic_options
Options common to different types of file system.

-o *specific_options*
specifies the file system specific options, if any. The list must have the following format:-o followed by a space, followed by a series of *keyword*[=*value*] options separated by commas with no intervening spaces.

special_file_name|mount_point
Identifies the file system by either the mount point of the special file (raw or block, depending of the command). Some commands need both

options. However, this option is not required when you want a command to act on all file systems (optionally limited by type) of the system.

Example of the generic syntax:

To create a HFS File System:

```
/usr/sbin/newfs -F hfs /dev/vg02/rlvol2
```

To create a JFS File System:

```
/usr/sbin/newfs -F vxfs /dev/vg02/rlvol2
```

To mount a HFS File System:

```
/usr/sbin/mount -F hfs /dev/vg02/lvol2 /usr
```

To mount a JFS File System:

```
/usr/sbin/mount -F vxfs /dev/vg02/lvol2 /usr
```

To verify a HFS File System:

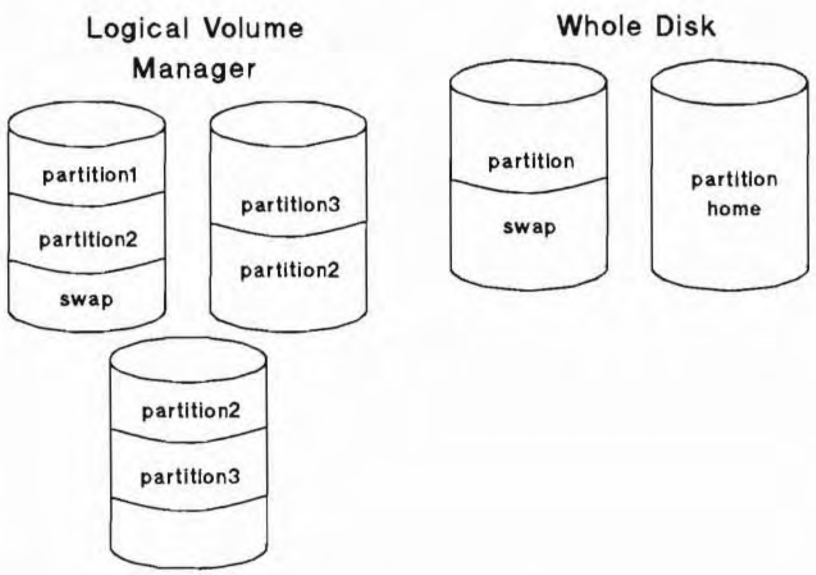
```
/usr/sbin/fsck -F hfs /dev/vg02/lvol2
```

To verify a JFS File System:

```
/usr/sbin/fsck -F vxfs /dev/vg02/lvol2
```

9-2. SLIDE: Understanding Disk Partitioning

Understanding Disk Partitioning



H6296 9-2 105

Student Notes

Before you can create a file system, you have to set aside a special area for it on the disk. This area is called a **disk partition**.

On HP-UX, a disk can be divided into areas that can accommodate file systems, raw I/O, and swap. A **root disk** will also contain a boot area. (The root disk or system disk is the disk that holds the root file system.) You can manage disk space more precisely using partitions than on a system without disk partitions.

Disk space can be partitioned with HP-UX 10.0 using two different methods:

- Whole Disk Layout
- Logical Volumes

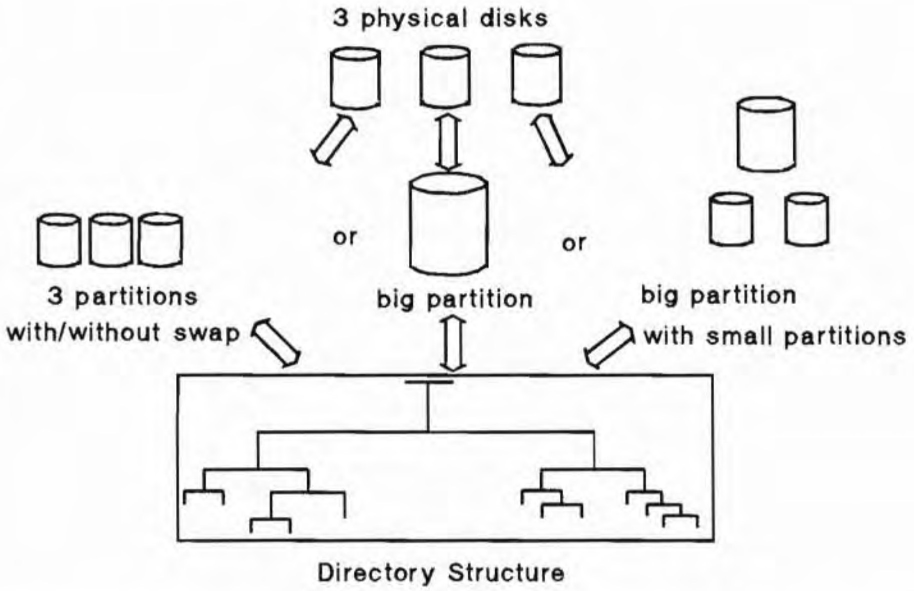
The Whole Disk Layout consists of dividing the disk into one, or two (three for a root disk) partitions:

- A file system
- A swap area (optional)

The advantages of employing a disk partitioning scheme generally outweigh the added complexity of managing such a scheme. Among the chief advantages is the ability to control the amount of disk space that can be used by a certain project or group of users. You can select appropriate sizes based on the application. Since a file system is contained within and limited to the size of the disk partition that it is created in, the file system cannot grow without bounds. By the same token, it does not make sense to use two disk partitions that overlap. If overlap were to occur, two file systems would be competing for the same disk space which would undoubtedly lead to disaster.

9-3. SLIDE: Disk, Directory Structure and Partitions

Disk, Directory Structure and Partitions



H6296 9-3 106

Student Notes

The diagram above shows how the directory structure is built from the partitions.

In this example there are three physical disks. The space on these disks may be used as three equally-sized partitions, or one big partition, or one large and two smaller partitions. No matter how the physical space is divided up, you still create file systems in partitions and mount them on directories. Depending on the type of your application you may want several small file systems, or one big file system, or a combination of both.

There is one restriction: One single file has to fit in one file system. It is not possible to split one file across two file systems. Therefore you must consider the largest file you plan to support and how much disk space you will need in each particular partition.

As you see, partitions allow you to be very flexible in the way you manage your file system disk space.

9-4. SLIDE: The HP-UX File Systems

The HP-UX File Systems

- HP-UX HFS
  - Based on 4.2 BSD UNIX file system also known as:
    High-Performance File System (hfs)
    Berkeley File System
    McKusick File System
  - Redesign of traditional AT&T file system to accommodate larger disks.
  - Same user interface as traditional AT&T file system.
- HP-UX JFS
  - The HP-UX Journaled File System, also known as:
    JFS
    Veritas File System
    VxFS File System
  - JFS-specific `fsck` reads log instead of scanning entire file system
  - Same layout as HFS

H6296 9-4 107

Student Notes

HP-UX HFS

The HP-UX file system is based on work done by Kirk McKusick at the University of California at Berkeley. The traditional UNIX file system developed by AT&T was developed when disks were small. As disk sizes grew by orders of magnitude, the initial design of the traditional UNIX began to show its age in respect to performance.

McKusick solved this performance problem by redesigning the underlying data structures that support the file system. With this approach, the user interface was unchanged so no code conversion was necessary for applications developed on the traditional UNIX file system. The simple and elegant user interface of the traditional UNIX file system coupled with the performance increases of the Berkeley group make the HFS file system a logical choice for HP-UX. HP's version of the file system has been tuned further for HP hardware.

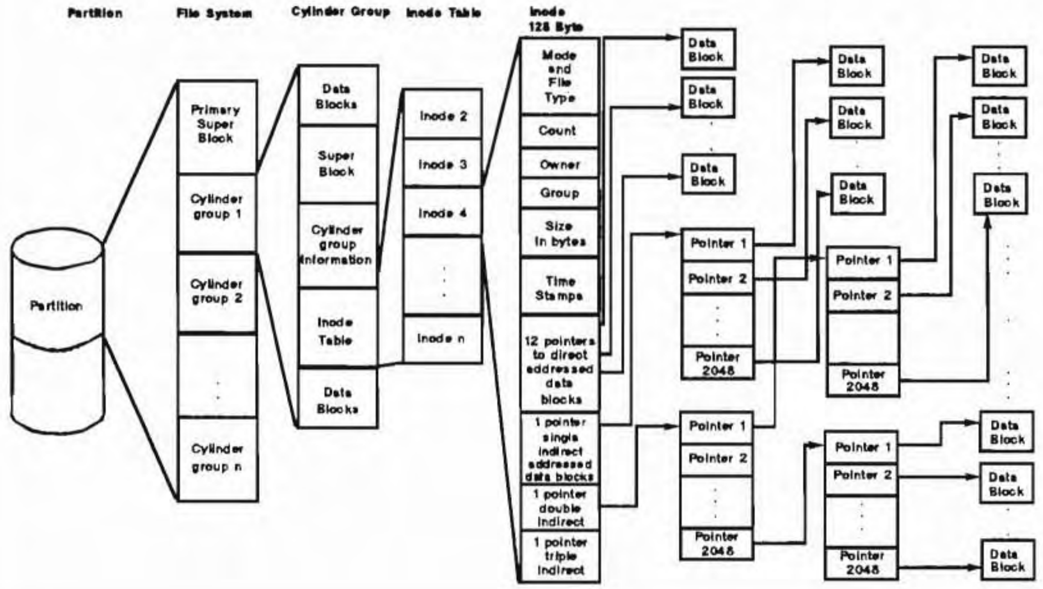
HP-UX JFS

The HP-UX Journaled File System (JFS) is an extent based journaling file system which offers fast file system recovery. The optional Data Center product also offers on-line features such as on-line backup, on-line resizing and on-line reorganization.

Note JFS is not a root or bootable file system in 10.0.

9-5. SLIDE: HFS File System Format

HFS File System Format



H6296 9-5 108

Student Notes

The HFS File System is laid out in a common format. The file system is made of these structures:

- primary superblock
- multiple cylinder groups.

The **superblock** is a contiguous 8 K block of disk space, which HP-UX uses to keep track of the current state of the file system.

Each **cylinder group** contains a copy of the primary superblock, a cylinder group information table, a part of the inode table, and data blocks.

9-6. SLIDE: The HFS Superblock

The HFS Superblock

The superblock contains:

- Summary information about file system updates
- Static information:
  File system size
  Number of inodes
  Locations of free space maps
  Number of cylinder groups
  Location of superblocks, cylinder groups, inodes, and data blocks
  Block size and total number
  Fragment size and total number
- Dynamic information:
  Total number of free data blocks
  Total number of free inodes
  File system clean flag

H6296 9-6 109

Student Notes

The HFS superblock is a contiguous 8KB block of disk space near the beginning of the file system. HP-UX uses the superblock to keep track of the current state of the file system.

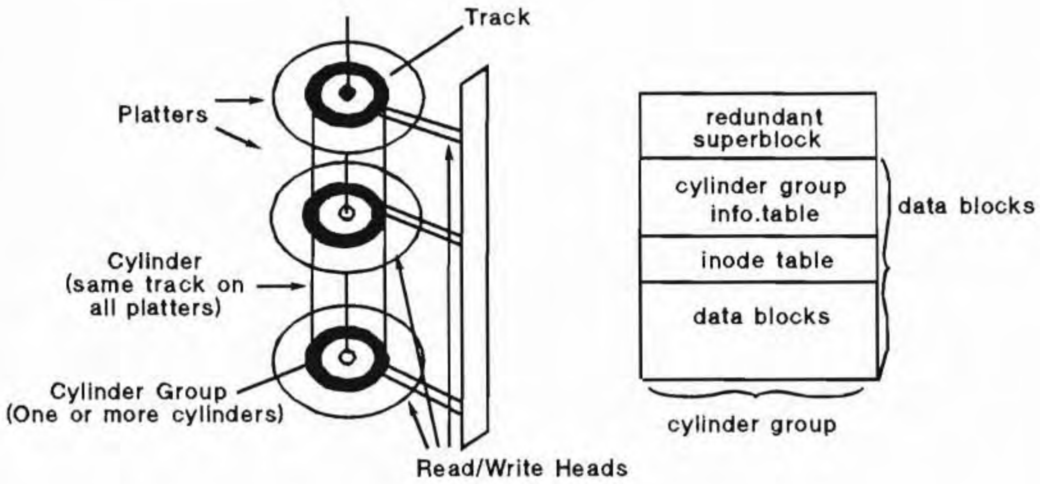
HP-UX uses information in the superblock for various file system maintenance procedures, for example, mounting a file system or performing a file system check.

Because the superblock is so important, HP-UX always keeps a copy of the superblock in main memory. The superblock on disk is updated whenever the sync command is executed. HP-UX also keeps a copy of the primary superblock's static information in each cylinder group. If the primary superblock is lost, damaged, or becomes corrupted in some way, then it can be reconstructed from any alternate superblock with a command called `fsck`.

A list of the locations of the redundant superblocks in each cylinder group can be found in a disk file created when the file system was made. This file is called `/var/adm/sbtab`. You should print a copy of this file and save it for future reference.

9-7. SLIDE: The HFS Cylinder Group

The HFS Cylinder Group



H6296 9-7 110

Student Notes

A **cylinder group** is a group of one or more disk cylinders.

A cylinder is a collection of tracks formed as the head-disk-assembly (HDA) positions all the heads on multiple platters (disk surfaces) at the same distance from the edge of the disk surfaces. The physical picture may help you visualize the concept of a "cylinder" of data being traced out by the array of heads.

Each cylinder group contains a copy of the primary superblock, a cylinder group information structure, an inode table, and data blocks. The cylinder group controls all access to a file and its associated data.

A copy of the superblock is located in each cylinder group so that any single track, cylinder, or platter can be lost without losing all copies of the superblock.

The **cylinder group information** contains the dynamic parameters of the cylinder group:

- number of inodes and data blocks
- pointers to the last used block, fragment, and inode
- number of available fragments
- used inode map
- free block map

The **inode table** contains entries for a set of inodes. Inodes contain information about individual files. The number of inodes allocated per cylinder group is determined when the file system is created and cannot be changed once the file system is made. Therefore, the default allocated by `newfs` is more than will be needed for average usage.

The **data blocks** are the actual data in a file. These are referenced by an addressing scheme in the inode, which will be discussed shortly.

Directory information is contained within normal data blocks and does not occupy a distinct region within a cylinder group. Any unused data block can be allocated for use as a directory. We will talk about directory information a little later.

Cylinder group information is updated whenever the sync command is executed.

9-8. SLIDE: Inodes in an HFS

Inodes in an HFS

- One inode for each possible file
- Each inode table entry is 128 bytes long
- A large fixed number of inodes is allocated when the file system is created
- The inodes for a particular file system are distributed among the inode tables of the cylinder groups in that file system
- An inode has a unique inode number, used to locate the cylinder group and offset into the inode table.

H6296 9-8 111

Student Notes

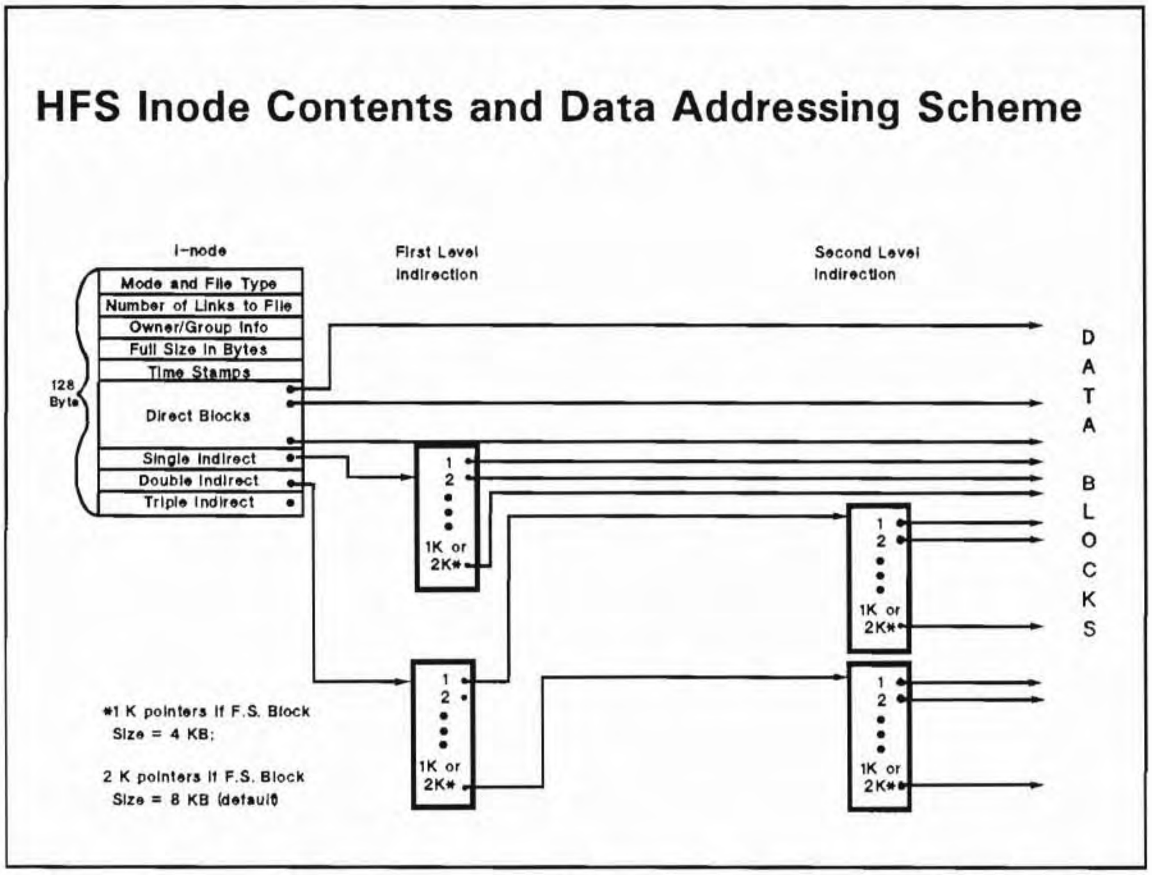
A file is accessed through information stored in its inode. The inode contains every piece of information about an HP-UX file except one. The missing piece is the file name (which is stored separately, in a directory). Everything else you would want to know about a file is in the inode. There is one inode for each existing file. Consequently, all files in a file system have an associated inode. To see the inode number for a file, use the `ls -i` *filename* command.

The inodes for each cylinder group are maintained in the inode table (within the cylinder group). Since the inode is the one place in the file system where (almost) all information about a file is stored, the inode table is accessed often.

The contiguous nature of the inode table makes accessing the table simple. Here, contiguous means that all of the information about files in a cylinder group is stored together in one place near the center of the cylinders. The inode table is physically located immediately after the cylinder group information structure on the same cylinder group.

Each inode is referred to by a unique number called the `i number` for the file it represents. The `i number` is used (by the file system code) to calculate which cylinder group contains the inode. The calculation also yields its relative position in the inode table. For instance, since an inode is 128 bytes long, the fifth inode in the list will start 512 bytes from the beginning of the inode table. Since the cylinder group information structure knows the length of the inode table, it is easy to determine when the end of the list has been reached.

9-9. SLIDE: HFS Inode Contents and Data Addressing Scheme



H6296 9-9 112

Student Notes

The inode for a regular file contains almost all the pertinent information about a particular file. The primary information contained in an inode is:

- mode or permissions of the file
- type of file (that is, regular, directory, special)
- number of hard links to the file
- current owner of the file
- group associated with the file
- actual file size in bytes (more may be allocated as the file grows)
- time stamps relating to file activity

- time/date of last file data change
- time/date of last file access
- time/date of last inode modification

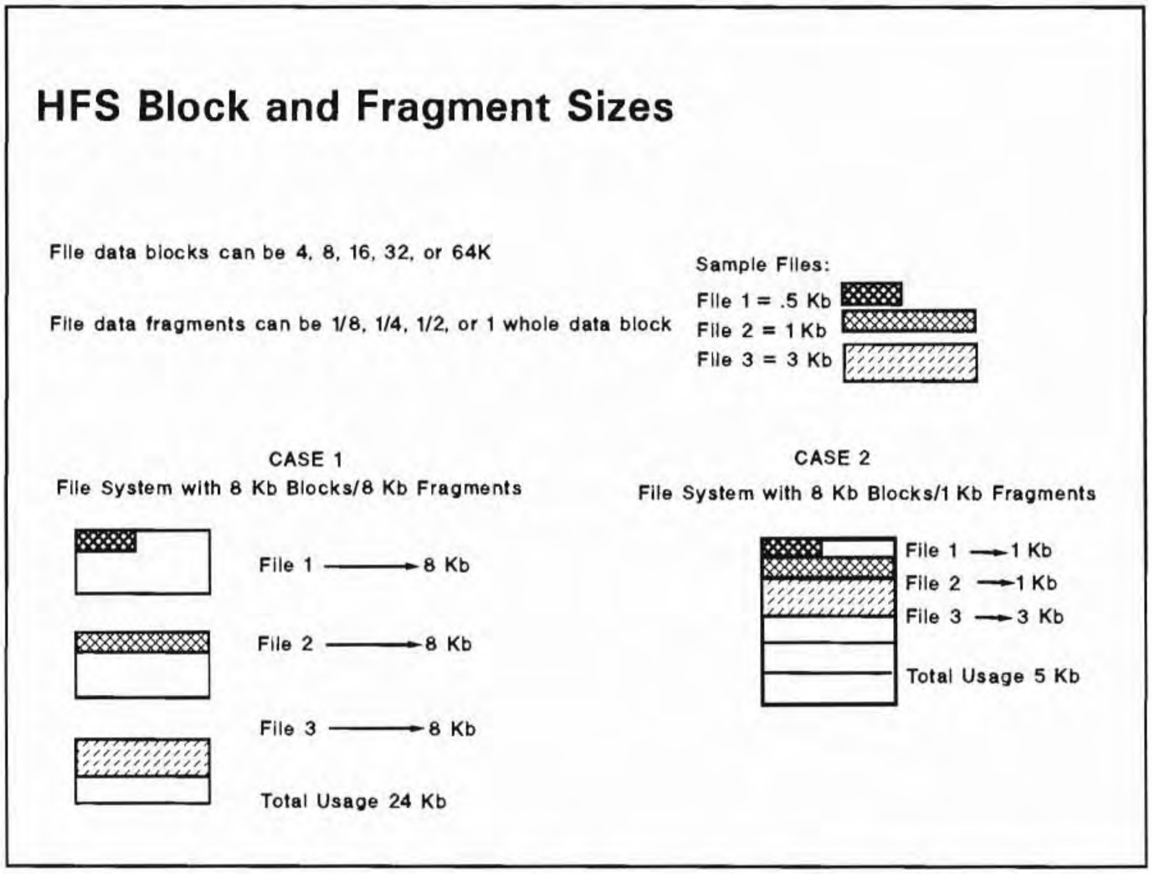
- disk addresses, or pointers to disk addresses, where the file's data is stored

Two things that are not kept in the inode are the file's name and its data.

HP-UX implements file-level security through the inode by maintaining file and directory permissions in the inode. Since the inode knows where the file's data is, all accesses to a file take place by first accessing the inode. This puts all the security data directly in-line with any file access. Note that the inode is the only thing that knows where to find a file's data.

The inode contains slots for 15 disk addresses or pointers. The addresses stored in an inode will be a fragment addresses (disk address). An entry is made in one of these slots when a file is created or when an existing file grows and occupies additional space. If the size of a file is decreased, resulting in the de-allocation of data fragments, the corresponding addresses are removed from the inode as soon as a data block is released.

9-10. SLIDE: HFS Block and Fragment Sizes



H6296 9-10 113

Student Notes

Each cylinder group contains data blocks in addition to the superblock, the cylinder group information table and the inode table. These areas are used to store the data of regular files, directories and special files. Free space in these areas is allocated in blocks. Blocks, which can be either 4 or 8 Kb in size, form the smallest physical unit that can be accessed by the HP-UX system on disk.

A data block is divided up into fragments, where the fragment size is 1/8, 1/4, 1/2 or 1/1 of the block size. The individual fragments within one block form the smallest logical unit that can be accessed by HP-UX. When allocating blocks and fragments for a file:

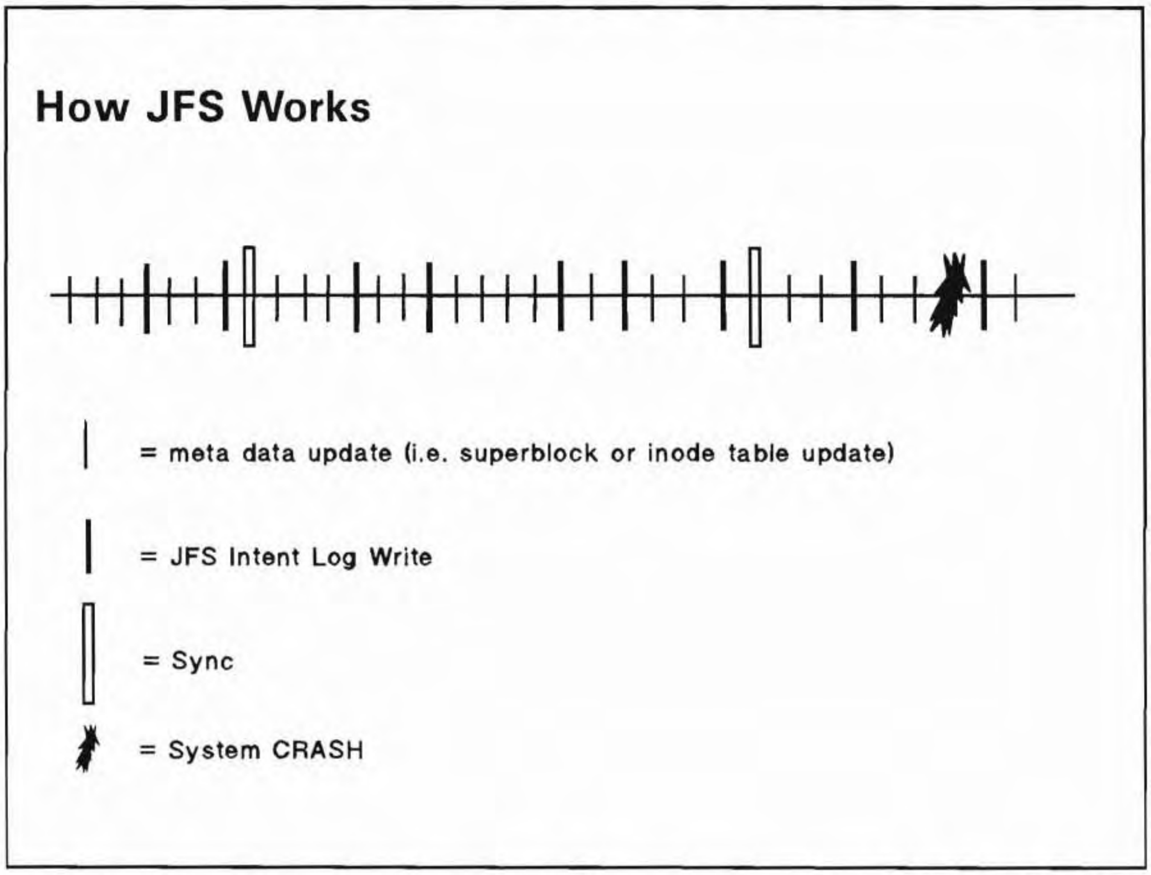
- A fragment will only be allocated at the end of a file (in other words, full blocks are allocated until less than a full block is left).
- A file will use fragments contained within one block.
- A block may contain fragments from more than one file.

- Multiple fragments can be used by a file, but only at the end of the file, and they must be contiguous within one block.

The block and the fragment sizes are specified at file system creation and cannot be modified without recreating the entire file system. Having large block or fragment sizes has both benefits and costs. If your application has large files, a large block and fragment size could significantly reduce the number of disk accesses, thereby increasing file system throughput. Eight K blocks with 8 K fragments are typically found in data base applications. Large block and fragment sizes are also typical in the file system containing the `/tmp` directory. Here, the files are typically volatile and temporary, so fast access is more important than disk space economy.

If your application uses small files (most HP-UX files are small), then a large block or fragment size wastes space. Accordingly, a combination of a 4 or 8 K block size with a 1 K fragment size is common for file systems containing directories such as `/home`.

9-11. SLIDE: How JFS Works



H6296 9-11 114

Student Notes

When data is added to a file on a JFS file system, it is grouped in extents rather than in blocks. JFS extents are contiguous file system blocks treated as a unit. The JFS extent is the incremental unit of allocation used when the file is to be extended. This allows for I/O in multiple blocks, which is much faster than block-at-a-time operations.

The JFS Superblock

The JFS superblock contains information on :

- The file system type
- Creation and modification times
- Label information
- Information about the size and layout of the file system
- Count of available resources

Copies of the superblock are kept in allocation unit headers, so the system can still be recovered, even if the primary superblock is destroyed.

9-12. SLIDE: The Intent Log

The Intent Log

Five Steps to complete a transaction:

- Update the free extent map for the directory, to increase directory size
- Change the allocated block
- Modify the directory inode to reflect the new size
- Update the free inode map

H6296 9-12 115

Student Notes

The intent log is a circular activity log. By default, it is 512 blocks in size. The intent log records the intention to perform the five steps described in the slide, so that `fsck` can read the log and reconstruct the file system. If the disk or system fails before steps have completed, then the intent log entries can be uses to reconstruct the file system. You can configure the size of the intent log when you create your file system by using the `logsize` option to the `mkfs` command.

9-13. SLIDE: Allocation Units

Allocation Units

allocation unit header

free resource summaries

free inode map

extended inode operations map

free extent map

inode list

padding

data blocks

H6296 9-13 116

Student Notes

An allocation unit is similar to an HFS "cylinder group". The number and size are be specified when the file system is created. All allocation units, except possibly the last one, will be of equal size. The last allocation unit can have a partial set of data blocks; this allows use of all available blocks on a partition. In order to align data blocks on a physical boundary, you can specify the gap to be left between the end of the inode list and the first data block.

The allocation unit header contains a copy of the primary superblock which is used to verify that the allocation unit matches the superblock of the file system. This copy of the superblock can be used if the primary superblock is destroyed.

Free resource summaries contain information about:

- Inodes with extent operations pending

- Number of free inodes
- Number of free extents within the allocation unit.

The free inode map is a bitmap that indicates which inodes are free and which are allocated.

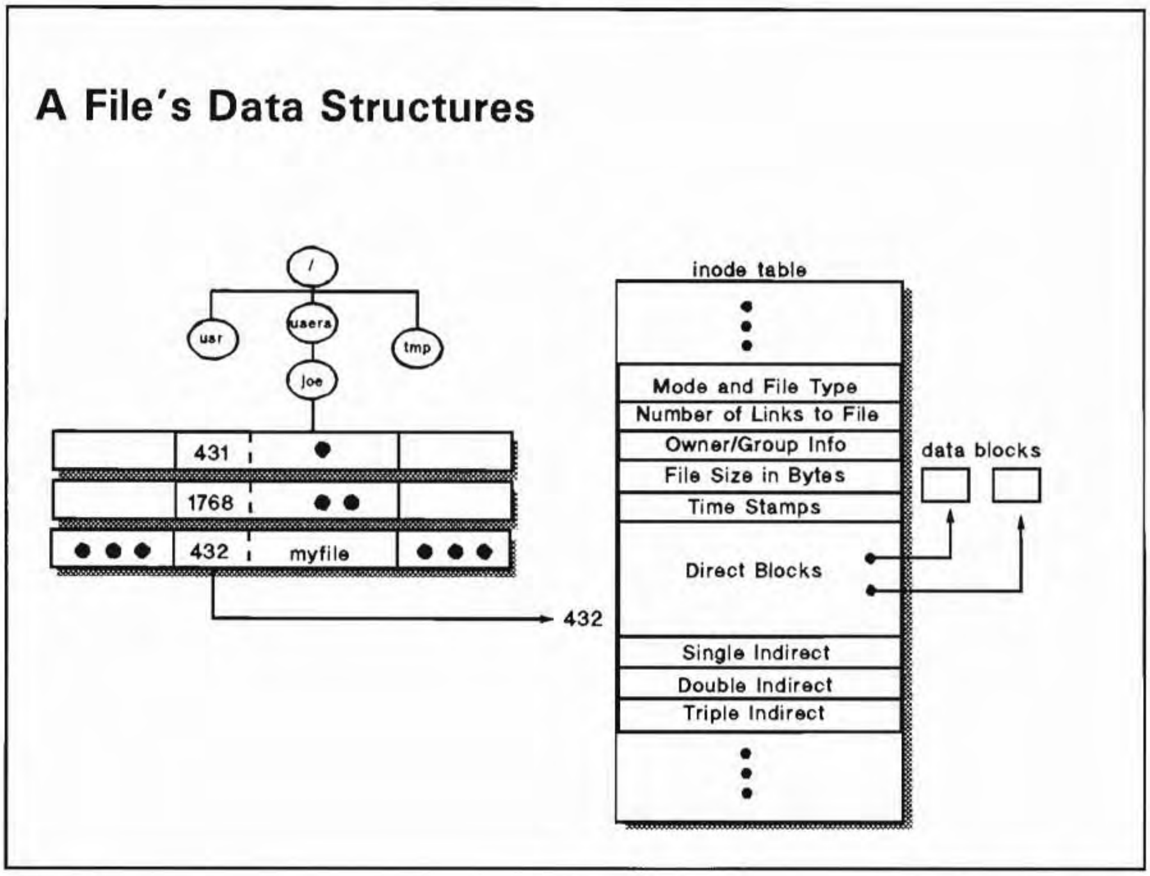
The extended inode operations map keeps track of inodes with pending operations which would remain pending for too long to reside in the intent log. This keeps the intent log from mapping. It has the same format as the free inode map, and its transactions are stored in the inode.

The free extent map one-block bitmaps are the true allocation mechanism. The remaining blocks are remapped into increasingly larger sized groups. When smaller extents are needed, large ones are broken up.

The inode list is a list of inodes for each file in file system. It is used for dynamic inode allocation. There is one inode for each file. The data for small directories, files and symbolic links is stored directly in the inode. This is also known as a direct file. The inode information includes:

- File size
- Link count
- Owner and Group IDs
- Access Privileges
- Pointers to file data.

9-14. SLIDE: A File's Data Structures



H6296 9-14 117

Student Notes

Files in an HP-UX file system are composed of three components:

Directory entry	The directory entry of a file has two pieces of information; the file's name and its inode number or `inum`. A directory itself is a file with a particular format, namely multiple directory entries. The directory entry "maps" a particular file name to an inode number. The inode number "maps" to an inode.
inode	The inode or information node contains the important information about the file, such as the information output from `ls -l`.
Data blocks	Another piece of information contained in the inode is the addresses of the data blocks. The data blocks are what actually hold the file's data.

A **directory** behaves like an ordinary file in the file system, except that no user can write into a directory. The function of a directory is to locate all files one level underneath it in the file system tree.

To accomplish this, it contains an entry for each file it must locate. The entry contains the file name and its inode number.

On HP-UX, you can have a file system that supports either short file names (up to 14 characters) or long file names (up to 255 characters). The 14-character length is the standard. The following discussion applies to the standard 14 character length.

A directory entry (also known as a directory slot) consists of four fields:

- binary inode number (4 bytes)
- length of this directory entry (2 bytes)
- length of file name (2 bytes)
- file name (up to 14 bytes with short file names and up to 255 with long file names)

In the example, there is a picture of a directory structure. If we tried to access `myfile`, the corresponding inode number 432 would be retrieved from the directory and used as a pointer to inode 432 in the inode table. The data block pointer of inode 432 would finally lead to the actual data of `myfile`.

When a directory is first created, it immediately contains two entries—dot (".") and dot dot (".."). These entries are used for relative path addressing. The entry for "." will contain the inode number for this directory itself, while the entry for ".." will contain the inode number of its parent. The only exception occurs when the directory is the top-level directory (root) of the file system. In this case, the inode number will always be 2. Since it has no parent, the number 2 is placed in its "." and ".." entries.

When a file is created, the file name and the inode number are placed in the first slot in the directory that has a 0 in the inode number field. When a file is removed, a 0 is placed in its inode number field. A 0 signifies an empty slot that is available for re-use.

We will see that there are important advantages to separating the information about a file (its inode) from the contents of the file (its data blocks) and its name (its directory entry). This architecture is the basis for several very nice capabilities for file management.

9-15. SLIDE: HP-UX Hard Links

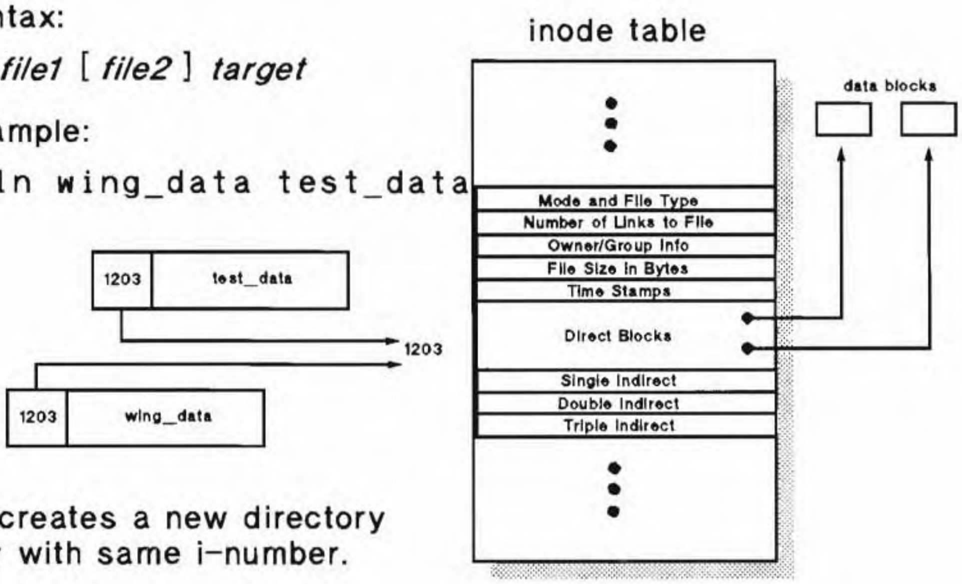
HP-UX Hard Links

- Syntax:

  `ln` *file1* [*file2*] *target*

- Example:

  `# ln wing_data test_data`



This creates a new directory entry with same i-number.

H6296 9-15 118

Student Notes

- Links are different directory entries indicating the same file.
- Links are sometimes called "hard" links.
- Links should be used when a file needs two or more different names.
- Example `ln` command:

```
# ln /users/projectA/src/control/test/data  /users/bob/data
```

Hard links can be used to link files that do not cross file system boundaries. It is quite useful to be able to link two files together.

Having two different directory entries point to the same file can be a bit confusing, but it is widely used in HP-UX. The `-l` option to the `ls` command is used to see if there are links to a file. If the number in the second field is greater than `1`, there are other "names" for that file.

In the example on the slide, a user application requires that a file be named two different names in the same directory. The file `wing_data` must exist for the command to work. The figure illustrates the state of the file system after the link command finishes.

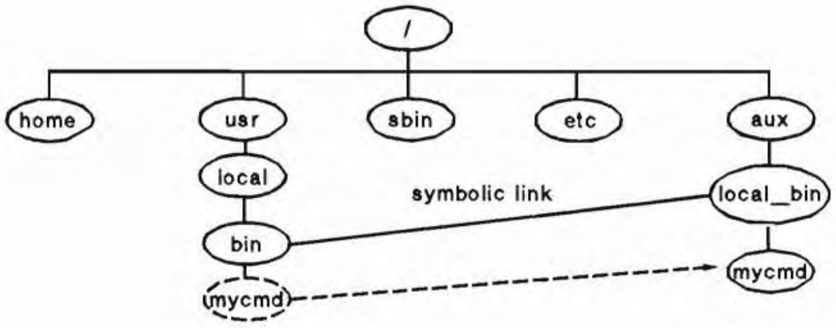
When the command is executed it creates a directory entry that has the new file name (the target name) associated with the same i-number as the original file. Both directory entries are then said to "point to the same file". A field in the inode is incremented to indicate the presence of the link. Only after the link count is reduced to zero does the actual data and inode get recycled.

9-16. SLIDE: Symbolic Links

Symbolic Links

- Link directories or files across file systems
- Should be used only when absolutely necessary
- Example:

```
# ln -s /aux/local_bin /usr/local/bin
```



H6296 9-16 119

Student Notes

If necessary, symbolic links can be used to link files (or directories) across file system boundaries. It is quite powerful to be able to link two directories together. This allows directories with a specific path name to be physically put on any file system. Linking directories together is helpful on a system if a partition is not large enough to hold an entire directory structure. Part of the directory structure can then be put on another file system.

The syntax of the `ln` command is:

`ln -s` *file1* `[` *file2* `]` *target*

Since having two different directories point to the same files can be a bit confusing, it is recommended that symbolic links be used sparingly.

In the example on the slide, a user application requires that several large files be put in the /usr/local/bin directory. The system administrator has determined that the file system that holds /usr/local/bin does not have enough available disk space to hold these files. Therefore the administrator decides to create a symbolic link from the existing /usr/local/bin directory to another directory on another partition with more space.

The symbolic link to a directory will be interpreted by the file system code so that files created in the /usr/local/bin directory will actually reside in the /aux/local_bin directory. Since the available disk space in the file system where /aux/local_bin is located is much larger than that of the file system where /usr/local/bin is located, the original problem of filling the original file system can be avoided.

If the directory /usr/local/bin does not exist, the problem is simple, and the following command will create the symbolic link:

```
# ln -s /aux/local_bin /usr/local/bin
```

This command creates a symbolic link named /usr/local/bin and links it to the directory /aux/local_bin. This means that any files written to the symbolically linked directory name /usr/local/bin will physically be put under the directory /aux/local_bin. However the symbolic link between the two directories will allow logical access from either directory structure. For example:

```
$ cp mycmd /usr/local/bin
```

will create a file /aux/local_bin/mycmd. However due to the symbolic link, a directory listing of either /usr/local/bin or /aux/local_bin will show the existence of the new file.

Note

If the directory /usr/local/bin does exist, the problem is not as simple. The command above will create a symbolic link inside the directory /usr/local/bin named /usr/local/bin/local_bin which points to the directory /aux/local_bin (whether the /aux/local_bin exists or not).

If we assume that the /usr/local/bin directory exists and contains files and the /aux/local_bin directory does *not* exist, the steps we must follow (in this example) to correctly create the symbolic link are:

1. Create the directory /aux/local_bin and change ownership and mode to match original /usr/local/bin attributes:

   ```
   # mkdir /aux/local_bin
   # chmod 775 /aux/local_bin
   # chgrp bin /aux/local_bin
   # chown bin /aux/local_bin
   ```

2. If you have NO DIRECTORIES in /usr/local/bin, move all the existing files to /aux/local_bin using the command:

   ```
   # mv /usr/local/bin/*  /aux/local_bin
   ```

3. If you get an error such as:

```
mv:  can't mv directories across file systems
```

you did have subdirectories that must be moved with another method such as:

```
# cd /usr/local/bin
# find . | cpio -pdvulm /aux/local_bin
```

After verifying everything else has been moved, you now may safely remove the directory /usr/local/bin and its contents.

```
# rm -r /usr/local/bin
```

4. Now you can safely create the link between directories:

```
# ln -s /aux/local_bin /usr/local/bin
```

and verify the contents using either path name.

9-17. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. List the types of file systems available on an HP-UX system.

2. List the dynamic information in an HFS superblock.

3. Describe a cylinder and a cylinder group.

4. List the contents of the cylinder group information.

5. Describe the inode table.

6. List the parts of an inode.

7. Explain three different ways that an inode references a data block.

8. Explain fragment and block allocation.

9. Explain the link process.

Module 10 — Managing Disk Space

Objectives

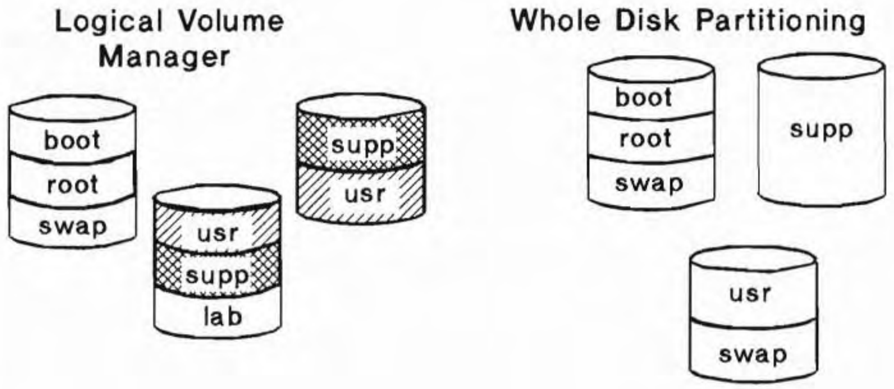
Upon completion of this module, you will be able to do the following:

- Describe disk partitioning and how it relates to the "sectioning" method of disk management.
- Describe the general features and benefits of LVM.
- Differentiate among volume groups, logical volumes, and physical volumes, and explain how they relate to each other.
- Create physical volumes, volume groups and logical volumes.

Module 10 — Managing Disk Space

10-1. SLIDE: Disk Partitioning versus Logical Volume Management

Disk Partitioning versus Logical Volume Management



H6296 10-1 120

Student Notes

Before you can create a file system, you have to set aside a special area for it on the disk. This area may be either a **disk partition** or a **logical volume**.

A disk partition or a logical volume may contain:

- Boot area
- File system
- Swap area
- Raw I/O

You have two different methods to manage your disk space:

Module 10 — Managing Disk Space

- Using the whole disk for File System/Swap
- Using the Logical Volume Manager

Partitioning with "Whole Disks" and "Sections"

Prior to the development of Logical Volume Management, disk space could be managed only in large "chunks." For example, on the HP 9000 workstations prior to HP-UX 10.0, an entire disk was defined as a single unit, and it could not be apportioned into smaller parts. This is known as the "whole-disk" method of space management; a partition basically corresponded to a whole disk.

In earlier releases of HP-UX for HP 9000 servers, whole disks could be subdivided into smaller units of predefined and fixed size. These smaller units were called "sections." In a disk that is managed by sectioning, the size of any partition must correspond to the size of one predefined section of a disk.

Also, in both the whole-disk and sectioning arrangements, a partition may not exceed the size of the whole disk on which it resides.

The fixed-size nature of both whole-disk and sectioning management schemes complicates the process of managing disk space, since user needs for contiguous space do not always fit readily into the "chunks" available.

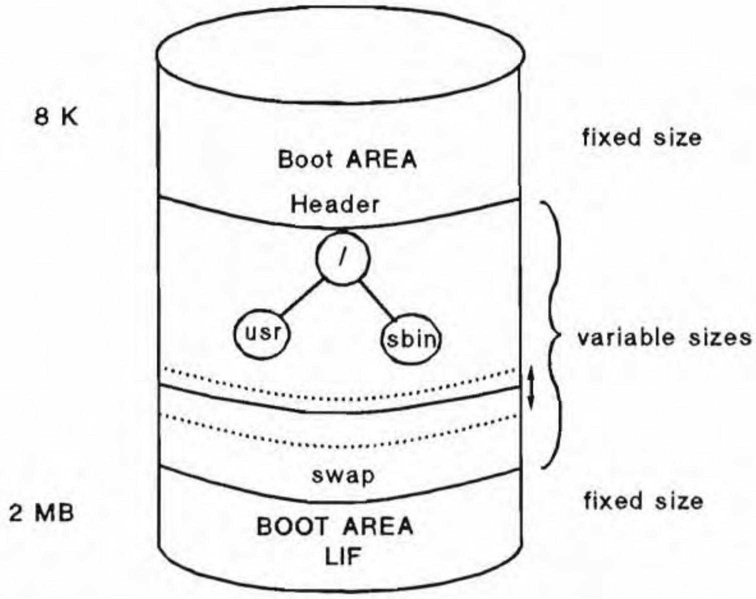
The Logical Volume Manager

Logical volumes are free of the sizing constraints of the old methods. You can manage disk space very precisely by using logical volumes. Unlike fixed-size sections, the size of logical volumes may be expanded and contracted at will.

LVM integrates **Software Disk Striping** (SDS). SDS was formerly available only on workstations, and is the capability to spread data across multiple disks to increase performance.

10-2. Whole Disk Layout

Whole Disk Layout



H6296 10-2 121

Student Notes

A **root disk** might have up to 4 partitions.

Boot Area Header	The Header (Pointer) to the actual boot files at the very end of the disk (System Disk only).
File system	The disk space where the files are located.
Swap	The Swap space is used by the **Memory Management System**
Boot Area Files	These files needed during boot up to load the Kernel.

During the installation process you will be asked to supply the size of the file system space.

When creating a disk that is *not* a root disk, you will specify the size of the file system, and eventually the remaining space (if any) may be used for swap.

10-3. SLIDE: What happens if . . . ?

What happens if . . . ?

What happens if . . .

- you need a partition which is exactly 250 Mbytes?
- you need a partition which is larger than a disk?
- you get a "file system full" message?

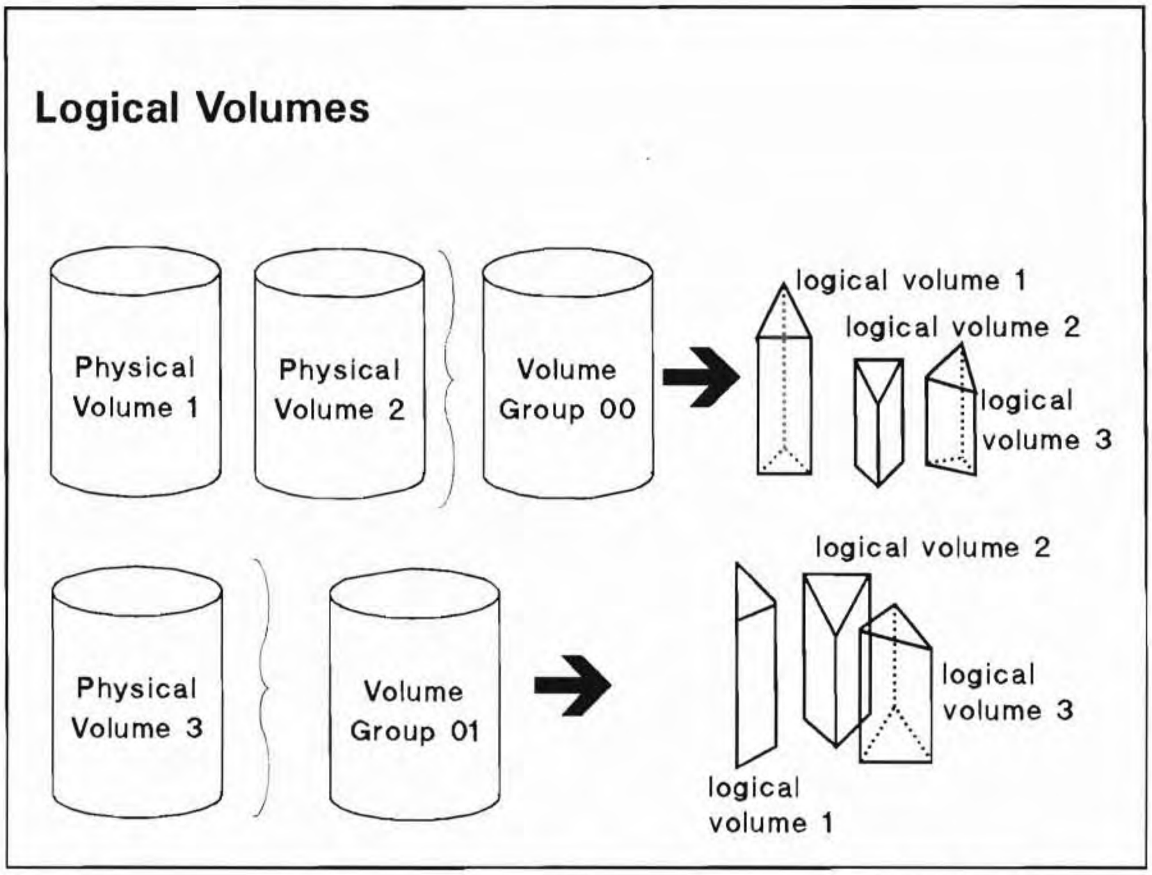
H6296 10-3 122

Student Notes

The system administrator often has to deal with one the situations shown on the slide. Without special file system architecture and management software,such tasks are very difficult or even impossible. For example, under a "sectioning" scheme, it is impossible to create a partition which is larger than the physical disk.

With the Logical Volume Manager, the system administrator can accomplish such tasks very easily.

10-4. SLIDE: Logical Volumes



H6296 10-4 123

Student Notes

The Logical Volume Manager (LVM) enables you to create sets of disks (**physical volumes**) called **volume groups**. You can then subdivide the space within a volume group into logical volumes.

Physical Volume

A physical volume means the *whole disk*.

Note

You can only manage whole disks with the LVM system. It is not possible to manage some sections of a disk as logical volumes and other sections of it as partitions.

Because different models of disks vary greatly in their physical and logical characteristics, some special information about any given disk is needed by the LVM subsystem.

Physical Volumes Device File Names

LVM physical volumes are identified by the device file names `/dev/dsk/`*cntmdp*, `/dev/rdsk/`*cntmdp*. Note that each disk has a *block* device file and a *character* device file.

Volume Groups

An LVM system consists of groupings of disks initialized for LVM and organized into volume groups. A volume group might consist of one or many LVM disks (physical volumes). Your entire system might consist of one or several volume groups.

Volume Group Device File Names

Each volume group has one directory in the `/dev/` directory which contains all device files for the logical volume in that volume group.

Volume groups have names that follow the pattern:

`/dev/vg`*nn*

where *nn* starts at 00 and increments by 1 in the order that volume groups are created.

When you create a volume group, the default volume group name is assigned, but it is possible to assign different names.

You have to create a directory `vg`*nn* in the `/dev` directory when using the manual method. If you are using SAM, SAM will do it for you.

Logical Volumes

Just as volume groups are groupings of one or more LVM disks, volume groups are also subdivided into logical volumes. Logical volumes can:

- encompass all or any portion of the space on an LVM disk
- span LVM disks

You apportion disk space in a volume group by creating logical volumes. The size of a logical volume is determined by its number of extents. You then assign file systems, swap, and dump to logical volumes.

A logical volume can contain a file system or swap space. You can choose the size of a logical volume, and you can increase the size of an existing logical volume, whereas you cannot change the size of the file system/swap on a whole disk under the outdated "sectioning" scheme.

Another advantage of logical volumes is that you can extend the size of a logical volume to be greater than one disk. This is referred to as *spanning disks*.

You can use HP-UX commands or SAM to create a file system in a logical volume of a specified size, and then mount the file system. At any time, you can extend this logical volume to allocate sufficient space for an expanded file system, swap or raw data.

Logical Volume Device File Names

Logical volumes are identified by their device file names, which take this form:

`/dev/vg`*nn*`/[r]lvol`*m*

By default, the number *m* starts at 1 and increments in the order that logical volumes are created within each volume group. LVM creates both *block* and *character device* files and places them in the appropriate volume group directory.

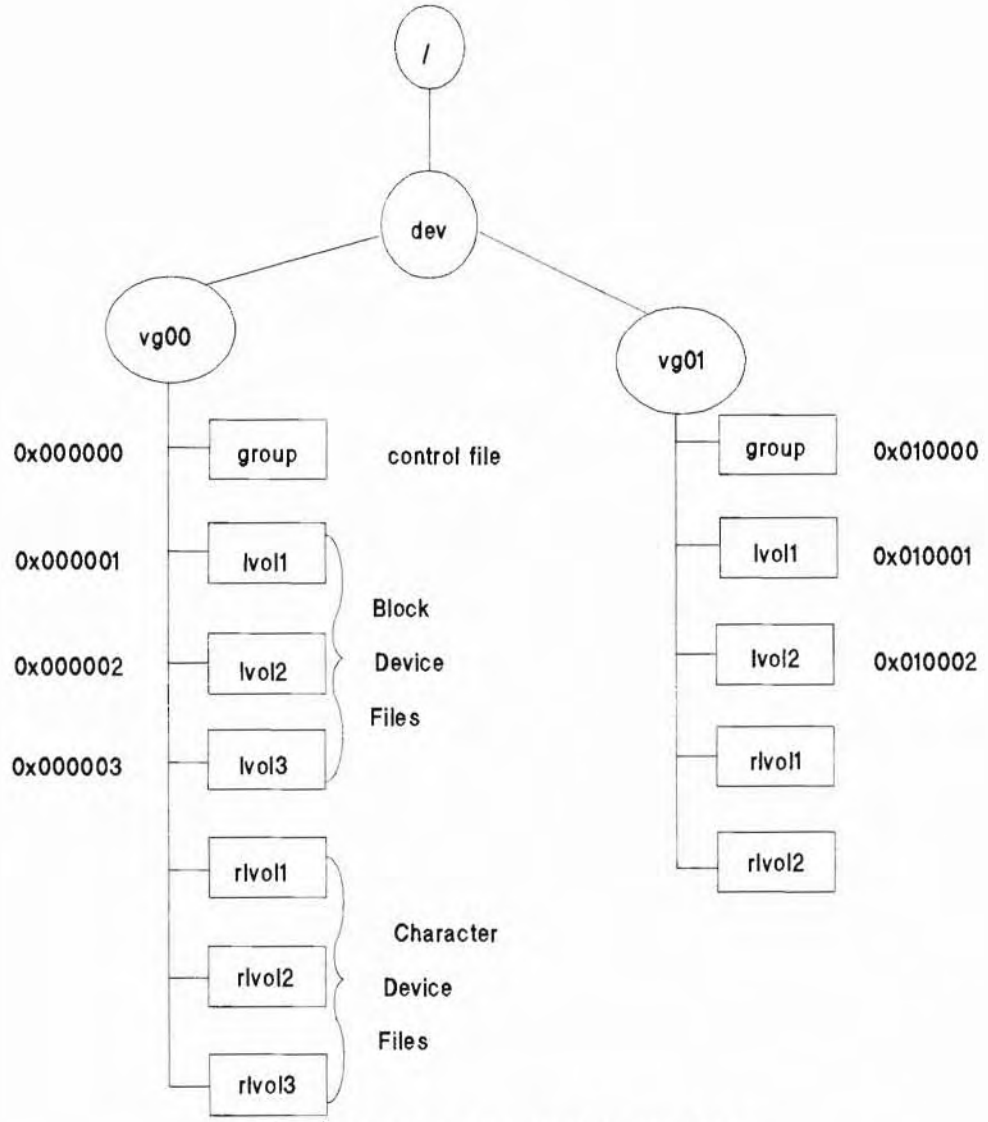
You can also assign a name to a logical volume when you create it. For example, you might want to create a logical volume to hold raw data for a database. You might name this volume `/dev/vg01/database_lv`. This logical volume would have two device files:

```
/dev/vg01/database_lv
/dev/vg01/rdatabase_lv
```

Device File Naming Conventions Summary

There are naming conventions that apply to physical volumes, volume groups, and logical volumes as explained above.

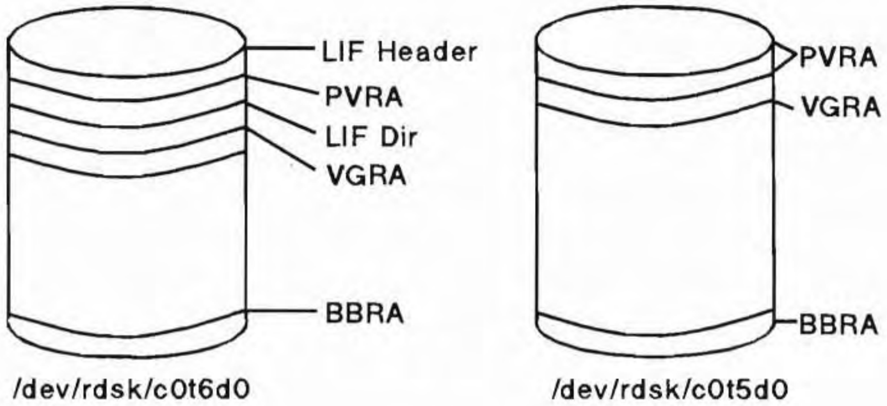
The following figure shows the standard device files for the **root volume group** (vg00) and a second volume group called vg01.



Device file naming convention for LVM

10-5. SLIDE: Creating Physical Volumes

Creating Physical Volumes




```
pvcreate - B  /dev/rdsk/c0t6d0
pvcreate      /dev/rdsk/c0t5d0
```

H6296 10-5 124

Student Notes

The space on LVM disks must be allocated for logical volumes either by using SAM or the LVM command `pvcreate(1M)`.

Before you can create and use logical volumes, you must first designate the disks as LVM **physical volumes** and then create **volume groups**. Once the volume groups are created, you can then create the logical volumes on them.

The first step is creating LVM **physical volumes**.

This procedure is done automatically when you create a volume group using SAM.

The disk you want to use as a physical volume must be installed and turned on. You use the *raw* device file name of the disk. If you are going to use the disk as a system disk, specify a boot section with the `-B` flag.

Options for the `pvcreate` command are:

`-b`	Allows you to specify the numbers that correspond to the indexes of all bad blocks
`-f`	Force creation of the physical volume even if there is already a file system present, without requesting a confirmation
`-B`	Make a bootable disk. Reserves space on the physical volume for boot related data.
`-t` *disk_type*	Get configuration information about the physical volume from `/etc/disktab`. Only necessary to specify if `pvcreate` can't get the information from the disk driver.
`-d` *soft_defects*	Specify minimum number of bad blocks that LVM should reserve in order to perform software bad block relocation. (Not supported on HP-IB disks).
physical_vol_path	The path name of the raw device to be used.

Physical volumes use the same device special files as traditional HP-UX disk devices.

Bootable and Non-Bootable Volumes

LVM stores information in data structures at the beginning of the Physical Volume. These structures contain the LVM information for that Physical Volume, Boot data, Volume Group information, and other information necessary to construct the Logical Volume layout. LVM disks are either bootable or non-bootable.

Non-Boot Disk

The non-Boot disk contains the following information:

- **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**, contains information specific to the bad block recovery mechanism.

The Boot Disk

The Boot disk contains the following information:

- **LIF header** contains the LIF header information. This area will also be present on Physical Volumes containing dump or swap devices. Created by `mkboot(1M)`.
- PVRA (same as for non-boot disk)
- **LIF Directory Area**, containing HPUXboot, `LABEL` files and other LIF files. Created by `mkboot(1M)`. `LABEL` files are created by `lvlnboot(1M)`.
- VGRA (same as for non-boot disk)
- User Data. The root File system must be the first logical volume.
- **Bad Block Relocation Area** (same as for non-boot disk)

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.

The `pvdisplay` Command

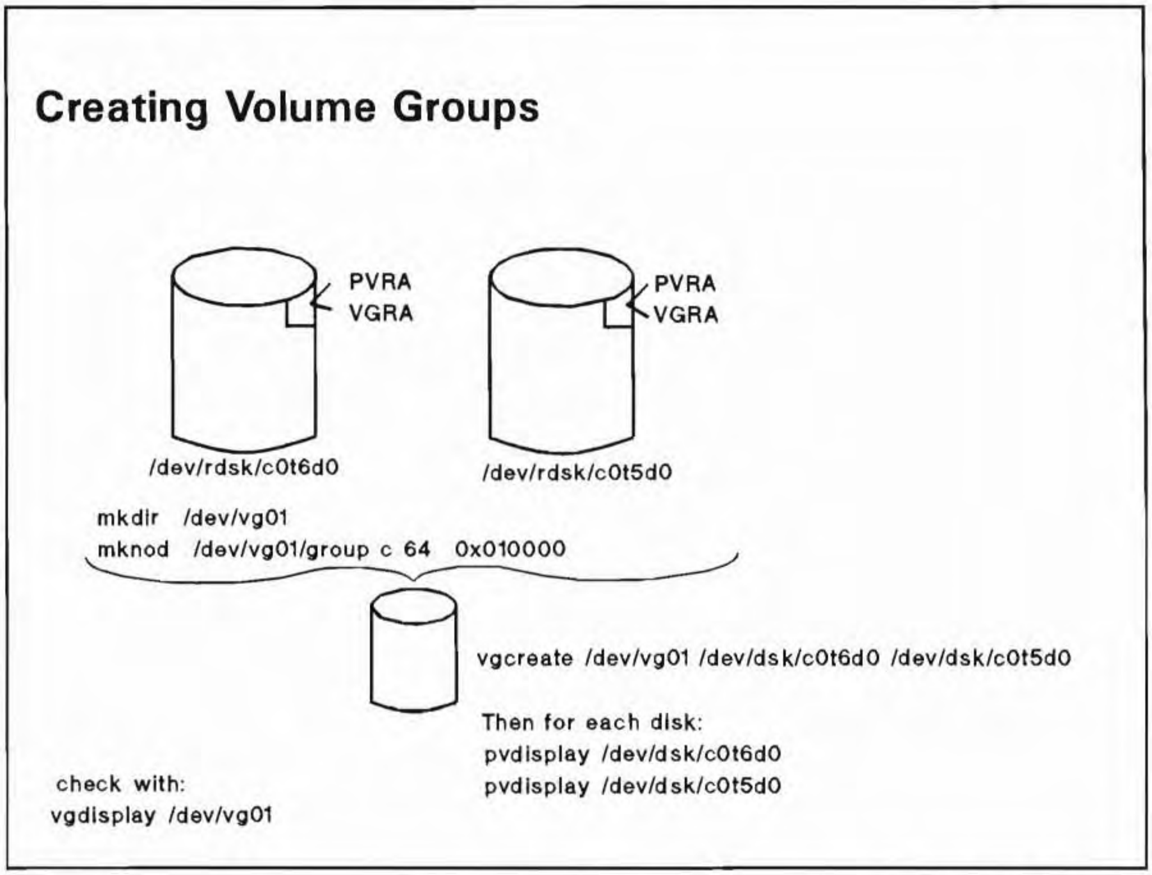
You can view the characteristics of an initialized LVM disk by using the `pvdisplay` command. This command is available for a disk *only if the disk is integrated into a volume group.*

`pvdisplay` displays information about the physical volume or volumes specified by the *physical_vol_path* parameter. If the `-v` (verbose) option is specified, `pvdisplay` displays a map of the logical extents that correspond to the physical extents of each physical volume.

```
$ pvdisplay /dev/dsk/c0t6d0

--- Physical volumes ---
PV Name    /dev/dsk/c0t6d0
VG Name                              /dev/vg00
PV Status                            available
Allocatable                          yes
VGDA                                 2
Cur LV                               4
PE Size (MB)                         4
Total PE                             632
Free PE                              462
Allocated PE                         170
Stale PE                             0
```

PV Name	Name of the physical volume
VG Name	Name of the volume group
PV Status	State of the physical volume: available or unavailable
Allocatable	Allocation permission for the physical volume
VGDA	Number of volume group descriptors on the physical volume
Cur LV (MB)	Number of logical volumes using the physical volume
PE Size	Size of physical extents on the volume
Total PE	Total number of physical extents on the physical volume
Free PE	Number of free physical extents on the physical volume
Allocated PE	Number of physical extents on the physical volume that are allocated to logical volumes
Stale PE	Number of physical extents on the physical volume that are not current

10-6. SLIDE: Creating Volume Groups



H6296 10-6 125

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 on them.

Step 1

Create a directory for the volume group. Use the naming convention `/dev/vg`*nn*, 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.

Create the control file named group in the directory `/dev/vg`*nn*. 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 form:

`0x`*hh*`0000`

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
```

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/vgnn /dev/dsk/cntmdp [/dev/dsk/cntmdp]
...
```

Remember to use the naming convention `vg`*nn* to name the volume group. Note that you are using the *block* device file to create the volume group.

`-e` *max_physical_extents*	Sets the maximum number of physical extents allocatable for LVM disks 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 32)
`-s` *physical_extent_size*	Sets the size, in megabytes, for each physical extent in a volume group (default 4).

Step 4

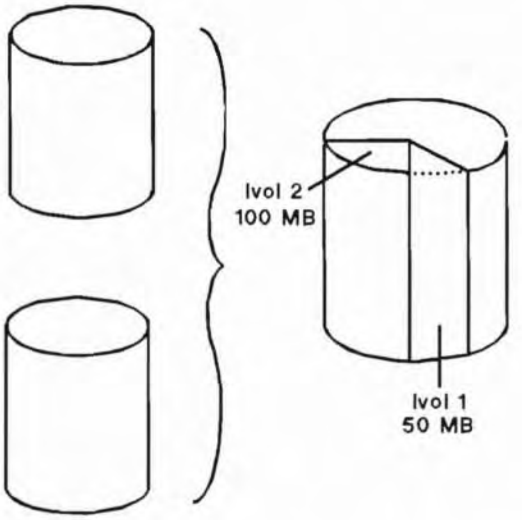
You can verify that you have created the volume group by using the `vgdisplay` command:

```
$ vgdisplay /dev/vg01

--- Volume groups ---
VG Name                 /dev/vg01
VG Write Access         read/write
VG Status               available
Max LV                  255
Cur LV                  4
Open LV                 4
Max PV                  16
Cur PV                  4
Act PV                  4
Max PE per PV           1016
VGDA                    8
PE Size (Mbytes)        4
Total PE                1754
Alloc PE                0
Free PE                 1754
Total PVG               0

$ vgdisplay -v /dev/vg01

--- Volume groups ---
VG Name                 /dev/vg01
VG Write Access         read/write
VG Status               available
Max LV                  255
Cur LV                  4
Open LV                 4
Max PV                  16
Cur PV                  4
Act PV                  4
Max PE per PV           1016
VGDA                    8
PE Size (Mbytes)        4
Total PE                1754
Alloc PE                0
Free PE                 1754
Total PVG               0
```

```
--- Physical volumes ---
PV Name                 /dev/dsk/c0t1d0
PV Status               available
Total PE                477
Free PE                 477

PV Name                 /dev/dsk/c0t2d0
PV Status               available
Total PE                477
Free PE                 477

PV Name                 /dev/dsk/c1t0d0
PV Status               available
Total PE                477
Free PE                 477

PV Name                 /dev/dsk/c1t3d0
PV Status               available
Total PE                323
Free PE                 323
```

10-7. SLIDE: Creating Logical Volumes

Creating Logical Volumes



lvcreate -L 50 /dev/vg01
lvcreate -L 100 /dev/vg01

check with:
lvdisplay /dev/vg01/lvol1

H6296 10-7 126

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 an LVM unit of measure called an **extent**.

LVM extents are 4 Mb unless you specify otherwise. Extents can range in size from 1 Mb to 256 Mb, and the size chosen must be in a power of 2. Large extents can be more wasteful of disk space, whereas smaller extents allow a finer granularity in disk allocation.

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 size. The default is no size.
`-l` *logical_extents_number*	The number of logical extents in the logical volume. The default is zero.
`-p` *permission*	Read-write access permissions of the logical volume. The default is `w`.
`-n` *name*	A custom name you want to assign to the logical volume. The default name follows the naming convention. Also called Logical Volume Path.
`-r` *Relocate*	The bad block relocation policy. In effect by default. `-r n` turns it off.
`-C` *contiguous*	Contiguous allocation policy. By default, not contiguous. `-C y` turns it on. Contiguous allocation means that physical extents of a logical volume are allocated in an unbroken, ordered sequence on one physical volume. The default is `n`. Logical volumes for `root` or primary swap *must* specify `-C y` and `-r n`.
`-i` *Stripes* `-I` *StripeSize*	*Stripes*: Number of physical volume, where the PE are located (striping). *StripeSize*: Size of stripe size. Valid stripe sizes are 4Kb, 8Kb, 16Kb, 32Kb and 64Kb. The default stripe size is 16Kb.

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 Mbyte.

```
lvcreate -L 100 /dev/vg01
```

Displaying Information About Logical Volumes

Use the `lvdisplay(1M)` command to see a logical volume. Give the path name of the logical volume as the argument.

```
$ lvdisplay /dev/vg00/lvol1

--- Logical volumes ---
LV Name                     /dev/vg00/lvol1
VG Name                     /dev/vg00
LV Permission               read/write
LV Status                   available/syncd
Mirror copies               0
Consistency Recovery        MWC
Schedule                    parallel
LV Size (Mbytes)            48
Current LE                  10
Allocated PE                10
Stripes                     0
Stripe Size (Kbytes)        0
Bad block                   on
Allocation                  strict
```

The information about mirror copies, consistency recovery, and schedule pertains to mirroring capabilities provided by the optional product HP MirrorDisk/UX.

10-8. SLIDE: What's Next?

What's Next?

Using Partitions or Logical Volumes to:

- create and mount file systems
- add swap area(s)

Manage Logical Volumes to:

- extend, reduce and remove a VG
- extend, reduce and remove a LV
- change a volume group availability
- move physical extents from one physical volume to another
- display information of the LVM Subsystem

H6296 10-8 127

Student Notes

After creating partitions or logical volumes you will want to use them for file systems or swap.

We have already mentioned that the LVM System is very flexible. You can manage it in many ways, as indicated on the slide.

10-9. LAB: Hands-On with Logical Volume Manager

Directions

Write the commands you would use to perform these tasks. Then, follow your instructor's directions to actually perform the tasks.

1. Assume that you have added a new SCSI drive to your system. The Instance is 1 and the Target is 2. You already have one volume group, and you now plan to set up a second volume group consisting of this disk. Write the steps you would follow and commands you would use to do this.

2. Create a logical volume of 50 megabytes in the volume group `/dev/vg01` and name it *yourname*`fs`.

3. What command would you use to display all the volume groups on your system?

4. What command would you use to display all the logical volumes on your system?

10-10. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. List two benefits of using LVM.

2. Differentiate between a volume group and a logical volume.

3. What are the two reserved areas on a non-bootable disk?

4. What two areas are unique to bootable disks?

5. What command do you use to initialize a disk as an LVM physical disk?

6. What are the steps for creating a volume group?

7. Write the commands you would use to perform these tasks:

a. Make a disk an LVM disk
b. Create a volume group
c. Create a logical volume

Module 11 — Creating and Using File Systems

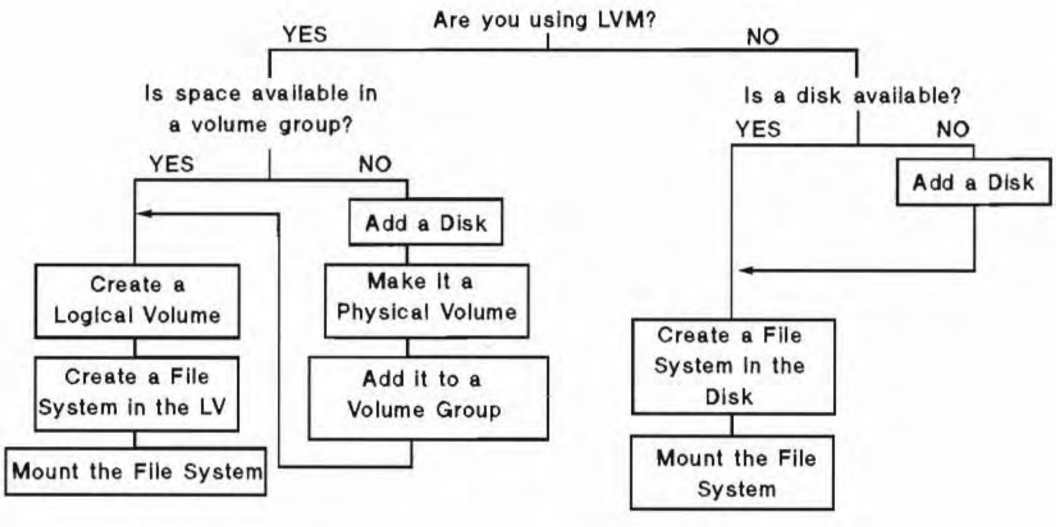
Objectives

Upon completion of this module, the student will be able to:

- Create a file system.
- Mount or unmount a file system.
- Automatically mount a file system.

11-1. SLIDE: Overview of File System Creation

Overview of File System Creation



H6296 11-1 128

Student Notes

You can expand your file system's overall capacity by creating new file systems and attaching (mounting) them to 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. You could create a new file system on another logical volume.

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.
- You want to create a file system with unique characteristics of ownership and/or mirroring.

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.

If you are creating your new file system on a new disk drive, you must connect the device to your system first. For help in this area, read the Installation Manual for the device that you are connecting.

11-2. SLIDE: Creating a New File System

Creating a New File System

1. Create the new file system

   `newfs -F hfs` *char_device_filename*

2. Create a mount point directory for the file system

   `mkdir` */mountpoint*

3. Mount the new file system

   `mount` *block-device-file mount-point-directory*

4. Add the file system to the `/etc/fstab` file

H6296 11-2 129

Student Notes

If you are going to create a file system on a Logical Volume, the Logical Volume should have been already created. These steps assume that has been done. LVM has allocated the physical extents for the logical volume on one of the disks.

If you would like to know the disk's product ID, you can find out the disk's product ID by using the `diskinfo(1M)` command and giving the disk's *character* device file name. The output will look something like:

```
#diskinfo /dev/rdsk/c0t5d0

SCSI describe of /dev/rdsk/c0t5d0
            vendor:  HP
        product id:  2213A
```

```
             type:  direct access
             size:  648192 Kbytes
 bytes per sector:  512
```

The disk's product ID is 2213A, and the size is 648192 Kbytes.

You use the `newfs(1M)` command to create the file system. The `newfs` command requires:

- The `-F` option followed by the type of file system: `hfs` or `ufs`.
- The *character* device file.

As `newfs` successfully creates your new file system, it reports its actions in a form that resembles:

For an HFS:

```
Warning:  256 sector(s) in last cylinder unallocated
Unknown driver type lv, using default rotational delay
/dev/vg00/rlvol5:   59455 sectors in 387 cylinders of 7 tracks,
22 sectors
        60.9Mb in 25 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, 17432, 19728, 22216,
 24704, 27192, 29680, 32168, 34656, 37144, 39440, 41928, 44416,
46904,
 49392, 51880, 54368, 56856, 59152,
```

`newfs` has now created an empty file system that consists of structures for the files and directories that you will create.

You must make an empty directory to serve as the mount point for your new file system. You can create a mount directory anywhere in your existing file system. Do not select a non-empty directory to mount the file system. If you do, the files there will not be accessible until you unmount the file system.

Use the `mount(1M)` command to attach the new file system to the root file system. You must use the *block* device file to mount the file system. The mount command does not print any feedback when it is successful.

You can confirm that the file system is mounted by using the `mount` command with no arguments. You can also use the command `bdf` which will show you a list of the mounted file systems, along with information about available space on the devices.

11-3. SLIDE: The newfs Command in Detail

The newfs Command in Detail

Syntax:

newfs [-F *FStype*][-V][-o *specific_options*] *char-dev-file*

For HFS-specific usage:

newfs -F hfs [-V][-L|-S][-d][-v][-R *swap*][-B][-O *disk_type*]
[*mkfs_options*]*Char-dev-file*

For JFS-specific usage:

newfs -F vxfs [-V][-v][-R *swap*][-B][-O *disk_type*]
[*mkfs_options*]*Char-dev-file*

H6296 11-3 130

Student Notes

Except for certain transfers of raw data (creation of a tape archive, for example), HP-UX cannot use media until a file system exists on it. The best way to create a new file system is with the `newfs` command.

`newfs` is the command for creating a "new" file system. Whether you create a file system in a logical volume or in a whole disk, the command works basically the same, with a few exceptions.

The syntax of the command is described in more detail below:

newfs [-F *FStype*] [-V] [-o *specific_options*] *character_device_file*

For hfs-specific usage:

`newfs -F hfs [-V] [-L|-S] [-d] [-v] [-R` *swap*`] [-B] [-O` *disk_type*`] [`*mkfs_options*`]` *character_device_file*

For JFS-specific usage :

`newfs -F vxfs [-V] [-v] [-R` *swap*`] [-B] [-O` *disk_type*`] [`*mkfs_options*`]` *character_device_file*

character_device_file	This is the character device file for the device on which you are creating the file system. It must be supplied.
`-F` *FStype*	This option specifies the file system type on which `newfs` should operate.
`-V`	Echo the completed command line without executing the command.
`-d`	The `-F` (force) option supported by previous versions of this command has been replaced by the -d (disregard) option in the HFS-specific version of the command. The functionality of the new `-d` option is identical to the old `-F` option (i.e., disregard the existence of a mounted file system).
`-R` *swap*	Reserve *swap* Mbytes of swap/boot space past the end of the file system.
`-B`	Reserve space for boot programs past the end of the file system. If `/usr/lib/uxbootlf` is present on the system, then sufficient space to accommodate that file is reserved, otherwise 691 Kbyte sectors are reserved.
`-O disk_type`	Specify the disk type in /etc/disktab . Any parameters specified in the command-line will override the corresponding values in /etc/disktab . Any values not given in the command-line or in /etc/disktab will be defaulted.
`-L`	Creates a file system with long file names, 256 characters maximum.
`-S`	Creates a file system with short file names, 14 characters maximum.
`-v`	Verbose mode, lists newfs actions.

`newfs` calls the `mkfs(1M)` utility. `mkfs` options that can be used with `newfs` to override `mkfs` defaults are:

`-s` *size*	Specifies the total size of the file system in blocks. If this option is not specified, `newfs` will calculate the maximum size file system possible on the disk being used.
`-b` *block-size*	Specifies the block size of the file system in bytes. If defaulted this is usually 8 K.
`-f` *frag-size*	*Only for hfs*. Specifies the fragment size of the file system in bytes. If defaulted this is usually 1 Kbyte.
`-m` *% free-space*	*Only for hfs*. This indicates the percentage of space reserved from the users. If the amount of free space in the file system falls below this percentage, the superuser is the only person who can write to the file system. If defaulted this is usually 10 percent
`-i` *bytes/inode*	*Only for hfs*. This specifies the number of inodes that will be created. In other words for every x number of bytes, one inode will be created. If defaulted this is

usually 1 inode per 6144 bytes of file space. If fewer inodes are desired, a larger number should be used; to create more inodes a smaller number should be given.

Typically, the default values for the disk being used are adequate for file system purposes. There may be an occasion, however, when it's necessary to specify a different value. In such a case, the options to `newfs` are very useful. If an option is not specified, `newfs` will either calculate an appropriate value or extract a default value from `/etc/disktab`.

Before using `newfs`:

- Make sure that the media is not mounted. Use the `mount` or `bdf` command to check.

Different examples are described below:

```
# newfs -F hfs /dev/vg01/rlvol5
```

Creates a new HFS file system in the /dev/vg01/rlvol5 logical volume.

```
# newfs -F hfs -b 4096 -f 2048 -m 5 -i 4096 /dev/rdsk/c1t4d0
```

Creates a new JFS file system in the /dev/vg01/rlvol5 logical volume.

```
# newfs -F hfs -b 4096 -f 2048 -m 5 -i 4096 /dev/rdsk/c1t4d0
```

The device file is `/dev/rdsk/c1t4d0`. The remaining options provide overrides to the default values for this type of disk. The block size is 4K, the fragment size is 2K, there is 5% reserved space, and 4K of file system space per inode.

```
# newfs -F hfs -R 96 /dev/rdsk/c1t4d0
```

Creates a new file system on a disk and leaves 96 Mbyte for swap.

```
# newfs -F vxfs -b 4096 /dev/rdsk/c1t4d0
```

Creates a JFS file system with block size=4096 bytes

`newfs` appends the superblock locations for the HFS file systems to the file `/var/adm/sbtab`. It is a good idea to print a copy of this file after you create a new file system. An example of the `/var/adm/sbtab` file is shown below:

```
/dev/vg00/rlvol1: super-block backups (for fsck -b#) at:
 16, 2504, 4992, 7480, 9968, 12456, 14944, 17432, 19728, 22216,
 24704, 27192, 29680, 32168, 34656, 37144, 39440, 41928, 44416,
46904,

/dev/vg00/rlvol3: super-block backups (for fsck -b#) at:
 16, 2504, 4992, 7480, 9968, 12456, 14944, 17432, 19728, 22216,
 24704, 27192,

/dev/vg00/rlvol4: super-block backups (for fsck -b#) at:
 16, 2504, 4992, 7480, 9968, 12456, 14944, 17432, 19728,
```

```
/dev/vg00/rlvol5: super-block backups (for fsck -b#) at:
 16, 2504, 4992, 7480, 9968, 12456, 14944, 17432, 19728, 22216,
 24704, 27192, 29680, 32168, 34656, 37144, 39440, 41928, 44416,
46904,
 49392, 51880, 54368, 56856, 59152, 61640, 64128, 66616, 69104,
71592,
 74080, 76568, 78864, 81352, 83840, 86328, 88816, 91304, 93792,
96280,
 98576, 101064, 103552, 106040, 108528, 111016, 113504, 115992,
118288, 120776,
 123264, 125752, 128240, 130728, 133216, 135704, 138000, 140488,
142976, 145464,
 147952, 150440, 152928, 155416, 157712, 160200, 162688, 165176,
167664, 170152,
 172640, 175128, 177424, 179912, 182400, 184888, 187376, 189864,
192352, 194840,
 197136, 199624, 202112, 204600, 207088, 209576, 212064, 214552,
216848, 219336,
 221824, 224312, 226800, 229288, 231776, 234264, 236560, 239048,
241536, 244024,
 246512, 249000, 251488, 253976, 256272, 258760, 261248, 263736,
266224, 268712,
 271200, 273688, 275984, 278472, 280960, 283448, 285936, 288424,
290912, 293400,
 295696, 298184,

/dev/vg00/rlvol6: super-block backups (for fsck -b#) at:
 16, 2504, 4992, 7480, 9968, 12456, 14944, 17432, 19728, 22216,
 24704, 27192, 29680, 32168, 34656, 37144, 39440,
```

11-4. SLIDE: Mounting the New File System

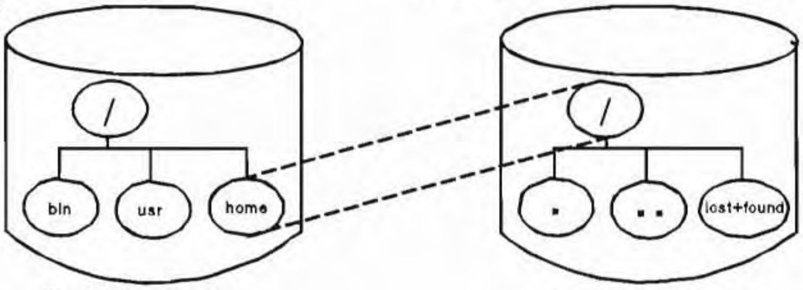
Mounting the New File System

- Create a sub-directory with the mkdir command
- Syntax:

  mount block-device-file directory

mount # Displays actually mounted FS

Example:



Root File System New File System

H6296 11-4 131

Student Notes

HP-UX cannot use a file system unless it is mounted. After the file system has been created on the device, it must be incorporated into the root file system. This is done by using the `mount` command to logically associate the root (/) directory on the new file system with a directory on the root file system, the **mount point**.

For example:

```
# mount /dev/vg00/lvol5 /home
```

The new file system whose device file is `/dev/vg00/lvol5` is mounted onto `/home` in the root file system.

After the file system is mounted, any reference to /home is the same as / on the new file system. For example, the following command puts a copy of /etc/profile into the root directory of the new file system.

```
# cp /etc/profile /home
```

The mount command with no arguments displays a list of the currently mounted file systems and their respective mount points.

Example:

```
$ mount
/ on /dev/vg00/lvol1 read/write on Tue Nov  9 09:50:37 1994
/var on /dev/vg00/lvol3 read/write on Tue Nov  9 09:53:42 1994
/usr on /dev/vg00/lvol5 read/write on Thu Nov 9 09:53:42 1994
/tmp on /dev/vg00/lvol4 read/write on Tue Nov  9 09:53:42 1994
/home on /dev/vg00/lvol5 read/write on Tue Nov  9 09:53:42 1994
```

Options to mount include:

-r	mount "read only"
-f	"forcible" mount
-a	mount "all" file systems in /etc/fstab
-F *FStype*	specify a file system type
-v	Reports the output with the FStype displayed, in addition to the old output.
-e	Verbose mode. Write a message to the standard output indicating which file system is being mounted.
-p	Print the list of mounted file systems in a format suitable for use in /etc/fstab.

Special options for JFS

For Best Data Integrity:

blkclear	Clear extents when unused
datainlog	Log synchronous writes
mincache=closesync	Flush when file is closed

For Enhanced Performance:

delaylog	Delay writes of the intent log
nodatainlog	For disks without bad block revectoring

For Temporary File Systems:

tmplog	Delay writes of the intent log

Instead of using the mount -a command, you may use the new mountall(1M)'' command.

If a user is working in the directory that is your mount point (for example, has done a `cd` into that directory), you will not be able to mount your new file system on that mount point. In addition, any files that exist in the directory that is your mount point will be hidden after you mount a file system onto that directory.

You can chain together mounted file systems with certain restrictions when other kinds of file systems are used. We have only talked here about the `HFS` file system. There are others. The `CDFS` file system will be covered later and the `NFS` file system is covered in a separate course, *HP-UX Network Administration*.

You should make certain that the `/sbin`, `/dev`, `/etc` are *not* mounted file systems and we already know that the `root` file system is automatically "pseudo mounted" at boot time, so it is not considered an explicitly mounted file system, because it cannot be "un-mounted" from the system.

Note

The `mount` command uses the block device file, whereas `mediainit` and `newfs` use the character device file.

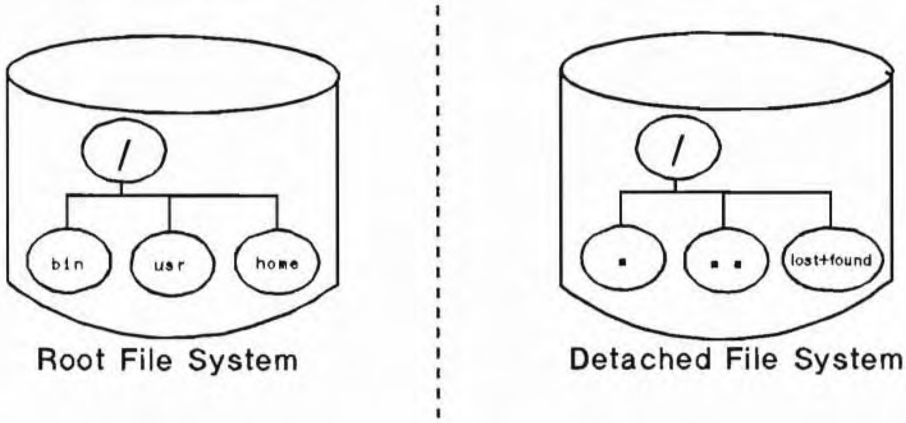
11-5. SLIDE: The umount Command

The `umount` Command

Syntax:

umount *block-dev-file*

umount *mount-point*



Root File System | Detached File System

H6296 11-5 132

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` and *not* "unmount". The command uses the block device file or mount-point directory.

Options to `mount` include:

-a	umount "all" file systems in /etc/mnttab
-F *FStype*	specify a file system type
-v	report the output with the *FStype* displayed

Instead of using the umount -a command, you may use the new umountall(1M) command.

A file system cannot be unmounted if any files are open or if any user's current working directory is a directory in that file system. You can use the fuser command to identify which processes are using a file or file structure:

```
# fuser -u /dev/vg01/lvol1
```

This lists process IDs and login names of processes using /dev/vg01/lvol1.

```
# fuser -u /etc/passwd
```

This lists process IDs and login names of processes that have the passwd file open.

```
# fuser -ku /dev/vg01/lvol1
```

This terminates all processes that are preventing logical volume lvol1 of volume group vg01 from being unmounted, listing the process ID and login name of each as it is killed. Always umount all mounted file systems *before* bringing the system down or you may cause corruption to the file systems. The umount -a command will unmount all the file systems in /etc/mnttab. The shutdown script unmounts all file systems before bringing the system down.

You cannot umount the root file system.

You cannot umount a file system that has file system swap enabled on that disk without rebooting the system.

11-6. SLIDE: Automatically Mounting File Systems

Automatically Mounting File Systems

- Place an entry in the `/etc/fstab` file
- File systems will be mounted when the system is booted, or you can use `mount -a` or `mountall`
- You can mount file systems by absolute directory names.

Sample `/etc/fstab`:

```
/dev/vg00/lvol1  /         hfs  defaults 0 1
/dev/vg00/lvol3  /opt      hfs  defaults 0 2
/dev/vg00/lvol4  /home     hfs  defaults 0 2
/dev/vg00/lvol5  /usr      hfs  defaults 0 2
/dev/vg00/lvol6  /var      hfs  defaults 0 2
/dev/vg01/lvol1  /home/work vxfs rw,suid,delaylog 0 3
```

H6296 11-6 133

Student Notes

Placing an entry in `/etc/fstab` causes your file system to be mounted every time the system boots. The `/sbin/init.d/hfsmount` script contains the command `mount -a`. The command `mount -a` mounts all file systems named in the `/etc/fstab` file. The `/etc/fstab` file is *not* maintained by the system and should be edited by the system administrator when changes are made to the system.

Fields in the `/etc/fstab` file are:

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.
- `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` - (Not for JFS). Sets options rw, suid and noquota. When used, this must be the only option specified. You may not specify additional options along with `defaults`.
- `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 #)

Special options for JFS

These may appear in the *options* field of `/etc/fstab` if the Journaled File System is used.

For Best Data Integrity:

- `blkclear` - Clear extents when unused
- `datainlog` - Log synchronous writes
- `mincache=closesync` - Flush when file is closed

For Enhanced Performance:

- `delaylog` - Delay writes of the intent log
- `nodatainlog` - For disks without bad block revectoring

For Temporary File Systems:

- `tmplog` - Delay writes of the intent log

See `fstab(4)` in the *HP-UX Reference* manual for more information.

11-7. SLIDE: CD-ROM File Systems (CDFS)

CD-ROM File Systems (CDFS)

- Allows mass distribution and easy retrieval of large amounts of information
- You can read data from a CD, but you cannot write to it
- To use CDFS volumes:
  1. Configure the appropriate driver into the kernel
  2. Create the necessary device files if necessary
  3. Mount the CDFS volume with the `mount` command
- CDFS Examples:
  1. LaserRom Manual
  2. The update process

H6296 11-7 134

Student Notes

CD-ROM is an acronym for Compact Disk-Read Only Memory. Compact Disks (or CDs) contain approximately 550 MB of data per disk. The information on the CD is virtually permanent; you can read data from a CD, but you cannot write to it. Data on a CD is prepared and mastered using a specialized publishing process. The steps to use a CDFS volume are shown on the slide.

You can have either an HP-IB or SCSI CD-ROM drive. Reconfigure the kernel using `sam` or the manual method.

Once the kernel has been configured, reboot the system to create the necessary device files (or type the `insf` command). The device should be created for the whole disk. Then, create an appropriate directory (in the example below, `/cdrom`) as a mount point for the drive and mount the CDFS volume with the `mount` command.

```
# mount /dev/dsk/c1t2d0 /cdrom
```

Once mounted, HP-UX commands may be used to perform any operation as long as that operation does not attempt to modify files on the CDFS volume. The data on the CDFS volume is read-only.

A sample entry in the `/etc/fstab` file for automatically mounting a cdfs volume:

```
/dev/dsk/c1t2d0 /cdrom cdfs ro 0 0 # Local CD-ROM drive
```

Other CDFS or HFS volumes may be mounted on top of the mounted CDFS volume.

There are a few commands that are not applicable to CDFS volumes, either because of the read-only nature of the medium, or because they are inherently HFS-oriented. These commands are unsupported under CDFS, and include the following (not a complete list):

- `mv`
- `fsck`
- `fsclean`
- `mediainit`
- `mkfs`
- `ncheck`
- `newfs`

11-8. LAB: Hands-On, Creating a File System

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

If you are do not have your own system, your instructor will assign you two logical volumes to use in the exercises.

1. Run SAM. Choose `Disks and File Systems-->` on the Control Box. Choose `File Systems` from the Functional Area list. Then, choose the `Add Local File Systems...` Action from the menu bar and `Using the Logical Volume Manager` from the submenu.

Within SAM, perform the following sequence of steps. As you carry out each step, SAM asks you to supply certain information. From the nature of this information, can you deduce which command SAM is invoking at each step?

1. Enter a mount directory.
2. Choose `HFS File System` and `Modify File System Defaults`.
   a. Choose the `Now` option under `When to Mount`.
   b. Choose the `Every System Boot` option under `When to Mount`.
   c. Turn on the `Create new file system` check box and activate (OK).

2. Create a file system using the `newfs` command on the partition assigned to you by your instructor.

3. Create a directory called `/disk`*n*, if it does not already exists, where *n* is the number of your group. Mount the newly created file system to that directory. Then copy the file `/etc/passwd` to the "root" directory of your new file system.

4. Unmount your file system. Can you access your copy of `/etc/passwd`? Why or why not?

5. Edit the `/etc/fstab` file and add a line that will automatically mount your newly created file system at boot time. If you are the working together with other groups on the same system, make sure that you are the only one who edit the `fstab` file.

6. What happens if you mount your new file system to a directory with files in it? How can you recover?

7. You accidentally destroyed an important file `/home/susan/goodfile`. Although a backup tape is *not* available, you do have another disk which is an exact replica of your system disk (before `/home/susan/goodfile` was removed). List out the steps to restore this file from the HFS file system on your auxiliary disk.

Module 12 — Maintaining the File System

Objectives

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

- Explain how an HFS and a JFS file system handles file modifications.
- Identify three causes of file system corruption.
- Explain the purpose of the sync command.
- Check for file system corruption.
- Repair file system corruption.

Module 12 — Maintaining the File System

12-1. SLIDE: File System Maintenance

File System Maintenance

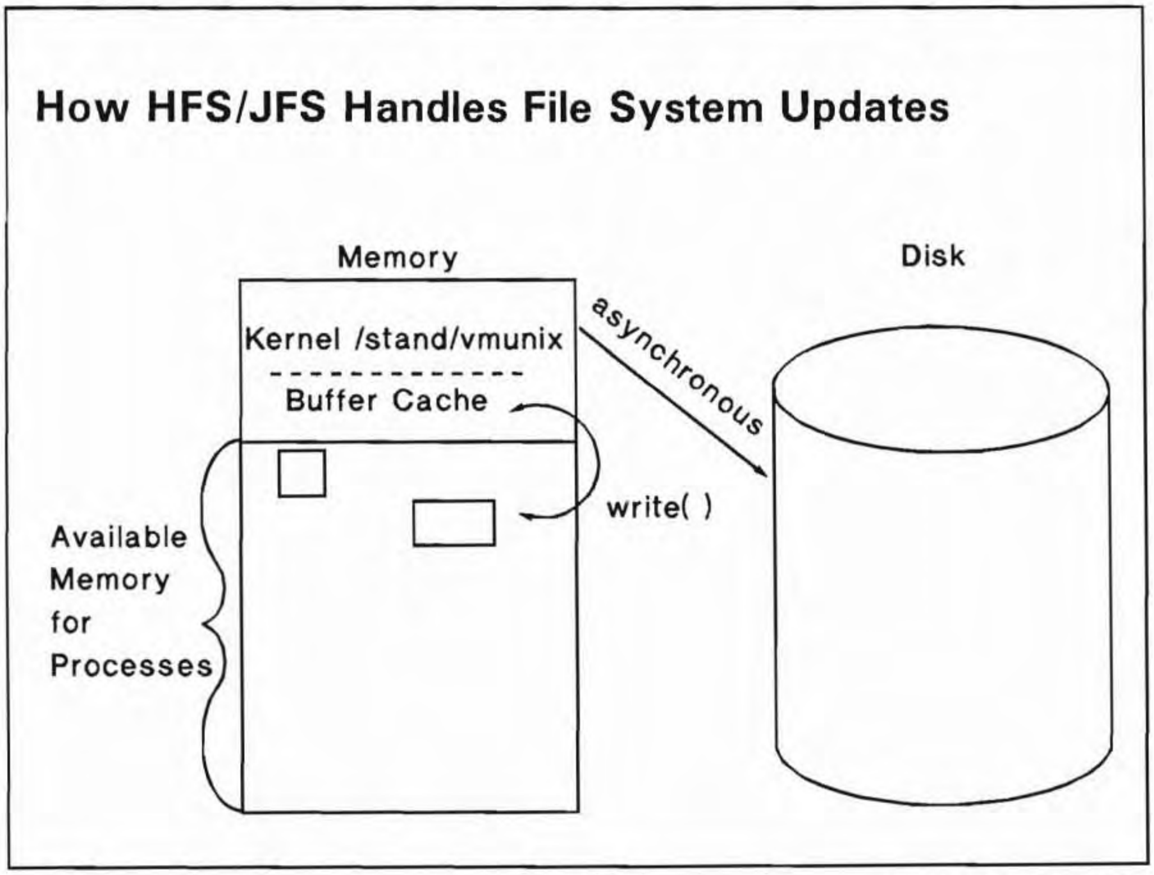
- Routine Maintenance
  - Check file system integrity
  - Employ regular backup procedures
  - Monitor disk usage

H6296 12-1 135

Student Notes

One of the principal responsibilities of a system administrator is preservation of the user's data. Since the data storage structure utilized by HP-UX is the file system, it's imperative that the storage environment of the file system be checked regularly for possible problems. The integrity of the file system mustnever be compromised. There are many things the administrator can do to help maintain the integrity of the file systems.

12-2. SLIDE: How HFS/JFS Handles File System Updates



H6296 12-2 136

Student Notes

The Buffer Cache

When data needs to be written, the actual write does not occur immediately. The data is initially copied to an in-memory buffer called the buffer cache. This is a much faster operation than if the system had to perform an actual write to the disk. The data, along with the inode information, is written to the disk sometime later, usually when the buffer cache fills up and the system needs to clear some buffer space. If the system is halted without writing the buffer to disk, the file system could become corrupted. You should never allow work to continue if you suspect that the file system is corrupted.

Here are some of the benefits and advantages of having a buffer cache:

- The use of the buffer cache allows for uniform disk access, because the kernel does not need to know the reason for the I/O. The kernel just always writes buffers to the disk, not parts of buffers or real numbers. So system design is simpler, from a disk I/O standpoint.
- By using a buffer scheme programs are more easily ported to other UNIX systems. Disk I/O may be different on different UNIX machines, but the programs don't have to know that. They simply write to buffers, without having to worry about how the disk is set up.
- Using a buffer cache reduces the amount of disk traffic, thereby increasing overall system throughput and decreasing response time. In other words the system runs faster.
- Re-use of data files in the buffer cache can also speed up a system.

HFS Data Structure Update

Every time a file is modified, the HP-UX operating system performs a series of file system updates. For example, if data is added to a file which increases its size, additional block(s) may need to be allocated for the new data. The block(s) must be deleted from the free block list, the inode of the file must be updated, the superblock must be updated to reflect the new size of the file system, and countless other updates must occur. Now consider that this type of update sequence must occur for every change in the file system.

Updates occur to the superblock, inodes, data blocks, and cylinder group information in the following ways:

Primary Superblock	The superblock of a mounted file system is written to the disk whenever a `umount` command is issued, or when a `sync` command is issued and the file system has been modified. The root file system is mounted during boot and cannot be unmounted.
Inodes	An inode contains information specific to the file it describes. An inode is written to the file system upon closure of the file associated with the inode, when a `sync` or `fsync` command is issued, when the file system is unmounted, or as soon as the file is written if `O_SYNC` is set for the file.
Data blocks	In-core blocks are written to the file system whenever they have been modified and released by the operating system. More precisely, they are buffered or queued for eventual writing. Physical I/O is deferred until the buffer is needed by HP-UX, a `sync`(1m) command is issued, an `fsync`(2) is issued for the file, or `O_SYNC` is set for the file. If a file is opened with the `O_SYNC` flag set, the `write`(2) system call does not return until completed.
Cylinder group information	The cylinder group information is updated whenever a `sync` is executed, or when the system needs a buffer and the cylinder group is written.

JFS Data Structure Update

With JFS you need five steps to complete a transaction:

- Update the free extent map for the directory, to increase directory size
- Change the allocated block
- Modify the directory inode to reflect the new size
- Update the free inode map

12-3. SLIDE: sync and syncer

sync and syncer

syncer

- syncer is executed at system boot in the startup script /sbin/init.d/syncer file
- The syntax of the syncer program is:

```
syncer [ seconds ]
```

sync(1m)

- Writes buffer contents to disk
- Keeps the file system current
- Is normally invoked on a regular basis by the syncer program

H6296 12-3 137

Student Notes

As we have seen, data is written to an in-core buffer cache before it is written to disk. A physical write from the buffer to disk is delayed until:

- The system needs the buffer for another operation.
- The last byte of the block is modified.
- The file system is unmounted.
- The sync command is executed and the buffer has not been accessed by the last sync call.

syncer(1m)

syncer is normally started from the /sbin/init.d/syncer script at system startup. You should not execute the syncer command directly; it should be executed at system boot time via

/sbin/init.d/syncer. syncer will also update the /etc/mnttab file unless it is invoked with the -s option.

sync(1m)

sync executes the system call sync(2), which flushes all previously unwritten system buffers, including modified superblocks, modified inodes, and delayed block I/O, out to disk. This ensures that all file modifications are properly written to disk before performing a critical operation such as system shutdown. You can execute the sync command at any time. To ensure that it is run automatically at periodic intervals, the syncer command should be used.

12-4. SLIDE: The `fsck` Command

The `fsck` Command

- Checks file system consistency and makes repairs
- Is multi-pass
- Should be run on an unmounted (or quiet) file system
- Can be run interactively or non-interactively
- Interruptable with (Control) + (C)

```
generic usage :
     fsck [-F FSType] [-V] [-m] [-o specific_options] [special ...]

hfs-specific usage :
     fsck [-V] [-F hfs] -p|-P [-f] [file_system]
     fsck [-V] [-F hfs] [-b block#] [-y | -n] [-f] [file_system]

jfs-specific usage:
     fsck [-V] -F vxfs [-pnNyY] [-o full,nolog] special
```

H6296 12-4 138

Student Notes

`fsck` is the principal file system maintenance tool available with HP-UX. It verifies the structural integrity by checking data which is intrinsically redundant in a file system. The redundant data is either read from the file system or computed from known values.

`fsck` should be run on quiescent file systems. Ideally, the file system should be unmounted. Since this is not possible for the root file system, you should bring your system to a single-user run-level with the `shutdown` command before running `fsck` on root. `fsck` won't run on a mounted file system unless the `-f` option is specified.

For HFS:

`fsck` is a multi-pass program, meaning that it examines the file system a number of times, each iteration examining a different feature of the file system. Each pass `fsck` makes through the file system is known as a *phase*.

During each phase, any inconsistencies noted in the file system are reported and `fsck` asks if corrective action should be taken. `fsck` then waits for a response. If a yes response is provided, `fsck` attempts to repair the inconsistency. With a no response, `fsck` ignores the inconsistency found and continues its checking. There are very few occasions when a no response should be given.

For JFS:

`fsck` for a JFS just reads the log instead of scanning the entire file system. The `fsck` just takes seconds instead of minutes.

You can also perform a complete `fsck` with specific options if you need to.

The `fsck` options:

`-F` *FSType*	This option specifies the file system type on which fsck should operate.
`-V`	The -V option is the verbose option and it echoes the complete command line.
`-m`	The -m option performs a sanity check and fsck will return zero if the file system is suitable for mounting. If the file system needs additional checking, the return code is 32. If the file system is mounted the return code is 33. Error codes larger than 33 indicate that the file system is badly damaged.
-o *specific_options*	specifies the file system specific options, if any.
`-n`|`-N`	Answer "no" for all the questions.
`-y`|`-Y`	answer "yes" for all the questions

If you invoke `fsck` without giving it a *file_system*. `fsck` will check those file systems marked "hfs" in the `/etc/fstab` file. The order is determined by the number in the *pass_number* field (sixth field).

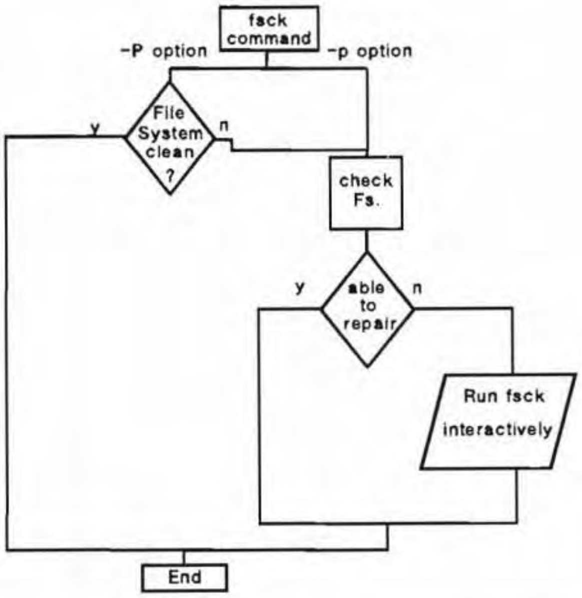
You can do a Ctrl + c out of `fsck`. `fsck` does not make any repairs to a file system until the command successfully completes. (Repairs are written to RAM as the command is running.) Thus, if you kill the command, nothing will happen.

Note

There will be multiple man pages for fsck. The generic man page will be named fsck and the file system specific man page will be called `fsck_FSType(1M)`, for example `fsck_hfs(1M)`.

12-5. SLIDE: fsck in Preening Mode for HFS

fsck in Preening Mode for HFS



H6296 12-5 139

Student Notes

The first of the two ways that `fsck` can be invoked is preening mode. This is a non-interactive mode. This option fixes many problems, but never removes data. `fsck` decides what to do to fix a problem. For each problem it fixes, it prints a message identifying the file system and the corrective action taken. If it cannot solve a problem, it terminates. If `fsck` terminates, you should run it interactively so that you can fix the problems.

If the `-p` (lower-case) option is used without specifying a *file_system*. `fsck` reads the specified pass numbers in `/etc/fstab` to inspect groups of disks in parallel, taking maximum advantage of I/O overlap to preen the file systems as quickly as possible.

The `-P` (upper-case) option operates in the same manner as the `-p` option except those file systems which were cleanly unmounted will not be checked. (`fsck` checks the clean byte.) This can greatly decrease the amount of time required to reboot a system which was brought down cleanly. `fsck -F hfs -P` is usually the command which is run on non-root file systems at bootup time. It is invoked in the `/sbin/bcheckrc` script run by `init`. The root file system is checked via `/sbin/pre_init_rc`.

Note

It is a good idea to run `fsck -F hfs -p` (lower-case) before doing a full backup. If `fsck -p` completes successfully, perform your backup as normal. If `fsck -F hfs -p` aborts with errors, backup the bad file system, then attempt to repair the file system. If the repair completes successfully, backup again.

12-6. SLIDE: Other Options to fsck for HFS

Other Options to fsck for HFS

`fsck [-F hfs] [-V] [ -b` *block#* `] [ -y | -n ] [ -f ] [`*file_system* `]`

-V	The -V option is the verbose option and it echoes the complete command line.
-b	Uses the block specified immediately after the flag as the super block for the file system
-y	Assume a "yes" response to all questions asked by fsck
-n	Assume a "no" response to all questions asked by fsck
-f	Force mode to check mounted file systems

H6296 12-6 140

Student Notes

If you invoke fsck with no options, it runs interactively. In interactive mode, fsck poses a question when an inconsistency is found and waits for your response.

Other options available to fsck are shown on the slide. They are described in more detail below:

-V The -V option is the verbose option and it echoes the complete command line.

-b *block#* This option tells fsck to use *block#* as the superblock for the file system check. This is useful if the primary superblock is lost or corrupted. You could try this option if you seem to be getting a lot of errors with fsck and it doesn't make sense to you. All block were written by newfs to the file /var/adm/sbtab. If you can't access this file, you can try the alternate superblock #16, which is always the first alternate superblock.

-y This option causes fsck to answer "yes" to all questions posed by the command. It is possible that data will be removed as a result of a "yes" answer. Consequently, if this option is to used, the file system should be examined with the `-n` option first so the possible consequences can be assessed.

`-n` This option causes `fsck` to answer "no" to all questions posed by the command. Since this option never results in the loss of data, it may appear safe. However, since "no" is supplied as the answer to all questions, `fsck` takes no corrective actions and inconsistencies are not resolved. It is recommended that this option be used when you want to assess the state of the file system; however, you should invoke the command again to resolve inconsistencies if any are found.

Example:

```
# fsck -n /dev/vg00/lvol3  |  tee /tmp/fsck.log
```

In this example, the diagnostic output is directed to the file `/usr/tmp/fsck.log` as well as to the screen. This output can be analyzed to determine corrective action.

`-f` `fsck` complains when the file system is mounted and gives a warning message and asks for continuation. The `-f` option suppresses this question.

Note

The -F option supported by HP-UX 9.x and earlier versions has been replaced by the -f option. The -F option for releases 9.x (and prior) forced fsck to check a mounted file system.

Note

Do not reboot the system unless `fsck` tells you to reboot. A reboot will sync the disks and thus write out the bad data. If you must reboot, use the `reboot -n` command which does not issue a `sync`.

On the next page is a listing from an interactive fsck output.

```
# fsck

fsck: /dev/vg00/lvol1: root file system
continue (y/n)? y
** /dev/dsk/c0t6d0
** Last Mounted on /
** Root file system
** Phase 1 - Check Blocks and Sizes
BAD DIRECT ADDRESS, SHOULD BE ZERO: inode.di_db[1] = 138361
 I=20507
 OWNER=volker MODE=100440
SIZE=347 MTIME=Feb 10 17:56 1993
CORRECT? y

INCORRECT BLOCK COUNT I=20507 (1 should be 0)
CORRECT? y

** Phase 2 - Check Pathnames
```

```
UNALLOCATED  I=20511
 OWNER=root MODE=0
SIZE=0 MTIME=Jan  1 01:00 1970
NAME=/home/volker/.login

REMOVE? y

** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
UNREF FILE  I=28
 OWNER=root MODE=100666
SIZE=230188 MTIME=Nov  9 16:58 1993
RECONNECT? y


CLEAR? y

LINK COUNT FILE I=20512
 OWNER=volker MODE=100600
SIZE=720 MTIME=Apr 22 14:53 1993  COUNT 1 SHOULD BE 2
ADJUST? y

FREE INODE COUNT WRONG IN SUPERBLK
FIX? y

** Phase 5 - Check Cyl groups
241 BLK(S) MISSING
BAD CYLINDER GROUPS
FIX? y

** Phase 6 - Salvage Cylinder Groups
14662 files, 0 icont, 265298 used, 47260 free (4524 frags, 5342
blocks)

***** FILE SYSTEM WAS MODIFIED *****

***** REBOOT HP-UX; DO NOT SYNC (USE reboot -n) *****

# reboot -n
```

12-7. SLIDE: `fsck` with a JFS File System

`fsck` with a JFS File System

Fast file system recovery

- JFS logs the changes it is going to make to an intent log.
- `fsck` reads the intent log and repairs the file system from it
- Old style full checking is also available

JFS-specific usage

`fsck [-V] -F vxfs [-pnNyY] [-o full,nolog]` *special*

H6296 12-7 141

Student Notes

A transaction that creates a file and expands a directory on a JFS file system does the following:

- Update the free extent map for directory to increase directory size
- Change the allocated blocks
- Modification of the directory inode to reflect new size
- Modification of the inode for new file
- Update to the free inode map

The system could crash before all six steps have completed, leaving the file system in an inconsistent state. The intention to perform these steps is logged before the file system is changed. If the system crashes before all steps are complete, `fsck` can read the intent log and reconstruct the file system.

Besides being faster, this will work on an mounted file system.

The JFS specific options are:

-o *specific_options*	specifies the file system specific options, if any. For JFS you should ask *full* or *nolog* specific options.

Example of a full fsck:

```
# fsck  -F vxfs -o full,nolog /dev/vg01/lvol4
checking structural files
pass1 - checking inode sanity and blocks
pass2 - checking directory linkage
pass3 - checking reference counts
pass4 - checking resource maps
OK to clear log? (ynq)y
set state to CLEAN? (ynq)y
#
```

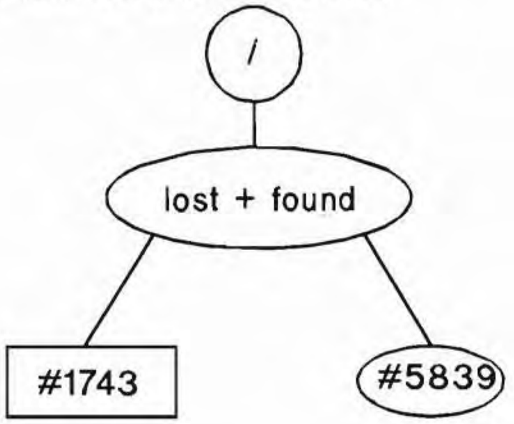
Example of a fsck on a corrupted JFS :

```
# mount /dev/vg01/lvol4 /home/tina
vxfs mount: /dev/vg01/lvol4 is corrupted. needs checking
# fsck -F vxfs /dev/vg01/lvol4
log replay in progress
replay complete - marking super-block as CLEAN
# mount /dev/vg01/lvol4 /home/tina
#
# mount /dev/vg01/lvol3 /usr/audio
vxfs mount: /dev/vg01/lvol3 is corrupted. needs checking
# fsck -F vxfs /dev/vg01/lvol3
log replay in progress
replay complete - marking super-block as CLEAN
# mount /dev/vg01/lvol3 /usr/audio
```

12-8. SLIDE: The lost+found Directory

The lost+found Directory

- Created by newfs (it should exist in every file system).
- fsck copies problem or orphaned files to this directory.
- Check this directory after each invocation of fsck and try to determine the origin of any files there.
- Can be created with the mklost+found command.



H6296 12-8 142

Student Notes

Every file system should have a `lost+found` directory at the root of its file system (that is, `/lost+found`). The `lost+found` directory is created by `newfs`. However, you should verify that the directory exists before using `fsck` to check the file system. If `lost+found` does not exist, you can rebuild it with the `/usr/sbin/mklost+found` command.

`fsck` places any problem files or directories in the `lost+found` directory. After `fsck` completes, you should examine the contents of the directory. The files that are placed there should be moved back to their original directories. The name assigned is the inode number, so it may be difficult (and sometimes impossible) to determine where the files actually belong, but you should try to find owners. Run the `file` command on a file. If the file contains text, look at its contents to try and determine the owner. If the file contains executable code check to see if it has an SCCS identification string. If it does, the `what` command will list SCCS identification information. If the file does not have an SCCS identification string, use the `strings` command to print the literal strings from the file. These strings may help identify the owner. Do not execute an executable file found in the `lost+found` directory to try and figure out what it is. It may be the program that corrupted the disk.

12-9. LAB: Hands-On Maintaining the File System

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. For the system you are using in class, on which file systems would the command `fsck -F hfs -p` perform a file system check? In what order would these file systems be checked? Why is the order potentially significant ?

2. Run `fsck` on the file system you created and mounted earlier.

12-10. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. Describe the buffer cache

2. What are the advantages of the buffer cache

3. What one disadvantage of the buffer cache does the System Administrator have to worry about?

4. What does `fsck` do with a HFS File System?

5. What does `fsck` do with a JFS File System?

6. What is the `lost+found` directory used for?

Module 13 — File System Management

Objectives

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

- Check available free disk space.
- Implement some space management procedures.
- Moving data from one disk to another.
- Extend file systems.
- Move a volume group from one system to another.
- Remove unused software.

Module 13 — File System Management

13-1. SLIDE: Disk Usage

Disk Usage

Syntax:

```
bdf [ -i ] [ file system | file ]

du [ -a | -s ] [ file ... ]
```

Examples:

```
# bdf

File system        kbytes    used  avail capacity  Mounted on
/dev/vg00/lvol1  100047   84736   5306     94%      /
/dev/vg00/lvol3  200094  136280 438046     76%      /usr
/dev/vg00/lvol4   60280    9348  44904     17%      /tmp
/dev/vg01/lvol1  483392  483392      0    100%      /home

# du -s /home
91464   /home
```

H6296 13-1 143

Student Notes

The system administrator is responsible for monitoring the amount of free disk space on the system. The easiest way to do this is with the bdf command. The fields have the following meaning:

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
capacity	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 give information relating to the availability of inodes in the file system.

iused	Number of inodes currently in use on the file system
ifree	Number of free inodes on the file system
iused	Percentage of inodes used on the file system

When you want to see more details, for example how much space is used beneath the `/user` directory, you can do this with the du command. It shows the amount of space in blocks of 512 byte. 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:

-a	Print entries for each file encountered in the directory hierarchies.
-s	Print only the grand total of disk usage for each of the specified *file* operands.

For further information refer to the du(1) manual entry.

13-2. SLIDE: Routine Management

Routine Management

- Log Files that grow without bound:

`/var/adm/wtmp`	logs successful login attempts
`/var/adm/btmp`	logs unsuccessful login attempts
`/var/adm/sulog`	logs use of `su` command

  And others....

- Remove core files
- Remove large, old files
- SAM can help

H6296 13-2 144

Student Notes

Disk space is often at a premium. The System Administrator should monitor disk free space regularly, and take steps to prevent a situation of running out of disk space. There are some proactive measures that the System Administrator can take, including monitoring files that continuously grow, removing of core files, trimming of log files, and removal of large files that have not been used in a long time.

Log Files That Grow Without Bound

One of the most common methods of recovering disk space on a system is to trim log files, both ASCII and non-ASCII.

There are many log files on a system that grow without bound, including `/var/adm/wtmp` and `/var/adm/btmp`. Files such as these should be monitored by the system administrator and appropriate action should be taken on a regular basis.

You can empty these files periodically with:

$ > *logfile*

You could do this with the `cron` facility or much more easily with SAM.

SAM monitors a basic list of ASCII logfiles and another list of non-ASCII logfiles. You can also add any other logfiles that might be produced by your applications to the list of logfiles that SAM monitors. SAM gives you the capability to trim specified logfiles to zero, to a recommended size, to a certain number of lines, or to a percentage of the current size. Logfiles should be monitored on a regular basis.

Here is a list (not necessarily complete) of files that tend to grow without bound. Your applications may also produce unbounded files.

Table 13-1. Table of files that grow without bound

File Name	Written to by	Read by
/var/adm/wtmp	/usr/bin/login	/usr/bin/last
	/sbin/init	
	/usr/sbin/acct/accton	/usr/sbin/acct/accton
/var/adm/btmp	/usr/bin/login	/usr/bin/lastb
/var/adm/sulog	/usr/bin/su	
/var/adm/cron/log	/usr/sbin/cron	
/var/spool/mqueue/syslog	/usr/sbin/sendmail	/usr/bin/mailq
/var/spool/mqueue/sendmail.st		
/var/spool/uucp/.Admin/*	Many of the UUCP commands	
/usr/spool/uucp/.Log/*	such as : /usr/lbin/uucp/uuxqt	
	/usr/lbin/uucp/uucico	
/var/adm/lp/log	/usr/sbin/lpsched	/usr/bin/lpstat
	/usr/bin/lp	
/var/adm/messages	/usr/sbin/dmesg	
/var/adm/sw/swinstalllog	/usr/sbin/swinstall	
/var/adm/sw/swremove.log	/usr/sbin/swremove	
/var/adm/sw/sw*.log	All /usr/sbin/sw* commands	

Note

Never empty a log file with:

```
# rm logfile
# touch logfile
```

If you do this, the permission and modes will become incorrect.

Core Files

Core files can appear anywhere on the system. Sometimes it will be apparent that one has been created; often it will not be apparent. A core file is a core image of a terminated process that is created when certain signals are received. The most common causes are memory violations, illegal instructions, floating point exceptions, bus errors, and user-generated quit signals. The core file is written in the process' working directory.

Core files can be used to determine what the process was doing at the time of the termination. However, they are frequently left around to clutter the file system and so they should be periodically located and removed. One way to do this is to use the `find` command:

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

You should periodically execute this command, or schedule it to be run periodically. SAM also has the capability to find and remove core files.

Large, Old Files

Often large files are created for specific purposes, and the creator subsequently forgets to remove them when they are no longer needed. A number of these files can mean wasted disk space on your system. You should periodically locate files that are large and have not been accessed (or modified) for some amount of time that you determine, based on your knowledge of your system use. One way to do this is:

```
find / -size +nbytesc -atime ndays
```

Where *nbytes* is the number of bytes and *ndays* are the number of days (file has been accessed in *n* days.)

You would probably want to print a listing of the files that you locate, and then contact the creators of the files to see if they can be removed.

You can also use SAM to locate a list of files of a specified size and access time.

13-3. SLIDE: What If You Run Out of Disk Space?

What If You Run Out of Disk Space?

- Extend a file system
  - Into existing free space
  - Onto a new disk (add a new disk)
- Move data to a new file system
- Move a disk to a new volume group
- Move a volume group to another system
- Remove optional software
- Remove unused file sets
- Format manual pages

H6296 13-3 145

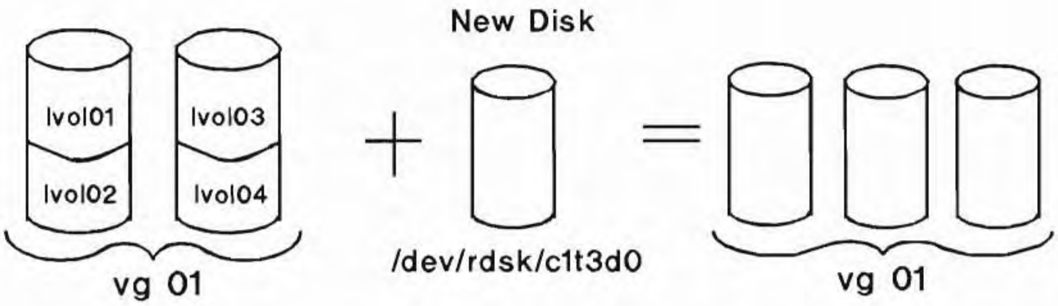
Student Notes

One of the most common tasks for a system administrator is to free some disk space and/or move data around. There are several things you can do when you're running out of disk space. They are shown on the slide and we will cover all of them in the following topics.

Try to avoid the situation where a file system becomes completely full. Especially on multiuser systems or network-servers it can be quite painful. It is a good advice to take action earlier, for example when the file system is filled up to 90%.

13-4. SLIDE: Extend a Volume Group

Extend a Volume Group



pvcreate /dev/rdsk/c1t3d0
vgextend /dev/vg01 /dev/dsk/c1t3d0
Check with: vgdisplay vg01

H6296 13-4 146

Student Notes

In order to extend a file system, you may first need to extend the volume group by adding a new disk. This slide shows how to extend a volume group.

Let's assume that logical volume 3 needs some space. That is, it has 200 Mbytes and we want it to have 400 Mbytes.

First we have to create the data structures for the LVM system on the new disk. This is done with the `pvcreate` command.

```
# pvcreate /dev/rdsk/c1t3d0
```

Now we have to extend the volume group where we want to add disk space.

```
# vgextend /dev/vg01 /dev/dsk/c1t3d0
```

Options to the vgextend command are:

-x *extensibility*	Sets allocation permission for additional physical extents on the physical volume.
volume_group_name	Used to extend an existing physical volume group by adding all the physical volumes in the *physical_volume_path*. Will create the volume group if it does not exist.
physical_volume_path	Names of one or more physical volumes to be added to t he group.

Checking Volume Group Configuration with pvdisplay and vgdisplay

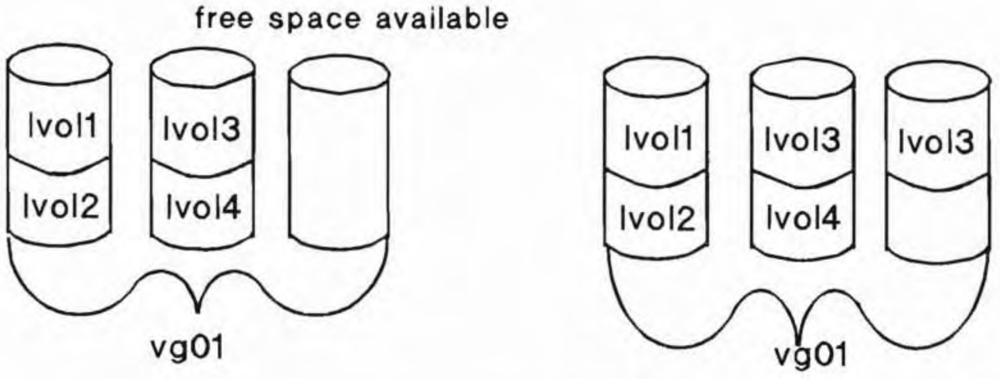
It always a good idea to check that everything works the way you had expected. This can be done with pvdisplay and vgdisplay :

```
# pvdisplay /dev/rdsk/c1t3d0
---Physical volumes ---
PV Name                /dev/dsk/c1t3d0
VG Name                /dev/vg01
PV Status              available
Allocatable            yes
VGDA                   2
Cur LV                 6
PE Size (Mbytes)       4
Total PE               124
Free PE                124
Allocated PE           0
Stale PE               0

# vgdisplay /dev/vg01
--- Volume groups ---
VG Name                /dev/vg01
VG Status              available
Max LV                 255
Cur LV                 4
Open LV                4
Max PV                 16
Cur PV                 3               <—new disk; when we started we
Act PV                 3               had only 2 disks
Max PE per PV          1016
VGDA                   8
PE Size (Mbytes)       4
Total PE               1754
Alloc PE               1754
Free PE                0
Total PVG              0
```

13-5. SLIDE: Extend a HFS File System

Extend a HFS File System

1. umount /dev/vg01/lvol3
2. lvextend -L 400 /dev/vg01/lvol3
3. extendfs /dev/vg01/lvol3
4. mount /dev/vg01/lvol3



H6296 13-5 147 © 1995 Hewlett-Packard Co

Student Notes

Let's assume that logical volume 3 of volume group vg01 is 200 Mb in size and we need 400Mb. This is very easy to do with LVM on an HFS file system.

You must have some disk space available into which you can extend the file system. This may mean that you need to add a new disk. Or, you may be able to take advantage of some unused part of a disk that is already in the volume group.

Before you can extend the logical volume you first have to unmount the file system. This makes the logical volume inactive.

```
# umount /dev/vg01/lvol3
```

Now you can extend the logical volume to 400Mb by using the `lvextend(1M)` command. You can also specify that you want the increased disk space allocated to a specific disk (physical volume). Otherwise, LVM will determine where to allocate the space.

```
# lvextend -L 400 /dev/vg01/lvol3
```

You can check the logical volume with `lvdisplay`

```
# lvdisplay /dev/vg01/lvol3

--- Logical volumes ---
LV Name                     /dev/vg01/lvol3
VG Name                     /dev/vg01
LV Permission               read/write
LV Status                   available/syncd
Mirror copies               0
Consistency Recovery        MWC
Schedule                    parallel
LV Size (Mbytes)            400
Current LE                  100
Allocated PE                100
Stripes                     0
Stripe Size (Kbytes)        0
Bad block                   on
Allocation                  strict
```

The information about mirror copies, consistency recovery, and schedule pertains to mirroring capabilities provided by the optional product HP MirrorDisk/UX.

The size of the file system is still the same. HP has enhanced the standard LVM utilities with the `extendfs` command (for HFS only). You have now the ability to extend the file system instead of backing up the file system, make a new file system and restoring the files.

```
# extendfs /dev/vg01/lvol3
```

The `extendfs` command will increase the size of the file system including the creation of additional superblocks and the update of existent superblocks.

Now it's time to mount the file system.

```
# mount /dev/vg01/lvol3
```

You can check the size with `bdf`.

Note

Expanding a Base-JFS File system is not available. This functionality is available with On-line-JFS through the `fsadm` command.

13-6. SLIDE: Moving /home to a New File System

Moving /home to a New File System

```
# newfs -F hfs /dev/vg01/rlvol1
# mkdir /home.new
# mount /dev/vg01/lvol1 /home.new
# cd /home
# find .  | cpio -pdumv /home.new
# cd /
# rm -rf /home/*
# umount /home.new
# mount /dev/vg01/lvol1 /home
# rmdir /home.new
```

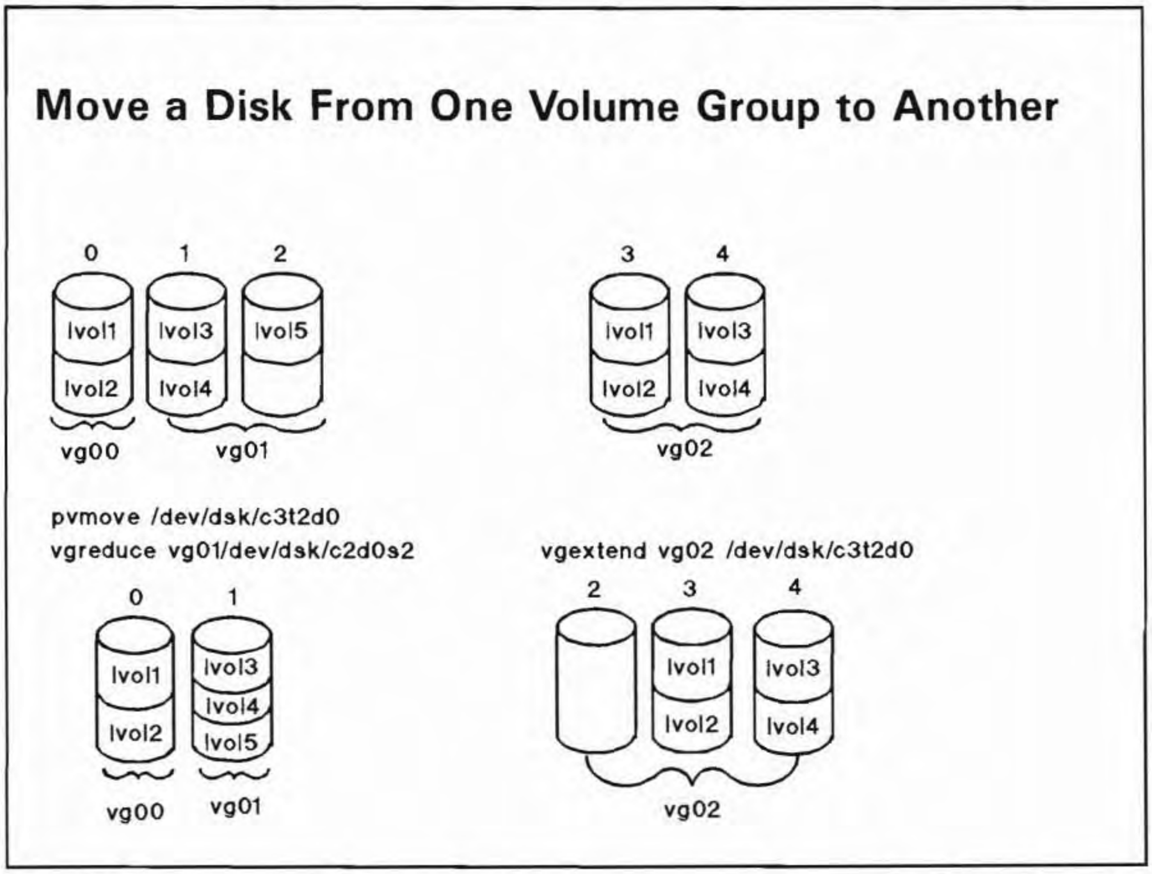
H6296 13-6 148

Student Notes

Many times when a file system becomes full, the system administrator will want to move a part of the file system to a new file system. The steps to perform this task are shown on the slide.

We assume here that we have run out of space on the root file system, and need to move to a larger partition. Another reason for doing this is to keep the users data away from the root file system. We also assume that the destination partition was not used before now.

13-7. SLIDE: Move a Disk From One Volume Group to Another



H6296 13-7 149

Student Notes

The slide shows an example of how to move a disk from volume group 01 to volume group 02. The steps are explained in detail below:

You can move logical volume data from one disk to another disk within a volume group, or you can move all the data from one disk to another disk in a volume group. In the example we move all the data from disk c3t2d0 which belongs to volume group 01, and on to the other disk. We assume that there is enough space on disks c0t5d0 to hold all the data.

```
# pvmove /dev/dsk/c3t2d0
```

Now the disk c3t2d0 is empty, we can disconnect it from the volume group 01, and we have a lonely disk c3t2d0.

```
# vgreduce  /dev/vg01 /dev/dsk/c3t2d0
```

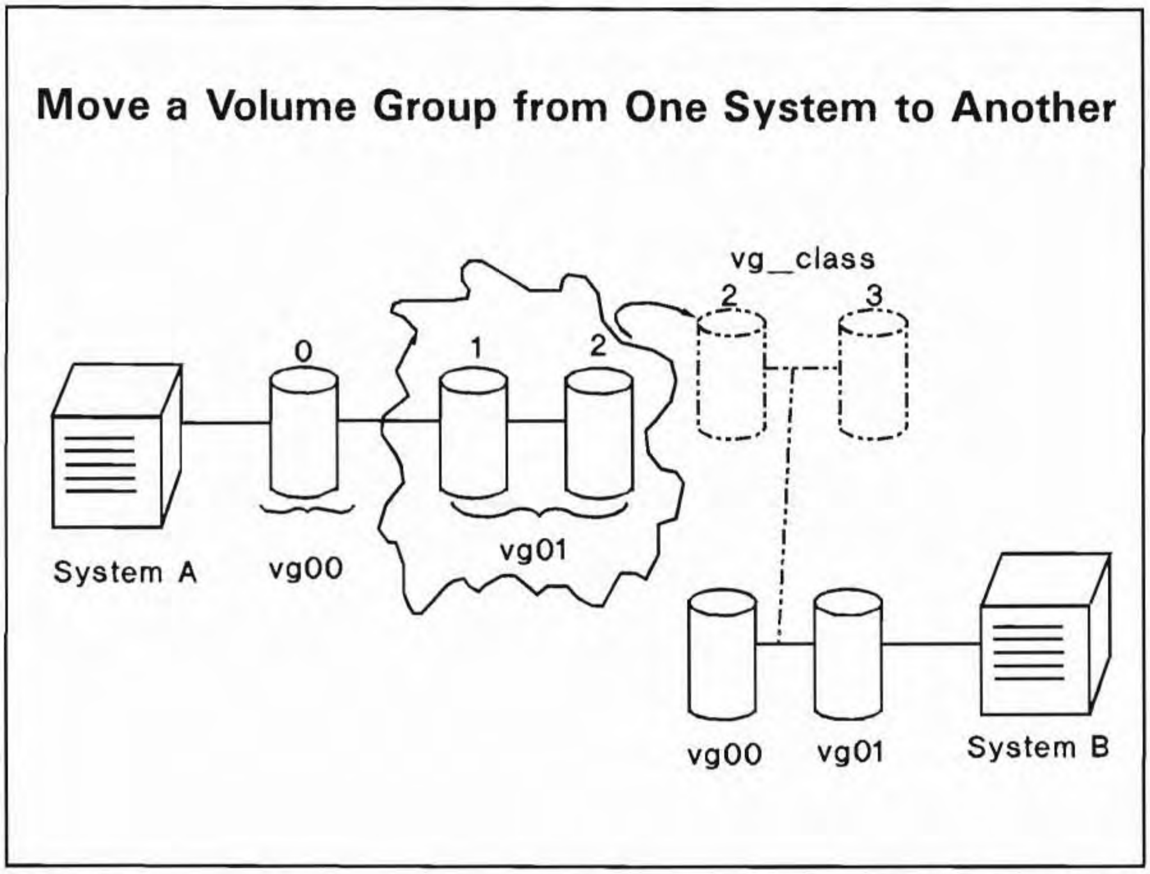
We can add disk c3t2d0 to volume group 02 using the `vgextend` command. The result is that volume group 02 now has 3 disks.

```
# vgextend vg02 /dev/dsk/c0t3d0
```

You might want to check your new configuration with `vgdisplay -v`

You now have disk c3t2d0 in volume group 02 and you can create or extend logical volumes and file systems in volume group 02.

13-8. SLIDE: Move a Volume Group from One System to Another



H6296 13-8 150

Student Notes

You may need to move your disks from one hardware location to another on the same system, or to an entirely different system. The file `/etc/lvmtab` must be changed to reflect the new hardware locations and device files for the disks.

You cannot edit `/etc/lvmtab` directly. You must use the `vgexport` command on the origination system to remove the definition of a volume group from the system without removing it from the disks. The disks can then be brought to the destination system and used there. You must execute `vgimport` on the destination system to have `/etc/lvmtab` properly updated. `vgimport` will also set up LVM special files for the new volume group.

Example

To move two disks in the volume group /dev/vg01 to different hardware locations or a different system:

1. Make the volume group unavailable:

```
# vgchange -a n /dev/vg01
```

2. Remove the volume group information from /etc/lvmtab. Use the -m option to specify a mapfile to hold the information that is removed from the lvmtab file. This file can be used when you set up the volume group at the new destination:

```
# vgexport -v -m map_class vg01
```

3. Save the *map_class* file on a tape, or transfer it across the network to the destination system. Shutdown the system, disconnect the disks, and reinstall the disks at the new destination.
4. Add the disks to the destination system. Be sure to note the device instance number so you can refer to the device files created for them. For this example, the instance number is 1, and the disk address 2 and 3. Restore the *map_class* file if you stored it on a tape.
5. On the destination system, create a new volume group directory and group file. This step is the same as when you are creating a new volume.

```
# mkdir /dev/vg_class
# mknod /dev/vg_class/group c 64 0xhh0000
```

*hh*is the hexadecimal representation of the volume group number

Note The next command is used only if you want to restore all the structure of the volume group (logical volume, file systems and files) on the new system. Otherwise you would have to run a pvcreate command.

6. Issue the vgimport command:

```
# vgimport  -v -m map_class /dev/vg_class /dev/dsk/c1t2d0 /dev/dsk/c1t3d0
```

7. Activate the newly imported volume group:

```
# vgchange -a y /dev/vg_class
```

8. Mount the file systems that previously used on the first server.

```
# mount /dev/vg_class/lvol3 /test_newvg
```

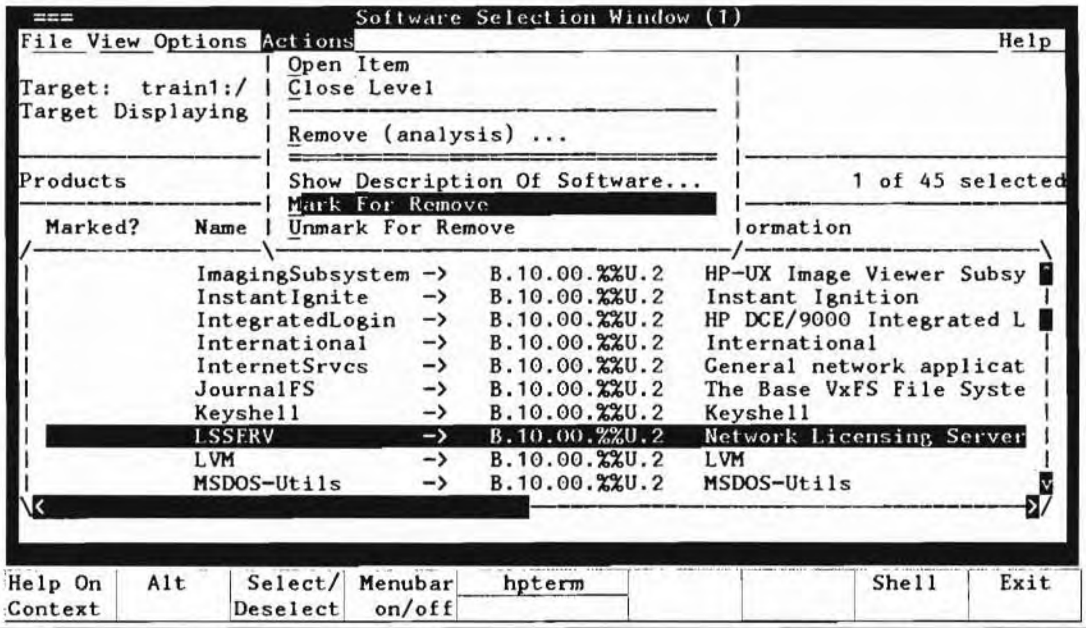
13-9. TEXT PAGE: Summary of LVM Commands

LVM provides numerous commands to configure and manipulate Logical Volumes. The following are functions of those commands (all of them are in `/sbin` directory) :

`extendfs`	Extend an off-line file system
`lvchange`	Changes the characteristics of a logical volume
`lvcreate`	Creates a logical volume in a volume group
`lvdisplay`	Displays information about logical volumes
`lvextend`	Increases the number of physical extents allocated to a logical volume
`lvlnboot`	Prepares a logical volume to be a root, swap or dump volume
`lvmerge`	Merges previously mirrored volumes into one logical mirrored volume
`lvreduce`	Decreases the number of physical extents allocated to a logical volume
`lvremove`	Removes one or more logical volumes from a volume group
`lvrmboot`	Removes a logical volume link to root, swap or dump volume
`lvsplit`	Splits a mirrored logical volume into two logical volumes
`lvsync`	Synchronizes logical volume mirrors that are stale in one or more logical volume
`pvchange`	Changes the characteristics of a physical volume in a volume group
`pvcreate`	Creates a physical volume that can be used as part of a volume group
`pvdisplay`	Displays information about one or more physical volumes within a volume group
`pvmove`	Moves allocated physical extents from one physical volume to another
`vgcfgbackup`	Saves LVM configuration for volume group
`vgcfgrestore`	Restores LVM configuration onto the volume group
`vgchange`	Sets the status of a volume group to on or off
`vgcreate`	Creates a volume group
`vgdisplay`	Displays information about volume groups
`vgextend`	Extends a volume group by adding physical volumes to it
`vgexport`	Exports a volume group from a system
`vgimport`	Import a volume group onto the system
`vgscan`	Scans the system's physical volumes for volume groups
`vgreduce`	Reduces a volume group by removing one or more physical volumes from it
`vgremove`	Removes the definition of one or more volume groups from the system
`vgsync`	Synchronizes logical volume mirrors that are stale in one or more volume groups

13-10. SLIDE: Removing Optional Software: swremove

Removing Optional Software: swremove



H6296 13-10 151 © 1995 Hewlett-Packard Co

Student Notes

It is recommended that when you do your installation or update you install all products on your distribution media. After everything is installed, you can remove the file sets that you do not want. The file sets that are installed on your system are listed in the directory **/var/adm/sw/products**.

You may leave all products installed on your system, and then at a later date you may need to recover some additional space. If you know that there are specific software products, including parts of HP-UX, that you do not need, you can use this tool to remove their file sets, thereby recovering disk space.

To use the interactive utility **swremove(1M)** command to remove the unwanted optional file sets, simply type:

```
# swremove
```

You will have to wait a short time while the products or bundles that are loaded on your system disks are scanned for size and dependency information. During the delay you will see the following message on the screen:

```
Contacting the target  ...
```

Then the screen shown in the slide will appear, listing the products loaded on your system, along with their revision and their sizes. At this point you can select a product for deletion and ask for (Selected) option : `Mark for Remove` if you want to remove all the products, `Open Item` if you want to show the subproducts or file sets and select one or more of them. Choose `Remove` and `Close Level` on the menu bar to come back on the Main Menu. When you select a product for removal it will again take some time for the system to traverse the list of dependencies for the file sets in the product you select for removal. This is usually ten or twenty seconds of intense disk activity before the y is posted to the screen.

Note

Do not press y twice, thinking that it did not recognize the first one, because this may cause you to select another product you did not intend. The program moves the cursor to the next line after a selection is made. A second affirmative response will select the next partition.

After you have selected all products or file sets to remove, choose `Remove` and `Remove (analysis)` on the Menu bar to start removing.

If a dependency exists for a software selected for deletion, you will get a pop-up window that warns you about it and give you the list of the dependent software. So you must unmark what is currently marked OR mark what is dependent on it in the software selection window.

Once you have selected all the software you want to delete, you must choose `Remove` on the Menu bar. You will be asked to confirm your choice to start the removal process. A pop-up window will appear. The system will analyze your choice. After analysis has completed choose `OK` to begin the removal, or `cancel` to return to prior selection screen.

Of course the number and sizes of files you choose will affect the number of kilobytes to be removed. This is you last chance to back out. You must answer `yes` to start removing files. Once you have confirmed your choice to start the removal process, a new window will appear to provide a status report on the removal process.

When you confirm that you want to remove the file sets, you will be able to observe the process on a screen which presents you with a continuous status report about the process of software and partition removal. It will display the progress and any error messages until the process completes.

The removal may take several minutes as individual file sets are removed. You will see the status at any time.

At the end of the removal, select `Exit` if there are no other products and file sets to remove.

13-11. SLIDE: Find Unused File Sets with `freedisk`

Find Unused File Sets with `freedisk`

Syntax:

```
freedisk [ -a n ] [ -v ]
```

-a	check access in previous *n* days instead of since install date.
-v	verbose output

Example:

```
#freedisk -a 50

    ----------------------------------------------------------
     freedisk Utility 1.6

    This utility helps you find file sets to remove to recover
disk space.
     ...
```

H6296 13-11 152

Student Notes

The main focus of the `freedisk` tool is recovering disk space by identifying file sets which have not been used since they were installed and then optionally using `swremove(1M)` to remove those file sets.

Backups and archives of your system may modify access time of system files, which may cause `freedisk` to erroneously report that there were no unused file sets. If you suspect that this is the case, `freedisk` has a command-line option to correct for this situation. Calculate the number of days since the backup or archive that accessed the files and invoke `freedisk` with an `-a` *nn* option in which *nn* is the number of days. Instead of looking for file sets that have been not been accessed since the file set was loaded, `freedisk` will look for file sets that have not been accessed in the last *nn* days.

The tool has limits on how small a value for *nn* that it will accept. For example, if the system has not been rebooted in 20 days and you specify the arguments `-a 19`, `freedisk` will generate an error. It does so to avoid improperly identifying some file sets as not in use, even though they really are.

The error occurs because some file sets might be completely unused except for a few daemons in them. If the daemon has not been restarted (as it would be following a reboot) in the past 19 days, that file and its file sets might appear unused.

Example

```
#freedisk -a 50

-----------------------------------------------------------------

    freedisk Utility 1.6

This utility helps you find file sets to remove to recover
disk space.

If you attempted to update your system but were not able to
due to insufficient disk space, this utility can help locate file
sets which can be removed.

If you would like to remove the kernel build file sets which
are not normally removable, this utility can also perform that
task.


Also, run freedisk with an argument of '-?' to see a list of
options.

-----------------------------------------------------------------

Press [Return] to continue...

-----------------------------------------------------------------

freedisk Utility

The freedisk utility provides two phases of file set removal:

Phase 1 - Identifies file sets which appear to be unused.
Allows you to remove all, some or none of them with
swremove(1M).

Phase 2 - Allows removal of kernel build file sets, which
are not normally removable.  (Most administrators should
skip this phase)

-----------------------------------------------------------------

Press [Return] to continue...

-----------------------------------------------------------------

freedisk Utility

You can quit the utility at any time by pressing [CTRL]-[C].

Some of the functionality for finding other unused or large
files which used to be available in previous versions of this
tool
is instead available in sam(1M) in the Routine Tasks functional
```

```
area of that tool.

NOTE:    File Sets you choose to remove using freedisk can be
reloaded using swinstall(1M), provided you have the
appropriate update media.

----------------------------------------------------------------

Do you want to continue running freedisk?

Press [y] for yes or [n] for no, then press [Return] y

----------------------------------------------------------------

Phase 1: Check Unused File Sets

This phase attempts to locate file sets which have not been
accessed in the last 50 days.  (A file set is a collection of
files
which are related to one another.)

First freedisk will examine all the file sets to determine
which appear to be unused.  This is done by checking whether any
files in
the file set have been accessed any time in the last 50 days.

Next freedisk will begin an interactive session of
swremove(1M).  Using that tool you can choose to remove all, some

or none of these file sets.

----------------------------------------------------------------

Do you want to proceed with Phase 1?  (Answer 'n' to skip this
phase)

Press [y] for yes or [n] for no, then press [Return] y

----------------------------------------------------------------

Please wait...

It will take several minutes to collect the necessary
information.
----------------------------------------------------------------

Counting file sets...

----------------------------------------------------------------

No unused file sets were found.
----------------------------------------------------------------

Press [Return] to continue...

----------------------------------------------------------------

Phase 2:  Remove File Sets Used for Kernel Builds

This phase of freedisk allows you to remove file sets that
```

```
contain files that are needed only for building kernels.

In most cases, most system administrators should skip this
phase.  It is offered only as a convenience to administrators
that do
not build kernels or update systems to new OS revisions and wish
to reclaim disk space.

If you are uncertain about whether to perform this phase, you
should skip it.

----------------------------------------------------------------

Do you want more information on this phase?  (Answer 'n' to skip
this phase)

Press [y] for yes or [n] for no, then press [Return] n
```

Note

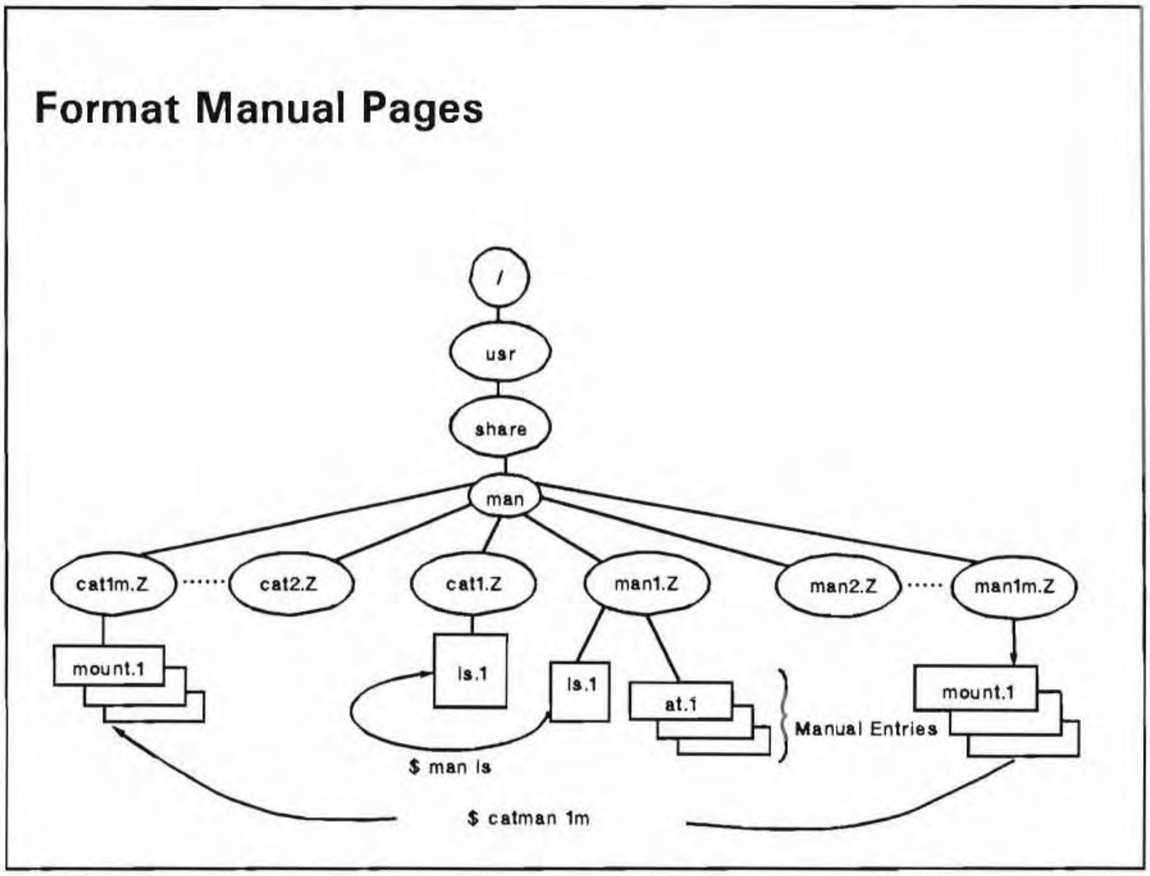
The freedisk tool has a second mode (Phase 2) that allows system administrators to remove file sets that contain files that are needed only for building HP-UX kernels. This optional removal occurs regardless of file set usage

This capability is intended for sophisticated system administrators that have systems on which they never intend to re_build a kernel for any reason at all. This feature can save 8 Mb or more of disk space.

If you will be performing any of the following activities on the system in the future, you should not remove the kernel build libraries with freedisk:

- Installing kernel patches
- Loading file sets with additional kernel modules
- Performing kernel parameter tuning of any kind
- Loading a new version of the HP-UX Operating System using swinstall

13-12. SLIDE: Format Manual Pages



H6296 13-12 153

Student Notes

Before we explain catman it helps if you understand how the manual entries are stored on the system. All the entries, both unformatted (source) and formatted, are stored in sub-directories under the /usr/share/man directory. Both unformatted and formatted entries can appear in compressed or uncompressed form. This gives us four possibilities.

	unformatted	formatted
compressed	/usr/share/man/man*X*.Z/*	/usr/share/man/cat*X*.Z/*
uncompressed	/usr/share/man/man*X*/*	/usr/share/man/cat*X*/*
		if exist or :
		/usr/share/cat/cat*X*.Z/*

X is the section: 1, 1m, 2, 3, 4, 5, 7, 9.

The unformatted entry is an `nroff` source.

`catman` (with no options) creates the formatted versions of the on-line manual pages from the compressed `nroff` source files. (Compressed files end in `.Z`.) When you invoke `catman`, each manual entry in the `man`*X*`.Z` directory is examined and those whose formatted versions are missing or out-of-date are created, compressed, and put it into the `cat`*X*`.Z` directory. Before invoking `catman`, it is best if you remove all of the `catX` directories (leaving only the `catX.Z` directories).

If you want to create uncompressed entries, make sure the `catX` directories exist and invoke `catman -z`. This command puts an uncompressed version of each entry in its appropriate `cat`*X* directory.

You can choose to format only certain sections of the manual pages by invoking `catman` *section*, where *section* is the number of the section you wish to format, for example sections 1, 1m and 2 as shown by:

```
# /usr/sbin/catman 11m2
```

There are several user commands that have names used to reference the manual pages that appear in more than one section of the manual. Separate pages may exist for a user command (section 1), system administration command (section 1m), library routine (section 3), system call (section 2), file format (section 4), special file (section 7), or miscellaneous entry (section 5). Several examples are: `mkdir(1)`, `mkdir(2)`, `mknod(1M)`, `mknod(2)`, `mknod(4)`, `mount(1M)`, `mount(2)`, and `mount(3)`.

To read a specific manual page, type the `man` command with the section number as the first parameter. Several examples follow:

```
$ man 1 mkdir  OR $ man 2 mkdir
```

You can choose not to format any manual entries with `catman`. If you do not format entries, then the first time you try and access a manual entry you will get this message:

```
Reformatting entry.  Wait...
```

You will wait a short period of time before the manual entry appears on your screen. This is usually less than 10 seconds, but may be much longer for very large manual pages such as "sh-posix". After the first time the page is read with `man` *some-command*, the entry for *some-command* is added to the appropriate `cat`*X* or `cat`*X*`.Z` directory, and in subsequent accesses, the user does not have to wait. This is a "build-as-you-use" alternative. The system fills the `cat`*X* directories as the users access commands with `man`. For this to work, the `cat`*X* directories must exist. If they do not, you can create them with the following script:

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

Note

If you execute catman it requires several megabytes of disk space and on some systems can take some time to complete. By doing so you get improved performance, but you will have the manuals twice. To recover some disk space, you should delete the unformatted directories after using catman.

Running catman allows you to use the `man -k keyword` command to locate documentation, executables and libraries by keyword lookup. A reference to the keyword in the one line description of the command is used as a search key for the `usr/share/lib/whatis` database built by the catman command. All commands (files, man pages) with that keyword are also listed. This is like a "conceptual" cross reference to commands based on the keyword used.

13-13. LAB: Hands-On File System Management

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. You have just run out of space on one of the disks. Use SAM to search for files larger than 50000 bytes, with a "modification since" time of 1 day, in the `/usr/bin` directory. Also search the `/etc` directory for core files.

2. Copy the file `/var/adm/sw/swinstall.log` to your home directory and add it to SAM's list of ASCII log files to check. Have SAM check the logfiles on the system, and trim *your* `swinstall.log` file. Keep 25% of the file.

3. Add one disk, if it is available. And extend the volume group specified by your instructor.

4. Extend one logical volume specified by your instructor. Add 50 Mb.

13-14. REVIEW: Check Your Understanding

1. List and define two commands to monitoring the free disk space on the system

2. What are some different solutions to recover space on your file system ?

3. Give two HP-UX commands which enable you to remove some file sets?

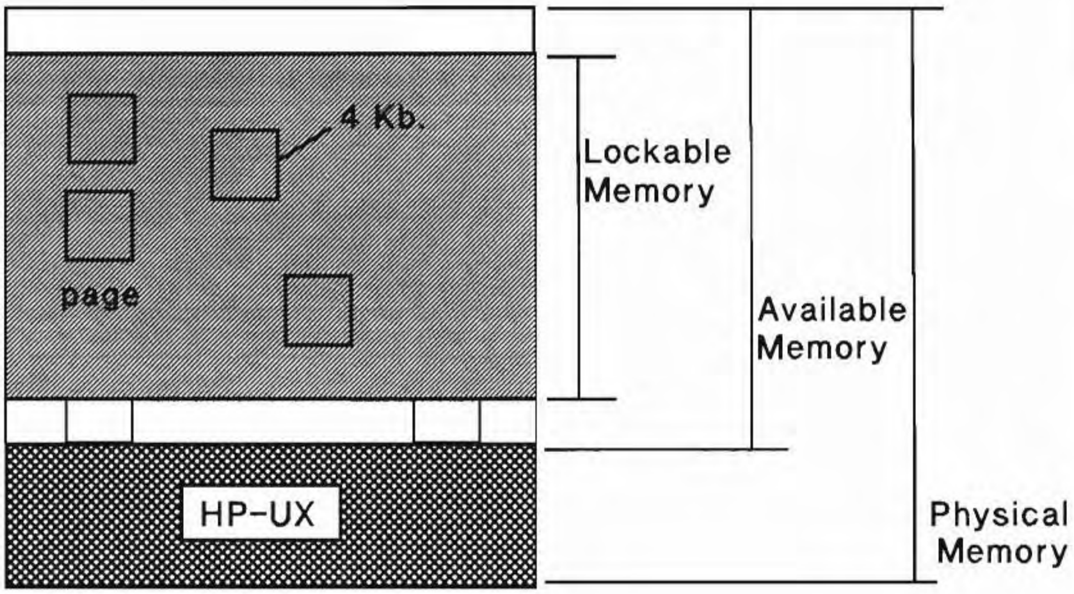
Module 14 — Managing Swap Space

Objectives

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

- Explain the concept of physical and virtual memories as applied to a demand page virtual memory system.
- Explain the difference between physical memory, available memory, and lockable memory.
- Identify the amount of memory your system has and how much of that memory is lockable.
- Define swap space and describe why it is necessary.
- Define various terms that describe swap types.
- Evaluate your own swap space needs.
- Set up device swap space on your system.
- Set up file system swap space on your system.
- Describe two methods for managing swap space on your system.

14-1. SLIDE: System Memory



H6296 14-1 154

Student Notes

Physical memory is the RAM (random access memory) installed in your computer. At system startup, the system displays on the system console the amount of physical memory installed:

```
real mem = xxxxxxx
```

Not all physical memory is available to HP-UX processes. Some memory is reserved for kernel code and data structures. The amount of memory remaining is referred to as **available memory**, and is used by the system for demand paging. During system startup, the system displays on the system console the amount of available memory:

```
avail mem = xxxxxxx
```

Module 14 — Managing Swap Space

All or part of available memory can be locked by a subsystem or by user processes. **Lockable** 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 unlockable memory must be available to prevent deadlock.

During system startup, the amount of memory that may be possibly locked is displayed on the console:

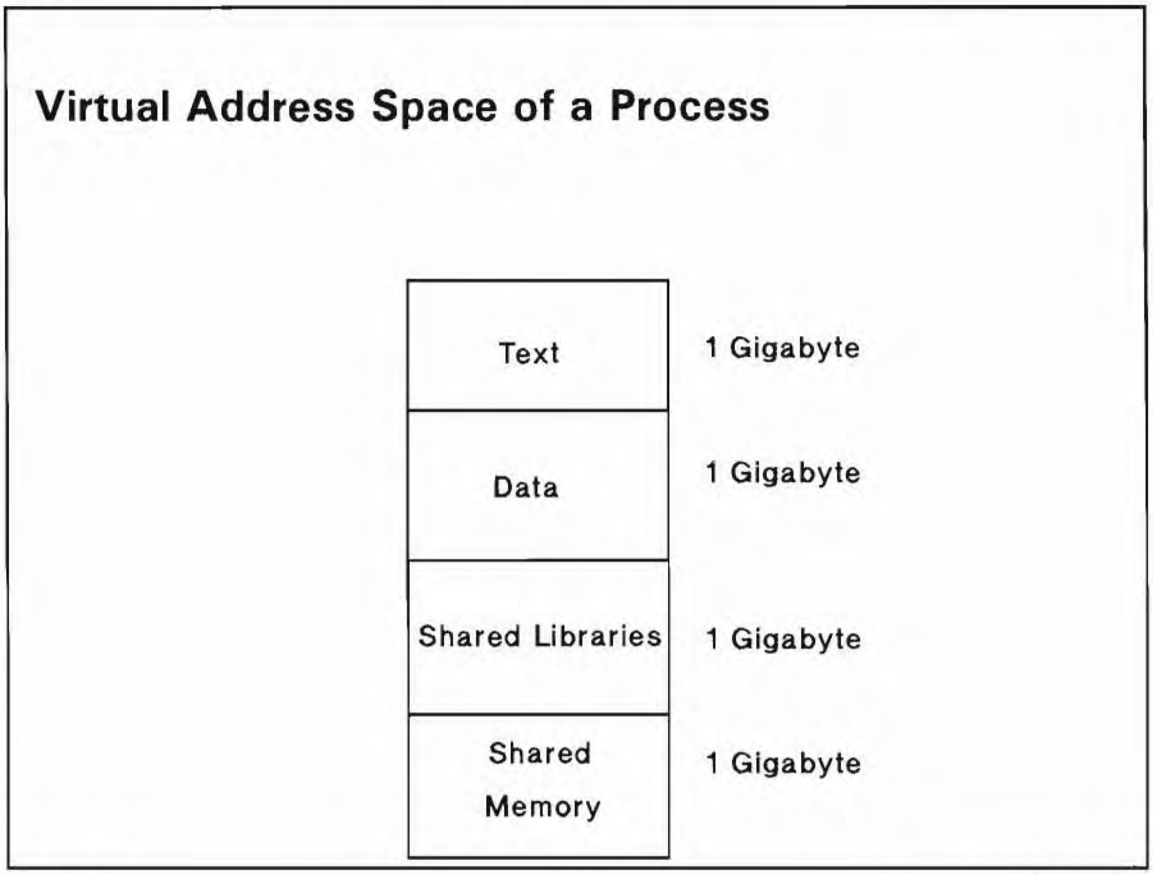
```
lockable mem = xxxxxxx
```

Available memory minus the memory locked by subsystems or user processes is the memory that is actually usable for virtual memory demand paging. The system parameter `unlockable_mem` reserves the amount of memory that cannot be locked.

Note

The `dmesg` command will show you the messages output by your system when you boot up. These messages include the amounts for `real mem`, `avail mem`, and `lockable mem`, reported in bytes.

14-2. SLIDE: Virtual Address Space of a Process



H6296 14-2 155

Student Notes

PA-RISC systems have one very large global address space, which is 262,144 gigabytes in size. The global virtual address space is divided into 65,536 four gigabyte spaces. Each space is logically divided into four 1-gigabyte quadrants.

Every process has one of these spaces with 4-gigabyte virtual addresses. Each of the quadrants is described below.

text segment	contains the code and is shared, that means another process which runs the same program uses the same segment, but has a different instruction pointer.
data segment	contains the data of this particular process.
shared library segment	shared with every process, and contains instruction code. This code will linked at run time to the process' code. This is code from the shared libraries.

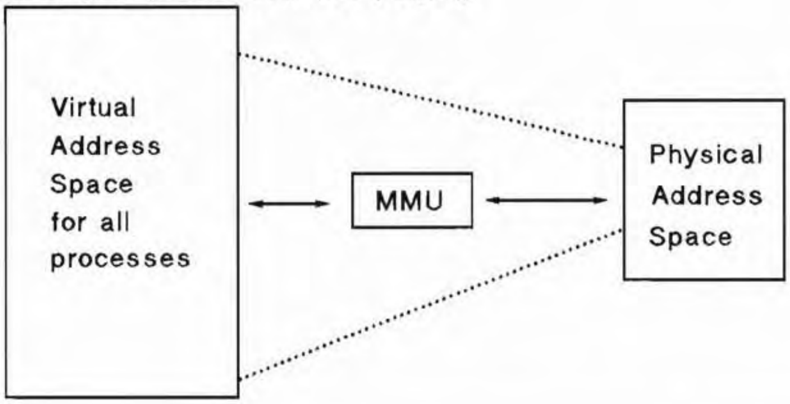
shared memory segment	used by processes which want to interchange data with other processes. Databases usually exchange data between their processes with shared memory segments.

14-3. SLIDE: Memory Management

Memory Management

The Memory Management Unit (MMU) manages:

- Demand-paged virtual memory system
- Virtual address space
- Physical memory
- Deactivating at processes and paging



H6296 14-3 156

Student Notes

The term **memory management** refers to the rules that govern physical memory and allow for efficient sharing of the system's resources by user and system processes. The hardware and software that enforces these rules is collectively called the **Memory Management Unit** (MMU).

The MMU allows the total size of user processes to exceed physical memory by using an approach called **demand-paged virtual memory**. By using the rules associated with the concept of virtual memory, when a process executes, parts of the process are brought into main memory only as needed, that is, on demand. The system uses a combination of deactivation scheme and paging to manage virtual memory.

Processes are now deactivated, (that is, will be taken off of the run queue), rather than swapped. The deactivated process' pages will be pushed out to disk over time by the pager.

Module 14 — Managing Swap Space

The **deactivation scheme** involves processes which are deactivated when the system detects thrashing as well as when the system experiences memory pressure, whereas **paging** involves moving smaller units (called pages) between RAM and mass storage when needed, or *on demand.*

For example, by using paging, a process can be written that is 2GB of code and data. This process can be executed on a system that has only 32MB of RAM. In the above example, the process' virtual or logical address space is 2GB. **Virtual address space** or logical address space is all valid memory addresses; whereas **physical address space** is the actual RAM purchased, 32 MB in the example.

In HP-UX, the basic virtual memory unit is a **page**. That is, processes are stored in blocks of memory called pages of physical memory. A page is 4 Kbytes in size.

The MMU also ensures that processes do not illegally access each others' address space. Since the MMU operates at the granularity of a page, memory locking, protection, and sharing are supported at the page level.

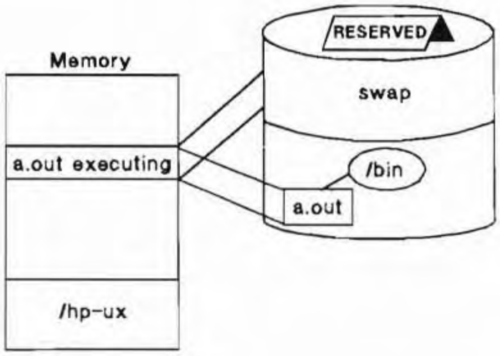
14-4. SLIDE: What Is Swap Space?

What Is Swap Space?

Used by:
- Virtual memory system

Used for:
- Swapping
- paging

Monitored by:
- vhand
- statdaemon
- swapper



H6296 14-4 157

Student Notes

Swap space is an area on a high-speed storage device, reserved for use by the virtual memory system for paging processes.

Physical memory is a finite resource on a computer. This means that only so many processes can fit into physical memory at any one moment in time, even though many more processes may actually be ready to run or execute. Because demand for physical memory may exceed actual supply, the concept of swap was introduced to computer systems. When demand for physical memory is high, portions of processes are pushed out to the swap area on disk. When physical memory demand is low, portions of processes are brought back into memory from the swap area.

The vhand daemon monitors each page of available memory in order to discover pages (a page is 4K bytes) that have not been recently accessed. If demand is high and pages exist that have not been recently accessed or that are deactivated, vhand pushes out or *pages out* these pages to the swap device.

Module 14 — Managing Swap Space

The statdaemon keeps track of how busy the vhand daemon is. As the statdaemon notices that vhand is unable to free up memory quickly enough to meet demands for memory, the swapper is called in order to help. vhand pushes pages of a process out to the swap device; the swapper deactivates the process: The process will be taken off of the run queue.

Most processes are shared text processes. A shared text process is a process that allows its code to be shared among all users currently executing its code. Each user, however, will have a copy of his own data, bss (Block Started by Symbol, an uninitialized data area), and heap. The user's data, bss, and heap will have space allocated in the swap area in case of page out. Since text is shared, it cannot be altered and therefore the original copy of the text from the executable file is always up to date. Swap space is not required for text, since we can always page in text from the executable file. A process' space on the swap device is reserved at process start up time; therefore, swap must be large enough to hold all executing processes.

HP-UX swap space management allows you to allocate swap as needed, while the system is running, without having to regenerate the kernel.

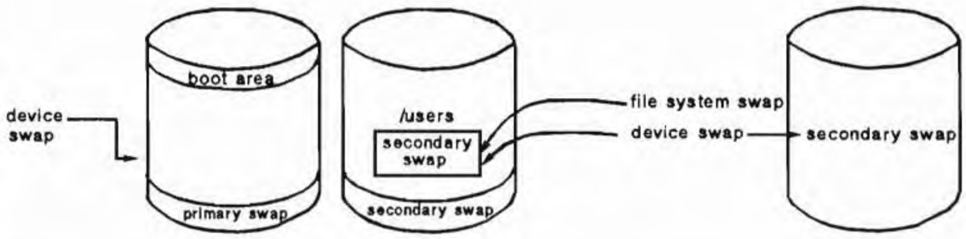
System performance can degrade if the system spends too much time paging - a condition known as thrashing. Remedy thrashing by adding more physical memory to your system.

With the self-deactivation mechanism and deactivation based on thrashing conditions as well as low memory conditions, processes which were not swapped before even though they deserved to be will now be deactivated.

14-5. SLIDE: Types of Swap Space

Types of Swap Space

- Device Swap - a disk or logical volume of a volume group that is used exclusively for swap
- Primary swap - device swap which is located on the system disk and is available at boot
- Secondary swap - device swap used in addition to primary swap
- File System Swap - file system that also supports swapping



H6296 14-5 158

Student Notes

HP-UX has introduced some swap terms to better define its particular brand of swapping:

Device swap	A disk or logical volume of a volume group that is used exclusively for swap. A swap logical volume can be increased as needed.
Primary swap	A special type of device swap which is available at boot. Primary swap is located on the same disk as the root file system, and is initialized by the kernel at boot time.
Secondary swap	Device or file system swap that is used in addition to primary swap and usually located on a disk other than the root disk.
File system swap	File system swap, unlike device swap, is a file system that not only supports files and their data structures, but also has space available for swapping.

Pseudo-Swap — A swap reservation policy that eases requirements for physical swap space for systems with very large system memory.

Device swap space resides in its own reserved area—an entire disk or partition (reserved area or logical volume). Device swap is not taken from file system space. Device swap (except primary swap) can be configured dynamically. It is fixed in size unless you are using Logical Volume Manager.

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.

Pseudo-swap is enabled by a kernel parameter, `swapmem_on`. By default this parameter is set to 1, enabling pseudo-swap reservation.

With very-large-memory systems, it becomes less desirable to have enormous amounts of disk space reserved for swap. Using pseudo-swap, this requirement is relaxed. The use of pseudo-swap avoids the waste of resources that could occur if, for example, a 1-Gb process were running and had reserved 1 Gb of disk space reserved for swap, and then did not need to use the swap space.

Use of pseudo-swap allows you to configure less swap space on the disk than the total size of memory. If the system runs out of physical swap space (a very rare occurrence), it will begin to lock pages in memory. Up to 75% of available RAM can be used as lockable memory. If, however, the lockable memory is all consumed, you will still receive an "out of swap space" error.

Another term that may be seen from time to time is **dynamically allocatable swap**. This is disk space, device swap or file system swap, that can be allocated while the system is running. The only type of swap that is not dynamically allocatable is primary swap.

14-6. SLIDE: Evaluating Swap Space Needs

Evaluating Swap Space Needs

When do you evaluate swap space needs?

- At System Installation Time
- If adding a large number of additional users or applications
- If the error message appears on the system console. The message could be :
  `Sorry pid 1622 was killed due to no swap space.`

RULE OF THUMB FOR INCREASING SWAP SPACE
Total Swap space needed =
Swap space requirements of all applications +
Current amount of swap space

How much swap do you have? Use `swapinfo`

H6296 14-6 159

Student Notes

A system administrator should always be monitoring system swap space, but there are times when this question becomes urgent. Before you install your system, you should try to get an idea of how much swap space you will need. Most application programs need a minimum amount of swap space to operate properly, and this figure is usually contained in the documentation that comes with the application.

Another event that may force you to re-evaluate swap space size is when an error message such as:

```
Sorry pid 1622 was killed due to no swap space.
```

appears on the system console indicating that processes are being terminated because there is no swap space available.

Swap space usage will vary with system load. Since what we are trying to estimate is maximum swap size needed, we need to understand swap size requirements when demand on the system is heaviest. In other words, at what moment in time will the programs that use the most data, bss, and shared memory all be running concurrently? (Bss is data that is uninitialized at the beginning of process execution. **bss** stands for Block Started by Symbol.) We are interested in only data size, bss size and shared memory because these are all reserved swap space when a process is created. Swap space for text is not reserved.

The best way to acquire information on process size is with the `size` command.

```
# size /usr/bin/vi
195035 + 24364 + 782976 = 1002375
```

`size` returns the text, data, and bss sizes for a program. Combine the second and third sizes to estimate the size this program will occupy in the swap area. Don't forget that each user executing a process is allocated his own data and bss. For example, if there are 23 copies of `vi` executing at the system's busiest time, multiply `vi`'s data and bss by 23 ((30376 + 351556) x 23) to account for all `vi`'s swap requirements. Repeat this calculation for each program on the system executing at the system's busiest moment.

Add the shared memory segment sizes to the data and bss sizes for an estimate. Add 256KB to this estimate in order to account for space used by the `exec()` system call during program initialization. Then to be conservative to account for page boundaries and the like, round up your estimate. Modify this estimate to be a multiple of 2MB. 2MB is the default size of `swchunk`, the kernel parameter for swap allocation unit.

How Much Swap Space Do You Have?

Total available swap on a system consists of all swap space available on all devices and file systems enabled as swap. The swapping subsystem reserves swap space at process creation time, but does not allocate swap space from the disk until swap time.

`swapinfo` prints information about device and file system swap space. It tells you the type of swap by location, how much of it is available, how much is used, and how much is free. Under type, you may see `dev`, `fs`, or `hold`. `hold` means that the operating system is reserving an amount of swap space based on possible requirements of currently running processes.

`swapinfo` options

`-t`	Add a totals line with a type of *tot*.
`-a`	Show all device swap areas, including those configured into the kernel but currently unused.
`-d`	Show information about device swap areas only.
`-q`	Quiet mode. Print only a total Kb AVAIL.

Adding Memory and Swap Space for X-Terminals

If you have X-Terminals connected to your system, you have to increase the size of system memory and swap space. The amount of additional Memory and Swap depends of course on the type of the application. But here is a rule of thumb for the system resources needed for each X-Terminal on the server.

Table 14-1.

X-Terminal running	HP-VUE	Application	Sum
System Memory:	2 MB	2-4 MB	4-6 MB
Swap Space:	9 MB	3-9 MB	12-18 MB

14-7. TEXT PAGE: Swap Space Computation

Swap Space Computation Formula

A. SUM (all shared code sizes) of all running processes as shown by `ps -el`. Do not count the page daemon, swapper, or statdaemon processes. The `file` command will show you if a file contains shared text, and you can find the size of the text by means of the size command.

B. SUM (all data and stack sizes) of all processes. By using the `size` command, you can calculate the size of initialized data and uninitialized data (BSS). This represents only part of the total swap space requirements of the process. In addition, you must calculate the amount of dynamic heap and stack space that the program might require. If you are familiar with the program's runtime logic, you might be able to calculate this by looking at requests made to `sbrk` or `malloc`.

   You can approximate by running the program with a typical input stream, and determine the total virtual memory size in number of pages. The virtual memory size can be obtained by running `ps -el` and looking in the `SZ` (size in 512-byte blocks) field for the program you are interested in. Subtract the code size calculated in step 1 from this to get the total data and stack size.

C. SUM (all existing shared memory segment sizes) for shared memory segments created by users via `shmget`. `Ipcs` can be used to show active shared memory segments.

D. Size of the scratch area used by `exec` to hold arguments. The default size of this area is 256 Kbytes; it can be changed by using `uxgen`.

E. Fragmentaticn and overhead.

   Fragmentaticn is the difference between the swap space needed at any given time and the actual amount allocated. The parameter `swchunk` controls swap space allocation.

   Overhead is additional disk space needed to store system-related information when a process is swapped out.

   There is no easy way to figure out an accurate value for fragmentation and overhead. We suggest you take an arbitrary value such as 6Mb.

F. Swap space = A + B + C + D + E.

Module 14 — Managing Swap Space

Swap Space Computation Worksheet

A. For shared code, fill out the code space needed by the process.

```
Process ID                                  Code size.

-----------------                           ------------------

-----------------                           ------------------

-----------------                           ------------------

-----------------                           ------------------
```

B. For each shared process listed above, fill out the data and stack space needed by the process. For each nonshared process, add the process's code size to its data size and enter the amount (that is, total from executing size).

```
Process ID                                  Data size.

                                  (minimum data space = dmmin x
1Kb
                                        block default = 32K%b)

-----------------                           ------------------

-----------------                           ------------------

-----------------                           ------------------

-----------------                           ------------------
```

C. For each shared memory segment, give the shared segment size.

```
Shared Memory                               Segment Size.

-----------------                           ------------------

-----------------                           ------------------

-----------------                           ------------------

-----------------                           ------------------
```

D. Scratch area used for arguments during exec. Default is 256 Kb:

```
-----------------

-----------------
```

E. Fragmentation and overhead. We suggest using an estimate such as 6 Mb.

F. TOTAL AMOUNT OF SWAP SPACE NEEDED IS:

A + B + C + D + E = Total swap space

___ + ___ + ___ + ___ + ___ = ________________

Example

Let's assume that you are going to run four FORTRAN compiles with optimization. This example shows how to calculate the additional swap space that might be required to handle this. Let's start with one compile and gather some statistics.

Note the following example may not be indicative of how much swap space your compiles would use, because the amount of swap space required by HP-UX compilers depends on the size of the program being compiled. Further, compiling with optimization may consume almost twice as much swap space.

```
# fc -o test.f > out [1] 3663

# ps -l

  F S   UID   PID  PPID  C PRI NI   ADDR    SZ    WCHAN TTY     TIME COMD
  1 S   867 22055     1  1 168 20  a0a308   121    1515308 ttyd3p4  0:07 csh
  1 S   867  3663 22055  0 158 20  5f3900    28   4c8024 ttyd3p4 0:00 fc
  1 R   867  3664  3663  4 229 20  d2ae30  6963          ttyd3p4 0:25 f77comp
  1 R   867  3680 22055 49 190 20  f9d260   182          ttyd3p4 0:00 ps

# file /usr/lib/f77comp

/usr/lib/f77comp:        S800 shared executable

# size /usr/lib/f77comp

5720($MILLICODE$) + 141600($LIT$) + 1726120($CODE$) + 8($CODE20$) +
567($UNWIND$MILLICODE$) + 47152($UNWIND$) + 1824($UNWIND$) +
16($UNWIND20$) + 132($RECOVER$) + 164192($GLOBAL$) + 93032($DATA$) +
9424($DATA$) + 8($PFACOUNTER$) + 24400($BSS$) = 2214204
```

From the above information, you can determine that:

- The code size for /usr/lib/f77comp is 5720+1726120+8= 1731848 bytes
- The data size for /usr/bin/f77comp is 93032+9424= 102456 bytes
- The BSS size for /usr/bin/f77comp is 24400
- (BSS and initialized data = 102456 + 24400 = 126856 bytes)

The ps command tells you that this FORTRAN compile uses 6963 pages or 14260224 bytes. Subtracting the code size from this and leaves 12528376 bytes for the total data, heap, and stack. The difference between this data size and the fixed (obtained from size above), is the amount of dynamic space used by the FORTRAN compiler for this program.

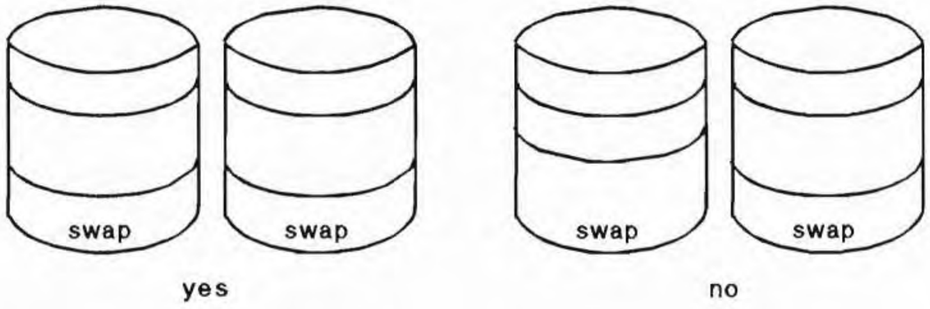
Process ID	**Code Size**	**Data size**
3664	1731848	12528376

Therefore, you need 1.7 Mb of swap space for the compiler text (of which there will be only one copy) and 12.5 Mbyte of swap space for the per process data. This would mean that the system would need approximately 1.7+4 * 12.5 (or 51.7 Mb) of additional swap space for the four compiles to execute in parallel.

14-8. SLIDE: Guidelines for Selecting Device Swap Areas

Guidelines for Selecting Device Swap Areas

- Two swap areas on different physical disks are better than one single swap area of equal size
- Only one swap partition (logical volume or reserved space) per disk
- Device swap areas should be of similar size
- Consider the speed of the disks



H6296 14-8 160

Student Notes

For device swap, you must allocate logical volumes or portions of disk. The size of space reserved on a whole disk is specified on the `newfs` command (`-R` option). You can create logical volumes in the size you require.

The guidelines for selecting swap areas are given above. Most of the recommendations are for performance reasons. You can set up swap space any way you like, but your system will run slower if the above rules are not applied.

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 will cause excessive head movement on that disk and slow the system down.

If you are using LVM, you should 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 space in the smaller device swap area is used, the larger swap area is all that is available and interleaving is no longer possible.

14-9. SLIDE: Guidelines For Selecting File System Swap Areas

Guidelines For Selecting File System Swap Areas

- Avoid using busy file systems such as the root file system
- Use bdf to check file system space availability
- Set priorities appropriately
  - Faster devices over slower devices
  - Infrequently-used file systems over busier file systems
- It is preferable to enable swap on file systems that are located on separate disks or, in the case of LVM, separate physical volumes.

H6296 14-9 161

Student Notes

When you need more swap space and you have no disk space available for additional device swap, you can dynamically add file system swap to your system.

These guidelines are put forth to help the System Administrator decide on 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's best to assign highest priorities to the swap areas that afford the fastest performance. This means, give device swap areas priority over file system swap areas, give faster devices priority over slower devices, and give lower use file systems priority over higher use file systems. 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.

Note

Enabling file system swap allows more processes to run simultaneously. This may result in slower system response time.

Enabling two file systems on the same disk (or, physical volume), can result in excessive head movement and will result in slower system performance.

14-10. SLIDE: Enabling Swap

Enabling Swap

1. Add entry to /etc/fstab

```
block device   directory type   options     backup   pass  comment
                                          frequency  number
/dev/dsk/c0t4d0   ...... swap   pri=1         0        0       # a.
/dev/dsk/c0t3d0   ...... swap   end,pri=1     0        0       # b.
/dev/vg00/lvol3   ...... swap   pri=1         0        0       # c.
/dev/vg01/lvol2 /extra   swapfs min=10,lim=4500,pri=2 0   0    # d.
```

Key to comments:

a. swap on whole disk
b. swap on a reserved space, on the end of a disk
c. swap on a logical volume
d. file system swap

2. Use swapon -a to enable entry

H6296 14-10 162

Student Notes

It is a good advice to add swap as shown on the slide, nevertheless it is possible to add swap temporary with the **swapon** command.

Device Swap

Syntax for adding device swap:

/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` Use space after end of 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 for use with a disk that has both file system and swap space on it (`newfs [-R swap]`).

`-f` forcibly enable device swapping to a device where a file system exists (*destroys the file system!*)

device block special file which is to be used for paging.

Example:

```
# swapon /dev/vg01/lvol3
```

File System Swap

Syntax for adding file system swap:

/sbin/swapon [-m *min*] [-l *limit*] [-r *reserve*] [-p *priority*] *directory*

-m *min* *min* indicates the number of file system blocks the swap system will initially take from the file system.(default 0). The size of a file system block is defined during creation of the file system (see `newfs`).

-l *limit* *limit* specifies the maximum number of blocks the swap system is allowed to take from the file system.(default no limit)

-r *reserve* *reserve* specifies the number of file system blocks in addition to the space currently occupied by the file system that are reserved for file system use only.

-p *priority* same as for device swap.

directory The directory where the file system swapping occurs. The system create a directory named `paging` under the root of the file system for swap. So only one swapfs is available per file system.

Note You cannot unmount a file system when file system swap is active. It is a good idea to specify the limit and increase it step by step

Examle

Let's assume that we have a file system with 8k blocks, and we want to have a maximum of 40Mb swap, with 80Mb always free for the file system `extra`:

```
/sbin/swapon -l 5000 -r 10000 -p 3 /extra
```

Once you have enable dynamic swap, use the `bdf` command to check the file system for the amount of swap space that is being used:

```
bdf -b /extra
```

Note

It is not a good idea to enable file system swap on more than one directory per file system. The swapper will compete for swap space out of the same file system and this will be very inefficient. Also, do not enable swap on file systems that are heavily used, such as root (/) or `/home`.

When you use SAM to add device or file system swap, you may specify that it be enabled at boot. SAM will make the appropriate entries in the `/etc/fstab` file.

`/etc/fstab` entries

`/etc/fstab` contains a list of mountable file system entries. Each file system entry is on a separate line. Fields in the `/etc/fstab` file are:

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`.

backup frequency — reserved for future use

pass number — unused with `swap` an `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. The command `swapon -a` is run as part of the `/sbin/init.d/swap_start` script, which executes when you boot the system to state 1 (in `/sbin/rc1.d` directory) The `-a` option instructs the system to read `/etc/fstab` for swap information, and to active all swap areas.

You can use the `/usr/sbin/swapon -a [-t dev|fs]` command at any time to force the system to read `/etc/fstab` and enable the swap types it finds there.

14-11. LAB: Hands-On, Enabling Swap

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. How much memory does your training system have? How much is lockable, avail, and real?

2. Add your logical volume of 50mb as swap space.

3. Add a free disk partition as swap. Use either SAM or HP-UX commands. Also, make sure that this partition will be enabled as swap each time the system is rebooted.

4. What is the command to add a file system swap in a mounted file system?. How would you make sure that this swap will be enabled each time the system is rebooted?

14-12. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. Define physical memory and virtual memory.

2. Define physical memory, available memory, and lockable memory.

3. Explain the paging process?.

4. What are the two types of swap space that a system administrator can set up?

5. What are the two types of device swap?

6. What are the advantages of device swap?

7. What are the advantages of file system swap?

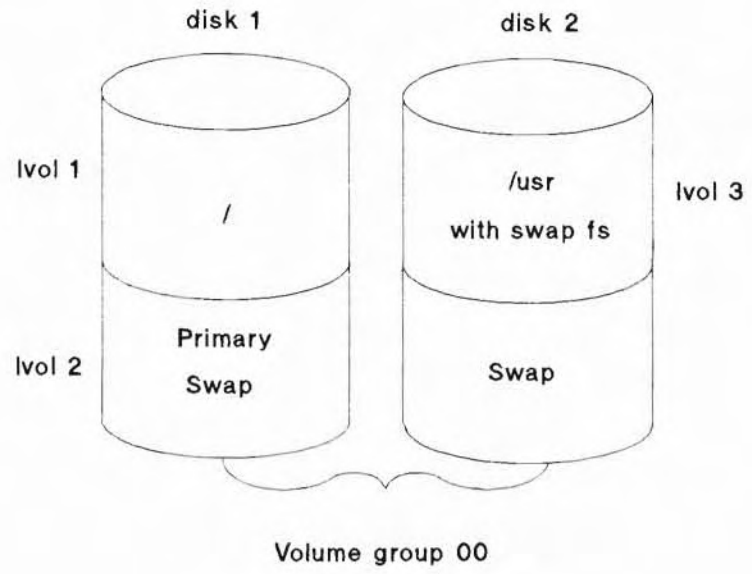
8. Name the two methods for allocating swap space?

9. Describe two guidelines for setting up device swap.

10. Describe how to set priorities for swap.

11. What is the name of the file used to automatically setup either device or file system swap?

12. What would the `/etc/fstab` file look like for these disks?


disk 1
disk 2
lvol 1
/
/usr
with swap fs
lvol 3
lvol 2
Primary
Swap
Swap
Volume group 00

Module 15 — Reconfiguring the Kernel

Objectives

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

- Describe reasons why it might be necessary to reconfigure the kernel.
- Perform the steps to reconfigure the kernel using HP-UX commands and SAM.
- Reconfigure the kernel for an additional device or subsystem.
- Describe some important tunable parameters.

Module 15 — Reconfiguring the Kernel

15-1. SLIDE: Why Reconfigure the Kernel?

Why Reconfigure the Kernel?

Reasons for reconfiguring:

- To add devices that use drivers not in kernel
- To modify kernel parameters
- To change location of:
  system console
  primary swap section
  dumps device

Methods for reconfiguring:

- SAM
- HP-UX commands

H6296 15-1 163

Student Notes

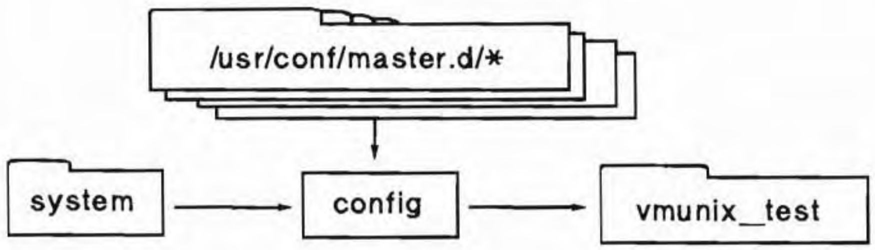
For most systems, the default kernel configuration supplied by Hewlett-Packard is sufficient and you will not have to modify the HP-UX kernel configuration. However, if you do need to change the default configuration, you will have to modify the kernel. Reasons for reconfiguring the kernel include:

- adding or deleting devices
- adding or deleting subsystems
- changing tunable parameters

You can modify the kernel either by using SAM, the System Administration Manager utility, or by using HP-UX commands to modify the `/stand/system` file, create the new kernel, and then reboot the system

15-2. SLIDE: How Is a Kernel Created and Activated?

How Is a Kernel Created and Activated?



H6296 15-2 164

Student Notes

The kernel is an executable program (called `/stand/system`) that deals directly with the hardware, manages the low-level functions for the operating system, and insulates the operating system from the details of the hardware.

The drivers configured into the kernel facilitate communication between the HP-UX operating system and I/O devices. The kernel is the device dependent portion of the operating system because it contains code (drivers) designed to control the operation of devices and I/O adapters. The rest of the operating system is device independent; it knows nothing about how devices operate. The device independent portion of the operating system can read and write devices by using device files, also called special files.

The reconfiguration process starts with the `/stand/system` file. The `system` file is the system description file. It specifies what drivers are used in the kernel, and optionally specifies root and values for system parameters. The files in `/usr/conf/master.d` directory and the `/stand/system` file are the input to config. /usr/conf/master.d contains files with all the information regarding supported devices. /usr/conf/master.d/* are supplied as part of the HP-UX operating system and should not be modified by anyone who does not fully understand its structure and purpose.

To reconfigure the kernel, you edit the /stand/system file, then run config on the new /stand/system.

```
#cd /stand
#config system
Compiling conf.c ...
Loading vmunix_test...
#
```

config generates a new kernel, /stand/vmunix_test. config also generates two other output files. One of these files, conf.c, is a C program that defines the configuration table for the various devices on the system. The other output file is config.mk. config.mk holds instructions on how to create the HP-UX kernel. make, using config.mk's instructions, compiles and links conf.c and outputs /stand/vmunix_test, the new HP-UX kernel. make is a standard HP-UX utility that can be simply described as a program construction tool. If you need to generate a new kernel without generating a new conf.c file, you just execute:

```
make -f config.mk
```

15-3. SLIDE: The /stand/system file

The /stand/system file

The /stand/system file contains:

- Device drivers and pseudo-drivers
  - Installed software drivers and I/O interface cards
  - SCSI disk and DAT device drivers
  - Network management
  - PTY Pseudo Drivers
  - I/O drivers
- Optionally specifiable parameters and locations
  - Kernel Devices
  - System Parameters
  - Swap information

H6296 15-3 165

Student Notes

The /stand/system is the configuration description file, used to create a new or customized kernel with either config or SAM.

Note SAM does not directly use the /stand/system file but instead uses a /stand/build/system.SAM file and creates a /stand/build/vmunix_test for the new kernel. Be sure to answer yes when SAM asks you to overwrite /stand/system.

Device Drivers and Pseudo-Drivers

Typically after installation only a minimum number of drivers are installed so you may need to add other drivers. Your system will work just fine if you have drivers configured into your kernel that are not actually on your system. In other words, you can have entries in your `/stand/system` for devices that are not really on your system. Remember though, that your kernel will be bigger and will take up more memory that cannot be reclaimed.

Only one entry is needed in your `/stand/system` for each category of devices. For example, if you have several `cs80` devices on your system, you only need one `cs80` entry in your `/stand/system` and if you have several SCSI devices on your system, you only need one `scsi1` (HP-PB SCSI Interface Module) entry in your `/stand/system` file.

The device drivers are simply listed.

Kernel Devices

Kernel devices are found via the `/usr/conf/master.d/*` file. You can optionally specify their locations in the `/stand/system` file.

swap — No more than one swap specification is allowed. If a swap specification is not given, the system will be configured to swap on the root device at the end of the file system.

```
swap hw_path
swap lvol Configure swap on a logical volume.
```

dumps — The location of the dumps device. One or more dump specification is allowed. If a dump specification is not given, then the primary swap area will be used.

```
dump hw_path
dump lvol Configure dump on a logical volume.
dump none Configure the kernel with no dump device.
```

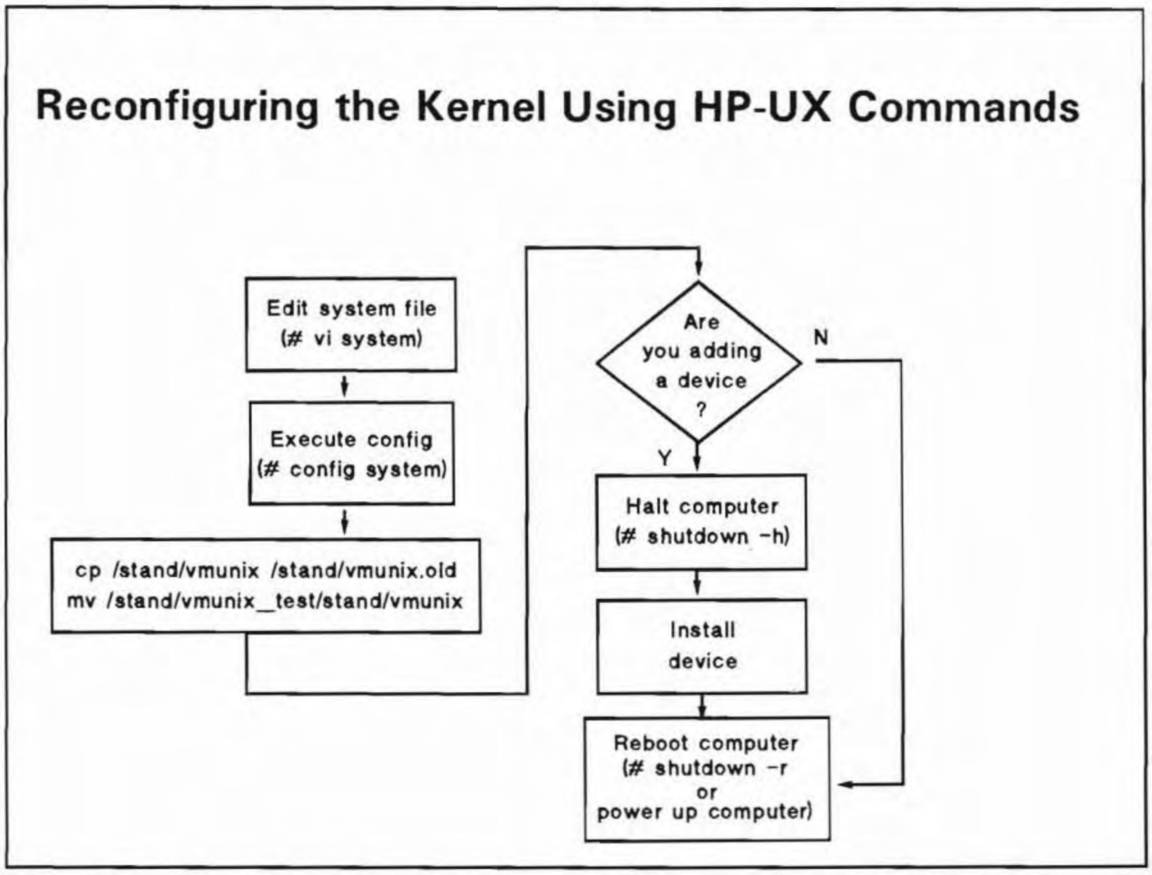
Operating System Parameters

These parameters are usually left as installed unless there is a compelling reason to change them such as an application program exhausting a certain system resource.

For system parameters, each line contains two fields:

`parameter_name` *number or formula*

15-4. SLIDE: Reconfiguring the Kernel Using HP-UX Commands



H6296 15-4 166

Student Notes

Before you begin the process of reconfiguring the kernel, bring your system to single-user mode and then change the working directory to /stand.

```
# cd /
# shutdown 0
   .
   . wait for shutdown to complete
   .
# cd /stand
```

The commands to perform the steps on the slide are:

1. Modify /stand/system to include your changes:

Module 15 — Reconfiguring the Kernel

The /stand/system is a description of an HP-UX system configuration. If possible, make modifications to your existing /stand/system. You might want to make a backup copy first in case you want the original again.

```
# cp /stand/system /stand/system.old
# vi /stand/system
```

2. To use the config command to process /stand/system:

```
# config /stand/system
```

3. Verify that config created vmunix_test in your local directory

```
# ls
```

4. Back up the existing kernel.

```
# cp /stand/vmunix /stand/vmunix.prev
```

5. If you are installing a new device, shutdown the system and install the device now.
6. Move the new kernel to /stand/vmunix and reboot the system:

```
# mv /stand/vmunix_test /stand/vmunix
# shutdown -r
```

15-5. SLIDE: Reconfiguring the Kernel to Add a Device

Reconfiguring the Kernel to Add a Device

```
  # vi /stand/system
:
  edit /stand/system as required to add autox0
:

# config /stand/system
# mv /stand/vmunix /stand/vmunix.prev
# mv /stand/vmunix_test /stand/vmunix
# cd /
# shutdown -h
```

if this applies to your system

H6296 15-5 167

Student Notes

The /stand/system file contains an entry for all peripheral devices, I/O cards, and major software subsystems such as LAN, RJE, X25, etc. To add a device, card, or subsystem to the kernel, you must add an entry for the device, card, or subsystem to your /stand/system. To determine what entry to add to /stand/system you should examine the /usr/conf/master.d/* files. The /usr/conf/master.d/* files are the device information tables that contains the device name, handler name, element characteristics, functions for the device, major device number, and minor device number. If what you are trying to add is not in the /usr/conf/master.d/* file then it is not supported by Hewlett-Packard.

The example on the slide assumes you want to add a Magneto-Optical Autochanger to your system. To configure your kernel for this card, you would check the /usr/conf/master.d/core-hpux to make sure it's supported and then add autox0 to your /stand/system. (Either the "name" or the "handle" will work in /stand/system.) Then follow the steps to configure your kernel as shown earlier in this module. They are listed again here for reference.

```
#grep autox0 core-hpux
core-hpux:autox0    30   230
# vi /stand/system
   .
   .  edit /stand/system as required to add autox0
   .
# config /stand/system
# mv /stand/vmunix /stand/vmunix.prev              if this applies to your system
# mv /stand/vmunix_test /stand/vmunix

# cd /
# shutdown -h
```

15-6. SLIDE: Configuring Operating System Parameters

Configuring Operating System Parameters

- Message related parameters
- Semaphore related parameters
- Shared memory related parameters
- Logical Volume Manager (LVM)
- Accounting Code Parameters
- File system related parameters
- Process parameters
- Miscellaneous parameters
- Networking parameters
- Swap related parameters

H6296 15-6 168

Student Notes

Configurable operating-system parameters are values set in the `/stand/system` file to customize kernel characteristics. Knowledgeably setting them can optimize performance but you should proceed cautiously.

Note Many parameters are interrelated and imprudent changes might cause unpredictable system behavior!

Kernel parameters can be directly modified using SAM. If you modify the kernel using HP-UX commands, the form of the parameters is:

parameter (*integer* or `formula`);

For example:

```
maxuprc                 50
maxusers                32
```

15-7. TEXT PAGE: Operating System Parameters

These are the parameters and their descriptions. For details on each of these parameters, refer to the appendix in the *System Administration Tasks* manual.

Operating System Parameters

Message related parameters

`mesg`	Enable/disable IPC messages (workstations only)
`msgmap`	Dimensions the resource map used to allocate. the buffer space for messages.
`msgmax`	Message maximum size.
`msgmnb`	Max number of bytes on the message queue.
`msgmni`	Number of message queue identifiers.
`msgseg`	Size, in bytes, of the units in which messages are allocated space.
`msgssz`	Message segment size.
`msgtql`	Number of message headers.

Semaphore related parameters

`sema`	Enable/disable semaphores (workstations only)
`semaem`	Maximum value by which a semaphore can be undone.
`semmap`	Dimensions the resource map which shows the free holes in the sem array.
`semmni`	Number of semaphore identifiers.
`semmns`	Total number of semaphores in system.
`semmnu`	Number of semaphore undo structures.
`semume`	Semaphore undo entries per process.
`semvmx`	Semaphore maximum value.

Shared memory related parameters

`shmem`	Enable/disable shared memory (workstations only)
`shmmax`	Maximum shared memory segment in bytes.
`shmmni`	Shared memory maximum number of identifiers.
`shmseg`	Maximum number of shared memory segments that can be attached to a process at any given time.

Logical Volume Manager (LVM)

`maxvgs`	Maximum volume groups on system
`no_lvm_disks`	No volume groups on system (workstations only)

Accounting Code Parameters

Configurable Parameters for Accounting

`acctresume`	Resume accounting due to disk usage.
`acctsuspend`	Suspend accounting due to disk usage.

Specifying a Threshold Value

minfree	File system parameter

File system related parameters

Configurable File System Buffer Parameters

bufpages	Pages of static buffer cache
dbc_min_pct	Minimum dynamic buffer cache
dbc_max_pct	Maximum dynamic buffer cache
nbuf	Number of buffer headers.

Configurable Open or Locked Files Parameters

maxfiles	Soft limit to the number of files a process can have open.
maxfiles_lim	Hard limit for the number of files a process can have open.
nfile	Maximum number of open files at any one time in the system.
nflocks	Possible number of file/record locks in the system.
ninode	Maximum number of open inodes which can be in-core.

Configurable Asynchronous Write Parameter

fs_async	enable/disable asynchronous disk writes

Process Parameters

Configurable Parameters for Process Management

maxdsiz	Maximum size of the data segment (in bytes) of an executing process.
maxssiz	Maximum size of the stack segment (in bytes) of an executing process.
maxtsiz	Maximum size of the shared text segment (in bytes) of an executing process.
maxuprc	Maximum number of simultaneous processes per user.
nproc	Maximum total number of processes that can exist simultaneously in the system.
timeslice	Time slice allocation between competing processes

Miscellaneous parameters

Miscellaneous configurable parameters

dst	Whether to convert to daylight savings time.
timezone	Number of minutes west of the Greenwich meridian.
maxusers	Defines macro MAXUSERS, which determines the size of system tables.
ncallout	Maximum number of timeouts that can be scheduled by the kernel at any one time.
ndilbuffers	Maximum number of DIL open device files at any one time.
npty	Number of pseudo-teletypes.
nstlbe	Maximum translation look-aside buffer entries
rtsched_numpri	Number of real-time scheduling priority levels
scroll_lines	Used on CIO computers only. Defines the scrolling area on each Internal Terminal Emulator (ITE) port configured into the system.
unlockable_mem	Maximum amount of memory that will always be available for virtual memory and/or system overhead.

Networking parameters

`netisr_priority`	Realtime process priority for networking.

Swap Related Parameters

`allocate_fs_swapmap`	Fixed or dynamic swap-data-structure allocation
`maxswapchunks`	Maximum amount of swap space allocated to the system.
`nswapdev`	Maximum number of devices that you can use for device swap.
`nswapfs`	Maximum number of file systems that you can use for dynamic swap.
`page_text_to_local`	Enable/disable text swap on client
`remote_nfs_swap`	Enable/disable swap to remote NFS
`swapmem_on`	Enable/disable pseudo-swap reservation
`swchunk`	The chunk size for swap. Must be a power of two. (This parameter cannot be configured with SAM.

Note

You can damage your system by changing these parameters improperly. Be sure you know the implications before you change them. Never set system parameters outside the given range. These parameters interact and should be changed in a balanced way.

15-8. SLIDE: Using SAM for Kernel Configuration

Using SAM for Kernel Configuration

SAM can configure:

- Drivers
- Subsystems
- Configurable Parameters
- Special Devices

H6296 15-8 169

Student Notes

This slide shows the kernel configuration tasks that SAM can perform for you. As you can see, SAM is quite versatile. In most cases when you need to reconfigure the kernel, you can use SAM instead of manually executing commands.

SAM will make the changes in the "template" file `/etc/conf/stand/system`. If you wish, you may specify a different "template" file for SAM to use as input when it creates the new kernel.

SAM will always show you a list of whatever you are configuring, with the "current" and "pending" values. The "pending" values are those that you have changed in the "template" file, and which will be the new kernel values when you have created a new kernel and rebooted the system. The "current" values are the values in the "template file" that you have selected.

SAM helps you to perform these kernel reconfiguration tasks:

Drivers

To get to the list of drivers, choose `Kernel Configuration ->` from SAM's Control Window. Then choose `Drivers` from the Kernel Configuration Functional Area. Remember that you must be superuser to execute the SAM program.

You will see a list of available drivers and whether they are "In" or "Out" of the current kernel. Actions you may perform are:

- Apply Template Values
- Clear Template Values
- Load a Template File
- Apply Lite RAM Configuration
- Add Driver to Kernel
- Remove driver from Kernel
- View Driver Details
- Create a New Kernel

Subsystems

To get to the list of subsystems, choose `Kernel Configuration ->` from SAM's Control Window. Then choose `Subsystems` from the Kernel Configuration Functional Area.

You will see a list of available subsystems and whether they are "In" or "Out" of the current kernel. Actions you may perform are:

- Apply Template Values
- Clear Template Values
- Load a Template File
- Apply Lite RAM Configuration
- Add Subsystem to Kernel
- Remove Subsystem from Kernel
- View Subsystem Details
- Create a New Kernel

Note

If you add a subsystem, you should use the `swinstall` utility. The `swinstall` utility adds all necessary software, configures your system for the software, adds necessary drivers, rebuilds the kernel, and reboots your system. Adding a subsystem with SAM merely modifies the kernel to run the subsystem. It does not install the necessary software or perform other essential tasks.

Configurable Parameters

To get to the list of parameters, choose `Kernel Configuration ->` from SAM's Control Window. Then choose `Configurable Parameters` from the Kernel Configuration Functional Area. You will see a list of available parameters, and their current values. Actions you may perform are:

Apply Tuned Parameter Set
Apply Template Values
Clear Template Values
Load a Template File
Apply Lite RAM Configuration
Modify Configurable Parameter
Create a New Kernel

Dump Devices

To get to the list of Dump devices, choose `Kernel Configuration ->` from SAM's Control Window.

Then choose `Dump Devices` from the Kernel Configuration Functional Area. You will see a list of available devices, the current hardware path, and the pending hardware path. Actions you may perform are:

Add Dump Device
Apply Template Values
Clear Template Values
Load a Template File
Apply Lite RAM Configuration
Modify Dump Devices
Insert New Dump Devices
Remove Dump Devices
Create a New Kernel

Creating the Kernel

When you have completed all the kernel changes you wish to make using SAM, you should choose the `Create a New Kernel` action.

When the new kernel has been created, you will be given the choice of:

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

SAM Limitations

Note that SAM has some limitations. It cannot be used to:

- Change the location of the swap device.
- Add or delete devices that use customer created drivers.

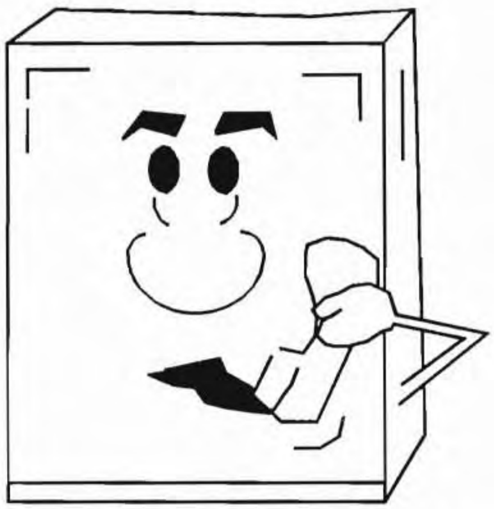
These tasks must be performed using HP-UX commands.

Note

SAM does not make a backup copy of the kernel, so you should be sure to copy `/stand/vmunix` to`/stand/vmunix.prev` before beginning the kernel reconfiguration procedure, whether you use SAM or HP-UX commands.

15-9. SLIDE: What If the New Kernel Won't Boot?

What If the New Kernel Won't Boot?



Call /stand/vmunix.prev

H6296 15-9 170 © 1995 Hewlett-Packard Co.

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. The only way to completely test whether your new kernel is good or not is to attempt to boot from it.

If the system will not boot from your new kernel, or if it boots but does not run acceptably (for whatever reason), you need a way to get back to where you were before you started. That way involves booting from the backup copy of the kernel, that you made before beginning the whole procedure.

Booting from the Backup Kernel

If your new kernel doesn't boot, you will have to boot from your backup kernel instead. You DID make a backup kernel, didn't you?

To boot the backup kernel, `/vmunix.prev`, redirect the boot sequence provided by the Boot Console User Interface by pressing the `ESCAPE` key until the message,

```
Terminating selection process.
```

is displayed.

The automatic boot sequence has now been halted and you are in fully "attended" or interactive mode.

Ask to boot from the primary boot path and interact with ISL. Type the boot command:

```
ISL> hpux boot disk(;0)/stand/vmunix.prev
```

15-10. LAB: Hands-On, Reconfiguring the Kernel

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. Move to the `/stand` directory. Move the current `/stand/system` to `/stand/system.`*your_name*. Then, create a new `/stand/system` file that shows the current system configuration.

2. Now, edit `/stand/system` and add the appropriate entry for a CD-ROM file system if one is not already present. Also you should make sure your system can access the CD-ROM disk interface.

3. Now, perform the commands to build a new kernel. Once again, assume the hardware has already been added to the system.

4. Invoke SAM. Choose the `Kernel Configuration` Functional Area from the Control Window. Look through SAM's options on kernel configuration, but don't reconfigure your kernel with SAM.

5. Add the appropriate entry for the parallel interface.

Module 16 — System Backup

Objectives

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

- Explain why backups are important, and what should be backed up.
- Describe several backup strategies for a system.
- Differentiate between different backup and restore methods.
- Backup and restore files with `fbackup/frecover`, `cpio`, `tar`, and `dd`.
- Find files with the `find` command.
- Backup the LVM configuration.

Module 16 — System Backup

16-1. SLIDE: Why Backup?

Why Backup?

How Much Data Can You Afford To Lose?

Data is sometimes lost by:

- File system corruption
- Accidental removal of files
- Hardware failures
- System crash

Regular backups:

- Minimize data loss
- Keep users happy
- Provide stability and order

H6296 16-1 171

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.

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.

16-2. SLIDE: What Do You Back Up?

What Do You Back Up?

- Backup the entire file system (full backup)
- Backup part of the file system
  - Files that have changed since the last backup (incremental or delta backups)
  - A subtree of the file system
  - The applications
  - Users' Files
- Backup the database configuration
- Backup the LVM configuration

H6296 16-2 172

Student Notes

Before we can discuss *how* to back up the system, we must first discuss *what* to back up and *when* to make backups. You can selectively back up parts of the system, or you can back up the entire system. Most of the time, you should implement a backup strategy based on a combination of these two strategies.

There are three things to consider when choosing a backup strategy that is right for you:

- amount of media needed
- amount of time needed
- how often backups are needed

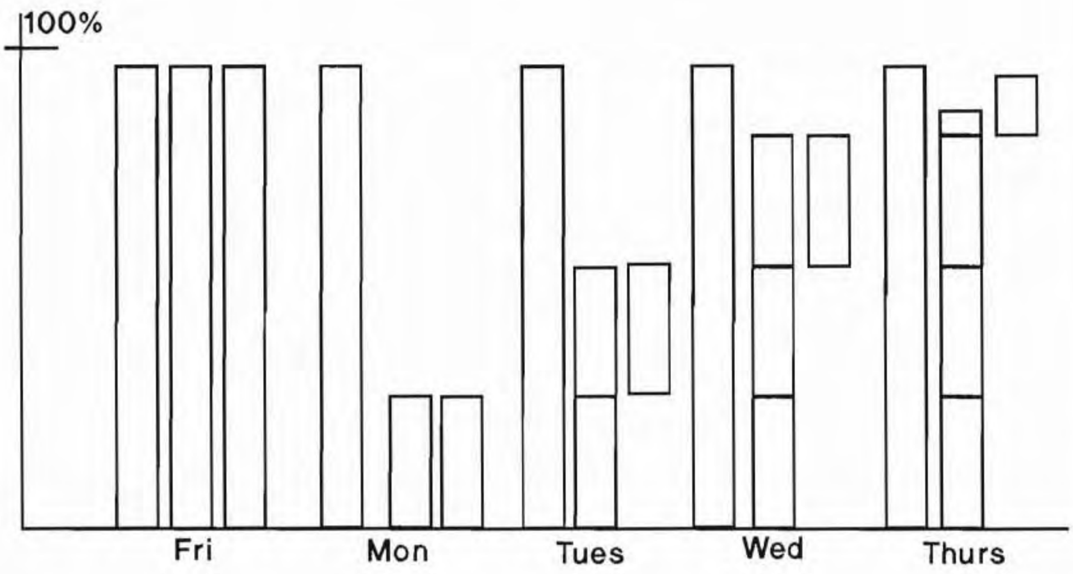
The amount of media and the amount of time require that you be able to approximate the amount of space being used by files on the system. You can use the `bdf` command to get the number of disk blocks in use. The `bdf` command reports in 1024-byte blocks.

Databases might provide special commands to backup the configuration. You should use these commands before running the regular backup.

The Logical Volume Manager (LVM) configuration is backed up using the command `vgcfgbackup`. It is not backed up during a regular backup using file backup commands.

16-3. SLIDE: How Often Do You Backup?

How Often Do You Backup?



H6296 16-3 173

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 things when determining how frequently to back up a particular file (or type of file):

- How often do the contents of the file change?
- How critical is it that the file's contents be up to date?

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.

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. 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 have changed since the first of the month.
- Every day except Friday, back up all of the files in the selected set that have changed since the last Friday or first of the month, which ever is most recent.

There are three levels associated with the above schedule (the once per month level, the once per week level, and the once per day level). The once per month level is a full backup. The other two are incremental backups. The problem is how to distinguish between the two types of incremental backups. This is accomplished with backup levels.

You can have up to ten backup levels (0 - 9). Your backup strategy varies based on the level of activity on your system and the capacity of your media.

Recovery Example Using Three Backup Levels.

To implement the earlier example of monthly, weekly, and daily backups, you would use the following backup levels:

level 0	full monthly backup
level 1	weekly backup on Friday
level 2	daily backup, except Friday

This table illustrates the level numbers for implementing this example.

Table 16-1.

Date of the month:	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**...**	**1**
Day of the month:	**Su**	**M**	**T**	**W**	**Th**	**Fr**	**Sa**	**Su**	**M**	**T**	**W**	**Th**	**F**	**Sa**	**Su**	**...**	
backup level	0	2	2	2	2	1	2	2	2	2	2	2	1	2	2	...	0

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:

1. Restore the monthly full backup from Sunday the 1st.
2. Restore the weekly incremental backup from Friday the 6th.
3. Restore the incremental backup from Wednesday the 11th.

Recovery Example Using Two Backup Levels.

The following example illustrates a weekly full backup and daily incremental backup, two backup levels. When implementing your backup strategy using SAM, only two levels of backups are supported. The figure illustrates the level numbers supported by SAM:

Table 16-2.

Date of the month:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...	1
Day of the month:	**Su**	**M**	**T**	**W**	**Th**	**Fr**	**Sa**	**Su**	**M**	**T**	**W**	**Th**	**F**	**Sa**	**Su**	...	
backup level	0	1	1	1	1	1	1	0	1	1	1	1	1	1	0	...	0

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:

1. Restore the full backup from Sunday the 8th.
2. Restore the incremental backup from Wednesday the 11th.

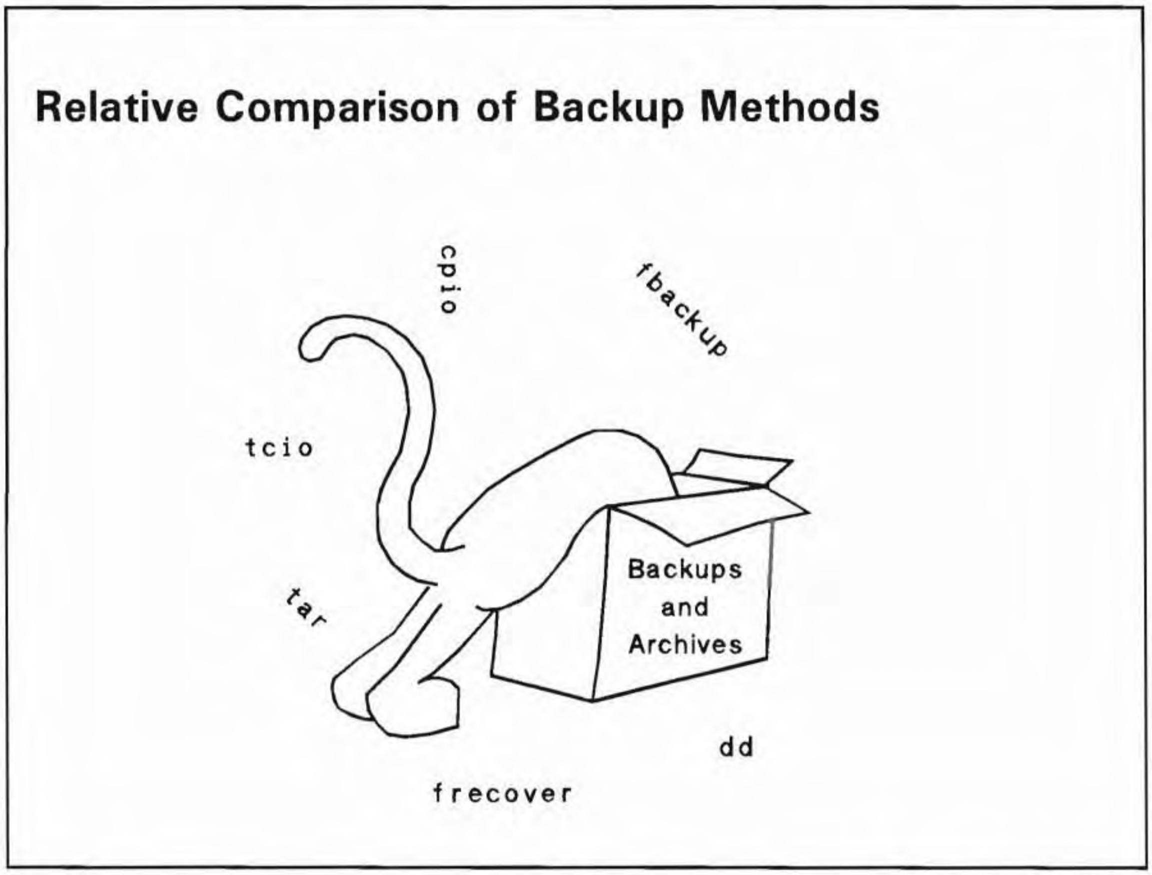
16-4. TEXT PAGE: Worksheet

System Backup Worksheet

Table 16-3.

Type of data	Location of Data	Importance of Backup	Size	Media	Frequency (daily, weekly, hourly)	Level (full, incremental)	Scheduled Time	Command
Operating System	vg00	low	150mb					
lvm config	all disks	med	bytes					
config files	vg00	med	bytes					
Applications								
appl	disk2	low	300mb					
Databases								
db1	vg02	v,high	300mb					
db2	disk3	high	1.5gb					
User File Sytems								
/home	vg03	high	100mb					

16-5. SLIDE: Relative Comparison of Backup Methods



H6296 16-5 174 © 1995 Hewlett-Packard Co

Student Notes

There are many commands and utilities you can use to save files to backup media:

`cpio` cpio is very commonly used in the UNIX world. It is a general purpose UNIX logical file backup utility, and was the backup method of choice before `fbackup` became available, because it also allows for backups to span multiple backup media. `cpio` is option oriented, it has the capability for selective file restores, and it is much faster than `tar` for large amounts of data. `cpio` also can backup special files such as device or network files.

`cpio` is POSIX compliant when you use certain options to create or read archives. The default behavior is POSIX. An option allows backwards compatibility.

`tar` `tar` is the oldest of all the utilities and thus the most portable. It is also a general purpose logical file backup utility, is found on all UNIX systems, and can be read by several non-UNIX systems as well. `tar` is much easier to use than `cpio` and allows appending to backup media. Prior to the HP-UX 7.0 release, `tar` did not permit the backup to span multiple media. This limitation may still be found on other vendors' system. Thus, `tar` has historically been used for backing up small numbers of files easily. `tar` also allows selective restore of files.

`tar` is POSIX compliant when you create or read tar archives. The default behavior is to create and read POSIX format archives of files. You must use a key, `O`, to get the old behavior for backwards compatibility. You will notice other differences in `tar` behavior with the `v` key.

`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.

`fbackup`

The `fbackup` command was written specifically to be used in HP system backups. `fbackup` allows the use of 9 different backup levels, thus making the backup process extremely flexible. `fbackup` and its companion `frecover` allow an "across the network" back up. The commands are considered "Network smart", in that they allow you to specify a device on a remote host as part of the backup/recover command line.

Another significant advantage of `fbackup` is that it is supported on all HP 9000 HP-UX systems as a feature of SAM. This means that workstation and server systems can use the same tools for easy backup, recovery, and exchange of files.

`fbackup` is an HP proprietary backup method usable on HP-UX systems only.

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.

Table 16-4. Relative Comparison of Backup Methods

	`fbackup`	`cpio`	`tar`	`dd`
Flexibility	High	Medium	Low	Low
Portability	HP-UX only	Unix	Unix/some Non-Unix	N/A
Directory Relative Recovery	Yes	If created with relative paths	If created with relative paths	No
Application	Full and incremental system backups	Copying directory structures and transferring files to other systems	Easy to use and highly portable	Duplication of bit images on to different media (data conversion)

16-6. SLIDE: The Command fbackup

The Command fbackup

Syntax:

fbackup -f *device* [-0-9][-u][-i path][-e path][-g *graph*]

Examples:

```
# fbackup -u8f /dev/rmt/0m -g /var/adm/fbackupfiles/graphfile
# fbackup -f /dev/rmt/0m -i . -e ./subdir
```

Remote backup:

```
# fbackup -f quimby:/dev/rmt/0m -i . -e ./subdir
```

H6296 16-6 175

Student Notes

The `fbackup` command is the primary tool for creating both full and incremental backups. The `fbackup` command is quite flexible and allows each system administrator to develop a backup strategy that best suits the needs of the installation.

The general form of the command is shown on the slide. The first example on the slide backs up files and directories indicated in the `graphfile` to tape. The second example backs up all files under the current directory except those contained under `subdir`.

fbackup options

-f *device*	the device to which output will be sent
[-0-9]	backup level—default is 0

[-u]	update /var/adm/fbackupfiles/dates file (only if used with -g)
[-i *path*]	include path (file or directory) in the backup
[-e *path*]	exclude path (file or directory) from the backup
[-g *graph*]	file that contains a list of files and directories to be included or excluded from the backup
[-I *path*]	write an index to file *path*

The fbackup command does *not*, by default, write to the standard output and the -f *device* part of the command is not optional. The *device* can be a regular file or a device (special) file. You can specify - as the *device* to have fbackup write to the standard output. For most systems, a magnetic tape drive is used as the backup device. In this case, use the appropriate device special file for the raw (character) device in the /dev/rmt directory. If you specify a device special file on another system: -f quimby:/dev/rmt/0m, you must be allowed to run a remsh command on the remote system. To do that you must put the name of the local system, and/or the name of the user who runs the command, in the $HOME/.rhosts or in the /etc/host.equiv file.

Backup Levels

The [0-9] option allows the user to define and use what are referred to as 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*"? Certainly, any file on the system can be considered to have changed if the file's time stamp is compared to the beginning of time. In other words, an incremental backup that copies files that have been modified more recently than the beginning of time is actually a full backup!

Level 0	is pre-defined 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 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. The reason for selecting level 5 for the incremental backups is to allow for some flexibility in the backup scheme should changes be required.

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.

Inclusion and exclusion is accomplished either through the use of graph files (`-g` option), or through the inclusion of parameters on the command line (`-i` and `-e` options). A graph file is a file that contains ASCII text. 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 being with an "`e`" indicate files that will be excluded.

```
i /usr
e /usr/lib
i /home
e /home/guest
```

The combination of backup levels and graph files provide 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.

Backup Levels and Graph Files

As described above, `fbackup` is capable of determining when a backup of a specific level was last made. This implies that there must be a data file used by `fbackup` to retain the necessary information. Such a data file does exist and is readable ASCII text. The file is `/var/adm/fbackupfiles/dates`.

The file `/var/adm/fbackupfiles/date` contains information about when the last backup at each backup level was performed. The dates file contains:

- the graph file used for the backup
- the level of the backup
- the date of the backup
- the start and end time for the backup

This information is used by `fbackup` to determine which files defined in the graph file are included in the backup. The `fbackup` command uses the following search sequence on the dates file to determine the base backup on which to build an incremental backup:

1. matching graph file
2. next lowest level number
3. most recent date

If no lower level is found, a full backup at the specified level is performed. If there are duplicates of a lower level found, the most recent is used as the base for the incremental backup.

Other Common Options

Since it may be desirable to create a backup without updating the `dates` file, `fbackup` requires the `-u` option to be specified on the command line if the user wants to update the `dates` file.

The `[ -I` *path* `]` options causes `fbackup(1M)` to write an index to file *path*.

There are other options that may be of interest to you. The manual page for `fbackup(1M)` provides explanations for all the options.

Examples

```
# fbackup -f /dev/rmt/0m  -8 -u -g /var/adm/fbackupfiles/graphfile
```

Perform a level 8 backup to `/dev/rmt/0m` using the graph file `/var/adm/fbackupfile/graphfile`. Update the `/var/adm/fbackupfiles/dates` file.

```
# fbackup -f /dev/rmt/0m -i . -e ./subdir
```

Backup everything under the current directory except the `subdir` to `/dev/rmt/0m`.

```
fbackup -f quimby:/dev/rmt/0m -i . -e ./subdir
```

Backup everything in the local path except the subdirectory `subdir`, to `/dev/rmt/0m` on a remote system named `quimby`.

Special Note for Cartridge Tapes

If you use a cartridge tape, you must use the `tcio` command to "buffer up" the data transfer between `fbackup` and the cartridge tape drive. `tcio`, like `cpio`, has three major options:

`-o`	go out to a device
`-i`	come in from a device
`-u`	utility to send commands to the cartridge tape drive

Here is an example of how you would use the `tcio` command:

- Backup `/home` directory to cartridge tape

```
# fbackup-f - -i /home | tcio -o /dev/rct/0s0
```

16-7. SLIDE: The Command frecover

The Command frecover

Syntax:

```
frecover -r | -x [-g graph] [-f device]
frecover -I path | -R path [-f device]
```

Examples:

```
# frecover -rf /dev/rmt/0m
# frecover -x  -g /var/adm/fbackupfiles/graphfile -f /dev/rmt/0m
# frecover -I /tmp/index -f /dev/rmt/0m
```

H6296 16-7 176

Student Notes

The real reason for employing a backup strategy is to enable the recovery of lost files. As mentioned earlier, lost files can be the result of inadvertent removal by a user, or a file system disaster. 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 four basic modes of operation for `frecover`:

`frecover -r`	Recover everything that is on a backup volume.
`frecover -x`	Extract certain files from a backup volume.
`frecover -I` *path*	Read the index from the backup volume and write it to *path*. This retrieves a table of contents.

`frecover -R` *path* — Restart an interrupted recovery. The option to restart an interrupted recovery will not be discussed here.

Unlike `fbackup`, `frecover` does have a default for input. The default is `/dev/rmt/0m`. If a different input source (device) is to be used, a `-f` *device* option may be specified on the command line. As with the `fbackup` command, `-f -` can be used to specify that the standard input should be used.

`frecover -r` and `frecover -x` have some options in common. Some of the most commonly used are `hoFX`:

```
frecover -x | -r [ -hoFX ]
```

`-h` — Used to recover (or extract) only directories and not the files contained in them.

`-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.

`-F` — Causes `frecover` to strip all the leading directories from the path names of files being recovered. If `/usr/bin/vi` and `/bin/sh` were on the backup and were recovered using the `-F` option and the current working directory were `/home/root`, the resulting files would be `/home/root/vi` and `/home/root/sh`.

`-X` — Makes all recovered files relative to the current working directory. Suppose the current working directory was `/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 an `frecover` session.

An option that is unique to the `frecover -x` mode is `-g` *graph*. This allows you to use a graph file in the same way as with `fbackup`. The file format for the *graph* file is the same. Lines that begin with an `i` indicate a path that is to be included in the recovery. Lines that begin with an `e` indicate a path that is to be excluded from the recovery. This is useful for partial recoveries.

The first example on the slide recovers all files from medium density reel or DDS tape. The second example recovers all files indicated in graphfile. The third example retrieves an index of files from the tape and puts it in `/tmp/index`.

16-8. SLIDE: Review of the `find` Command

Review of the `find` Command

Syntax:

`find` *path-list expressions*

path-list	the list of directories to search recursively
expressions	search criteria and actions

Examples:

```
find /

find /home -name .profile

find . -size +2000

find / -atime +30
```

H6296 16-8 177

Student Notes

Before discussing any of the other backup commands in detail, we will review a command that is frequently used in conjunction with backup commands; `find`.

The `find` command can be used to find a group of files that match a certain criteria. The `find` command is one of a very few commands that will perform an automated search through the file system. This ability makes it extremely versatile. Unfortunately, since the basic operation of `find` is to conduct a search, it is very slow and consumes considerable system resources.

The format of the command is:

`find` *path-list expression*

The *path-list* is a list of path names, typically one directory. Often "." is specified which results in a search starting in the current directory. A *path-list* of "/" results in a search of the entire file system. The path names are searched recursively for files that satisfy the criteria specified in *expression*. When `find` locates a match, it performs the tasks also specified in *expression*. One of the most common tasks is to print the path name to the match.

The *expression* is made up of keywords and arguments that can specify search criteria and tasks to perform upon finding a match. One of the things that can make `find` seem complicated is that the keywords used in *expression* are all preceded by a "-", so it can look like the arguments precede the options.

Examples

- Find all files in the file system and print their names on the screen.

```
$ find  /
```

- Find all the files named `.profile` searching recursively down beginning at the directory `/home`.

```
$ find  /home  -name  .profile
```

- Find all the files beginning at the current directory that are larger than one megabyte (2000 512-byte blocks).

```
$ find  .  -size  +2000
```

- Find all the files in the entire file system that have not been accessed for 30 days.

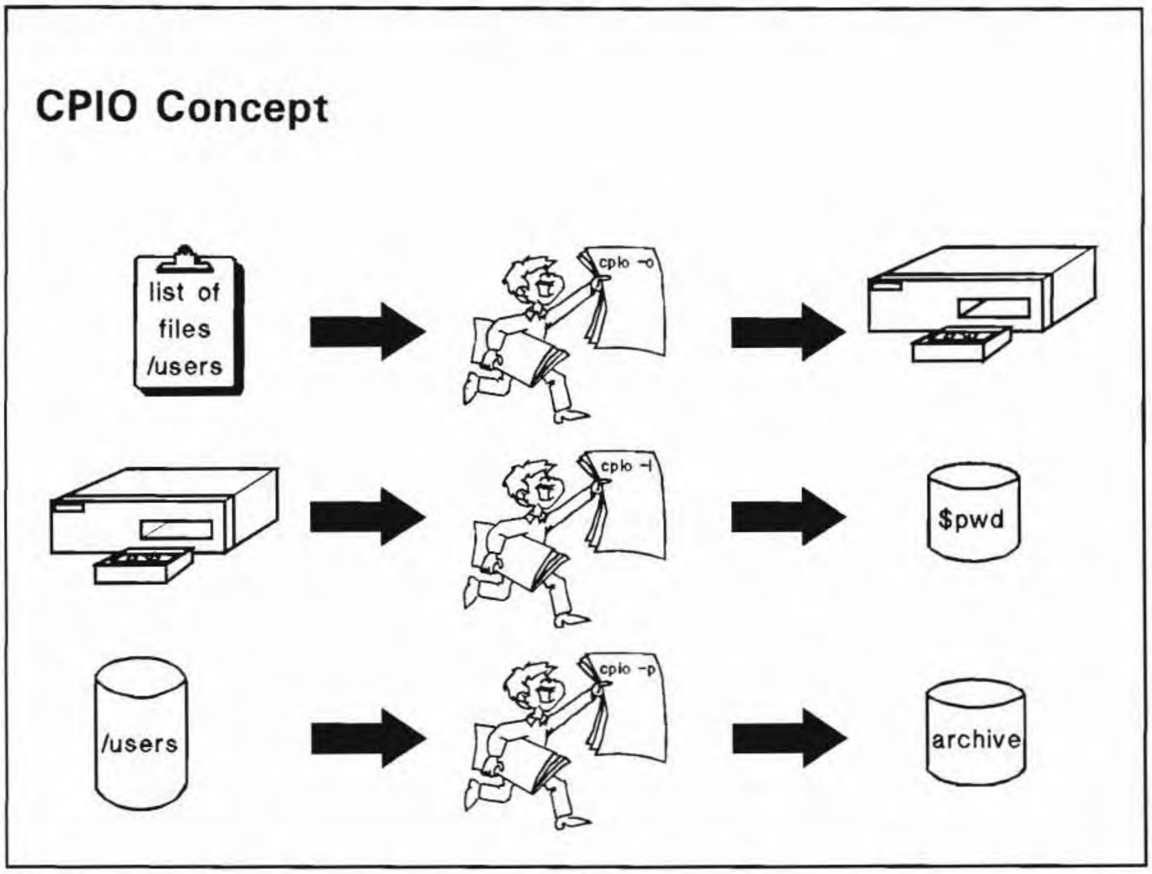
```
$ find  /  -atime  +30
```

- Find all files named `core` (core dumps) in the entire file system and remove them. Notice the space following the curly braces, it is required.

```
$ find  /  -name  core  -exec  rm  {}  \;
```

The list of expressions and options for the `find` command can be found on `find(1)` man page in the *HP-UX Reference* manual.

16-9. SLIDE: CPIO Concept



H6296 16-9 178

Student Notes

cpio stands for "copy input to output". It uses standard input as its source of file names, and standard output as the archive output. Since the defaults are standard input for a file list, and standard output for the archive, you have to specify the tape as a device, and you must provide a list of files to store. This is generally accomplished by piping the output of `find` into `cpio`.

The `cpio` command operates in file system or stream format. That is, it talks to block devices like hard disks, as well as raw character devices such as tape drives. The mode of operation is specified with a flag; `-i` or `-o` for stream format, and `-p` for file system format.

The `cpio` utility with piped input from the `find` command can be used to create a backup on tape, as well as to make copies of files to other locations in the file system.

16-10. SLIDE: The Command cpio

The Command cpio

Syntax:

```
cpio -o [ cvxB ]
cpio -i [ cvxdumB ] [ patterns ]
cpio -p [ vdumxl ] directory
```

- Reads a list of file names from standard input
- Copies contents of each file to standard output

Examples:

```
# find /  | cpio -ocx > /dev/rmt/0m       (backup)
# cpio -icxvmd "*/home/*" < /dev/rmt/0m  (restore)
```

H6296 16-10 179

Student Notes

The cpio command (with the -o option) reads a list of file names from standard input and copies the contents of the files specified to standard output. Thus, we must use cpio within a pipeline with some other HP-UX command generating the list of file names.

The cpio command is used in three major ways:

-o	Read standard input and copy each file to standard output (make a backup)
-i	Read standard input for the backup data and recreate it on the disk (restore backup)
-p	Read standard input for file names and recreate those files in another directory (pass option)

There are also several additional options that can be used with each major option:

Table 16-5.

-o	-i	-p	Function
-c	-c	—	Write header in ASCII format (if used with -o, it must be used with -i).
-x	-x	-x	Handle special (device) files.
—	-u	-u	Unconditionally restore (normally, an older file will not replace a newer file).
—	-m	-m	Retain current modification date (important for version control).
—	-d	-d	Recreate directory structure as needed.
-v	-v	-v	Print a list of files copied to the terminal.
—	-t	—	Print only a table of contents of the input.
-B	-B	—	Block input/output at 5120 bytes to the record.
-h	—	—	Follow symbolic links as if they were normal files.

Magnetic tape examples of the `cpio` command are as follows:

- Create a relative backup on `/dev/rmt/0m`. Files can be restored to any disk directory.

```
# cd /home
# find .  | cpio -ocx> /dev/rmt/0m
```

- Create an absolute backup on `/dev/rmt/0m`. Files can be restored only to the `/home` directory from where they were copied.

```
# find /home  | cpio -ocx > /dev/rmt/0m
```

- Restore the backup to the working directory if the tape was created in relative format.

```
# cpio -iudmcx < /dev/rmt/0m
```

- Copy the subtree rooted at current working directory to subtree rooted at `/tmp`.

```
# find .  | cpio -pdm /tmp
```

- If you use a cartridge tape, you must use the `tcio` command to "buffer up" the data transfer between `cpio` and the cartridge tape drive. `tcio`, like `cpio`, has three major options:

-o	go out to a device
-i	come in from a device
-u	utility option

  The `-o` and `-i` options of `tcio` correspond to those used with `cpio`. The `-u` option is for sending commands to the cartridge tape drive.

  Examples of the `tcio` command are as follows:

  — Make relative backup on cartridge tape `/dev/rct/update.src`:

```
# find . -print | cpio -ocx | tcio -o /dev/rct/update.src
```

— Restore relative backup to current working directory:

```
# tcio -i /dev/rct/update.src | cpio -iudmcx
```

16-11. SLIDE: The Command tar

The Command tar

Syntax:

tar *key* [*args ...*] [*file* | -C *directory*] ...

file	If file is a directory name, it refers to the files and (recursively) subdirectories contained in that directory.

Examples:

```
# tar cvf /dev/rmt/0m file1 file2

# tar xvf /dev/rmt/0m

# tar cvf /dev/update.src  -C /home . -C /etc .

# tar rvf /dev/rmt/0m file2
```

H6296 16-11 180

Student Notes

tar saves and restores archives of files on a magnetic backup media (DDS format DAT, mag tape, or cartridge tape), optical disk (MO), flexible disk, or regular file. Its actions are controlled by the *key* argument. The *key* string can be preceded by a hyphen (-) (as when specifying options in other HP-UX commands), but it is not necessary.

Where *key* could be the following:

f *dev-file*	Causes tar to use next argument as name of device where the archive will occur
c	Creates a new archive
r	Appends to end of archive
x	Extracts from archive
t	List table of contents of archive
v	Verbose, type the name of each file affected

If the f key is used, it causes tar to use the next argument as the name of the archive instead of /dev/rmt/0m (which is the default). If the name of the file is -, tar writes to the standard output or reads from the standard input, whichever is appropriate. Thus, tar can be used as the head or the tail of a pipeline.

The -C option can be included in the *file* list. It causes tar to change to the given directory and archive the files there. This allows multiple directories not related by a close or common parent to be archived using path names.

When the end of tape is reached, tar prompts the user for a new special file and continues. If a nine-track tape drive is used as the output device, it must be configured in Berkeley compatibility mode.

For a complete list of tar keys and options, see tar(1).

The tar command has new default behavior. tar is POSIX compliant when you create or read tar archives. The default behavior is to create and read POSIX format archives of files. You must use a key, O to get the old behavior for backwards compatibility. You will notice other differences in tar behavior with the v verbose key.

If you have a cartridge tape, you have to use the tcio command to "buffer up" the data transfer between tar and the cartridge tape drive. tcio has three major options:

-o	go out to a device
-i	come in from a device
-u	utility option

16-12. SLIDE: The Command dd

The Command dd

Syntax:

dd [*option=value*] ...

Examples:

Disk to disk copy

```
# dd if=/dev/rdsk/c1t6d0 of=/dev/rdsk/c1t5d0 bs=512k
```

Tape to disk copy of foreign tape with EBCDIC to ASCII translation

```
# dd if=/dev/rmt/0h of=/home/dave/ascii.data conv=ascii bs=80
```

H6296 16-12 181

Student Notes

The dd command is useful in some limited situations, but is technically *not* a backup command.

Options:

if=*file*	Input file name
of=*file*	Output file name
bs=*n*	Set both input and output block size to the same size. Supersedes ibs= and obs= options.
count=*n*	Copy only n input blocks
conv=*value*	Data conversion option. *value* could equal ascii, ebcdic, ibm, lcase, ucase, etc.
skip=*n*	Skip *n* input blocks before starting copy
seek=*n*	Seek *n* output blocks before starting copy

One useful function of the dd command is to copy from a disk to a disk. In the first example, dd is being used to copy an entire disk from target 6 to the disk at target 5. The block size that is specified (512 kilobytes) is an optimum number for doing disk-to-disk copies due to the size of the buffers allocated to the disk drivers. Note that while this is very fast, it requires a destination disk the same size or larger for successful operation.

The dd command can also be used to read a foreign magnetic tape format. In the example above, dd is reading a file off the tape using a block or record size of 80 bytes and translating the contents from EBCDIC to ASCII format. The final ASCII version is stored in the file /home/dave/ascii.data on the disk.

Another example follows:

```
# dd if=/dev/rdsk/c1t6d0 of=/dev/rmt/0h count=4480 bs=32k
# dd if=/dev/rdsk/c1t6d0 of=/dev/rmt/0h count=4480 bs=32k skip=4480
# dd if=/dev/rdsk/c1t6d0 of=/dev/rmt/0h count=4480 bs=32k skip=8960
```

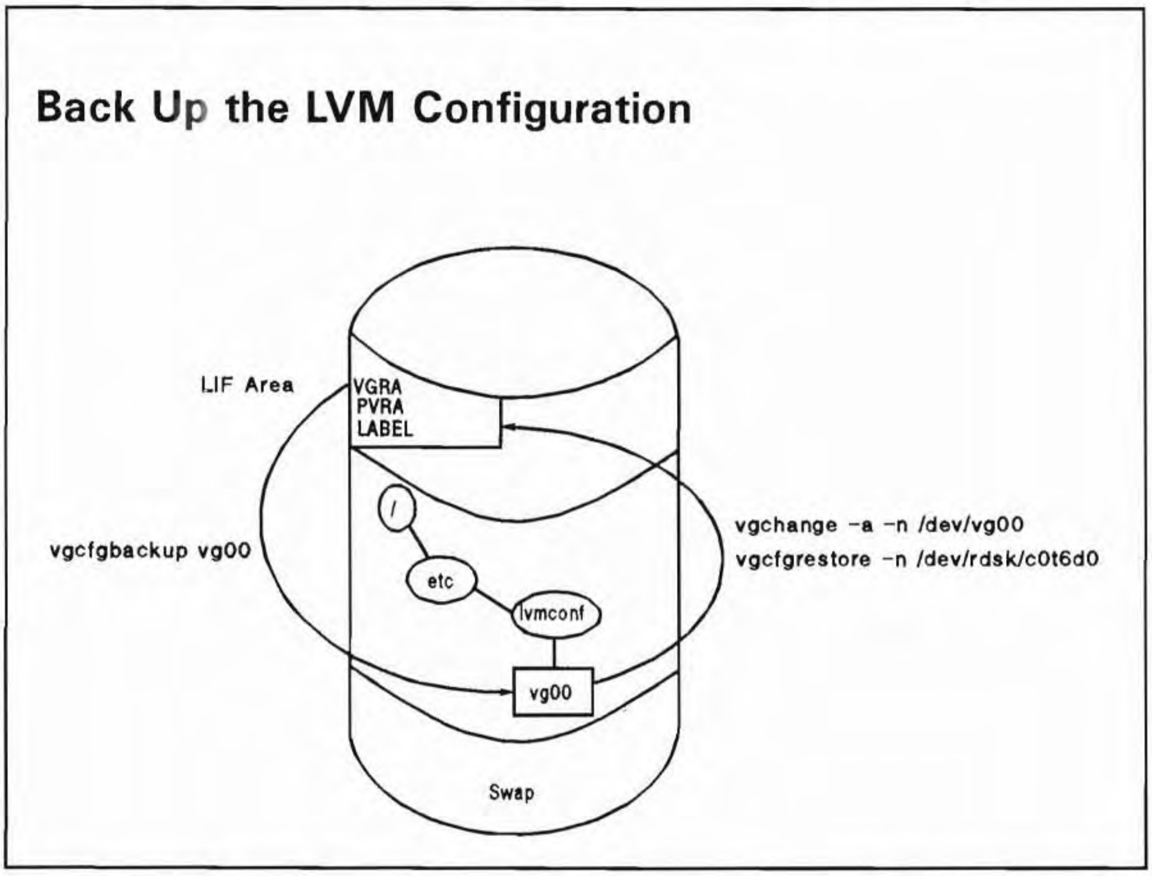
This example shows backing up a large disk section to a mag tape. The disk input is too large to fit on a single reel, so a separate dd command must be used for each of the three reels used for the backup. To restore this dd backup, execute the following:

```
# dd if=/dev/rmt/0h of=/dev/rdsk/c1t6d0 count=4480 bs=32k
# dd if=/dev/rmt/0h of=/dev/rdsk/c1t6d0 count=4480 bs=32k seek=4480
# dd if=/dev/rmt/0h of=/dev/rdsk/c1t6d0 count=4480 bs=32k seek=8960
```

The dd command is *not* recommended for file system backups because:

- It cannot do selective restores (restore nothing or all are the only choices)
- A direct image copy is produced. On HP-IB disks, good blocks as well as bad blocks are saved.

16-13. SLIDE: Backup the LVM Configuration



H6296 16-13 182

Student Notes

It is important to safeguard your LVM configuration because, unlike fixed disk sections that begin and end at known locations on a given disk, each volume group configuration is unique, frequently changes, and uses the space on several disks.

Although the commands that change the LVM configuration (such as `vgextend`), will automatically back up the LVM configuration, you might also want to periodically backup your LVM configuration using the `vgcfgbackup(1M)` utility.

The `vgcfgbackup` utility backs up volume-group configuration information into binary files, one file per volume group. It backs up the data structures that maintain the current LVM configuration.

It does *not* backup the LIF header and files or the bad block directory.

Running `vgcfgbackup`

You should have all disks in the volume group on-line. The command saves the configuration of the volume group to a file `/etc/lvmconf/`*vgname*`.conf`.

```
vgcfgbackup /dev/vgnn
```

Running `vgcfgrestore`

You can use the `vgcfgrestore(1M)` utility to restore the LVM configuration if a disk has corrupt LVM configuration data. First make the volume group unavailable, then run `vgcfgrestore`.

```
vgchange -a n /dev/vgnn                       make the volume group unavailable
vgcfgrestore -n /dev/vgnn /dev/dsk/c0t6d0
```

Note

These utilities back up and restore *only* the LVM configuration. They do not back up user data. You must still backup and restore user data using HP-UX utilities such as `fbackup` and `frecover`.

Recovering `lvmtab` - Using `vgscan`

The `/etc/lvmtab` file is the heart of the LVM configuration. This file is read by all LVM commands. It is not readable or editable on-screen. `/etc/lvmtab` is generated the first time you create an LVM entity using SAM or LVM commands. It is updated every time you change the LVM configuration; it lists all volume groups and the physical volumes that are matched to each volume group. If this file is destroyed or corrupted, the proper activation of volume groups and disks cannot occur.

You use `vgscan(1M)` to re-create the `/etc/lvmtab` file. It searches every LVM disk on the system for logical volumes, then groups them into volume groups by searching the `/dev` directory and matching major and minor numbers.

16-14. LAB: Hands-On, Backing Up the System

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

If you do not have your own tape drive, you can simulate a tape drive by backing up to a file. Name the file `/dev/rmt/`*yourname*.

1. Create a graph file in `/var/adm/fbackupfiles` specifying the inclusion of the following files in your backup:

   include all files and dirs underneath `/var/spool/lp/model` EXCEPT `thinkjet` and `remote`

   exclude all files and dirs underneath `/etc`, except `/etc/passwd`

   exclude the tape device file in `/dev/rmt`

2. Perform a level 0 backup of the designated files to your tape drive. (Remember to use the -u option of `fbackup` to record the timestamp of the backup.)

3. Make a modification to `/etc/passwd`.

4. Recover a copy of the `/etc/passwd` file into the `/tmp` directory. Make sure that you DON'T overwrite the newer copy of `/etc/passwd` which you modified in the step above.

5. Perform a level 1 backup of the files included in your graph file to your simulated tape drive. How many files were backed-up this time?

6. Use the `find` and `cpio` commands to backup the directory `/usr/share/lib/term` to your simulated magtape. Now look at the "tape" table of contents. Restore these files from magtape to the directory `/cpio_files`. Verify that the restoration was successful. Then remove these files. Remember to use the relative form of the `find` command when performing the copy.

7. Use the `tar` command to backup the directory `/usr/share/lib/term` to your simulated magtape. Now look at the "tape" table of contents. Restore these files from magtape to the directory `/tar_files`. Remember once again to use the relative form of the `tar` command and to remove the restored files when you are done.

Module 17 — Job Scheduling

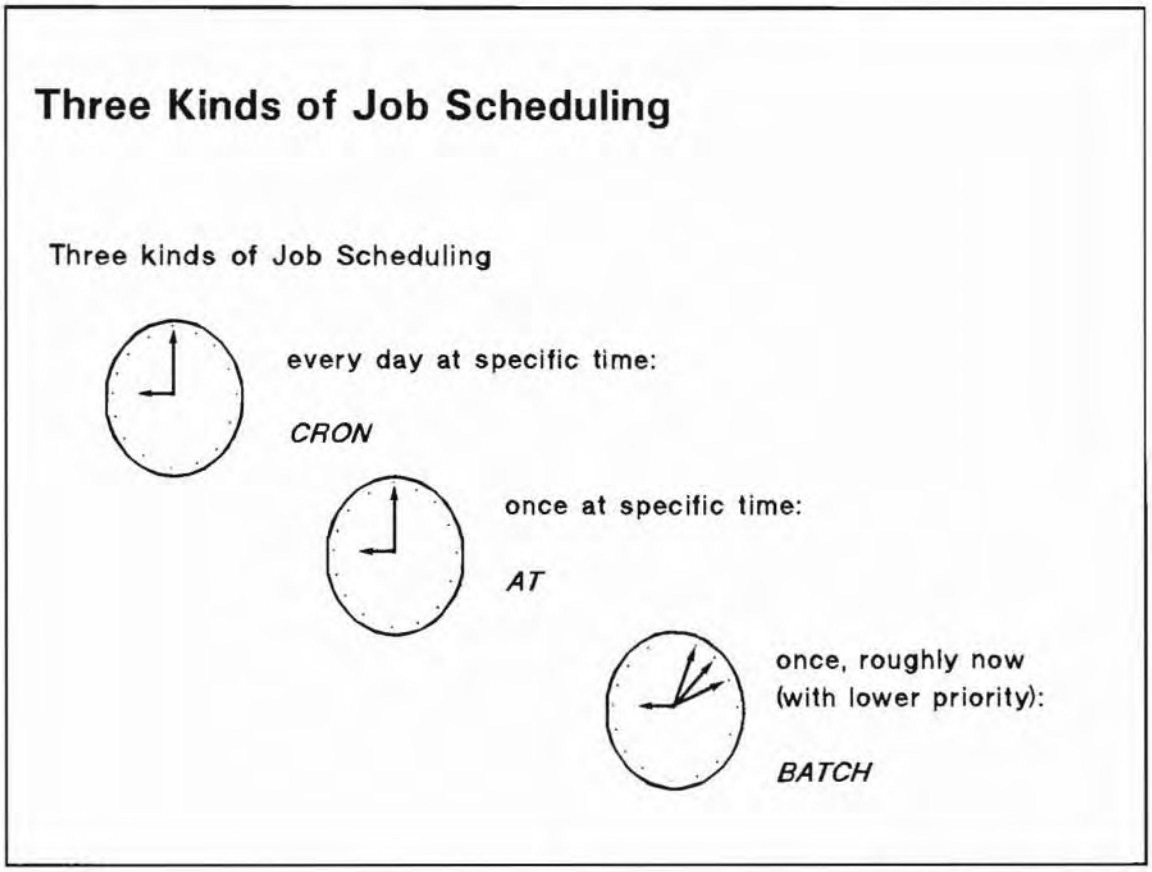
Objectives

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

- Use the `at` and `batch` commands.
- Schedule programs for repetitive invocation with `cron`.

Module 17 — Job Scheduling

17-1. SLIDE: Three Kinds of Job Scheduling



H6296 17-1 183

Student Notes

It is often desirable to execute a program at a later time, or to schedule a program to run at specific time. For example, if a backup program needs to execute everyday at 4 am, there should be mechanism to schedule the program to run so that an individual need not be there to directly invoke the program. HP-UX supplies three mechanisms to schedule programs:

`at`	schedule a job to run "at" a specific time.
`batch`	schedule a job to run in the background at a lower priority; the system decides when to run the job
`cron`	schedule one or more jobs at periodic intervals specified by the user.

`at` and `batch` are generally used to schedule a one-time invocation of a command at a later time, whereas `cron` is typically used to schedule commands to run at regular intervals, such as daily or weekly backups.

17-2. SLIDE: Scheduling Programs for One Time Invocation

Scheduling Programs for One Time Invocation

- at schedules a program to run at a specific time

```
$ at 3:00 pm Mar 3
echo "Happy Birthday" > /dev/console
Ctrl +d
$
```

- batch executes a program in batch mode

```
$ batch
nroff file.frc > file.fmt
Ctrl +d
$
```

H6296 17-2 184

Student Notes

Access to the at command is controlled by the files:

- /var/adm/cron/at.allow
- /var/adm/cron/at.deny

(Note: There are equivalent files for cron, which will be covered later.)

First the at command checks for the existence of the at.allow file. If at.allow exists, then permission to use at is granted to those users whose names are in the file. If at.allow does not exist, at checks for the existence of at.deny. If at.deny exists, then permission to use at is denied to those users whose name appears in the file. If both files exist, precedence is given to at.allow since at checks for the existence of this file first. If neither file exists, only root can use at.

If you create an empty at.deny file, you grant usage to all users. Similarly, if you create an empty at.allow file, you deny usage to all users.

The format for both files is one user name per line.

Table 17-1. Who Can Use cron and at?

*.allow	*.deny	Who Can Use?
—	—	superuser
exists	ignored	everybody in *.allow
—	exists	everybody who is not in *.deny
empty	ignored	superuser
—	empty	everybody

17-3. SLIDE: Scheduling Programs for Repetitive Invocation

Scheduling Programs for Repetitive Invocation

1. Ensure `cron` process is running

   `$ ps -ef | grep cron`

2. Root must add login name to `/var/adm/cron/cron.allow`

   `# vi /var/adm/cron/cron.allow`

3. User creates `cronfile` containing programs to be submitted

   `$ vi cronfile`

4. User submits `cronfile` to `cron` process with `crontab` command

   `$ crontab cronfile`

H6296 17-3 185

Student Notes

The cron process is normally started during boot-up by the `/etc/rc.config.d/cron` script. Therefore, as long as you are in multi-user state, cron should be executing on your system. If for some reason it is not, start it running by typing:

```
# cron
```

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 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`.

Once you have access to the `crontab` command, create a file in your home directory. You may name the file anything, but it may be easier to remember what the purpose of the file is, if you name it something like `cronfile`. This file contains the commands you wish to submit to `cron` and the times you want them to be executed. We will look at the format of this file on the next slide.

To submit your job to `cron`, type:

$ crontab *cronfile*

Table 17-2. Who Can Use `crontab` and `at`?

*.allow	*.deny	Who Can Use?
—	—	superuser
exists	ignored	everybody in *.allow
—	exists	everybody who is not in *.deny
empty	ignored	superuser
—	empty	everybody

17-4. SLIDE: The cronfile

The cronfile

- Consists of lines of six fields each. The first five specify:
  - minute(s) after the hour (0-59)
  - hour(s) of the day (0-23)
  - day(s) of the month (1-31)
  - month(s) of the year (1-12)
  - day(s) of the week (0-6 with 0=Sunday)
- An "*" indicates all legal values
- Fields are separated by spaces or tabs
- The last field is the program to be executed

Example:

```
0 * * * *          /usr/bin/date  >  /dev/console
* 0-23 * * *       /usr/bin/who  >>  /home/bugs/whofile
0 1 * * *          /usr/bin/last  |  /usr/bin/lp ; > /var/adm/wtmp
0 6 * * 1,3,5      /usr/bin/lastb |  /usr/bin/lp ; > /var/adm/btmp
```

H6296 17-4 186

Student Notes

The crontab command allows you to submit jobs to cron to be executed at a later date and time. The commands and their associated execution times are entered in a cronfile.

The entries in cronfile must be in a specific format to be interpreted successfully by cron. Each entry in the file is a line containing six fields separated by white space or tabs. The first five fields contain integers which represent the date and time a command is to be executed. They are shown on the slide. Each of these fields may contain an asterisk which represents all legal values, or a list of entries separated by commas. Each entry may be either a number or two numbers separated by a dash, which specifies a range.

The last field is a string that is executed by the shell at the specified times. A percent character in this field (unless escaped by \) is translated to a new-line character.

You could use cron to help regulate your backups.

Example

```
0 23 * * 0      /home/bugs/fullback
3  4 * * 1      /home/bugs/incrback
```

The script fullback will be run at 11 p.m. every date, every month, on the 0th day of the week (Sunday), and the script incrback will be run at 4:03 a.m. every date, every month, on the first day of the week (Monday).

Note

You must redirect the standard output and standard error of your commands. If you do not do this, any output generated will be mailed to you.

You should 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.

17-5. SLIDE: Modifying Your `cronfile`

Modifying Your `cronfile`

1. List contents of existing `cronfile`

```
$ cd /home/bugs
$ crontab -l > cronfile
```

2. Make changes to `cronfile`

```
$ vi cronfile
```

3. Re-submit your `cronfile` to `cron`

```
$ crontab cronfile
```

4. To remove your cronfile, use the `-r` option

```
$ crontab -r
```

H6296 17-5 187

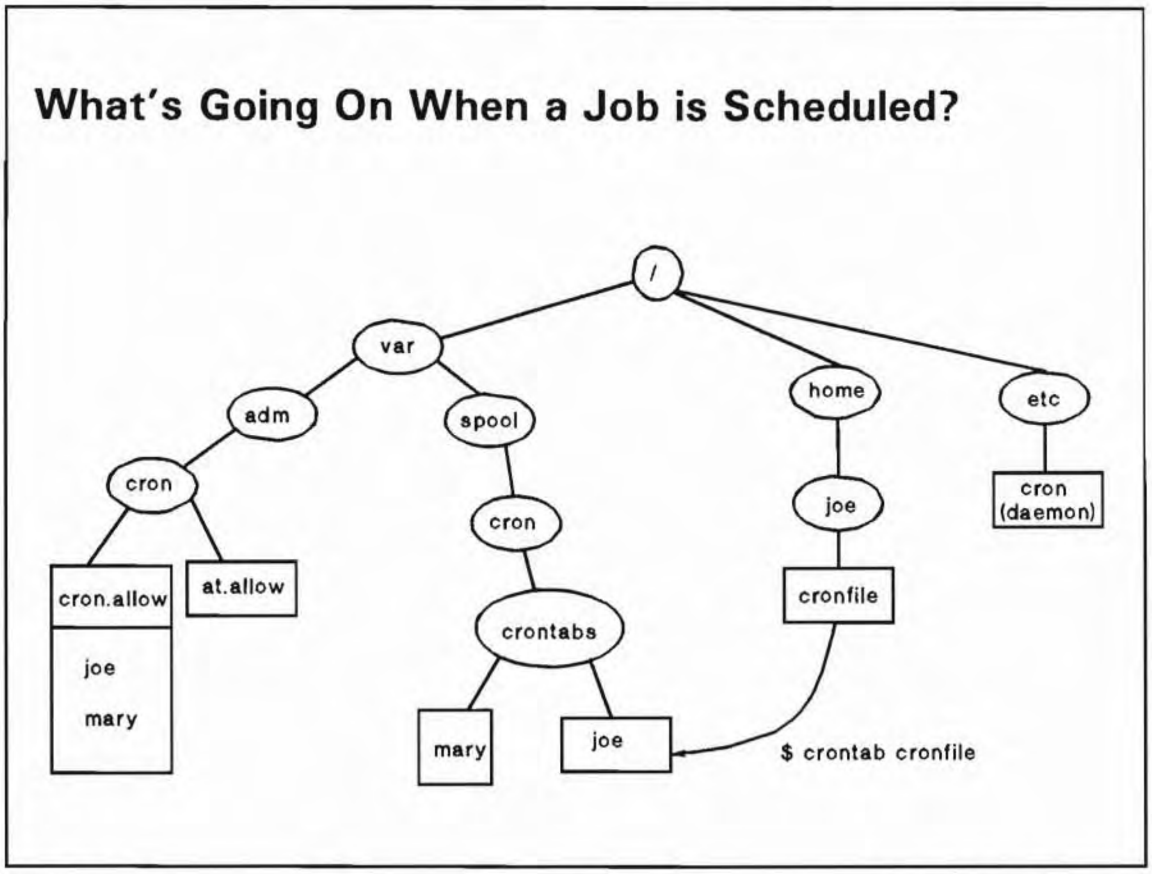
Student Notes

Each user who is authorized to issue jobs to `cron` has *one* file in the directory `/var/spool/cron/crontabs`. The name of this file is the user's login name. Once the user's `cronfile` is activated with the `crontab` command, any future use of this command will cause the crontab file in `/var/spool/cron/crontabs` to be replaced.

To change information in your crontab file, retrieve the existing crontab file, modify it, and then re-submit the new crontab file. `crontab` will replace the old file with the new one.

Note that you must use the `crontab -r` command to remove `cron` jobs. *Do not* remove cron jobs by deleting the file in `/var/spool/cron/crontabs`, since this may leave `cron` in an undefined state.

17-6. SLIDE: What's Going On When a Job is Scheduled?



H6296 17-6 188

Student Notes

When you use the `crontab` command to submit your cronfile to the `cron` daemon, the `crontab` command actually copies your local file to the system `cron` directory `/var/spool/cron/crontabs`. The copy of the file is named with your user name. Once you submit your file with the `crontab` command, your job is activated.

Whenever `cron` executes a command, it writes a record to `/var/adm/cron/log` unless otherwise specified. This file grows without bound, and should be checked and emptied periodically.

17-7. LAB: Hands-On, Scheduling Jobs

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. Time schedule a command which runs at lunch time and echoes out to your terminal - "Lunch time - Where are we going?"

2. Time schedule the `date` program to execute every minute with the display coming tc your screen. (Hint: Make sure you redirect the standard output to your screen device.)

3. Now add another process to your time scheduled list. Have this one execute a program to display to your screen who is logged in. This program should run every 10 minutes.

17-8. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. What type of activities might be useful to schedule with `cron`?

2. When will the following `crontab` entry be run?

```
2 * * * * /usr/bin/who >> /var/adm/who_log
```

Module 18 — Printers and the LP Spooler

Objectives

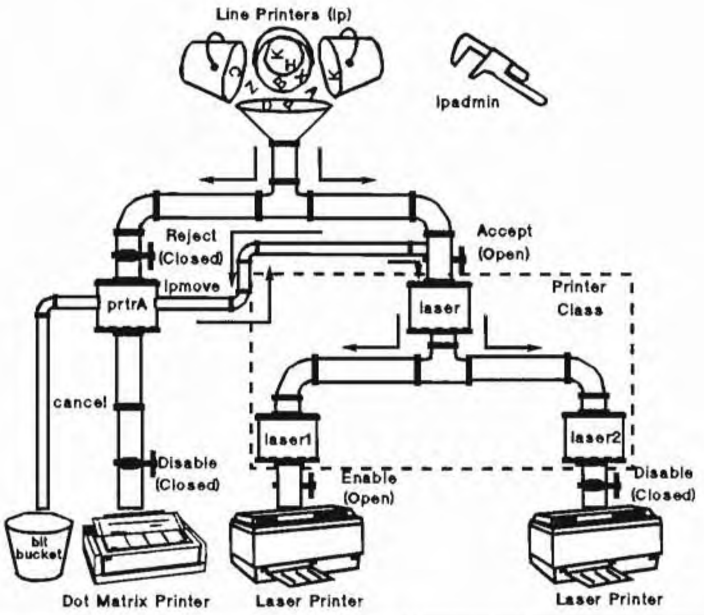
Upon completion of this module, you will be able to:

- Activate the LP (line printer) spooler.
- Add a serial printer to your system.
- Add a local and remote printer to the LP spooler system.
- Start and stop the LP spooler.
- Remove a printer or class of printers from your system.
- Check LP spooler status.
- Enable and disable a printer class.
- Set the default printer.
- Move printing requests to other destinations.
- Explain what print classes are.
- Describe what priority fences are and differentiate them from printer priorities and print job priorities.

Module 18 — Printers and the LP Spooler

18-1. SLIDE: An Overview of the Spooling System

An Overview of the Spooling System



H6296 18-1 189

Student Notes

On a multiuser system, access to the printers requires careful management and control. Since users could possibly send print requests to the same printer at the same time, there needs to be a way to:

- Make sure each file is printed separately.
- Determine which file will be printed first.

The **LP spooler** is a collection of utilities and commands that controls the print requests of users. The LP spooler stores print requests in the spool directory until a printer is available. When a printer is available, print requests are processed, one at a time, in the background.

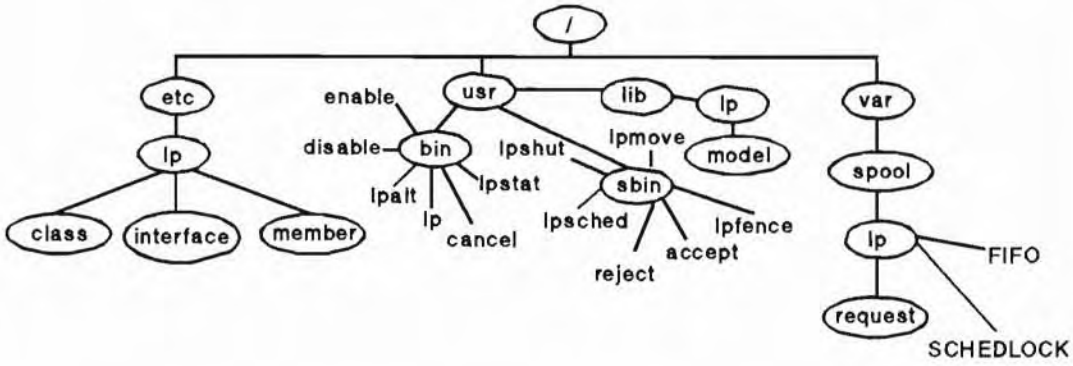
The spooling system can be customized by the administrator. For example, the administrator can choose where requests are spooled, or the administrator can also group several printers into a **class** of printers which can increase the overall efficiency of the system.

Module 18 — Printers and the LP Spooler

After the LP spooler system is installed, any user can submit a job to be printed, obtain the status of all printers or any single printer, cancel any print job, or declare printers to be in and out of service.

18-2. SLIDE: Directory Overview

Directory Overview



H6296 18-2 190 © 1995 Hewlett-Packard Co.

Student Notes

The following is a road map of your way through the directories of 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` Printer classes directory. This contains the files that define how printers are grouped.

`/usr/lib/lp/model` System-supplied interface programs. This directory contains the model shell scripts designed for particular printer models. When adding a printer to the spooling system you will select a model script from this directory. The `lpadmin` command copies this file into the `/var/spool/lp/interface` directory.

`/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, you should 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, such as LP spooler commands that general users can execute.
`/usr/sbin`	Contains administrator commands, such as 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 printers member file is the assigned device file.
`/usr/lib/lp/fonts`	Contains fonts for Laserjet printers.

18-3. SLIDE: What Happens When a File is Submitted With `lp`

What Happens When a File is Submitted With `lp`

`$lp -d`*LJ filename*	Print request submitted
`request id is LJ-####`	Spooler assigns unique request ID and queues request
	filename is linked or copied to spool directory `/var/spool/lp/request/LJ`
Printer becomes available	`lpsched` invokes the interface program `/etc/lp/interface/LJ`
File is printed	

H6296 18-3 191

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.

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 with the `lpadmin` command.

Do not confuse the printer priority with the printer fence level. The printer priority specifies the default priority a print request is queued to the printer with and is configured with the `lpadmin` command. A printer's fence level defines the minimum required priority for the spooled file to be printed and is configured with the `lpfence` command. A print request sent with a priority lower than the printer's fence level will sit in the printers queue until the print job's priority is raised or the fence level is lowered.

If a priority is not specified when issuing the `lp (1)` command, the request will be sent with the default. The default print job priority will be zero unless the System Administrator configured another default with the `lpadmin` command using option `-g"prioriy"`. The `lpalt` command can be used to modify the priority of a print job.

In summary, when a job is submitted, the print job priority is checked against the fence level of the printer to which it was sent. If the print job priority is higher than or equal to that of the printer's fence it will be printed, but not before other jobs of higher priority; otherwise it will just wait in the queue. If its priority is below that of the printer's fence then the job will not be printed until either the job print priority is raised, the printer's fence is lowered, or the job is moved to another printer with lower fence.

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?

For example, 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 to do by just searching for "9600" or "do_banner" in the file and do the modifications.

18-4. SLIDE: LP Spooler Administration Commands

LP Spooler Administration Commands

- Scheduling

```
# lpshut
# lpsched
```

- Queuing

    \# reject -r"*reason*" *printer_name*
    \# accept *printer_name*

- Printing

    \# disable -r"*reason*" *printer_name*
    \# enable printer_name

- Reporting Spooler Status

```
# lpstat -t
```

H6296 18-4 192

Student Notes

The commands on the slide are explained in more detail below.

```
# lpshut
```

`lpshut` stops the scheduler, which in turn stops all printers. Internally, the scheduler is a daemon process that sends files from destination queue directories to the interface scripts. Whenever the structure of the spooling system needs to be changed (for example, by commands such as `lpadmin` or `lpmove`), the scheduler must be stopped. All requests printing when `lpshut` is executed will be stopped and left in the queue. When the scheduler is started again, it will begin printing those files that were stopped as if they were a new request.

Note, even though the scheduler is in a halted state, `lp` can still queue jobs to the various destinations.

```
# lpsched
```

`lpsched` starts the scheduler. It is normally invoked during the boot-up procedure. `lpsched` is the opposite of `lpshut` and must be executed after modifying the spooling system structure in order to start proper operation again.

`# reject -r"`*reason*`"` *printer_name*

`reject` temporarily rejects requests to *printer_name*. The `lp` command is prevented from queuing further print requests to this destination. If a user executes `lp` to the destination printer_name, the user will get an error message similar to this:

`lp: can't accept requests for destination "`*printer_name*`"`

The `accept` command is the opposite of `reject`. It changes the status of a destination to allow the `lp` command to queue requests to this destination. This command is executed when the destination is first created. It need not be executed again unless the LP administrator decides that this destination should not queue up requests, for whatever reasons, and executes a `reject`.

`# disable -r"`*reason*`"` *printer_name*

This command disables the scheduler from sending request to the printer. Users, however, can still submit requests with the `lp` command.

`enable` is the opposite of `disable`. `enable` allows the scheduler to send request to printer_name.

```
# lpstat -t
```

`lpstat` allows you to check the status of the spooling system. `lpstat -t` prints the status of the scheduler and each printer configured in the spooler. For each printer, it tells you if that printer is enabled or disabled, is accepting or rejecting requests, and the request-id of the files that are queued for printing.

18-5. SLIDE: The `lpadmin` Command

The `lpadmin` Command

- Configures the LP spooling system
- Cannot be used when the LP scheduler is running

Syntax:

lpadmin -p*printer* [*options*]	*Names a printer to which the options refer*
lpadmin -x*dest*	*Removes dest from the system*
lpadmin -d*dest*	*Specifies or changes the system default destination*

H6296 18-5 193

Student Notes

The `lpadmin` command configures LP spooling systems to describe printers, classes and devices. It is used to add and remove destinations, change membership in classes, change devices for printers, change printer interface programs, and to change the system default destination.

You use only one of the options `p`, `x`, or `d`.

-p*printer*	Names a printer to which all of the options refer.
-x*dest*	Removes destination *dest* from the LP system.
-d*dest*	Marks *dest*, which already exists, as the default destination.

If you are using the p option, you can specify further options, in any order.

`lpadmin` options with `-p`

-v*device* Required. Specifies the full path name of the printer's device file. Note there is nothing to stop an administrator from associating the same *device* with more than one printer. For example, if a printer allows both pcl and Postscript printing, it can be set up with different model scripts.

-m*model* Required. Selects a model interface program for the printer. *model* is one of the model interface names in the `/usr/lib/lp/model` directory.

-c*class* Inserts the printer into the specified *class*. This is optional; printers do not have to belong to a class.

-g*priority* Sets the default priority for the printer. This is optional; the default is 0.

There are many more options to the `lpadmin` command. See `lpadmin(1M)` for more information.

Note

`lpadmin` creates and modifies files that are being used by the line printer scheduler, so you must be sure that the scheduler is stopped when you use the `lpadmin` command to add a new printer.

18-6. SLIDE: Priorities and Fences

Priorities and Fences

- Set a printer's fence priority

```
# lpfence laserjet 4
```

- Change the printer's default print priority

```
# lpadmin -plaserjet -g4
```

- Submit a print job with a print priority

```
$ lp -p5 file1
```

- Alter a queued print job's priority

```
# lpalt laserjet-200 -p5
```

H6296 18-6 194

Student Notes

It is important to understand the distinction between the priority of a print job, the default print priority of a printer, and the printer's fence level.

The printer fence defines the minimum required priority that a spooled file must have in order to be printed on that printer. The printer fence is set with the `lpfence` command. A print request with a priority lower than the printer fence will stay in the print queue until either the print job's priority is raised, or the printer fence is lowered.

```
# lpfence laserjet 4
```

The `lpfence` command assigns a minimum print priority for the printer. In this example, "laserjet" will accept any print request but will only print those print requests queued with a print priority of 4 or greater. A user's print request is sent to the printer with the default priority or they can specify a priority with the -p option to lp. The print requests default priority will be zero unless the system administrator set a higher default with the -g option to lpadmin.

The printer priority specifies the default priority for any print request that is queued to that printer which does not have a priority specified. It is specified at the time the printer is configured, and can be changed with the lpadmin command. The print job may be submitted with a different priority by using the -p option with the lp command.

```
# lpadmin -plaserjet -g4
```

This command changes the default priority value that a print job is sent to "laserjet" with. This means if a user issues the lp command without the -p option, the print job will be queued with a priority of 4. The range of priority for printers and print jobs is 0 to 7, 0 being the lowest and 7 being the highest. The default printer priority is 0 if not specified with either the lpadmin or lp command. Users would use the -p option with the lp command to submit jobs with a certain priority:

```
$ lp -p5 file1
```

This example submits a job to the default printer with priority 5. Print requests will print from the queue highest priority first. If many files have the same priority, they are printed FIFO, (First In First Out).

```
$ lpalt laserjet-200 -p5
```

If a print request was sent to a printer with a priority lower that the printer's fence, the job will just sit in the queue. The above example changes the priority of the print request_id "laserjet-200" from what it was to 5.

18-7. SLIDE: Other Things the Administrator Can Do

Other Things the Administrator Can Do

- Insert the printer "laserjet" into the class of printers "fastprint"

```
# lpadmin -plaserjet -cfastprint
```

- Remove a printer "qj1" from the class of printers "regprint"

```
# lpadmin -pqj1 -rregprint
```

- Move all requests from destination "lp1" to destination "laserjet"

```
# lpmove lp1 laserjet
```

- Set the system default printer

```
# lpadmin -dlaserjet
```

H6296 18-7 195

Student Notes

The commands on the slide are explained in more detail below.

```
# lpadmin -plaserjet -cfastprint
```

This command inserts the printer "laserjet" into the class of printers "fastprint". Thereafter print requests can be addressed to the destination "fastprint". Note that the scheduler must have been stopped before invoking this `lpadmin` command.

```
# lpadmin -pqj1 -rregprint
```

This command removes the printer "qj1" from the class of printers "regprint" (for "regular printers"). If "qj1" was the last member of the destination "regprint", the class itself would be deleted.

```
# lpmove lp1 laserjet
```

This command moves all requests from destination "lp1" to destination "laserjet". Furthermore, a `reject` command is internally executed for the destination "lp1". If one printer is out of order, for example, due to hardware failure, and there are still print requests waiting to be printed on that specific printer, you may want to move the jobs to another printer that is operating. Note that the scheduler must be stopped before executing `lpmove`.

```
# lpadmin -dlaserjet
```

This command makes "laserjet" the new system default destination. "laserjet" must already be an existing destination. The `-d` option may be used when `lpsched` is running.

Note Always shut down the scheduler before using the `lpadmin` command, except if using the -d option.

Adding and Removing a Printer With TSM

The command `tsm.lpadmin` is used to add (or remove) a printer to the LP spooling system when a printer is connected to the system through a terminal running the Terminal Session Manager (TSM). `tsm.lpadmin (1m)` is a shell script that uses `lpadmin`.

The command is used as follows:

```
tsm.lpadmin -p printer -m model
tsm.lpadmin -x printer
```

To use `tsm.lpadmin` you must be `lp` or `root`.

A second way to add or remove a printer which is connected to a system running TSM is to use SAM.

18-8. SLIDE: Configuring and Adding a Printer to the LP Spooler

Configuring and Adding a Printer to the LP Spooler

What you need to know:

- Is the printer driver in the kernel?
- Does a device file exist? What is its name?
- The name you are giving to this printer.
- Select the appropriate model script from `/usr/lib/lp/model`.
- Do you want to set a default printer priority?
- Do you want to set a printer fence level?
- Do you want to specify a device class?
- Will this be the system default printer?

H6296 18-8 196

Student Notes

A **local printer** is a printer that is physically connected to your system.

Adding a printer to the spooling system is not the same thing as adding a printer to the I/O system. You must do both in order to use a printer on your system.

Note

`lpadmin` creates and modifies files that are being used by the line printer scheduler, so you must be sure that the scheduler is stopped when you use the `lpadmin` command to add a new printer.

Steps for Configuring a Printer and Adding it to the LP

Spooler System

1. Ensure the printer driver is in the kernel. Look at the kernel entries in the `/stand/system`. If the printer driver is not part of the current kernel, you must add it, using either SAM or HP-UX commands.

2. Before adding your printer to the LP spooler system, make sure it is connected and the appropriate device file exists. To determine the device file to use, list the `/dev` directory and find an entry that matches the major and minor numbers of the printer.

   Make sure that the device files:

   - Has a permission mode of 644 (crw-r—r—)
   - Is owned by `lp`
   - Is in the group `bin`

   Once you have connected your printer to your system and have created or identified the appropriate device files, you can add the printer to the LP spooler system.

3. Shut down the LP scheduler.

   ```
   # lpshut
   ```

4. Check `/usr/lib/lp/model` and select the appropriate model script. Decide which model script matches the printer you are adding. You will use this model script name when issuing the `lpadmin` command. The `lpadmin` command will copy this model script into the "interface" directory.

   Once you choose the model script from the `/usr/lib/lp/model` directory, make sure the script:

   - Has a permission mode of 644 (-rw-r—r—)
   - Is owned by `lp`
   - Is in the group `bin`

5. Add the printer to the LP spooler.

   ```
   # lpadmin -plaserjet -v/dev/lp -mPCL3
   ```

   The `lpadmin` command configures the LP spooler to describe printers, classes, and devices. It is used to add and remove destinations, change membership in classes, change devices for printers, change printer interface programs, and to change the system default destination.

6. Allow printer to accept print requests.

   ```
   # accept laserjet
   ```

7. Enable the printer to print the requests.

   ```
   # enable laserjet
   ```

8. Optionally assign a default printer.

   ```
   #lpadmin -dlaserjet
   ```

9. Restart the LP scheduler.

```
# lpsched
```

Once you restart the scheduler with the `lpsched` command, you can verify that the scheduler is functioning with the `lpstat -t` command.

```
# lpstat -t
scheduler is running
system default destination: laserjet
device for laserjet: /dev/lp
laserjet accepting requests since Sep 10 11:15
printer laserjet is idle.  enabled since Sep 10 11:15
     fence priority: 0
```

Note

If you are adding a serial printer to your system, make sure there is not a `getty` running on the serial port.

Note that when you use the command `lpadmin`, exactly one of the following options must be supplied:

`-p`*printer*	Names a printer to which all of the options refer.
`-x`*dest*	Removes destination *dest* from the LP system.
`-d`*dest*	Makes *dest*, which already exists as the default destination.

There are many more options to the `lpadmin` command. See `lpadmin(1M)` for more information.

18-9. SLIDE: How to Add a Remote Printer to the LP Spooler

How to Add a Remote Printer to the LP Spooler

What You Need to Know:

- The name you are giving to this printer.
- The remote system name.
- The remote printer name.
- Select the remote cancel model from `/usr/lib/lp/cmodel`.
- Select the remote status model from `/usr/lib/lp/smodel`.
- 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?

H6296 18-9 197

Student Notes

A **remote printer** is a printer that is not physically connected to your system. To configure a remote printer into your line printer spooling system, you must be able to access the system that has the printer via a local area network (LAN).

There are special "interface scripts" on the remote system that issue cancel and status requests. These scripts are similar to the printer interface scripts. They have a model directory that holds sample scripts, and an interface directory that holds scripts currently in use. The model directories are `/usr/lib/lp/cmodel` and `/usr/lib/lp/smodel`. The interface directories are `/var/spool/lp/cinterface` and `/var/spool/lp/sinterface`.

Note

`lpadmin` is constructing and modifying files that are being used by the line printer scheduler, so you must be sure that the scheduler is stopped when you use the `lpadmin` command to add a new printer.

Preparing to Add a Remote Printer

1. Check `/usr/lib/lp/model` and select the appropriate model script.

   Once you choose the model script from the `/usr/lib/lp/model` directory, make sure the script:

   - Has a permission mode of 644 (-rw-r—r—)
   - Is owned by `lp`
   - Is in the group `bin`

2. Check `/usr/lib/lp/cmodel` and select the appropriate cancel model script.

   Once you choose the cancel model script from the `/usr/lib/lp/cmodel` directory, make sure the script:

   - Has a permission mode of 644 (-rw-r—r—)
   - Is owned by `lp`
   - Is in the group `bin`

3. Check `/usr/lib/lp/smodel` and select the appropriate status model script.

   Once you choose the model script from the `/usr/lib/lp/smodel` directory, make sure the script:

   - Has a permission mode of 644 (-rw-r—r—)
   - Is owned by `lp`
   - Is in the group `bin`

Steps for Adding A Remote Printer to the LP Spooler System

1. Shut down the LP scheduler.

   ```
   # lpshut
   ```

2. Add the printer to the LP spooler.

   ```
   # lpadmin -pmickey -v/dev/null  -mrmodel  -ocmrcmodel
   -osmrsmodel  -ormhped827 -orpmickey
   ```

   This command creates a printer called "mickey" on the local system. The printer "mickey" was already configured on the remote system "hped827" with the same name, "mickey."

   Since the lpadmin command enforces the use of the `-v` option, we use the system's "waste basket" `/dev/null` as output device filename.

3. Allow printer to accept print requests.

   ```
   # accept mickey
   ```

4. Enable the printer to print the requests.

```
# enable mickey
```

5. Restart the LP scheduler.

```
# lpsched
```

Once you restart the scheduler with the `lpsched` command, you can verify that the scheduler is functioning with the `lpstat -t` command.

`lpadmin` Options for Remote Printing

`-ob3`	Uses three-digit request numbers associated with the printer directory. This is for contact with BSD systems.
`-ocm`*cmodel*	The cancel model script is used to forward a "cancel" request to the remote system's line printer spooling system.
`-osm`*smodel*	The status model script is used to forward a "status" request to the remote system's line printer spooling system.
`-orp`*rpname*	The printer name *as it is defined on the remote system.*
`-orm`*remsys*	The name of the remote system to which the printer is physically connected.
`-orc`	Restricts users to canceling only their own requests.

Tasks Performed on the Remote System

The remote system, the one that actually has the printer, must run the `rlpdaemon`. The remote spooling printer daemon handles remote spool requests. `rlpdaemon` runs on a system that receives requests to be printed. It transfers files to the spooling area, displays the queue, or removes jobs from the queue.

To start the `rlpdaemon` daemon, the system administrator on the remote system must edit the `/etc/inetd.conf` and take away the comment symbol (#) so the `rlpdaemon` line looks like the below:

```
printer stream tcp nowait root /usr/sbin/rlpdaemon rlpdaemon -i
```

The system administrator on the remote system now needs to reboot the system or issue the `inetd (1m)` command to reread the `/etc/inetd.conf` file. For example:

```
# /etc/inetd -c
```

18-10. SLIDE: How to Add a Network Based Printer

How to Add a Network Based Printer

What You Need to Know:

- The name you are giving to this printer.
- The printer node name.
- The model or interface that the printer will use.
- The link-level address of the network card installed in the printer.
- The IP address (for TCP-IP protocol printers).
- Do you want to set a printer priority?
- Will this printer be placed in a class of printers?
- Do you wish to make this device your system's default printer?

H6296 18-10 198

Student Notes

A network-based printer is physically connected to the local area network (LAN). A network-based printer is *not* physically connected to a system. HP-UX line printer spooling system configuration instructions are provided with the printer or printer interface card product.

You can use SAM to configure a TCP-IP protocol printer or an NPX protocol printer. You must supply the following information:

- Printer name
- Printer Node Name
- Printer Model/Interface
- Printer Class (optional)
- Default Request Priority
- Whether to Make System Default Printer

If you are using an HP JetDirect card to connect to a TCP/IP Ethernet local area network, you must first load the HP JetDirect software onto your system. Then you must run an installation script, `hpnpinstall`, to set up the file structures required for card and spooler configuration.

You then use the configuration utility `/usr/lib/hpnp/hpnpcfg` to configure the card and spooling system. For more information, refer to the manual *HP JetDirect Card for UNIX*, Part Number J2340-90101.

18-11. SLIDE: SAM Versus Manual Commands

SAM Versus Manual Commands

Actions in SAM Functional Areas:

- Printers/Plotters
  - Add or Remove Local Printer/Plotter
  - Add Remote Printer/Plotter
  - Add Network Based Printer/Plotter
  - Enable/Disable Printer/Plotter
  - Modify Fence Priority
  - Set as System Default Printer/Plotter
  - Startup/Shutdown Print Spooler
- Print Requests
  - Cancel Request
  - Startup/Shutdown Print Spooler

H6296 18-11 199

Student Notes

SAM can also be used to administer the LP spooler system. The tasks that SAM can perform are shown on the slide.

When you choose `Printers and Plotters` from the SAM Control Window, you can choose from either the `Printers/Plotters` or `Print Requests` functional subareas.

In general, `SAM` is easier to use than the manual commands. You are led through each task with menus and templates. You are prompted for all necessary entries, and the entries are checked for plausibility. If an entry isn't correct, you get an error message and are prompted for a different response.

Though you may find SAM easier to use, it is less flexible than the manual commands. You cannot control many of the options available with commands. Also, there are some commands that are not available in SAM. For example, SAM cannot:

- submit a request
- move a request between printers
- accept and reject requests

18-12. SLIDE: The SCHEDLOCK File

The SCHEDLOCK File

- `lpsched` creates the file `/var/spool/lp/SCHEDLOCK` when it starts the scheduler
- `lpshut` removes the files `/var/spool/lp/SCHEDLOCK` when it shuts down the scheduler
- `lpsched` will only allow one scheduler to run at a time; when invoked, it:
  - Checks for the existence of `/var/spool/lp/SCHEDLOCK` to see if a scheduler is currently running
  - If `/var/spool/lp/SCHEDLOCK` exists, `lpsched` will not start up the scheduler

H6296 18-12 200

Student Notes

SCHEDLOCK prevents the execution of more than one scheduler. So, if the SCHEDLOCK file exists when you try to start the scheduler running, the `lpsched` command won't start up the scheduler.

The `lpshut` command automatically removes the SCHEDLOCK file when it terminates the scheduler. If the SCHEDLOCK file does not exist when you execute `lpshut`, then `lpshut` might not work. If `lpshut` won't shut down the scheduler, you can use the following command:

```
# kill -15 lpsched_pid
```

When you start the scheduler again with `lpsched`, it will create the SCHEDLOCK file.

18-13. SLIDE: Troubleshooting the Spooler

Troubleshooting the Spooler

Potential Spooler Problems:

- Spooler won't start
- Scheduler won't stop
- Paper jam/printer out of paper
- Runaway printout
- Printer won't print

H6296 18-13 201

Student Notes

Scheduler

If the scheduler won't start up when you execute the lpsched command, check to see if the SCHEDLOCK files 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
```

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 to restart printing from where it stopped.

Restart From the Beginning

1. Put printer off-line.
2. `disable` the printer.
3. Clear the jam/load paper
4. Put printer on line.
5. `enable` the printer
6. Printing begins from the beginning.

Restart from Stopping Point

1. Clear the jam/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*

Printer Won't Print

1. Move its print requests to another printer

`lpshut`	*shut down the lp scheduler*
`lpmove` *sourceprinter destinationprinter*	*move the print jobs*
`lpsched`	*restart the lp scheduler*

2. Ask Questions!
    a. Do other printers print?
    b. Has this printer printed before?
    c. Were there any error messages?
    d. 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 < \   this configures the printer
     /dev/ttynpn
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

Beginning in HP-UX 10.0, SAM has the ability to save and restore the actual spooling system configuration. This can be very helpful if e.g someone (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 much time consuming.

You should always save the current spooler state with SAM after adding or deleting a printer.

18-14. 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 you have a printer available, use SAM to configure the printer and enable it. Then, exit SAM and print a file like `/etc/passwd`. Finally, use SAM to remove the printer.

2. Determine the status of the LP scheduler. Shut it down if it is running.

3. Add the printer with the `lpadmin` command by specifying the name, device file, and model.

4. Assign the default destination to the printer.

5. Allow the scheduler to accept requests for your new printer.

6. Enable the scheduler to print to your new "printer."

7. Disable your printer, then send a file to the printer. Determine if the files are queued. Now cancel one of the requests from the queue. Enable the printer. What happens?

8. Remove your printer.

18-15. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. What functions does the spooling system provide, and why are they required?

2. Which of these functions are available to the administrator and which to normal users?

3. What is the difference between a device, a printer and a destination?

4. Is it possible, to install two printers for one device?

5. How would you cancel your own print request?

6. How would you cancel a print request owned by someone else?

7. How would the owner of a canceled request know that you had canceled it?

8. What is an interface program?

9. If you have stopped the scheduler (by `lpshut`), does print-out continues?

10. If you have stopped the scheduler (by `lpshut`), can you still use the `lp` command to add print requests to the queues?

11. How can you tell other users that a printer is "broken"?

12. How can you redirect print requests from the "printer1" queue to the "printer2" queue?

Module 19 — Terminals and Modems

Objectives

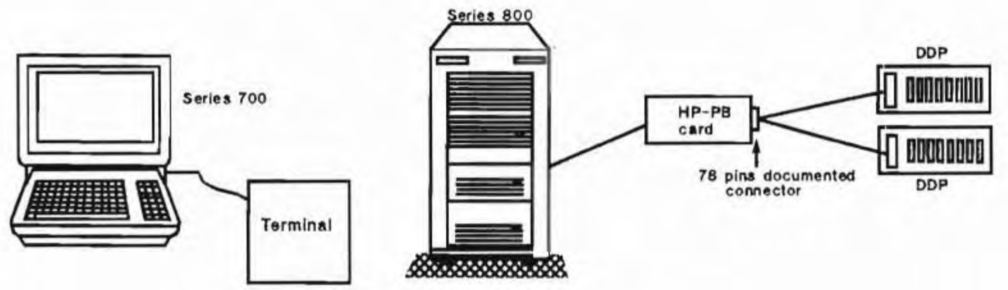
Upon completion of this module, you will be able to do the following:

- List three ways to connect terminals to a system.
- Explain the function of the `getty` process.
- Configure a Modem for dialin and dialout.

Module 19 — Terminals and Modems

19-1. SLIDE: Serial Connection of Terminals and Modems

Serial Connection of Terminals and Modems



H6296 19-1 202

Student Notes

There are actually three ways you can connect terminals to your system:

- To a direct serial connection on a MUX (server) or on the Core I/O (workstations).
- To a DTC (server).
- An X Terminal, on the LAN.

Serial terminals connect to a port on a MUX card on the servers, or to a serial port on the Core I/O on the workstations. A getty must be running on the port.

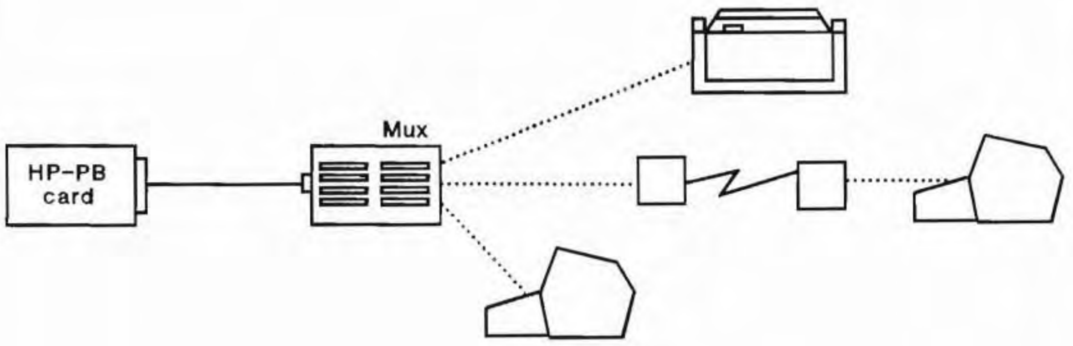
Xterminals connect via the LAN. Xterminals use the industry-standard X Window System, Version 11, Release 4. Xterminals support OSF/MOTIF and come standard with a license to use HP's VUE. X stations deliver the performance needed to display multiple complex graphic applications simultaneously. Pseudo device files are needed for each window used in a session. The login process is controlled through `xdm` or `vuelogin`.

A DTC terminal is considered a pseudo terminal since the DTC connects to the Local Area Network (LAN). Since the DTC connects to the LAN, only one interface card is used to connect up to 72 serial devices per DTC. Multiple DTCs can be connected to one LAN interface card. If you were using MUX cards, one interface card would be needed to handle every 32 serial devices (if you are using 32 channel MUX cards).

This module discusses the serial connection of terminals and modems.

19-2. SLIDE: Connecting a Terminal to a MUX (Servers)

Connecting a Terminal to a MUX (Servers)



H6296 19-2 203

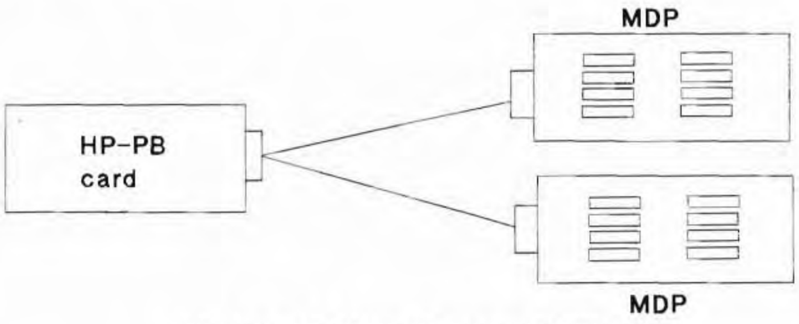
Student Notes

Multiplexers are the means to connect terminals, printers, and other asynchronous peripherals in either direct connect mode (RS-232-C or RS-423), or modem connect mode (RS-232-C), to systems that use the HP-PB backplane. Multiplexers come in 8-, 16- and 32-Channel varieties. MUXes come in three connection configurations:

1. 8, 16 or 32 RS-232-C peripherals in direct connect mode using data signals only (up to 15 meters)
2. 8, 16 or 32 RS-423-A or RS-422 peripherals in direct connect mode for long distance purpose (up to 122 meters)
3. 8, 16 or 32 RS-232-C peripherals connected locally but using data AND modem signals (up to 15 meters) or remotely through use of asynchronous modems.

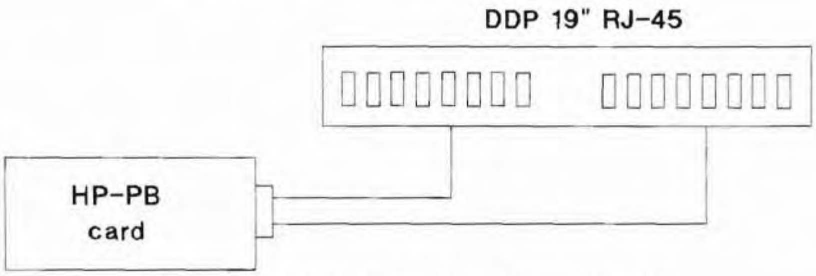
MUXes consist of one single-high HP-PB card and one or two distribution panels. The distribution panels will be either DDP (Direct Connect Distribution Panel) or ADP (Active Distribution Panel) depending on the MUX.

These figures show the various MUXes:



16-channel Direct-Connect MUX

16-channel Modem-Connect MUX



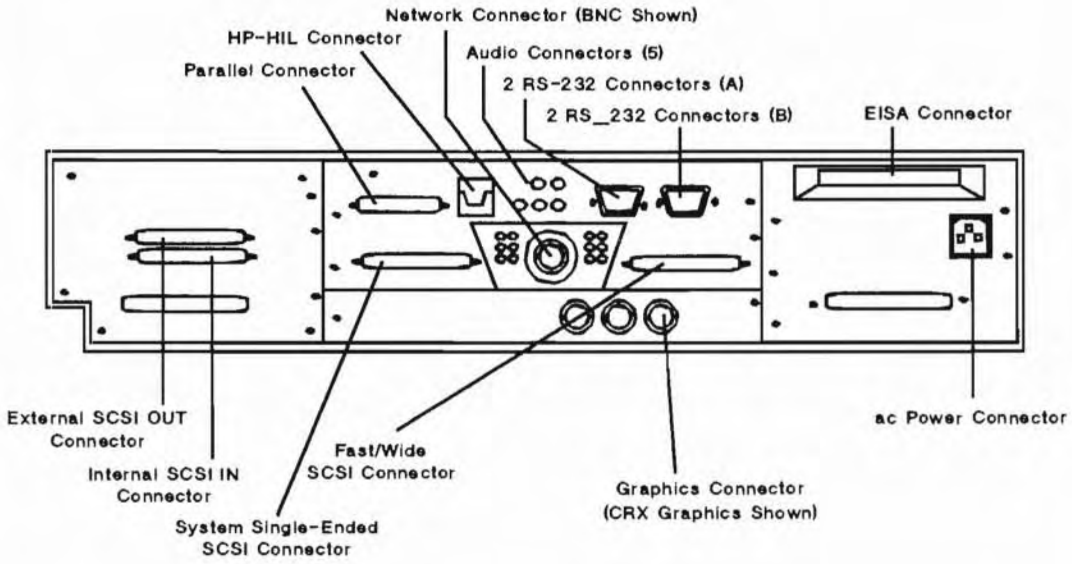
8-channel Direct-or Modem-Connect MUX

A MUX terminal is considered a direct connect terminal. There must be a device file for each terminal connected to the MUX. The device file's minor number will define which MUX card, which port on the MUX, and identify the connection type.

A `getty` process must be running for each direct connect terminal and dialin modem. The `getty` process is started when the system is booted.

19-3. SLIDE: Workstation Terminal Interfaces

Workstation Terminal Interfaces



H6296 19-3 204

Student Notes

Terminals connect to any supported RS-232-C interface.

19-4. SLIDE: The /usr/sbin/getty Process

The /usr/sbin/getty Process

- Normally invoked by `init` via the `/etc/inittab` file
- First command executed for each login
- Operates as follows:
  1. Displays the contents of `/etc/issue`
  2. Issues `login:` prompt
  3. Waits for you to type something then reads login name
  4. Establishes speed and case
  5. Invokes `/usr/bin/login` passing it the login name you typed

Example:

`/usr/sbin/getty [ -h ] [ -t` *timeout* `]` *line* `[`*speed* `]`

`-t` *timeout* is the timeout value (seconds)
`-h` do not force line hangup after timeout
line is the tty line in `/dev`
speed is the label to speed definition in `/etc/gettydefs`

H6296 19-4 205

Student Notes

In normal operating mode, a `getty` process is running on each port on which a user can log in. The `getty` process is normally invoked by `init` according to an entry in `inittab`.

The purpose of `getty` is to set terminal options, print the contents of the `/etc/issue` file (if it exists), print a `login` prompt, wait for input to that prompt, and, following a response by a user, `exec` `/usr/bin/login`.

You must supply `getty` with the device file name of the line on which it is to run. This device file should exist in the `/dev` directory.

Frequently, `getty` is invoked with a `-t` option followed by an integer value representing seconds. If this option is specified, `getty` opens a line and if nothing is typed in the number of seconds specified, `getty` exits.

getty may also be invoked with a -h option. This option is used primarily with modem connection lines. If specified, getty drops carrier when a user logs off. Another login prompt is not issued and the user must dial in again to establish a new connection.

getty also has a speed option. If specified, it serves as a label in to the /etc/gettydefs file. The definition in /etc/gettydefs instructs getty at what speed to run, what to use as a login prompt, what to set as initial tty line settings, and at what speed to try next if the initial speed is inappropriate. With HP-UX, a speed entry of 9600 should be used for terminals connected directly to the system. For dial-up ports, a label corresponding to the appropriate speed for the dial-up line should be used. If a speed value is not designated, a default of 300 baud is used.

The *action* field for a getty entry in inittab is usually respawn. Thus, whenever the getty process terminates, usually when the user logs out, a "wake-up" signal is sent to init. init immediately forks a new getty process. The result is that another login: prompt appears on the terminal connected to that port.

Example from /etc/inittab :

```
a0:4:respawn:/usr/sbin/getty -h tty0p0 H
```

a0	Unique label in /etc/inittab to identify this line
4	Run level(s), in which this line is valid
respawn	init action, here "Restart getty when it died"
/usr/sbin/getty	Program to start
-h	Do not force line hangup, because this is a direct connected terminal and not a modem device
tty0p0	The serial line on which to enable login. The full filename is /dev/tty0p0. getty requires the device file to be found in /dev.
H	The label to use to set serial line parameters found in /etc/gettydefs (/etc/gettydefs will be covered on the next slide).

The actions of getty are extensive and complex. For more information, see getty(1M) in the *HP-UX Reference* manual.

Note

As shipped, /etc/inittab invokes /usr/sbin/getty only for the system console in run-level 1 to 6. If your system has additional terminals on which you wish to support logins, you must add the appropriate getty entries to /etc/inittab. (SAM automatically creates these entries when you use it to add terminals.)

Note

If a user issues a telnet command or rlogin command the getty process is not used. The network daemon starts /usr/bin/login bypassing the getty process.

19-5. SLIDE: The /etc/gettydefs File

The /etc/gettydefs File

```
label#  initial-flags #  final-flags #login-prompt#next-label
```

label	Identifies the entry Matches against getty speed argument in /etc/inittab
initial-flags	Initial line and terminal settings Speed must be specified
final-flags	Final line and terminal settings Speed must be specified
login-prompt	Initial login-prompt printed on the terminal
Next-label	Entry to try next if "break" is typed

H6296 19-5 206

Student Notes

The file /etc/gettydefs contains information used by getty to set up speed and terminal settings for a line. Each entry in the gettydefs file contains a series of hash mark ("#") separated fields having the following format:

```
label#  initial_flags #  final_flags #login_prompt#next_label
```

The meaning of each field is as follows:

label — The string against which getty tries to match its second argument. It is often the speed, such as 1200, at which the terminal is supposed to run, but it need not be.

initial-flags — These flags are the initial ioctl(2) settings to which the terminal is to be set if a terminal type is not specified to getty. The flags that getty understands are the same as the ones listed in /usr/include/sys/termio.h. Normally only the speed flag is required in the *initial-flags.* getty automatically sets the terminal to raw input mode and takes care of most of the other flags. The *initial-flags* settings remain in effect until getty executes /usr/bin/login.

final-flags — These flags take the same values as the *initial-flags* and are set just prior to when getty executes login. The speed flag is required. The composite flag SANE takes care of most of the other flags that need to be set so that the processor and terminal are communicating in a rational fashion. The other two commonly specified *final-flags* are TAB3, so that tabs are sent to the terminal as spaces, and HUPCL, so that the line is hung up on the final close.

login-prompt — This entire field is printed as the *login-prompt.* Unlike the above fields where white space is ignored, white space is included in this field.

next-label — If this entry does not specify the desired speed, indicated by the users typing a (break) character, then getty will search for the entry with *next-label* as its *label* field and set up the terminal for those settings. Usually, a series of speeds are linked together into a closed set. For example, 2400 is linked to 1200, which in turn is linked to 300, which finally is linked to 2400.

The syntax of the gettydefs file must be exact. Spaces within each entry must appear as shown on the slide and a blank line must follow each entry (apart from the very last line).

Two sample entries in the file /etc/gettydefs might look like these:

```
9600    # B9600 HUPCL SANE CS7 PARENB ISTRIP IXANY TAB3
        # B9600       SANE CS7 PARENB ISTRIP IXANY TAB3
        #login: #300

console# B9600 SANE CLOCAL CS7 PARENB ISTRIP IXANY TAB3 HUPCL
        # B9600 SANE CLOCAL CS7 PARENB ISTRIP IXANY TAB3 HUPCL
        # Console login: #console
```

For more information on the gettydefs file see gettydefs(4), stty(1) and termio(7) in the *HP-UX Reference* manual.

Note — It is strongly recommended that after making or modifying /etc/gettydefs, it be run through getty with the -c option (for check) to be sure there are no errors.

The following procedure is recommended:

```
# cd /etc
# cp gettydefs gettydefs.new
# vi gettydefs.new
    .
    .
    .
```

```
# getty -c /etc/gettydefs.new | more
# mv gettydefs.new gettydefs
```

19-6. SLIDE: Configuring a Serial Terminal

Configuring a Serial Terminal

What You Need to Do:

- Connect terminal
- Determine what device file to use
- Check `/etc/gettydefs` for appropriate entry
- Create entry in `/etc/inittab`
- Add entry to `/etc/ttytype`
- Run `telinit q`
- Use `ps -ef` to verify `getty` is running on the port

H6296 19-6 207

Student Notes

To add a terminal, perform these steps:

1. Connect the terminal to the desired port.
   a. For servers, determine the hardware address for the MUX card since this will be needed to determine which device file to use.
   b. For workstations, be sure of where on the interface you are plugging the terminal. The HP 9000 has both parallel and RS-232-C ports. These ports frequently appear identical. The built-in parallel interface should be labeled PARALLEL. Make sure you plug your cable into an RS-232-C port.
2. Find the appropriate device file in the `/dev` directory.
   a. The device file should exist since the mux and asio drivers are autoconfigurable at boot time. To find the correct device file use the `ioscan` command to determine the Logical Unit (LU) number assigned to the MUX (on servers) or the Card Instance (I) (on workstations).

Example (workstation):

```
# ioscan -f

Class       I  H/W Path    Driver      S/W State   H/W Type
Description
=============================================================================
...
target      3  2/0.1.6     target      CLAIMED     DEVICE
disk        2  2/0.1.6.0   sdisk       CLAIMED     DEVICE      HP       C3010
lan         0  2/0.2       lan2        CLAIMED     INTERFACE   Built-in LAN
hil         0  2/0.3       hil         CLAIMED     INTERFACE   Built-in HIL
tty         0  2/0.4       asio0       CLAIMED     INTERFACE   Built-in RS-232C
tty         1  2/0.5       asio0       CLAIMED     INTERFACE   Built-in RS-232C
ext_bus     1  2/0.6       CentIf      CLAIMED     INTERFACE   Built-in Centronics Interface
audio       0  2/0.8       audio       CLAIMED     INTERFACE
...
```

If you connect the terminal to the second Built-in RS-232C, you should look for `/dev/tty1p0`.

Terminal device files naming convention are:

```
/dev/ttyLpP
```

where:

L	Logical Unit Number (servers) or Card Instance (workstations)
P	Port on the MUX; on workstations usually 0

Use the `lssf` command to verify the device file is configured correctly.

Example:

```
For Workstations:
# lssf /dev/tty1p0
asio0 card instance 1 hardwired at address 2/0.5 /dev/tty1p0

For Servers:
# lssf /dev/tty0p5
mux2 lu 0 port 5 hardwired address 56 /dev/tty0p5
```

If for some reason the device file does not exist you will need to create it with the **mksf** command.

Example:

```
# mksf -d mux2 -l 0 -p 5 -h  /dev/tty0p5
```

Most of the time the device drivers required for the terminal will already be a part of `/stand/system`. However, if you have edited your `/stand/system` you may want to verify that the driver you need is present. If the `/stand/system` does not contain the driver you need to operate your terminal, you must add the driver to `/stand/system` and remake the kernel.

Create the appropriate device file in the /dev directory. Communication ports (user terminals as well as modems) need to be identified by one or more device files, depending on the intended use of the port. Device file naming conventions vary, depending on the use of the device. Terminal (tty) files are required for terminals.

3. Check /etc/gettydefs and add a baud rate chain for the new terminal if necessary, or use an existing baud rate chain.
4. Edit /etc/inittab and add a getty for the new terminal.

```
tt05:2:respawn:/usr/sbin/getty -h tty0p5 9600
```

5. Add an entry to the /etc/ttytype file

   The /etc/ttytype file is a database that contains the terminal type of the terminal associated with each port on the system. It is used by the tset and login commands. /etc/ttytype entries have this form:

   model_number location

   where:

   model_number is the product number of the terminal as defined in /usr/share/lib/terminfo

   location is the device file associated with the

   For example:

```
2392      console           # System console
2392       tty00             # Bill's terminal
```

6. Invoke telinit q
7. Invoke ps -ef to see if the getty is running.

Note

A terminal getty can be temporarily turned off by changing the action from respawn to off. You will then need to invoke telinit q to kill the current getty process running. The getty will not be respawned until you change the off back to respawn and rerun telinit q. Turning a getty off is useful when troubleshooting a device file problem.

Removing a Terminal

If you want to remove a terminal from your system, follow these steps:

1. Determine the device file associated with the physical terminal. You can do this by logging into the terminal you wish to remove and issue the command tty.

```
# tty
/dev/tty0p4
```

2. Verify that no one is using the terminal by issuing a `ps -ef` command. You should find a `getty` running on the port.

```
# ps -ef |grep tty0p4
     root  1075     1  0  May 19  ?     0:00 /usr/sbin/getty -h tty0p4 9600
     root  2830  2430  3 10:46:20 console  0:00 grep tty0p4
```

3. Edit `/etc/inittab` and delete the line for this terminal.
4. Notify `init` that `/etc/inittab` has changed:

```
telinit q
```

19-7. SLIDE: Adding Modems

Adding Modems

- Configure modem
- Create device files
- For Dial-In line:
  - Edit `/etc/inittab`
  - Rerun `init` in desired run-level
- Connect modem
- For Dial-Out uucp line:
  - Edit `/etc/uucp/Devices`
  - Confirm entry in `/etc/uucp/Dialers`

H6296 19-7 208

Student Notes

Modem connections are commonly used when a user at a remote site needs access to a computer at another geographic location. When Local Area Networks are not available or not feasible, modems are an excellent solution. Modems can allow dialin and/or dialout capabilities.

Prepare the Modem

The first step in configuring a modem is to set the dip switch settings and/or memory of the modem appropriately. As an example, Hayes 2400 baud modem should be configured as follows:

Dip Switch Settings:

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
UP	UP	:	UP	UP	UP	UP	:	UP	UP
:	:	DOWN	:	:	:	:	DOWN	:	:

It is very important that you have the correct cable for attaching your modem. Verify the proper cable is being used. An HP 24542M cable is recommended.

Create the Device Files

You will need three device files for your dial-in and dial-out modem, two for the dial-out and one for the dial-in. The naming convention used for modems is:

```
/dev/ttydnn  -  For Dial-In
/dev/culnn   -  For Dial-Out (Call Unix Line)
/dev/cuann   -  For Dial-Out (Call Unix Autodialer)
```

Set Up for Dial-In Lines

For the dial-in line, you will need to start a getty on the modem device.

1. With the modem disconnected from the port, add an entry to `/etc/inittab` file. For example:

   ```
   01:2:respawn:/usr/sbin/getty -h ttyd02 2400
   ```

2. After saving the file execute a `telinit q` command. Verify that getty is running on port and that it is in a pending state, meaning there should be a ? mark in the tty field. If the modem is not in a pending state, then hardware should be checked out. These steps can be done as follows:

   ```
   # telinit q
   # ps -ef | grep ttyd01
     root       0  7477  7032  0 10:35:56 console  0:00 grep ttyd01
     root       0  7473     1  0 10:35:32?       0:00
   /usr/sbin/getty -h ttyd01 2400
   #
   ```

3. With the modem disconnected from the port and powered on, the DTR or TR light should be off. When the modem is connected to the port, the DTR or TR light should come on. Therefore the port is setting DTR or TR, not the modem. If the DTR or TR light remains on all the time, this indicates that the modem has DTR strapped high and the setting should be changed. You should check your modem users manual for procedure on how to change strapping.

4. After the modem is connected to the port, perform anotherps `-ef | grep ttyd01` and verify that getty is still in a pending state. If getty has a port number in tty field instead of a question mark, then the modem has carrier detect (CD) strapped high. Check your modem users manual for the procedure on how to change strapping.

5. At this point the modem is ready for dial-in use.

Set Up for Dial-Out Lines

To set modem up for callout uucp usage you must add these entries to the `/etc/uucp/Devices` file assuming the modem is connected at tty01:

```
ACU cul01 cua01 2400 hayes
Direct cul01 cua01 2400 direct
```

The first field identifies either Automatic Call UNIX (ACU) for a modem or Direct for system to system communications. The next two fields indicate the callout device files to use. The fourth field identifies baud rate, and the last field indicates type of modem. The last field is a label into the `/etc/uucp/Dialers` file.

To test dialout ability use the following command:

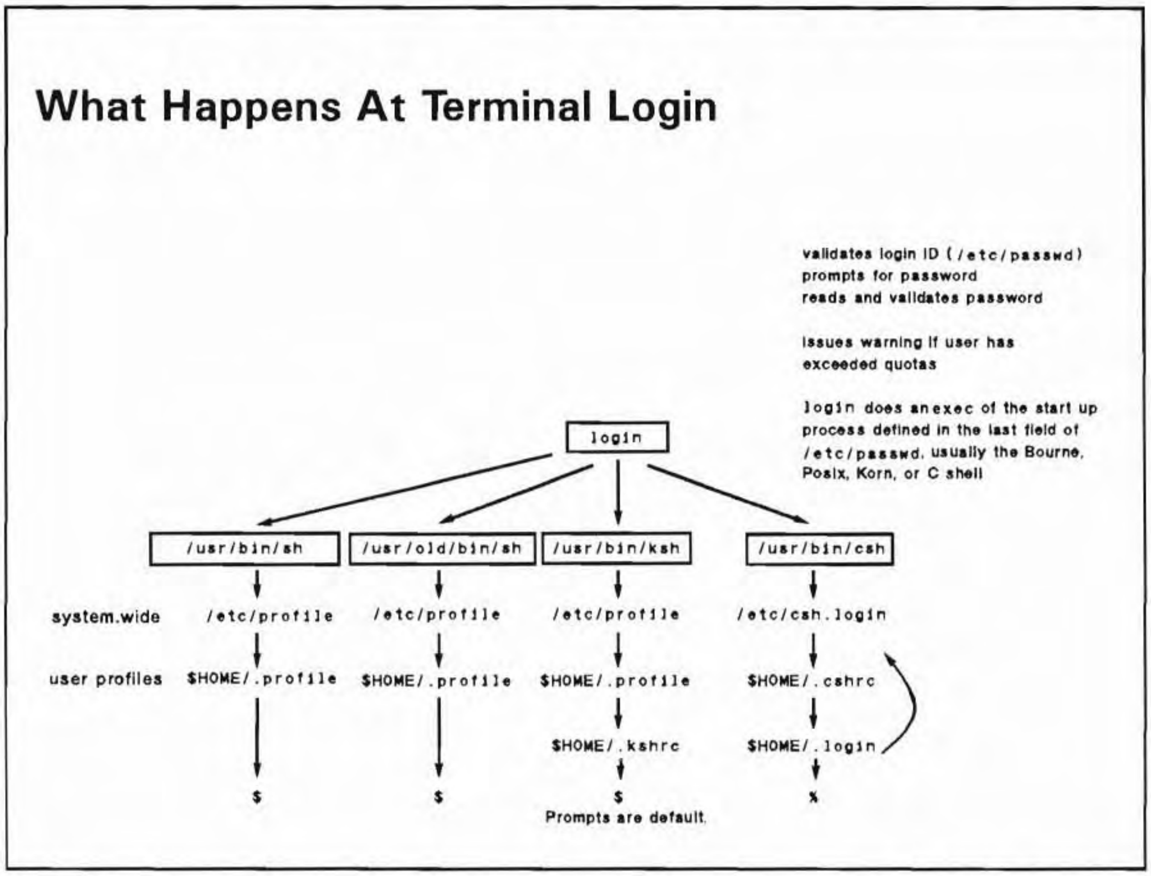
```
cu -s2400 -lcul01 direct
```

After this is entered you should get a message indicating that you are connected. Entering `AT` and return should give you a OK back, but if it doesn't this may be due to the modem having echo turned off. If so enter "ATDTphone number" and listen to the modem to see if it dials. If the modem doesn't dial then have the modem checked out.

The modem is now ready for callout using:

`cu -s2400` *phone number*

19-8. SLIDE: What Happens At Terminal Login



H6296 19-8 209

Student Notes

Allowing users to log into an HP-UX system is a three step procedure. The steps are:

`/usr/sbin/getty -> /usr/bin/login ->` *command*

We have already discussed the operations of `getty`. Now we will discuss `login`. The following steps describe what `login` does.

1. `login` searches the `/etc/passwd` file for the user name.
   a. If `/etc/securetty` is present, login security is in effect, meaning that root login is possible only on the ttys listed in this file. If a root login attempt is made on a not listed tty, `login` handles this login attempt as an invalid login, even if the correct password is typed in. `login` goes to step c.
   b. if the user name exists, `login` goes to step 2.

c. if the user name does not exist, `login`:

   i. Prompts the user for a password (`Password:`). This makes it more difficult for an intruder to find and use a valid user name.

   ii. Displays the message `Invalid login`.

   iii. If `/var/adm/btmp` is present, `login` logs all unsuccessful login attempts to this file.

   iv. If this is the user's third consecutive invalid login attempt, `login` exits; otherwise, `login` prompts the user (`login:`) for a user name and repeats step 1.

2. `login` checks to see if the user name's password field is set in `/etc/passwd`.

   a. If so, it prompts the user for a password (`Password:`) and goes to step 3.

   b. If not, then the user need not enter a password; go to step 4.

3. `login` compares the password to the user name's encrypted password in `/etc/password`.

   a. If the password matches, `login` goes to step 4.

   b. If the password does not match, `login` displays the message `Invalid login`. If this is the user's third consecutive invalid login attempt, `login` terminates; otherwise, `login` prompts the user for a user name (`login:`) and control passes back to step 1.

4. `login` sets the user name's numeric user ID, group ID, and home directory from the corresponding fields in `/etc/passwd`. `login` also updates the `/etc/wtmp` file, which keeps track of valid logins.

5. `login` runs the `/usr/bin/quota` command and warns the user if he or she has exceeded the quotas set by the system administrator.

6. `login` runs (via `exec` system call) whatever command is present in the command field of `/etc/passwd`. Typically, this field is set to the path name of the shell the user wishes to use; that is:

   a. `/usr/bin/ksh` (Korn shell)

   b. `/usr/bin/sh` (POSIX shell)

   c. `/usr/bin/csh` (C shell)

   d. `/usr/bin/keysh` (Key shell)

   If the command field is empty, `login` starts a POSIX shell by default.

Note

The Bourne shell is no longer standard in HP-UX 10.0. You will find the Bourne shell binary in `/usr/old/bin/sh` if you need it. The POSIX shell features are a superset of the Bourne shell's.

19-9. SLIDE: Setting the Terminal Type

Setting the Terminal Type

Syntax:

```
ttytype [-apsv] [-t type]
```

Examples:

```
# ttytype -ps
TERM = hpterm      (user typed in hpterm)
TERM='hpterm'; export TERM;
ERASE='^H'; export ERASE;
# ttytype -ps
TERM =             (user typed nothing in, just (Return) .
TERM='X-hpterm'; export TERM;
LINES=30; export LINES;
COLUMNS=104; export COLUMNS;
ERASE='^H'; export ERASE;
```

H6296 19-9 210

Student Notes

The examples are usually evaluated by `eval 'ttytype -ps'`, so that the variables are set in the current shell.

To communicate effectively with your terminal, HP-UX must know the type of terminal or graphics display you are using. The TERM environment variable supplies the terminal type to HP-UX and can be set with the `tset` command.

The default local login script prompts the user in the following way:

```
TERM = (hp)
```

Pressing (Return), sets the TERM environment variable to hp, the default value. This value works with HP terminals, but it may not let you take full advantage of your terminal or graphics display features. Entering a different value sets the TERM environment variable to that value.

The `ttytype` command automatically identifies the current terminal type by sending an identification request sequence to the terminal. This method works for local, modem, and remote terminal connections, as well as for the `hpterm` and `xterm` emulators.

HP-UX supports many terminal types. The `/usr/share/lib/terminfo` database tells HP-UX how to communicate with each terminal type. When you assign a value to TERM, the value must equal a value in the `terminfo` database.

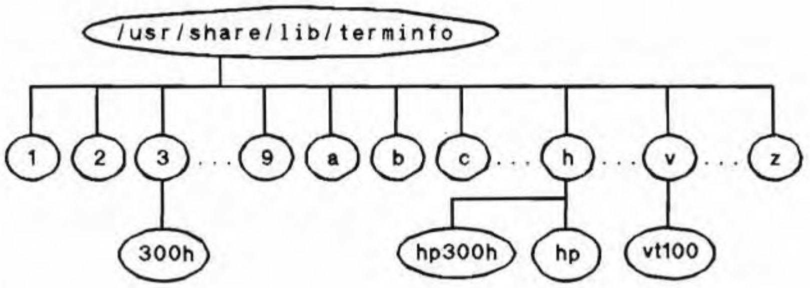
The `stty` command can be used to change various terminal characteristics, such as baud rate, erase and kill characters, etc. For example, the following stty command resets the erase and kill characters to # and @, respectively:

```
stty erase # kill @
```

`stty` also displays terminal settings if invoked with the -a option.

19-10. SLIDE: The terminfo Data Base

The terminfo Data Base



H6296 19-10 211

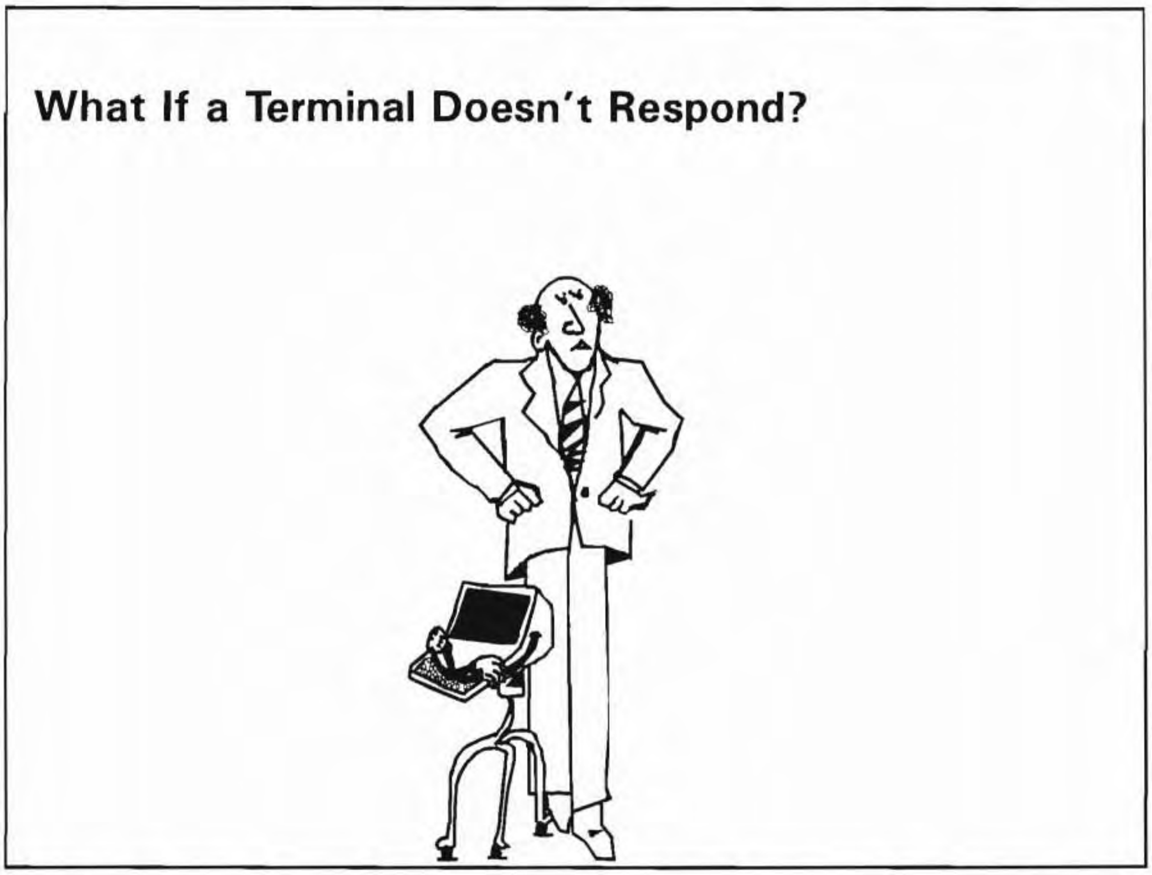
Student Notes

The contents of the environment variable TERM is a pointer to a file in the terminal capability database terminfo. Terminals are described in terminfo by giving a set of capabilities which they have and by describing how operations are performed.

For the sake of a faster access to the terminfo files, an additional level of directories was added. These directories are named after the first letter of the file names.

The terminfo database contains files describing HP terminals and many commonly used non-HP terminals. The set of files is updated by HP. Usually a system administrator need not modify these files. If you find you need to modify a terminfo file, translate it from an internal, compiled format into an open, source format with the untic command. The tic command compiles the modified source file and places the compiled file back into the terminfo database.

19-11. SLIDE: What If a Terminal Doesn't Respond?



H6296 19-11 212

Student Notes

There are many things that can cause a terminal not to respond (no characters are displayed except, perhaps, those which are displayed by the terminal's local echo setting). Here is a procedure you can use to find many of them.

1. *Simply check the cabling.*

   This seems to be obvious, but it is often the reason of terminals not working and it's easy to do. Sometimes the cable connections come loose when users tangle their feet in the cabling . . . or whatever . . .

2. Check the status of the system

   Is the system still up? If not, you've probably found your problem. You will need to reboot the system.

Is the system in single user state? If so, the only active terminal will be the system console. All other direct connect terminals will not respond. You will need to switch to a multi-user state.

To check what run state your system is in (from a working terminal) type:

```
# who -r
```

The output will look something like:

```
.        run-level    Feb 10 07:10    2    0    S
```

The current state of the machine is in the field immediately to the right of the time (third field from the right). For complete information on each of the fields, consult the `who(1)` manual reference page.

3. Check to see if an editor is running on the terminal

From another terminal, issue the command:

```
# ps -ef
```

Look in the column marked TTY for all processes associated with the terminal you are having problems with. For each entry, check in the column marked COMMAND to see if the process is an editor.

If you find that an editor is running at the terminal, it is probably in a text-entry mode. You will need to save the work and exit the editor. For directions on how to do this, consult the manual reference page for the appropriate editor. If saving the file is not possible, you will need to kill the editor process.

`kill` *process-id*

Caution The file being edited on the hung terminal will not be saved by the kill command.

4. Enter ctrl-q at the terminal keyboard

Terminals frequently use the XON/XOFF protocol to start and stop output to them. If output to the terminal was stopped because an XOFF signal ctrl+s was sent from the terminal to the computer, it can be restarted by sending the computer an XON signal (type ctrl+q from the problem terminal's keyboard). Sending the XON signal does not harm anything even if no XOFF signal was previously sent.

If the problem is an application program that's looping or not functioning properly, try pressing the break key and then try ctrl+c to see if you can get a shell prompt back (ctrl+c is the default interrupt character; you might use a different one). If you need to find out what the interrupt character for the affected terminal is, go to a working terminal and enter the command:

`# stty < /dev/`*device file name for the problem terminal*

Caution The stty command, above, should only be used with device file names for currently active terminal device files (use the `who(1)` command to see which device files are active). If you attempt to execute stty with a non-active device file, you will hang the terminal you entered the command from.

5. Reset the terminal

   The terminal itself may be stuck in an unusable state. Try resetting it. Consult your terminal owner's manual for information on how to do this. Powering the terminal off, waiting for a few seconds and powering it back on will reset the terminal (but there is probably an easier and better way to do it). Hewlett-Packard terminals normally reset with SHIFT+CTRL+RESET.

6. Check the terminal configuration

   The terminal might not be configured correctly. If your function keys are not highlighted, press the Menu key, middle of function keys, then press User System which is one key to the right of Menu. Choose F4, Modes and verify the following:

   a. Is the terminal in `Remote *` mode? It should be.

   b. Is `Block` mode turned Off? It should be.

   c. Is `Line` mode turned OFF? It should be.

   d. Is `Modify` mode turned OFF? It should be.

7. Check the physical connection

   Check to make sure that:

   a. All cables are firmly attached and in their proper locations.

   b. All interface cards are firmly seated in their slots.

8. Kill processes associated with the problem terminal

Caution Use extreme caution when killing processes. The processes will be immediately and unconditionally terminated, so be sure you are not killing a valid process that just happens to be taking a long time to complete. Be sure not to make typos when entering the PID numbers for the kill command. You could accidentally kill the wrong process.

If you have another terminal that is still working, go to that terminal and login (you will need to be superuser). Execute the command:

```
# ps -ef
```

The output will look similar to this:

```
UID    PID  PPID  C    STIME TTY      TIME COMMAND

root    95     1  0  Jul 20  ?        0:00 /usr/sbin/getty -h tty00 9600
root 14517     1  0  Jul 21  tty02  0:01 -ksh [ksh]
stevem 20133     1  0 11:20:24 tty01  0:00 -ksh [ksh]
stevem 21234 20133  0 12:22:05 tty01  0:01 rlogind
stevem 21235 21234  0 12:22:12 tty01  0:04 rlogin remote
```

Look in the column marked TTY for those processes that are associated with the terminal you are having problems with. Look at the column marked PID for those entries (these are the process IDs for the processes associated with that terminal). If all you see is a getty process running kill the process associated with the getty instead. Execute the following command, listing each process ID associated with the problem terminal:

`kill -9` *process-id* `[`*process-id*`...]`

If, in the example above, we wanted to kill the processes associated with terminal tty0p1, we would execute the command:

```
# kill -9 21235 21234 20133
```

This should kill all processes associated with that terminal. The `init` process will then respawn a getty process for that terminal (if it has been set up to do so, in the `/etc/inittab` file) and you should once again be able to login.

9. Attempt to re-login to the previously hung port.
10. Use `cat(1)` to send an ASCII file to the hung terminal's device file.

    Try using the cat command to send an ASCII file (such as `/etc/motd` or `/etc/issue`) to the device file associated with the problem terminal. For example, if your problem terminal is associated with the device file `ttyp01`:

    ```
    # cat /etc/motd > /dev/tty0p1
    ```

    You may have to Ctrl+c to terminate the `cat`.

    You should expect to see the contents of the file `/etc/motd` displayed on the terminal associated with the device file `/dev/tty0p1`. If you do not, continue to the next step.
11. Check the parameters of the device file for the problem terminal.

Device files have access permissions associated with them, just as other files do. The file's access permissions must be set so that you have access to the file. If you set the files permissions mode to 622 (crw—w—w-), you should be safe.

If the file's permissions are set to allow write access and the file isn't displayed on the terminal, check the major and minor numbers of the device file. You can list them with the ll command. Consult the *System Administration Tasks* manual for information on the format of minor numbers and for information on what the major number should be. The `lssf` command can be used to interpret the major and minor numbers and display the results.

12. Verify entry in `/etc/inittab`

Make sure your inittab entries are active (`telinit q`).

If you are just adding this terminal and have made a new entry in the `/etc/inittab` file by editing it, remember that this doesn't automatically make your new entry active. To do that you need to enter the command:

```
# telinit q
```

This tells the init process to scan the `/etc/inittab` file to update the information in its internal tables bases on the current run-level.

19-12. LAB: Hands-On Terminal Configuration

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. Use SAM to answer this question. Invoke SAM, then select `Peripheral Devices ->` from the Control Box, then select `Terminals and Modems` from the Peripheral Devices functional area. Choose the "Add Terminal" action from the Terminals and Modems list screen. If you were adding a terminal, what information would you need to provide so that SAM could successfully add the terminal?

2. Perform all the necessary steps to configure a terminal using HP-UX commands. If you are using a Server, assume you are adding a terminal to port 5 on logical unit 0. The only step you will not be able to perform is rerunning `init`. If you are using a workstation, assume you are adding a terminal to port 0 on Card Slot 204.

19-13. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. When you add a terminal to your system, what file should you modify to start the `getty`?

2. When adding a dial-in and dial-out modem, which files need to be modified?

Module 20 — LAN Setup Cookbook

Objectives

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

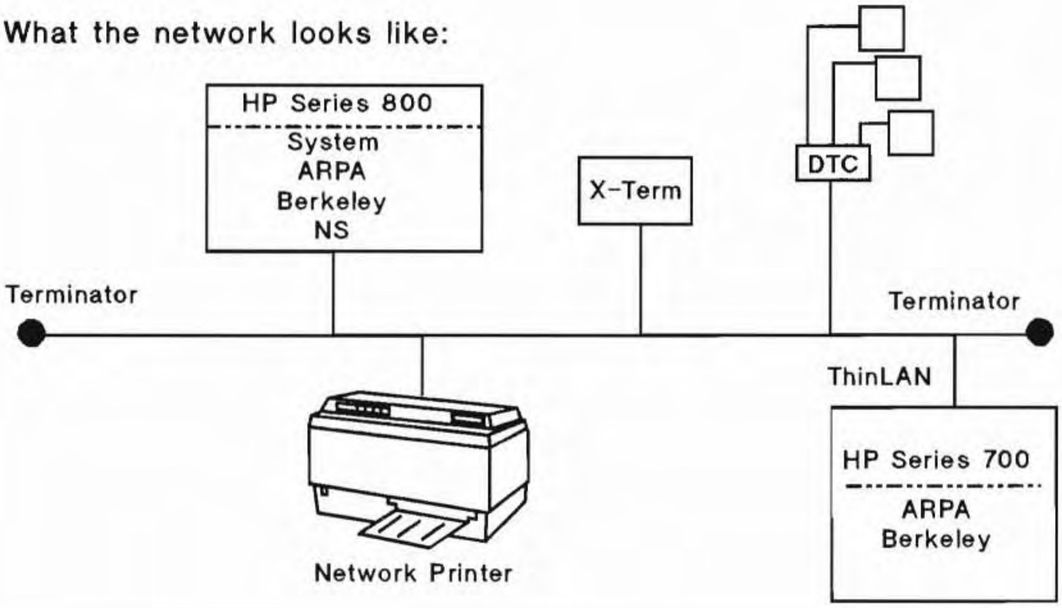
- Describe the functionality of the LAN.
- Describe important networking terms.
- State the steps for configuring a LAN.
- Describe Basic LAN Troubleshooting techniques.

Module 20 — LAN Setup Cookbook

20-1. SLIDE: A Sample LAN Configuration

A Sample LAN Configuration

What the network looks like:



H6296 20-1 213

Student Notes

This is a sample of what your LAN network may look like.

The software packages for LAN include:

ARPA	Advanced Research Projects Agency networking package
Berkeley	University of California at Berkeley networking package
NS	HP Network Services
OSI	Open Systems Interconnection
NFS	Network File System

NS is limited by proprietary nature. As such, it runs on HP equipment only (with a few exceptions). Berkeley is generally limited to UNIX-like operating systems. ARPA is, by far, the most widespread utility, spanning product vendors as well as operating systems. OSI is still gaining acceptance, but will play an increasingly important role as an industry standard.

NFS Services

Another important networking package available on HP-UX is NFS. NFS is the defacto UNIX remote file access standard today for networking file systems. NFS was developed by Sun Microsystems

20-2. SLIDE: Essential Network Configuration Information

Essential Network Configuration Information

- Host Name
- Internet Protocol Address (IP)
- MAC Address or Physical Address

H6296 20-2 214

Student Notes

It is always confusing to hear so many different networking terms. Below are the most common networking terms and their definitions. Determining the correct value for these terms is also confusing so we have included the commands needed in determining these values.

Host Name or System Name

This is the name by which people using the network can uniquely identify your system. You can obtain the system's official hostname from the person who administers that system or if you have a user account on the desired system you may log in and issue the `hostname` command:

```
# hostname
hped710
```

The output from the `hostname` command is that system's name.

If the system is a new system and has not been assigned a hostname (that is, the output of the `hostname` command is "unknown"), a hostname must be assigned in the file `/etc/rc.config.d/netconf`. The assignment would look like this:

```
HOSTNAME="hped710"
```

Don't try to directly assign the variable in this file. We will see the appropriate command to assign it.

The hostname must be a unique system name in your network environment. To verify uniqueness make sure the system name you want to use is not in `/etc/hosts` file. The system name, or host name, you choose must:

- Contain no more than 8 characters.
- Contain only letters, numbers, underscore (_), or dash (-).
- Start with a letter.
- NOTE: Uppercase letters are not recommended.

Internet Protocol (IP) Address

Internet Protocol is responsible for routing individual network packets of data. IP's job is simply to find a route for the packet and get it to the other end. IP defines an internet addressing scheme which can uniquely identify multiple networks as well as a node (system) within a single network. The "Dot notation" scheme is an industry standard. For example, 153.41.211.25. IP addresses must be unique within the network. If your network will connect with other networks, then the IP address must also be unique across all networks.

MAC Address or Physical Address

The MAC Address is the unique address of the LAN interface card.

This value is set at the factory and can not be changed. The link level address is used to establish the connection between the network and the computer. This is the lowest level of communications and it deals with the interconnection of processors. This number is usually given in hexadecimal, for example, 0x0800090012AB. Note the address will always start with "080009" for HP systems.

This number can be seen on the LAN interface card by removing it, but can be more easily obtained with `lanscan(1m)`:

```
# lanscan
Hardware Station        Crd Hardware Net-Interface  NM  MAC       HP DLPI Mjr
Path     Address        In# State    NameUnit State ID  Type      Support Num
2.0.2    0x0800097843FB 0    UP       lan0     UP    4  802.3      Yes     52
```

20-3. SLIDE: Steps for Installing a LAN

Steps for Installing a LAN

1. Connect the LAN cable to the LAN connector
2. Execute `/sbin/set_parms hostname`
3. Execute `/sbin/set_parms ip_address`
4. Edit `/etc/hosts` file and add a line for each system on LAN.
5. Reboot
6. Verify LAN Connectivity

H6296 20-3 215

Student Notes

A newly installed system will prompt for network configuration during the first boot. The shell script `/sbin/set_parms` gets executed from `/etc/rc0.d` if `/etc/rc.config` does not exist or if the default HOSTNAME is unset or an empty string. You may choose to setup your LAN at installation time or configure it later. To setup the LAN at a later time, execute `/sbin/set_parms` *argument* at your shell prompt, where *argument* can be any of:

- hostname
- timezone
- root_passwd
- ip_address
- addl_network
- font_c-s, or
- initial, for initial boot-time dialog sequence

- Connect the LAN cable to the LAN connector. Terminate both ends with terminators. (We assume that your LAN card is already installed in the system. If not, you must install the card.)
- The /sbin/set_parms program will prompt you for the information needed to configure the LAN. The /etc/rc.config.d/netconf file will be updating with a new value for each networking subsystems.

Below is an example of using the /etc/set_parms program.

```
# /etc/set_parms initial

##################################################################

                          Welcome to HP-UX!

Before using your system, you will need to answer a few
questions.

The first question is whether you plan to use this system on a
network.

Answer "yes" if you have connected the system to a network and
are ready to link with a network.

Answer "no" if you:

     * Plan to set up this system as a standalone (no
networking).

     * Want to use the system now as a standalone and connect to
a network later.

##################################################################

Are you ready to link this system to a network?

Press [y] for yes or [n] for no, then press [Return] y

##################################################################

   Before you begin using this system, you need to obtain the
   following information from your local network administrator:

      * Your system name (host name).

      * Your Internet Protocol (IP) address.

      * Your time zone.

    If you do not have this information, you may stop now and
restart your system once you have it.
```

```
########################################################################

Do you wish to continue?

Press [y] for yes or [n] for no, then press [Return] y

########################################################################

For the system to operate correctly, you must assign it a unique
system name or "hostname".  The hostname can be a simple name or
an Internet fully-qualified domain name.  A simple name, or each
dot (.) separated component of a domain name, must:

   * Contain no more than 64 characters.

   * Contain only letters, numbers, underscore (_), or dash (-).

   * Start with a letter.

NOTE:
   * Uppercase letters are not recommended.

   * The first component should contain 8 characters
     or less for compatibility with the 'uname' command.

The current host name is hpfrcu73.

########################################################################

Enter the system name, then press [Return] or simply press
[Return] to retain the host name (hpfrcu73):

You have chosen hpfrcu73 as the name for this system.
Is this correct?

Press [y] for yes or [n] for no, then press [Return] y

########################################################################

  Working...

########################################################################

########################################################################

The following procedure enables you to set the time zone.

Select your location from the following list:
```

```
  1) North America or Hawaii

  2) Central America

  3) South America

  4) Europe

  5) Africa

  6) Asia

  7) Australia, New Zealand

#################################################################

Enter the number for your location (1-7) then press [Return] 4

#################################################################

Select your time zone from the following list:

  1) Greenwich Mean Time
     British Summer Time

  2) Portuguese Winter Time
     Portuguese Summer Time

  3) Western European Time
     Western European Daylight Time

  4) Middle European Time
     Middle European Daylight Time

  5) Iceland time

  6) Turkey, Finland, Romania,
     Greece, Bulgaria

  7) Western Russia (Moscow)

  8) Unlisted time zone

  9) Previous menu

#################################################################

Enter the number for your time zone (1 - 9), then press [Return]
4
```

```
########################################################################

You have selected:

   Middle European Time/Daylight Time (MET-1METDST).

########################################################################

Is this correct?

Press [y] for yes or [n] for no, then press [Return] y

########################################################################

If you wish networking to operate correctly, you must assign
the system a unique Internet Protocol (IP) address.  The IP
address
must:

      * Contain 4 numeric components.

      * Have a period (.) separating each numeric component.

      * Contain numbers between 0 and 255.

      For example:  134.32.3.10

  Your current address is 153.11.41.173.  To retain this address,
  just press [Return].

########################################################################

Enter your IP address, then press [Return] or press [Return] to
select the current address (153.11.41.173):

You have chosen 153.11.41.173 as the IP address for this system.
Is this correct?

Press [y] for yes, [n] for no or [c] to cancel then press
[Return] y

########################################################################

  Working...

########################################################################

########################################################################

  You may configure some additional network parameters at this
```

```
time:

     * Subnetwork Mask and Default Gateway

     * Domain Name Service (DNS)

     * Network Information Services (NIS)

  Your local network administrator can tell you which if any of
these parameters should be configured for your system, and
provide
you the appropriate values.

  If you do not have these values now, you can configure them
later.

#################################################################


Do you want to configure these additional network parameters?

Press [y] for yes or [n] for no, then press [Return] n

#################################################################

Your changes will not take effect until you reboot the system.

#################################################################

Do you want to reboot now?

Press [y] for yes or [n] for no, then press [Return] n
```

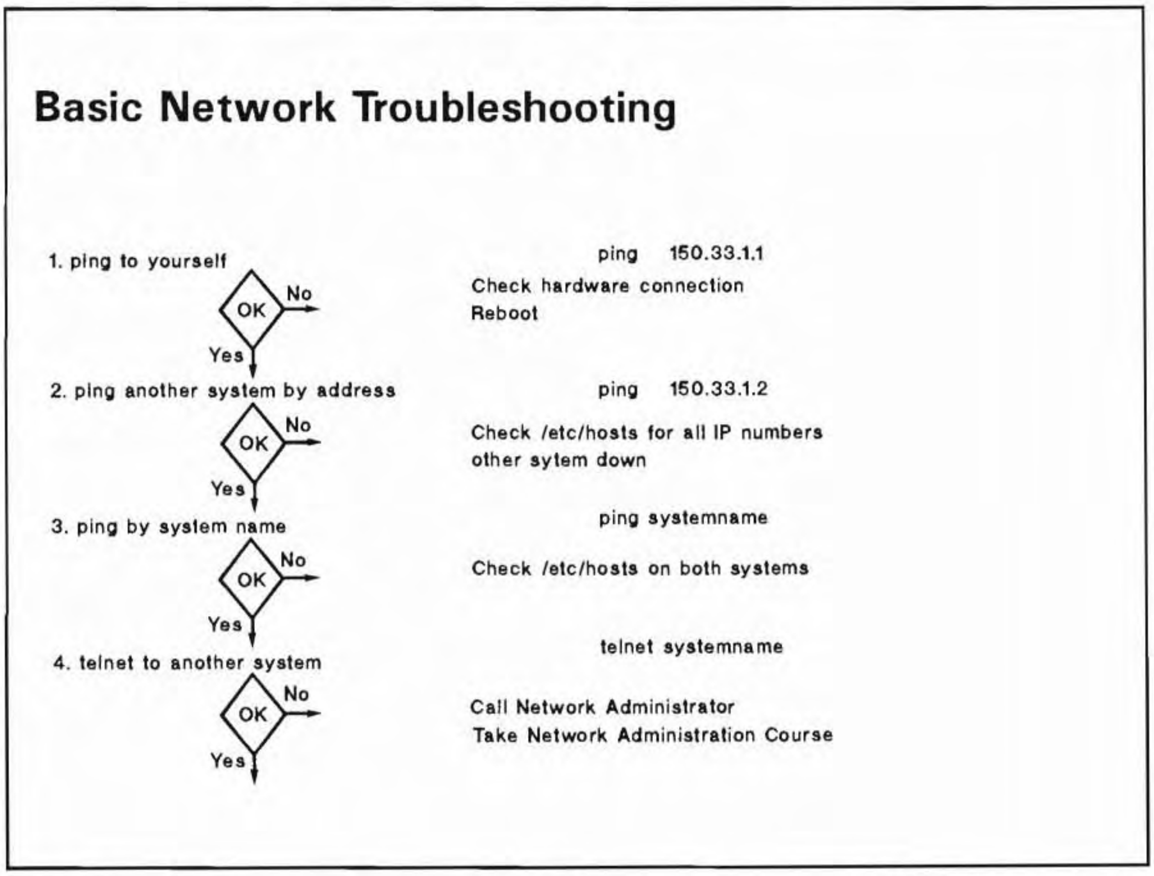
- Edit /etc/hosts file and add a line for each system on the LAN. The /etc/hosts file is the host name database. It associates internet addresses with official host names and aliases. It allows a user to refer to a host by a symbolic name instead of an internet address. An example hosts file is shown below. You would use /etc/set_parms to add your internet address and nodename to this file.

```
153.129.16.211    hped710    #HP-UX 9000/710
153.129.16.100    snake1     #HP-UX 9000/750
153.129.16.234    emerald1   #HP-UX 9000/890
```

- Reboot and start up the network. You will need to reboot if you assigned a new nodename. Use shutdown -r (or reboot if you are in single-user mode) to reboot your system. Rebooting causes the network to be brought up.
- Verify the LAN is operational

  After you have rebooted, it is wise to verify using the standard commands (next TOPIC) that the LAN is indeed installed and operational.

20-4. SLIDE: Basic Network Troubleshooting



H6296 20-4 216

Student Notes

When your network is not operational, you have to retrieve information about problems. There are several commands which can be used on HP 9000 systems to retrieve this information.

Commands Used to Test LAN Connection

Command	Information Gained ...
ping	Percentage of packets transmitted and received between systems; lists IP address of system being accessed.
telnet	Access a system using ARPA service; if successful, will receive login prompt
rlogin	Access a system using Berkeley service; if successful, will receive login prompt

Descriptions and examples of using each of these commands are presented below.

- Using ping

ping sends ICMP ECHO_REQUEST packets to network hosts. Each packet that is echoed back represents a successful transfer of information between the two systems.

```
# ping 153.11.41.234 -n 10

64 bytes from 153.11.41.234 icomp_seq=0. time=5. ms
64 bytes from 153.11.41.234 icomp_seq=2. time=6. ms
64 bytes from 153.11.41.234 icomp_seq=3. time=5. ms
                    .
                    .
                    .
64 bytes from 153.11.41.234 icomp_seq=28. time=4. ms
64 bytes from 153.11.41.234 icomp_seq=29. time=5. ms
64 bytes from 153.11.41.234 icomp_seq=30. time=5. ms

-------emerald1 PING statistics ------
10 packets transmitted. 10 packets received. 0% packet loss
round-trip (ms) min/avg/max = 4/5/6
```

- You may use ping with the hostname too :

For example, your /etc/hosts table contains the following entries:

```
153.11.41.211    hped710    #HP-UX 9000/710
153.11.41.100    snake1     #HP-UX 9000/750
153.11.41.234    emerald1   #HP-UX 9000/890

# ping snake1 -n 10

64 bytes from 153.11.41.100; icomp_seq=0. time=5. ms
```

```
64 bytes from 153.11.41.100; icomp_seq=2. time=6. ms
64 bytes from 153.11.41.100; icomp_seq=3. time=5. ms
                    .
                    .
                    .
64 bytes from 153.11.41.100 icomp_seq=28. time=4. ms
64 bytes from 153.11.41.100 icomp_seq=29. time=5. ms
64 bytes from 153.11.41.100 icomp_seq=30. time=5. ms

-------snake1 PING statistics ------
10 packets transmitted. 10 packets received. 0% packet loss
round-trip (ms) min/avg/max = 4/5/6
```

- Using `telnet` and `rlogin`

Of course, the final test for your LAN installation is the ability to access other systems. Depending on which type of networking service you are using, you should now try to test the link between the system configured and other systems listed in its `/etc/hosts` table.

If you are using ARPA Services, you could access the 890 (emerald1) system with the following commands:

```
# telnet 153.11.41.234

or

# telnet emerald1
```

A successful ARPA access results in receiving the `login` prompt.

If you are using Berkeley Services, you could access the 890 (emerald1) system with the following commands:

```
# rlogin 153.11.41.234

or

# rlogin emerald1
```

A successful Berkeley access results in receiving a `passwd` prompt from the desired system. The `rlogin` command assumes you are logging in on the remote system using the same user as on the local system.

Note

The user will not receive a `login` but a prompt if the `$HOME/.rhosts` file is configured on the remote system.

For example, `user1` has a login on `server_a` and the file `$HOME/.rhosts` on this `server_a` is

```
hpc1200
```

If `user1` types the `rlogin server_a` command from the `hpc1200` machine, he will not receive a `login`, but will immediately receive a shell prompt.

Module 21 — Installing HP-UX

Objectives

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

- Differentiate between an installation and an update.
- Perform the steps to install HP-UX 10.0 on an HP 9000 workstation or server.

Module 21 — Installing HP-UX

21-1. SLIDE: Why Install ?

Why Install ?

- New system, operating system not pre-installed
- Recovery from disaster
- New operating system with major changes

H6296 21-1 217

Student Notes

There are only a few instances when it will be necessary to *install* HP-UX.

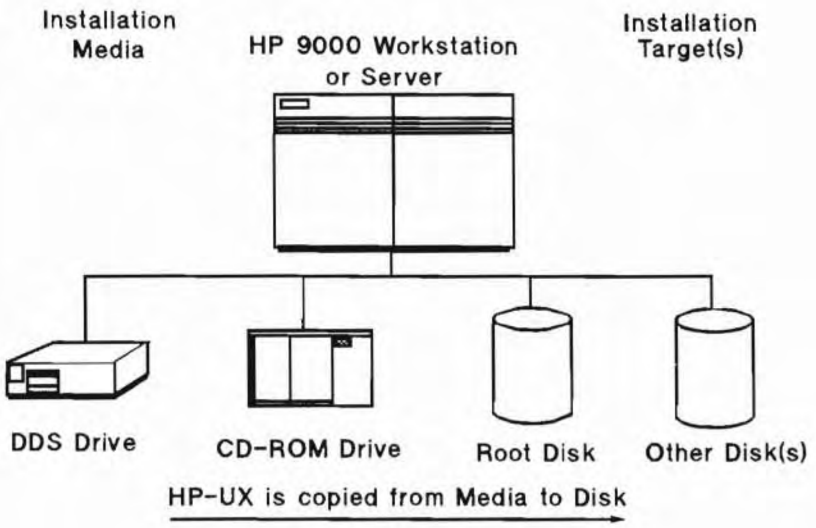
The installation process places HP-UX on a root disk, overwriting files that exist on that disk or disks. If you have no operating system, or an irrecoverably damaged one, then you must perform an installation.

Sometimes a new release of the operating system will have major new features that require an installation to be done in order to use them, rather than an update. An **update** simply modifies the operating system on the disk and does not completely overwrite files that already exist on the disk.

Each time a new release of the operating system is delivered, the installation and update procedures may change slightly. This chapter discusses the process of installing HP-UX Release 10.0. If you are installing a different release, you should consult the *Installing and Updating* manual for that specific release.

21-2. HP-UX Installation

HP-UX Installation



H6296 21-2 218

Student Notes

Installing the HP-UX operating system on your system disk will:

- Destroy any data that previously existed on the root disk.
- Construct an HP-UX file system on the root disk.
- Copy file sets from the installation media to the root disk.
- Build the initial HP-UX kernel on the root disk.
- Create a login for the system administrator.

You can install HP-UX from DDS format DAT, or CD-ROM. Installation media will vary depending on the peripheral devices available on your system.

Module 21 — Installing HP-UX

The installation process destroys any data that previously existed on the root disk or root logical volume. Next, an HP-UX file system is constructed on the root disk or logical volume.

Subsystem files from the installation media will then be copied to this newly constructed file system. An HP-UX kernel will be created. This kernel may need to be further customized to accommodate the applications that will run on this system. Lastly, a login is created for the system administrator.

21-3. SLIDE: Planning Your Installation

Planning Your Installation

- Read the *README BEFORE INSTALLING HP-UX* notice.
- Determine:
  - Hardware address of the installation source device (DDS tape or CD-ROM drive) and the root disk.
  - Will you install a Root Logical Volume?
  - Swap space needed.
  - Hardware IDs and codewords (CD-ROM only).
  - Which file sets and bundles to install.
  - Customization information (time zone, time, hostname, Internet address and (optional) some more networking information)

H6296 21-3 219

Student Notes

If you are installing HP-UX on a brand-new system, you should locate all system hardware and the corresponding addresses..

Read the *README BEFORE INSTALLING HP-UX* notice. If an installation manual is supplied with your installation media, read it as well. The installation manual *Installing and Updating HP-UX HP 9000* is an essential reference if you are installing for the first time.

Determine the hardware address of the installation device and the root disk. Determining these addresses is not as critical if your system has a standard configuration. However, it will allow you to verify that the installation is proceeding properly if you know the hardware addresses.

If your system is not using a standard configuration, you must know the hardware addresses of these devices in order to supply the appropriate information during the installation. Refer to the Appendix of *Installing and Updating HP-UX HP 9000* for the hardware addresses of the system disk and tape drive/CD-ROM with standard configurations.

Determine the amount of swap space needed.

Determine any required hardware IDs and codewords. If you are installing anything other than a 2-user Runtime product from a CD-ROM, you need a codeword. Note that codewords are only necessary for CD-ROM installations. The codeword serves as a "key" during the installation process. It unlocks the specific software products that you have purchased on the CD. The codeword is tied to your system's hardware IDs. Codewords can be obtained in one of the following ways:

- Your codeword may appear on the CD-ROM Certificate that was delivered with your product.
- If your certificate does not contain a codeword, you must obtain a codeword from Hewlett-Packard.

The procedure for obtaining a new codeword is to fill out the codeword certificate and FAX it to 800-541-2633 or 415-960-5670. A codeword will be returned to you via FAX.

If you do not have a certificate, you should call your local sales office and ask for the Sales Contracts Coordinator. The Contracts Coordinator can then contact the Codeword Delivery office and verify your sales order so that a codeword can be sent to you.

Determine which file sets and bundles you wish to install. If you are not planning to install everything from the software bundle that you ordered, determine which file sets and bundles you wish to install. A **bundle** is a group of file sets, and a **fileset** is a group of files.

Determine customizing information (time zone, time, hostname, IP address). If you plan to use networking, you will need to have a unique hostname for your system and a unique Internet Protocol (IP) address for your LAN interface. Many companies have an internal department that assigns IP addresses. You should also know the time zone of your locale. During the final stage of installation, you will be asked to supply this information.

After supplying this information you may need to set some more networking parameters.

- Subnetwork Mask and Default Gateway
- Domain Name Service (DNS)
- Network Information Services (NIS)
- X11 font client or server configuration

Your local network administrator can tell you which if any of these parameters should be configured for your system, and provide you the appropriate values.

If you do not know these values now, you can configure them later.

21-4. SLIDE: Steps to Install HP-UX Operating System

Steps to Install HP-UX Operating System

1. Load the media marked *INSTALL* into the installation device.
2. Identify the ESC (workstation) or the "ANY" (server) key on your keyboard.
3. System Startup
   a. Turn on power to disk drive, the installation device and system console, and then SPU.
   b. When you see the message (server):

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

   or (workstation)

```
Selecting a system to boot...
To stop selection process, press and hold ESC.
```

   PRESS A KEY / ESC ON THE KEYBOARD IMMEDIATELY!

H6296 21-4 220

Student Notes

Here is the detailed procedure for installing the HP-UX operating system on your disk:

Turn on the power to your disk drive, your tape or CD-ROM drive, and your system console.

If you are using an HP-UX *INSTALL* tape, make sure the tape is write protected. Load the *INSTALL* tape or CD-ROM in its drive. If you are loading an install tape, wait for the tape drive to be ready and check to make sure it is on-line.

Turn on the SPU to begin the boot-up process.

The first message asks you to press a key to override the autoboot process (server) or ESC (workstation). Autoboot will try to boot an operating system from the primary boot path, which is the hardware path to the system disk (as preset by the factory). You want to press a key to override autoboot so that you can boot from tape or CD-ROM.

After you interrupt the boot process, the system prompts you where to boot from. Specify the address of your installation device, for example (server)

```
Enter boot path, command, or ?> 52.1.0
Interact with IPL ?  n
```

or (workstation)

```
P0      scsi.6.0        QUANTUM LPS525S
P1      scsi.3.0        HP HP35470A
...
Select from menu : b p2
```

You do not need to interact with the IPL/ISL; the software on the installation medium will automatically boot and start the installation menus.

21-5. SLIDE: Install Main Menu

Install Main Menu

```
            Welcome to the HP-UX installation process!

   Use the (tab) and/or arrow keys to navigate through
   the following menus, and use the (return) key to
   select an item.  If the menu items are not
   clear, select the "Help" item for more information.

                    [      Install HP-UX        ]
                    [   Run a Recovery Shell    ]
                    [    Cancel and Reboot      ]
                    [    Advanced Options       ]

                           [   Help    ]
```

H6296 21-5 221

Student Notes

`Install HP-UX`	Choose this to continue normal install.
`Run a Recovery Shell`	Allows you to use the install media later (after a complete installation) to manually repair the root disk (if possible) in case of an unbootable system.
`Cancel and Reboot`	Cancel the installation, and reboot the system.
`Advanced Options`	Allows you to Edit the Installation Configuration file, to set / modify environment variables and to do a network source install instead of installing from tape or CD-ROM.

The configuration should be edited only by experienced system administrators with good knowledge of the process of installation and software configuration, since an incorrect config file may make another new installation necessary.

A **network source install** may make it easier to newly install more than a few systems. Defining a network source install at this point does *NOT* mean that all system software is copied over the network. Instead, this is meant to get the software for the first (preparatory) steps from a network install server. To complete the installation process, after preparing the disk you can choose a network source for copying the software from later during the installation process.

21-6. TEXT PAGE: Setting up Installation Parameters

Before the system installation can be continued, the system administrator has to give some information about the system and the desired configuration.

Setting up network configuration

The first menu sets the system's name and network parameters. This is especially necessary when installing from the network.

```
            HP-UX INSTALLATION UTILITY  --  NETWORK CONFIGURATION


                                       This system's hostname:  train1

   Internet protocol address (eg. 15.2.56.1) of this host:  192.6.2.33

                Default routing internet protocol address:  192.6.2.33

        The subnet mask (eg. 255.255.248.0 or 0xffff800):

   Internet protocol address of the install server system:

    [   OK   ]              [ Cancel ]                  [  Help  ]
```

Fill in your information. All of it can be changed after the installation is done, but you should use your correct values if known. When installing from a network source, this information must conform with your network's topology.

Hostname	your system's future name. If you don't know this, use any name shorter or equal eight characters. The system name can easily be changed later.
Internet address	The unique network address of your system. When your system is hardware connected to a local network, make sure this address isn't already used in your network. Choosing an address already in use may cause a failure of the installation as well as errors on all other systems in your network (for example when the duplicated address belongs to a network server system).
Default router	This address is normally set to your system's Internet address. When you plan to continue the installation using a network source outside your local network, you must specify the address of a system that has access to both your local network and the network the server system resides in.
Subnet mask	By default, this field remains empty. When subnetting is used in your network, this information is required. Ask your network administrator for the value to fill in.
Install server's address	This line appears only if you have chosen to use a network source instead of tape or CD-ROM. Fill in the Internet address of the server system. The named system must be set up as a server for HP-UX installation.

Select New Root Disk

In the next step, after filling in the network config information, the user is asked to select the new root disk. The system is scanned for all disks and the result is displayed in a list.

Select the disk by moving the cursor and use (TAB) to move to the (OK) button.

Use-model selection

The next menu lists the available use models as defined in the install configuration file. At the same time you can specify the use of the root disk and the swap size.

```
                    HP-UX Install Utility - Use Model Selection

    Here are the available use models as defined in the install
  configuration
    file.  For more information on a particular use model, position
  the cursor
    on the line containing that use model and press the '?' key.

    Available Use Models: (Scroll to see entire list.)

  /---------------------------------------------------------------------------\
    |     ============  Software Selections (select one) ===========
    | Load Minimal File Sets
    | Load All File Sets
    | Load Server File Sets
    | Load "Desktop HP-UX" software bundle
    |     ========  Disk Configuration Selections =====
    | Root on LVM
    |     Use Remaining Disk Space For /home
    | Root on Whole-Disk (non_LVM)
    | 68Mb Swap
    | 100Mb Swap
    | 36Mb Swap
  \---------------------------------------------------------------------------/
    [  OK  ]                         [ Cancel ]
  ----------------------------------------------------------------------------
```

Load Minimal Filesets	Selecting this loads only the absolute minimum of file sets to the disk. Use this item when installing on a small disk or if you are not sure what to install. The software loaded to disk allows you to update the system later.
Load All Filesets	This loads everything on the medium to the disk!
Load Server Filesets	This selection loads the minimal file sets plus those required for a diskless server. The system will be able to support booting systems not using their own disks over the network.

Load Software Bundle	Installs a predefined set of file sets to the disk; thus both conserving disk space by not loading software you probably won't need, and loading everything necessary to perform the most common tasks. When selecting this menu item, the system has a well balanced subset of the file sets on the medium, meeting most customers' needs.
Root on LVM	LVM stands for Logical Volume Manager. This is the standard use of the root disk. Unless your needs require the use of the whole disk, thus not using the Logical Volume Manager, you should select this item. Otherwise you give up the advantages of LVM, for example the extendability of the logical volumes.
Remaining Disk Space	The installation procedure divides your system disks in several parts of different size. These sizes are predefined and usually do not use the whole capacity of the disk. You can decide to use the remaining disk space to store the user files (`/home` is by default the directory where the users home directories are created in).
Root on Whole-Disk	When NOT using LVM, the disk is divided in just two parts. One is contains the file system, the other is used as swap space. The file system is not modifiable. In case the root file system size has to be modified later, the whole installation must be redone.
Swap	The software installation process has a few choices of swap space sizes; you should accept the default value. In a small disk configuration, you may choose a smaller swap size. When using LVM, the swap space can be extended later. When you have chosen to put root on the whole disk, the swap size is fixed.

Check System Configuration

After selecting the (OK) button in the previous menu, you get a summary about the next steps done by the installation process. This is the last chance to modify things. After leaving this menu with (OK), all data on the selected root disk will be destroyed.

```
                HP-UX Install Utility - System Configuration

  Data on the following disks will be destroyed:
    Hardware                 Product          Size      Disk      Volume
    Path                     ID               (Mb)      Usage     Group

/-------------------------------------------------------------------------\
  | 2/0/1.5.0                QUANTUM_LPS525S499         LVM       vg00
  |
  |
  |
  |
\-------------------------------------------------------------------------/

  The following file systems (swap) will be created:
    Mount                     Size      FS
    Directory                 (Mb)      Type     Origin

/-------------------------------------------------------------------------\
  | /                         48        HFS      vg00                     |
  | (swap)                    68        swap     vg00                     |
  | /home                     0         HFS      vg00                     |
  | /opt                      28        HFS      vg00                     |
\-------------------------------------------------------------------------/
  [ Modify Configuration... ]
 --------------------------------------------------------------------------
  [   OK   ]                       [ Cancel ]
```

Selecting `Modify Configuration...` returns to the previous menus, where you may modify the current configuration. Also, this menu point allows you to modify the sizes of the different file systems and swap size. When this is done, you return to this menu to confirm the selected configuration.

Interact with swinstall?

```
                HP-UX Install Utility - Confirmation Dialog

/--------------------------------------------------------------------------\
 | The HP-UX Software Distributor utility, swinstall, will actually load
 | the HP-UX software on your system.  The system configuration that you
 | have already specified should be sufficient and you should not need to
 | modify the software selections.  However, if you do have a need to view
 | or modify the software selections, you may choose to interact with the
 | swinstall utility at this time.
 |
 | Do you wish to interact with the swinstall utility to view or modify
 | the software selections?
 |
 | Select "Yes" if you want to view/modify the software selections that
 | are to be loaded onto your system.
 |
 | Select "No" if you want the software loading to proceed without
 | interaction.
 |
 |
\--------------------------------------------------------------------------/
              [  Yes  ]                        [  No   ]
```

21-7. TEXT PAGE: Starting the Installation

Starting Installation

After leaving the previous menu, the preparation of the system's root disk is started. During installation, you will see a lot of messages, like

```
       * Starting system configuration...
       * Will ignore empty volume: /home
       * Creating LVM physical volume: /dev/rdsk/c0t5d0
(2/0/1.5.0)
       * Creating volume group: vg00
       * Creating logical volume: vg00/lvol1 (/)
...
       * Creating logical volume: vg00/lvol6 (/var)
Volume Group configuration for /dev/vg00 has been saved in
/etc/lvmconf/vg00.conf
       * Making HFS file system for: /, (/dev/vg00/rlvol1)
...
       * Making HFS file system for: /var, (/dev/vg00/rlvol6)
       * Loading mini-system to hard disk...
x ./sbin/insf, 212992 bytes, 416 tape blocks
...
x ./stand/system.700, 453 bytes, 1 tape blocks
       * Installing boot area on disk.
Volume Group configuration for /dev/vg00 has been saved in
/etc/lvmconf/vg00.conf
insf: Installing special files for stape instance 0 address
2/0/1.2.0
...
insf: Installing special files for pseudo driver root
```

The creation of the file system and unloading of software takes about five to ten minutes. The function of this unload process is to create a mini system on the hard disk. This mini system is bootable and can be used to restart the installation in case of a failure of `swinstall`.

Specify the Software Source

```
======================================================================
                     USER INTERACTION REQUIRED:
To complete the installation you must now remove the HP-UX installation
tape and insert the HP-UX Core Operating System tape.

Once this is done, press the [Return] key to continue.
Or, if you would like to use a network source to load the
remainder of the operating system, type the word "network" followed by
```

(Return):

After completion of unloading the mini system, the installation process is stopped for user interaction. When doing a normal tape/CD-ROM install, the user interaction consists of

- unload the Install medium
- load the medium containing the core operating system
- press (Return) to complete the installation.

If you want to continue the installation using a network source, enter network followed by (Return). You will be asked for the network source system's Internet Address (IP Address) and for the directory where the so called depot resides. The default directory where the depot is located is `/var/adm/sw`. You must enter the appropriate location when asked for the depot's name.

Loading the Operating System

Finally, the installation starts loading the software. This is done in four steps:

- Analyze the selected installation configuration
- Install the software to the target
- Analyze the installed software
- Configure the installed software

Analyzing the selected installation configuration checks the existence of all files and directories, adds up all disk space required and checks the sums against free disk space.

After checking the configuration of all software, the system is rebooted.

21-8. TEXT PAGE: After Software Loading

Set Networking Information

Rebooting the system is necessary to boot the new kernel and to use the installed software instead of the software loaded to install the system.

After rebooting the system, it does not know about its network connectivity. This information must be supplied, if the system is connected to a local area network (LAN).

If you won't connect the system to a LAN, some of the steps can be skipped by answering no when asked whether you want to connect the system to a LAN.

On a workstation, the graphical interface X is started to ask for the networking information; server systems query from the console.

You will be asked

- for the systems *hostname* and *Internet address* (you already supplied both when preparing the installation; these values are used as the default)
- for the *time zone*
- to set the *root password*
- for optional parameters about your network:
  - whether you use a *subnet mask*
  - to enter the IP address of a *Domain Name Server*
  - to set a *Network Information Service (NIS) domainname.*
- and last, whether to make the system a font client or font server for HP-VUE. Making the system to a font client requires another system in your network which is set to a font server.

All these parameters are set using a script file, `/sbin/set_parms`. You can reset the networking parameters later by calling `/sbin/set_parms` again. `/sbin/set_parms` accepts several options defining what to set. If you want to set all of the above parameters, you can use the option `initial`.

Setting the networking parameters later requires a reboot of the system, because changes will not take effect until you reboot the system.

Complete boot process

After you've set the networking parameters, you must simply wait for the system to complete the boot process. Your system will become very busy for a few minutes.

A lot of boot up messages will be displayed. Many of them will indicate some form of error. They may be safely ignored at this time.

Finally at the console login, you may become the superuser by typing "root" (Return), and proceed to the post installation procedures in the module on update.

Verify Installation

The complete process of loading the operating system to disk is logged. You should check these logfiles for possible errors, and follow any instructions that are given for additional tasks that should be done.

The log files for the installation are found in the directory `/var/adm/sw`. Change to that directory and view the log file:

```
more swagent.log swinstall.log
```

The following items are message labels and their meanings. The actual messages are indented nine spaces.

`=======` Indicates that a task within `swagent` and `swinstall` is beginning or has completed. For example:

```
=======...Agent session started for user ...
```

`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.

`WARNING:` Usually indicates the program can continue. However, something went wrong or requires attention, either now or later. One example that can often be ignored is:

`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 you must take after the installation.

Verify Time is Correct

Several startup customization files are "sourced" to customize the system configuration at boot time (set the time zone and host name). These files are edited for you at the end of the installation procedure, when you provide the information about time zone, time, and whether the system is to be part of a network.

The file /etc/TIMEZONE is created in the installation process to set the TZ variable for you at startup time. It is "sourced" by the /sbin/rc script at boot time and the system-wide profiles (/etc/profile and /etc/csh.login) as users login.

The contents of /etc/TIMEZONE should be something like:

```
TZ=CST6CDT
export TZ
```

For more information on time zones see the file /usr/lib/tztab.

21-9. LAB: Hands-On, Installing HP-UX

Directions

Perform the following tasks. Write the commands you use, and the answers to any questions that are asked.

1. If there is time enough, your instructor will provide installation materials and you can perform an actual installation

Module 22 — Installing Additional Software

Objectives

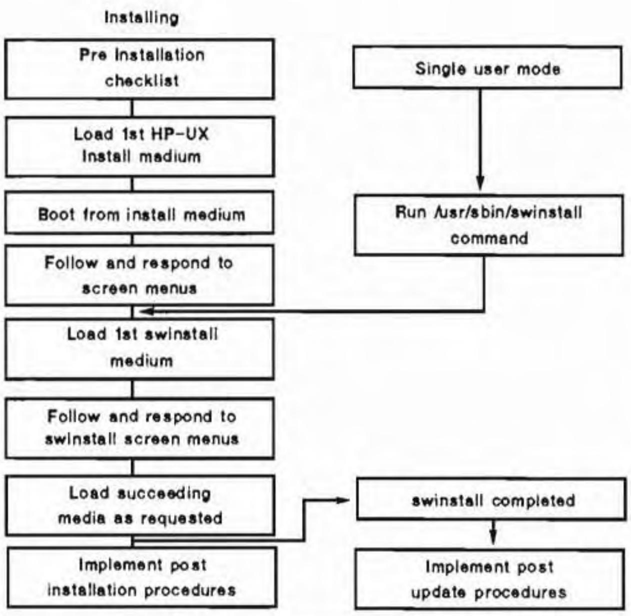
Upon completion of this module, you will be able to do the following:

- Describe the steps you should take before installing additional software.
- Perform the steps necessary to install software on your HP-UX system using the `swinstall(8)` command for an install using local media.
- List the files and directories to look through after a software installation.
- Perform the steps necessary to install software on your HP-UX system using the `swinstall(8)` command for an install from a Network Installation Server.
- Perform the steps needed to configure a Network Installation Server.

Module 22 — Installing Additional Software

22-1. SLIDE: Installing Versus Updating HP-UX

Installing Versus Updating HP-UX



H6296 22-1 222 © 1995 Hewlett-Packard Co

Student Notes

Use the update process to add new software. When you update HP-UX, you load software products from the update media (or from a special network server) and incorporate that software (in the form of products and filesets) into an existing HP-UX file system. If any new or replacement files affect the existing kernel configuration, the kernel is reconstructed during the update. If the kernel is reconstructed, the update process will reboot your system.

The update process is the same as the "swinstall" phase of the "install" process. The relationship between the two processes is shown on the slide.

You must have a functioning system in order to perform an update. In contrast to an installation, which overwrites files on the root disk, an update provides new functionality without destroying the existing system.

Module 22 — Installing Additional Software

As a quick review of system installation: The installation utility runs the `/usr/sbin/swinstall` program to load the rest of the products and filesets to the system disk.

A **fileset** is a logical grouping of files and a **product** is a logical grouping of filesets. A **product** (for example, the X Window System) is a logical grouping of filesets. When you load the filesets onto the disk from the update media, the files are copied to the disk in a hierarchical structure that is unrelated to the original fileset groupings on the update media.

Update Media

You can update HP-UX from DDS format DAT or CDROM. Update media will vary depending on the peripheral devices available on your system.

You may also update your system over a Local Area Network (LAN) provided that there is a computer system that has been configured as a network depot for software updates. Network installation and update are described in the *HP-UX Network Administration* Course.

22-2. SLIDE: Preparing for the Update

Preparing for the Update

- Read any *Read Me First* documents that come with the update media.
- Make a note of the device file name for your installation device.
- Ensure that your TERM environment variable is set correctly.
- Obtain codewords and hardware IDs if using codeword protected media.
- Ensure there is adequate disk space to perform the update.
- Clean up the file system using fsck.
- Back up the system.
- Check Free Disk Space
- Save a copy of the kernel and system configuration files.

H6296 22-2 223

Student Notes

Read Me First

Prior to updating, you should read any *Read Me First* documents that came with the update media. Such documents usually provide information about special conditions which may affect your update process.

Device File Names

You must know the device file name for your installation device (CD-ROM or DDS/DAT drive) so that you know whether or not you need to override the default drive that `swinstall` uses. `swinstall` uses `/dev/swtape` as its default. If this is not the device file for the installation device you intend to use, you may change the source device name in the command line with

```
# swinstall -s /dev/rmt/0m
```

Set TERM environment variable correctly

If your TERM variable is not set properly, your display could behave strangely during an interactive update process.

Codewords for CD-ROM

Codewords are used with the `swinstall` program to "unlock" software products stored on the CD-ROM disk in order to install or update these products on a specific computer system. Codewords are shipped to you from HP and are computed from information you supply with your order. This information includes the software product number that you have purchased and a unique hardware ID from the system you intend to install the software product on.

Adequate Disk Space

Before you update your system, analyze the amount of space available on your destination disk. Make sure there is enough space to accommodate the update. Consider these factors:

- The files on your system.
- The size of your disk(s).
- The products and file sets you will select to load.
- The minimum free space required on your system's mounted file systems.
- Swap space requirements on your system.
- Future expectations for space requirements on your system.
- Application being added.

The `swinstall` program will tell you if you have insufficient disk space to update your system. However, if you suspect you might not have enough disk space to update your system, it is best to handle the problem before you begin. You can free disk space by removing files from your system or by creating symbolic links. You could also choose to mount another file system.

Full System Backup

It is essential to perform a full system backup before you update your system. This will allow you to restore any or all of your files if something goes wrong with the update.

Clean up the File System

You should start with a file system that is in good shape. You should have been periodically running `fsck` to check disks. It would be a good idea to do this before you start an `swinstall` to be highly confident of the file system integrity.

Kernel and System Configuration Files

It is a good idea to save a copy of the kernel and system configuration files. If `swinstall` recreates the kernel, this step is not necessary, since it will be done automatically for you. If however, the `swinstall` fails for some reason, it's a good idea to be able to reboot your system from a "known good" kernel.

In most cases this will not be a "bad" thing to do. There are some problems with simply assuming you can reboot from a copy of the kernel `/stand/vmunix`. For example, when updating to another major release of the HP-UX the backup kernel may be an earlier revision and therefore incompatible with the rest of the HP-UX files and commands.

Ensure that any files that affect system configuration are properly backed up and correctly modified according to the installation instructions in your *Read Me First* documents.

22-3. SLIDE: Codewords and Hardware IDs

Codewords and Hardware IDs

CODEWORDS:

- Enable access to licensed software
- Computed based on hardware ID, software products purchased, and CD-ROM disk part number
- Supplied by HP in one of two forms:
  - Codeword Certificates for pre-computed codewords
  - Codeword Entitlement Certificates for post-computed codewords

HARDWARE IDs:

- Must be a unique identifying number related to your system hardware
- Are electronically readable by software

H6296 22-3 224

Student Notes

All HP software distributed on CD-ROM comes loaded on the CDs shipped to you as part of your HP-UX Software on CD Media subscription service. This service will provide two CD disks, one containing the operating system and subsystems and the other containing HP applications. Most of these products can only be accessed using a special codeword that unlocks the products. Codeword Certificates come in two forms: pre-computed and post-computed.

- The pre-computed certificate will have the hardware ID, the HP Sales Order number, and the part number(s) of the disks that work with this codeword.
- Post-computed codewords are obtained by faxing the hardware ID along with the other information requested on the fax form provided with the certificates. NOTE: this form is customized with a certificate ID number. You must use the fax form provided by HP for your software license. If this form is missing or has been misplaced, it is necessary to contact your local HP Sales Office. Ask for software contracts and have your HP sales order number(s) handy.

Whether or not codewords are pre-computed depends on how the order was placed. One factor is the hardware ID. If the order is placed specifying the hardware ID as some hardware other than the HIL ID module, the codewords cannot be pre-computed.

Hardware IDs

Hardware ID's can be:

- CPU hardware ID
- HP-IL ID module (if connected)
- HPIB CD-ROM drive (C1707A) serial number (if connected)

To obtain a hardware ID currently usable on a particular system, type:

```
# uname -a
    HP-UX train2 A.10.00 C 9000/715 2004922247 32-user license
```

This CPU's Hardware ID is 2004922247, and this number can be used to obtain a codeword.

Obtaining a Codeword

To resolve problems with codewords, please contact (800) 538-1733 to obtain assistance on codeword difficulties. Be sure to have your codeword certificate pages, especially the pages with the certificate ID number and the page with the peel-off stickers which contains the Sales Order number. All codeword certificates are printed with blue borders and titles with the words

```
* IMPORTANT: Retain for Reference *
```

on all the pages in red letters. Post-computed certificates will have a fax form attached which is used to obtain the codeword.

22-4. SLIDE: Steps to Update HP-UX with Local Media

Steps to Update HP-UX with Local Media

1. Execute `shutdown` to bring the system to a single-user state
2. If not already done, mount all file systems and perform a full backup
3. When using DDS tape drive, write protect your media
4. Insert media, and wait for drive to be ready
5. When using CD-ROM, mount your medium
6. Execute `/usr/sbin/swinstall` and follow the interactive menus

H6296 22-4 225

Student Notes

This slide describes the steps to update your system software if you plan to use physical media such as tape or CD-ROM. If you plan to use a network server to update your system software across a network, then some steps will be different. The network update is covered in another topic.

`shutdown` to single-user state

Depending on the file sets that are being loaded, the system may or not be rebooted during the update process. File sets that require kernel reconfiguration will reboot the system. If you unsure if you are updating this type of file set, it is best to be prudent and bring your system to a single-user run-level with the `/usr/sbin/shutdown(1M)` command:

```
# cd /
# shutdown
```

DDS/DAT: Write protect your media

If you are not using CD-ROM to update, the media you are using is vulnerable if not write protected. For the DDS/DAT tape media, push the write-protect tab towards the corner of the tape.

Insert media and wait for drive to be ready

The power should already be turned on for the update device. Insert the first medium into the device. Now wait for the drive to become ready. Some devices will indicate readiness by turning on a light and others will turn off a light. Consult the hardware installation manual for your particular drive.

CD-ROM: Mount your medium

When using CD-ROM to update your system, the medium must be mounted. The steps to do this:

- Create an empty directory
- Mount the CD-ROM to the new directory

Here is an example :

```
# mkdir /depot
# mount /dev/dsk/c0t3d0 /depot
```

The example assumes that the CD-ROM drive is already known on the system. See the module on Device Files for information on how to add a CD-ROM drive to your system.

Use `/usr/sbin/swinstall` to Load the File sets

Make sure you are in the root (/) directory before invoking `swinstall`.

You should type `/usr/sbin/swinstall` on the HP-UX command line to invoke interactive update process. The menus will lead you through the process of updating the system.

A command line user interface is provided for all features. For the most important user commands (`swinstall` and `swremove`) both a graphical and a terminal user interface (GUI and TUI) exists.

On `swinstall` and HP OpenView Software Distributor

The HP OpenView Software Distributor (SD) provides the capability of installing, updating and removing software products simultaneously on multiple local and remote systems. The complete software distribution process can be managed from a single point of administration.

Note Not all features of Open View SD are part of the default operating system. The push option for software distribution is only available with OpenView Software Distributor and must be purchased separately.

See the *HP OpenView Software Distributor Administrator's Guide* for more information.

If `swinstall` Cannot Be Started

If `swinstall` cannot be started, the reason is usually that a daemon process named `swagentd` is not started. You will see a message `Contacting the target` ... , when `swinstall` tries to connect to the `swagentd` daemon process on your system.

In case of a missing `swagentd`, `swinstall` tells you that the target can't be connected and asks the user to check whether the process is started. Usually `swagentd` is started during system startup to run level 2 or higher and is stopped during shutdown.

If your system is in single user state, `swagentd` is not running and must be started manually to successfully run `swinstall`.

To start the `swagentd` daemon manually, type

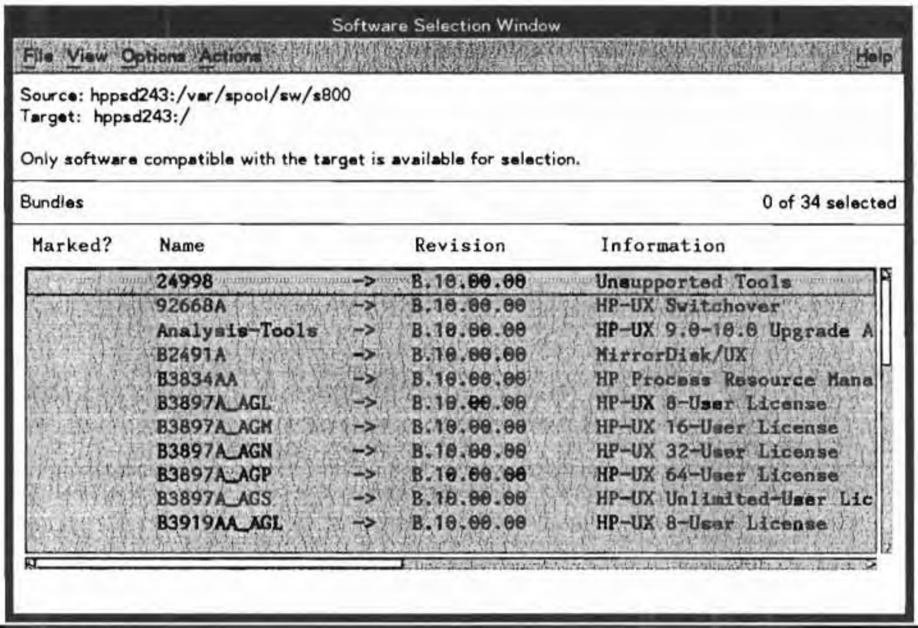
```
# /sbin/init.d/swagentd start
```

and rerun `swinstall`.

`/sbin/init.d/swagentd` is the run script for `swagentd`. You can stop `swagentd` by calling the same script with the `stop` option.

22-5. SLIDE: `swinstall` Main Menu

`swinstall` Main Menu



H6296 22-5 226

Student Notes

Starting `/usr/sbin/swinstall` on workstation systems allows `swinstall` to use the graphics display, as shown on the slide. You can also run `swinstall` inside the current window with normal ASCII terminal behavior by unsetting the "DISPLAY" variable. This is how it works on HP 9000 server systems.

Once you invoke `/usr/sbin/swinstall`, this window appears on your screen. It shows messages such as `Contacting the target ... Reading the software source ...` , an so on, as `swinstall` tries to read information about existing software from the target and source. As shown on the slide, within `swinstall` you may specify which source medium to use or you can add this information in the command line.

```
# /usr/sbin/swinstall -s /dev/swtape
```

`swinstall` reads information about available software from the specified medium and displays this after specifying the software source.

22-6. SLIDE: Select Software to Update

Select Software to Update

```
===                     Software Selection Window (1)
File View Options Actions                                              Help
                 | Open Item                    |
Source: train1:/de| Close Level                  |
Target:  train1:/ | ---------------------------- |
                 | Match What Target Has        |
Only software comp| ---------------------------- |for selection.
------------------| Change Source...             |--------------------
Products          | ---------------------------- |      1 of 54 selected
------------------| Install (analysis)...        |--------------------
  Marked?   Name  | ============================ |ormation
/-----------------| Show Description Of Software...|------------------\
|           Accou| Mark For Install             |ounting
|           AdvJo| Unmark For Install           | Advanced VxFS File S
|           Audio| Change Product Location...   |UX Audio Subsystem   |
|           Curse\------------------------------/ses library with colo |
|           DCE-Core          ->   B.10.00.%%U.2  HP DCE/9000 Core Client   |
|           DCE-CoreTools     ->   B.10.00.%%U.2  HP DCE/9000 Development   |
|           DFS-Core          ->   B.10.00.%%U.2  HP DCE/9000 DFS Client S  |
|           DesktopConfig     ->   B.10.00.%%U.2  System configuration for  |
|           DiskQuota         ->   B.10.00.%%U.2  DiskQuota
\<                                                                      >/
```

Help On Context	Alt	Select/ Deselect	Menubar on/off	hpterm			Shell	Exit

H6296 22-6 227 © 1995 Hewlett-Packard Co.

Student Notes

Here is an overview of how to perform the installation/update of software:

```
# swinstall
    |
    V
  Source Depot Path ...    /dev/swtape
    |
  press (OK)
    V
  Software Selection Window
    |
```

Select product(s)/fileset(s) with space bar or CR

```
|
+--> Action
|       |
|       +--> Open Item                    to display file sets of the current product
|       |
|       +--> Show Description Of Software       more information on current item
|       |
|       +--> Mark For Install             Mark selected items for Install ...
|
+--> Action
        |
        +--> Close Level   Leave level opened with Selected-->Open Item
        |
        +--> Match what target has   Select all file sets found on both, target
        |                            and source for Update
        |
        +--> Install (analysis) ...        Start Installation
```

You have already selected a Source Depot Path. Now the Software Selection Window is displayed.

The screen displays a list of available products or bundles. You may select

- bundles, which consist of one or more products and/or file sets. Bundles are sets of products useful to install together instead of one-by-one, for example an HP-UX desktop bundle for workstations.
- products, which form a logical set of file sets to add the specified functionality to the system, for example PrinterMgmt for everything needed to manage printers.
- or file sets, which are the smallest units to select, for example the manuals to the PrinterMgmt product.

To select an item, move the cursor to it and press SPACE or RETURN. You may select one or more items and then mark them for Install by pressing F4 to move to the Menu Bar and choose `Action` –> `Mark for Install`.

To show all subsets belonging to a bundle or product, `Open` it. To `Open` a bundle or product, select the item, then press F4 to move to the Menu Bar and mark `Action` –> `Open Item`. Opening an item is possible when only one item is selected, otherwise the `Action` menu does not show `Open Item`.

To return from the opened bundle or product view to the previous level press F4 –> `Action` –> `Close Level`.

To get more information on an item (if there is any), select the item and press F4 to move to the Menu Bar. Choose `Action` –> `Show Description Of Software`. This may take a while because the description is loaded from the source medium.

Showing the description allows you to view in detail what is installed on the target system, down to single filename level.

To update all parts of your current operating system with the new software found on the update media select `Action` —> `Match what target has` .

Note

By default, `swinstall` does *not* reinstall file sets if the same revision already exists on your system. If you want to reinstall the same revision (for example when some of the files are lost), you may change the installation options by `Menu` —> `Options` —> `Change Options` .

Installing a product or a file set may automatically install additional file sets necessary to run the selected item(s). `swinstall` reads the dependencies from the update media and auto-selects dependent file sets.

22-7. SLIDE: Start the Update

Start the Update

```
===                 Software Selection Window (1)
File View Options Actions                                             Help
                      Press CTRL-K for keyboard help.
Source: train1:/dev/rmt/0m
Target:  train1:/
===                      Install Analysis (2)

 After Analysis has completed, press 'OK' to begin the actual installation,
     or 'CANCEL' to return to prior selection screen(s).


 Target                :  train1:/
 Status                :  Ready
 Products Scheduled :  3 of 3

 [ Products... ] [ Logfile... ] [ Disk Space... ] [ Re-analyze ]
 -------------------------------------------------------------------------
 [   OK   ]                 [ Cancel ]                       [  Help  ]
            MSDOS-Utils       ->   B.10.00.%%U.2   MSDOS-Utils
            MailUtilities     ->   B.10.00.%%U.2   User mail agents and rel

Help On | Alt | Select/  |        | hpterm |  OK  |       | Shell | Cancel
Context |     | Deselect |
```

H6296 22-7 228

Student Notes

When you have finished selecting products/filesets, start the installation with `Menu` —> `Action` —> `Install (analysis)` .

The installation is divided into four phases: lab

Install Analysis	Checks dependencies, verifies everything can be installed correctly and defines the sequence of installation so that for example only one kernel rebuild should be necessary even if there are more file sets which require a new kernel.
Execution Phase	Performs pre_install tasks if necessary and installing the file sets.
Post_install	For example, kernel rebuild and system reboot

Configuration phase — To configure the installed file sets for your system. The configuration phase is started after the post_install phase, thus after a system reboot.

All information about the update is logged in `/var/adm/sw/swinstall.log`. You can open the logfile during the update by selecting (Logfile). The logfile is automatically scrolled while the update process continues.

During the process a window displays the progress of the update. After `swinstall` is finished, the system is rebooted if the installed file sets required a reboot or kernel regeneration. If a reboot is not necessary, you must leave `swinstall` manually with **Menu —> File —> Exit** .

22-8. SLIDE: Check Up on swinstall

H6296 22-8 229

Student Notes

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 for additional tasks that should be done.

The /var/adm/sw/swinstall.log

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. The actual messages are indented nine spaces. 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.

WARNING Usually indicates the program can continue. However, something went wrong or requires attention, either 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 you must take after the update.

Remove Files Not Removed During `swinstall`

The swinstall process may be unable to remove some files from the file system. This occurs when there is an attempt to remove a file from the disk when a process is running that corresponds to the file. The `swinstall` process will usually rename that file with a name formed by placing an "#" in front of the old filename.

For example, it may happen that the file to be removed was `/usr/sbin/swinstall`, and it was also executing. The program would rename it to `/usr/sbin/#swinstall`, and unload a new `/usr/sbin/swinstall` from the update media. The list of such files is written to `/var/adm/sw/cleanupfile`.

When the system is rebooted, the files named in this file are usually automatically removed during startup. If the removal during boot succeeds, the file `/var/adm/sw/cleanupfile` is deleted. If this file is still there, try to find out which files could not be removed and remove them manually with the command:

```
# rm -rf 'cat /var/adm/sw/cleanupfile'
```

You should remove all such files after the installation or update completes.

22-9. SLIDE: Removing Software

Removing Software

```
===                    Software Selection Window (1)
File View Options Actions                                       Help
                  | Open Item                    |
Target:  train1:/ | Close Level                  |
Target Displaying | ---------------------------- |
                  | Remove (analysis) ...        |
------------------| ============================ |----------------------
Products          | Show Description Of Software... |      1 of 45 selected
------------------| Mark For Remove              |----------------------
  Marked?    Name | Unmark For Remove            |ormation
/-----------------\------------------------------/----------------------\
|          ImagingSubsystem ->  B.10.00.%%U.2  HP-UX Image Viewer Subsy |
|          InstantIgnite    ->  B.10.00.%%U.2  Instant Ignition         |
|          IntegratedLogin  ->  B.10.00.%%U.2  HP DCE/9000 Integrated L |
|          International    ->  B.10.00.%%U.2  International            |
|          InternetSrvcs    ->  B.10.00.%%U.2  General network applicat |
|          JournalFS        ->  B.10.00.%%U.2  The Base VxFS File Syste |
|          Keyshell         ->  B.10.00.%%U.2  Keyshell                 |
|          LSSERV           ->  B.10.00.%%U.2  Network Licensing Server |
|          LVM              ->  B.10.00.%%U.2  LVM                      |
|          MSDOS-Utils      ->  B.10.00.%%U.2  MSDOS-Utils              v
\<----------------------------------------------------------------------->/

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

H6296 22-9 230

Student Notes

In some cases it might be necessary to safely remove software from your system, for example if you are in desperate need of disk space and want to remove unused software.

To remove software, use `/usr/sbin/swremove`. Its user interface looks exactly like `swinstall`'s. You select the software to remove, and the system checks dependencies between selected and remaining software.

22-10. LAB: Hands-On, Update Using `swinstall`

Directions

Perform the following tasks.

Your instructor will provide a depot and explain which fileset(s) or product(s) to update.

1. Perform an update using the media from your instructor. Update only those file sets she/he named. Installing other file sets may disrupt normal system operation or require a system reboot.

Appendix A — Introduction to HP VUE Administration

Objectives

Upon completion of this module you will be able to:

- Describe the basics of the HP Visual User Environment.
- Differentiate between HP VUE and HP VUE Lite.
- Outline the component parts of HP VUE.
- Describe the HP VUE architecture and explain the need for a Broadcast Message Server.
- Describe the system wide directory structure.
- Describe the home directory structure.
- Describe how applications are started.
- Describe and customize the HP VUE Login Manager.
- Describe and customize the HP VUE Session Manager.
- Add system-wide actions to HP VUE and HP VUE Lite.
- List basic troubleshooting files and tools provided with HP VUE.

Appendix A — Introduction to HP VUE Administration

A-1. SLIDE: What Is HP VUE ?

What Is HP VUE?

- A Powerful graphical environment
- A set of applications for interacting with the computer

Features:

- Windows and workspaces
- Icon-based file management
- Front panel and toolboxes
- Extensive online help
- Session management
- Easy customization of colors, fonts, window behavior, and other aspects of appearance and behavior
- Easy to use text editor and icon editor
- Multi-media applications for annotating files

H6296 A-1 231

Student Notes

HP VUE is a simple, graphical user interface to HP-UX (and DOMAIN/OS SR10.4). It is designed to make interaction with your computer easier and more productive. It provides a graphical environment featuring window-based, mouse driven functionality and a simple, flexible set of utilities. In this environment you can have a collection of windows and icons that allow you to start several programs simultaneously and interact with all of them. In addition, you may have multiple workspaces, each with its own collection of windows and icons. It is like having six monitors on your desk!

HP VUE is based on MIT's X Window System and OSF/Motif standards. It is the first step toward NewWave in the Unix system environment.

HP VUE requires a bit-mapped display and some type of pointing device, such as a mouse, although it can be used with just a keyboard if required.

Appendix A — Introduction to HP VUE Administration

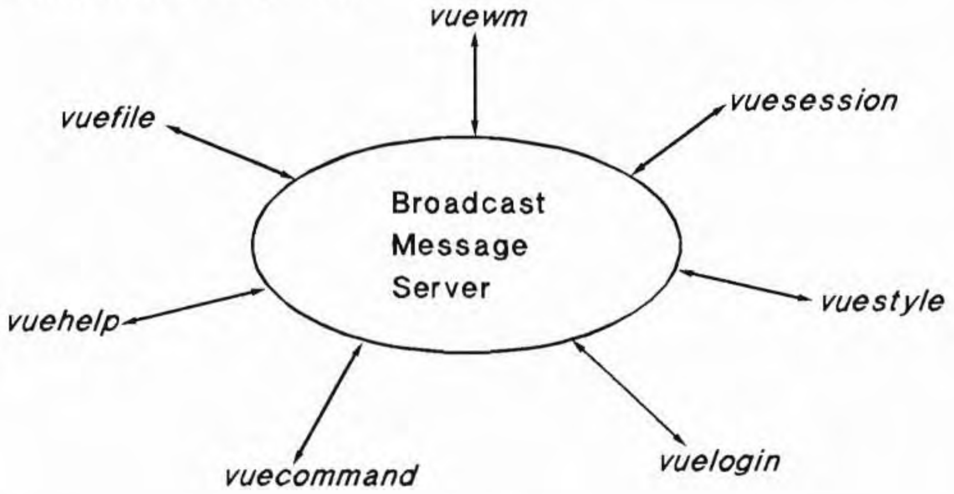
VUE Reference Manuals

- B1171-90061 *HP Visual User Environment 3.0 User's Guide*
- B1171-90044 *HP Visual User Environment Advanced User's Guide*

A-2. SLIDE: HP VUE Architecture

HP VUE Architecture

The BMS is the "hub" of the HP VUE architecture, by which components cooperate:



Student Notes

HP VUE consist of seven components known as "Managers" and two internal subcomponents: Broadcast Message Server (BMS) and SubProcess Control Daemon (SPCD). The Managers are:

`vuelogin`	Validates user login and password
`vuesession`	Starts and later saves the user's session
`vuewm`	The user's workspace and window manager
`vuefile`	Application launcher, file and directory browser
`vuestyle`	Interactive application resource customizer
`vuehelp`	On-line help and man page browser
`softmsgsrv`	Provides inter-VUE manager communication

Appendix A — Introduction to HP VUE Administration

The Login Manager

`vuelogin` is the first interface the users see; it is simple to use and easily customized. `vuelogin` provides services similar to `init`, `getty` and `login` on character terminals: prompting for login and password, authenticating the user, and running a user's session. Specifically, the Login Manager is responsible for:

- Reading initial configuration files
- Starting the X server
- Displaying the login screen and validating the login and password supplied by the user
- Invoking the HP VUE Session Manager

The Session Manager

A **session** in the VUE context is defined to be the lifetime of the HP VUE Session Manager - the time between logging in and logging out. The Session Manager provides the ability to save and restore sessions. The following are characteristics of sessions stored and restored by session manager:

- which applications are running
- what the application windows look like
- other settings such as the X server, mouse behavior, audio volume, etc.

Types of Sessions

- A **current session** is a session stored at logout.
- A **home session** is stored by the user at some other time during the session.

HP VUE components communicate with each other via two mechanisms:

Broadcast Message Server (BMS)	and standard Inter-Client Communications Conventions (ICCCM), defined by the X Consortium. ICCCM is used for communication of basic X-Windows information such as properties and client events. The ICCCM focuses on clients who cooperate visually. This method is insufficient for VUE components, who must cooperate semantically as well. For semantic cooperation types of actions such as dragging a data file over an icon, the BMS is used. The BMS is a message-routing service capable of invoking the intended receiver of a message if it is not running.
SubProcess Control Daemon (SPCD)	Used for inter-system tool sharing. The SPCD allows one process to invoke and control another process on a remote or local host. It also allows the parent process to communicate with the `stdin`, `stdout` and `stderr` channels of the subprocess. For example, VUE is installed on your local system. By installing SPCD on a remote system and setting up the two systems so they can access each other (nfsmount, xhost, etc), actions can be written for tools which reside on the remote system. You can then (after some set up) use these actions on the remote system just like you can use actions on your local system. VUE need not be installed on the remote system.

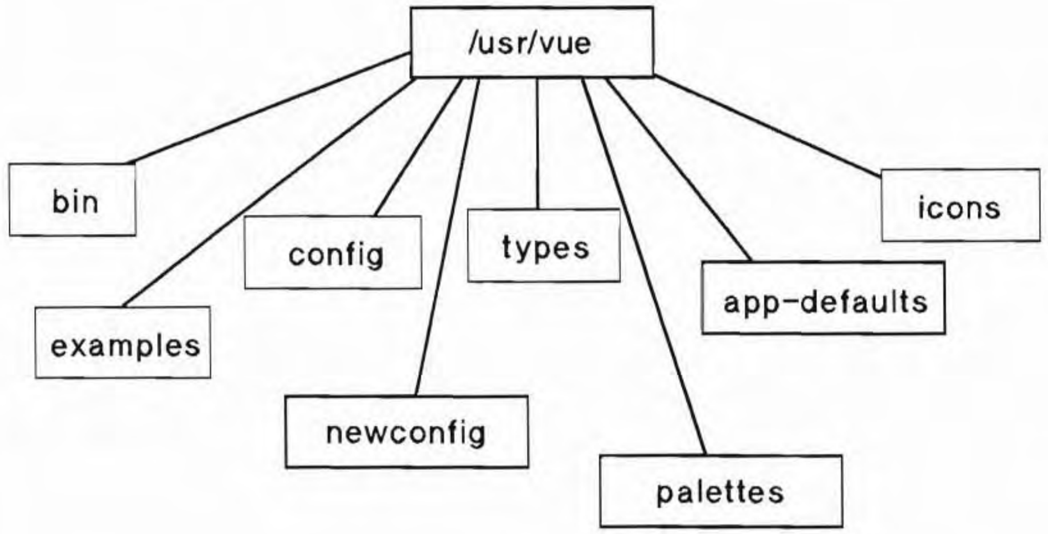
Pertinent files for this include:

`/usr/bms/config/softenv`	Environment variables propagated to subprocess started though SPCD.

`/usr/bms/bin/softinit`	Invocation specifications used by the BMS for starting messaging tools.
`/usr/bms/config/softtypes/config/C`	File type specifications used by Softbench and the BMS; not directly used by VUE.

A-3. SLIDE: The Directory Structure - System Wide

The Directory Structure - System Wide



H6296 A-3 233

Student Notes

This slide is designed to show you where all of the main system files and directories can be found. System Wide means all the files which are global to all users of VUE.

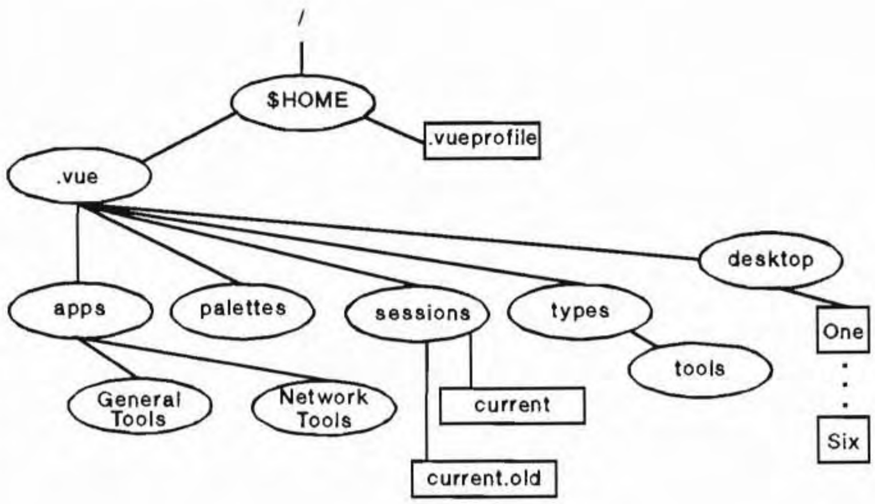
Here is a description of the main system wide directories:

`/usr/vue/bin`	HP VUE executables.
`/etc/vue/config`	Configuration files for login, session and workspace manager. The dynamic configuration files are in `/etc`. (`/usr` is a static directory)
`/usr/vue/types`	Built-in actions and filetypes.
`/usr/vue/app-defaults`	Application defaults for HP VUE clients.
`/usr/vue/icons`	HP VUE system wide icons.

`/usr/vue/palettes`	Palettes.
`/usr/vue/examples`	Printer configuration examples, old backdrops.

A-4. SLIDE: The User's Home Directory

The User's Home Directory



H6296 A-4 234

Student Notes

This slide shows the *per-user* files and directories used for configuring HP VUE. When modified, these will only affect the user in question.

- `$HOME/.vueprofile` contains environment files.
- `/$HOME/.vue/palettes` contains palettes you can create or modify.
- `/$HOME/.vue` subdirectory contains applications, tools, and palettes that can be used by the user.

A-5. SLIDE: What Is HP VUE Lite?

What Is HP VUE Lite?

- A different mode of operation, chosen at login time
- A subset of HP VUE
- Minimum set of processes
- Useful on hardware configurations with small amounts of RAM

H6296 A-5 235

Student Notes

HP VUE Lite is a collection of the most used features of HP VUE (Login, Workspace, Help, and Style Managers) in a "light weight" minimum set of processes. It features enhanced system performance by omitting full icon-based file management, full session management, and file annotation. It is intended for users who do not need access to the file system: for example, users running a single application with its own built-in file management feature. HP VUE Lite will also be useful to some hardware configurations with small amounts of RAM.

HP VUE Lite sessions are chosen from the login screen's "Options" menu.

Differences between HP VUE and HP VUE Lite:

- Session Manager uses different configuration files for HP VUE Lite, and the session is started from an editable startup script.

- VUE Lite does not have the File Manager.
- HP VUE Lite has a different Front Panel:
  - The terminal button has a configurable subpanel for starting other terminal emulators.
  - The Tools controls can be configured to start the application of your choice.
  - Since there is no File Manager, there are no drop zone controls.

The following table lists a comparison of HP VUE Lite to regular sessions.

Table A-1.

Feature	HP VUE	HP VUE Lite
Windows	yes	yes
Workspace	yes	yes
File management using icons	yes	no
Front Panel	yes	yes
Toolboxes containing applications	yes	no
Text editor	HP VUE Text Editor or `vi`	
Icon editor	yes	yes
Session management	yes	partial
Applications for customizing the workstation	yes	yes
Command line	yes	yes
Mailer	yes	yes

HP VUE Lite uses different session files than HP VUE regular sessions:

Table A-2. Comparison of HP VUE to HP VUE Lite Session Files

Purpose of File	Lite Session	Regular Session
system default session	/etc/vue/config/sys.ses.lite	/etc/vue/config/sys.session
system default resources	/etc/vue/config/sys.res.lite	/etc/vue/config/sys.resources
system default profile	/user/vue/config/sys.vueprofile	/etc/vue/config/sys.vueprofile
default window manager configuration	/etc/vue/config/sys.vuewmrc	/etc/vue/config/sys.vuewmrc
user session	$HOME/.vue/sessions/lite/vue.session	/$HOME/.vue/sessions/home or current/vue.session
user resources	$HOME/.vue/sessions/lite/vue.resources	$HOME/.vue/sessions/home or current/vue.resources
user X server settings	N/A	$HOME/.vue/sessions/home or current/vue.settings
user supplemental startup script	N/A	$HOME/.vue/sessions/sessionetc

A-6. SLIDE: Setting System-Wide Resources

Setting System-Wide Resources

Edit:

- `/etc/vue/config/sys.resources` for HP VUE sessions
- `/etc/vue/config/sys.res.lite` for HP VUE Lite sessions

Syntax of resource specifications:

client[client components*]resource:*   *value*

Resources will be in effect when a new user starts HP VUE for the first time.

H6296 A-6   236

Student Notes

A **resource** is a mechanism of the X Window System used to describe, or specify, certain attributes (appearance or behavior) of a window or application. The `RESOURCE_MANAGER` is a property of the root window. HP VUE uses the `RESOURCE_MANAGER` property to describe all resources that are not set to their default values.

Resources are loaded into the `RESOURCE_MANAGER` property by the Session Manager at the beginning of a session. Properties may be modified during a session by using the Style Manager or the EditResources action. The Session Manager stores resources at the end of a session.

The syntax of resource specifications is:

client[client components*]resource:*   *value* where:

client	can be either the actual client name or the class to which the client belongs.
client components	allow you to define the resource to specific parts of the client.
resource	can be either the resource name or class.

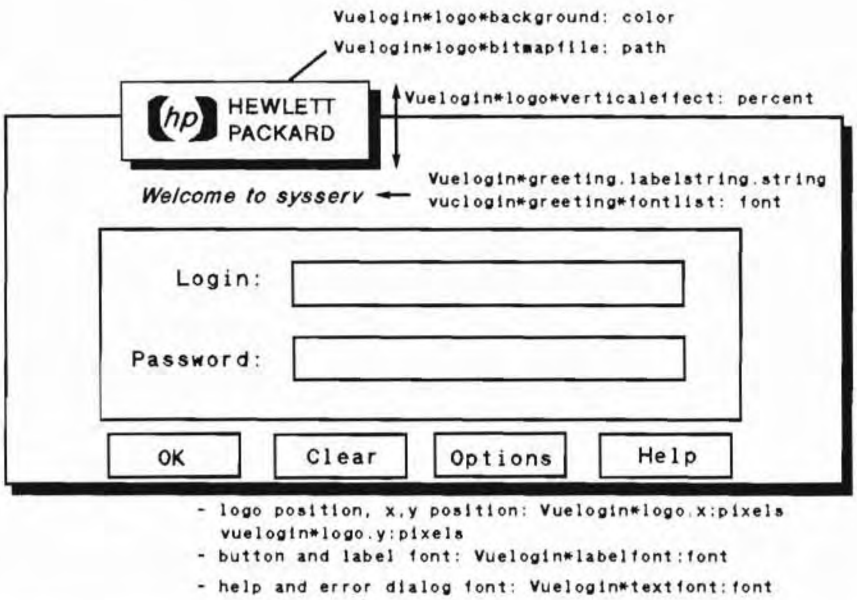
Resource files are text files and must obey the following rules:

- Each resource specification must be on a separate line.
- Text to the right of an exclamation point (!) is regarded as a comment.
- A pound sign (#) in the first column makes the whole line a comment.
- The resource name is separated from the value by a colon (:) and optional spaces or tabs.
- Do not have extra spaces or tabs after the value.

For detailed information on specific resources, their definitions and default values, see *HP Visual User Environment 3.0 User's Guide* (Part Number B1171-90079).

A-7. SLIDE: Customizing Login Manager

Customizing Login Manager



H6296 A-7 237

Student Notes

Login Manager is responsible for reading the initial configuration files, starting the X server, displaying the login screen and validating the login and passwords supplied by the user, and invoking the HP VUE Session Manager.

You can customize various aspects of the Login Manager appearance and behavior. After you customize the Login Manager, you must reread the Login Manager configuration files by either logging out and then back in, or killing the PID of the *parent* vuelogin process.

```
kill -HUP pid
```

Login Manager Files

Executable files located in /usr/vue/bin provide the Login Manager Functionality:

vuelogin	Performs configuration tasks and spawns another vuelogin process for each display in the system.
vuegreet	Provides the functionality of the login screen.
vuehello	Provides a transition effect between successful login and the beginning of the HP VUE session.

System-wide configuration files are located in /etc/vue/config:

Xconfig	Contains the resources for the behavior of vuelogin.
Xservers	Contains a list of servers to be run by vuelogin.
Xresources	Contains resources for the appearance of the login screen.
Xstartup	Program (usually a shell script) that is run as root after the user login and password are validated.
Xsession	Shell script that sets up the user environment variables, runs vuehello, and invokes the session manager.
Xreset	Shell script that is run as root upon termination of an HP VUE session.

Customizing the Login Screen

To change the appearance of the login screen, you set the appropriate resource in /etc/vue/config/Xresources, and then reread the Login Manager configuration files. The following table presents a summary of the Vuelogin resources:

Table A-3.

Screen Appearance	Resource	Value
Logo image	Vuelogin*logo*bitmapFile:	*path*
Logo position	Vuelogin*logo*verticalOffest: Vuelogin*logo*x: Vuelogin*logo*y:	*percent* *pixels*
General Logo Appearance	Vuelogin*logo**resource*:	*value*
Fonts	Vuelogin*textFont: Vuelogin*labelFont:	*font*
Content and Appearance of Greeting	Vuelogin*greeting.labelString: Vuelogin*greeting.fontList:	*string* *font*

To Make VUE Lite the Only Session Available

When the vuelite resource is set to true, the user is unable to log into a regular HP VUE session on that system. To set the vuelite resource to true, edit /etc/vue/config/Xconfig and modify this resource:

```
Vuelogin*vuelite:  true
```

Remember that the Xconfig file is only read by the Login Manager when it starts, so in order to make this resource take effect, you must restart the Login Manager by logging out and then back in.

A-8. TEXT PAGE: Examples of Customizing Login Manager

You can customize many aspects of the Login Manager appearance and behavior. The system-wide configuration files are located in /etc/vue/config.

Customizing the Login Screen

These lines in Xresources set colors for the login screen:

```
Vuelogin*background:       lightgray
Vuelogin*highlightColor:   #ef506f
```

These lines in Xresources specify a bitmap and background color for the logo:

```
Vuelogin*logo*bitmapFile:
/usr/local/lib/X11/vue/bitmaps/MYlogo.bm
Vuelogin*logo*background:   #ef506f
```

These line in Xresources specifies the login screen greeting:

```
Vuelogin*greeting.labelString:  This is %LocalHost%
```

Where the string %LocalHost% causes the hostname of the system to be displayed.

Customizing the Language Menu

To customize the language menu, you set the languageList resource in /etc/vue/config/Xconfig:

Vuelogin*languageList: *language [language ...]*

where *language* is a valid value for the LANG environment variable. The NLS directory /usr/lib/nls contains an entry for each supported value. The default language is the first language you specify.

Preventing Users from using the Failsafe Session

If you want to prevent users from using the failsafe session, modify the Vuelogin*failsafeClient resource in the Xconfig file. The value of this resource can be the path name to any XWindows client. For example, to guarantee that the user will always log in to HP VUE:

```
Vuelogin*failsafeClient:      /usr/vue/bin/vuesession
```

Customizing Login Manager Behavior

To Display a Message of the Day

The `vuehello` client displays the transition window after the user login and password are validated, before the session manager is run. By default, `vuehello` displays `Starting the HP Visual User Environment` and the copyright message (`/etc/copyright`) in the transition window.

The file `/etc/vue/config/Xsession` contains the variable `VUEHELLO`:

```
VUEHELLO=$VUEDIR/vuehello
```

Modify the line to include the `-file` option, which takes as its parameter the name of the file containing the message of the day:

`VUEHELLO=$VUEDIR/vuehello -file` *path*

Up to five files can be added; each uses a separate `-file` option.

To Run System-wide Scripts at Login and Logout

Separate files are executed at login and at logout. You may customize these files to contain the commands you want to run. Make sure the file is executable.

At login	`/etc/vue/config/Xstartup`
At logout	`/etc/vue/config/Xreset`

Setting Other Global Parameters

The file `Xconfig` contains behavior resources for the HP VUE Login Manager. It also specifies the location of other configuration files used by the Login Manager. Some of the other resources you can specify:

- Global environment variables (can also be specified in the Xsession file)

  `Vuelogin*environment:` *name=value [name=value ...]*

- Default user PATH environment variable

  `Vuelogin*userPath:` *path*

- Globally setting the LANG environment variable: You can set the LANG environment variable to any value supported by the operating system.

  `Vuelogin*language:` *language*

  Setting this variable in the `Xconfig` file causes the login screen to be localized, and the LANG is set for all users.

To set the language variable for individual users, you must set the variable in the `.vueprofile` file for that user. In this case, the login screen will not be localized.

A-9. SLIDE: Customizing The Default Session

Customizing The Default Session

Session Manager Files:

/usr/vue/config/sys.session
/etc/vue/config/sys.resources

Session Data:

vue.session
vue.resources
vue.settings

Session Files:

/$HOME/.vue/sessions/current
/$HOME/.vue/sessions/home
/$HOME/.vue/sessions/current.old
/$HOME/.vue/sessions/home.old

H6296 A-9 238

Student Notes

You can establish resources that will be in effect when a new user starts HP VUE for the first time. After the first time, the system-wide resource file will be ignored in favor of session-specific files. When a user logs into HP VUE for the first time, Session Manager uses the default session files `/etc/vue/config/sys.session` and `/etc/vue/config/sys.resources`. When the user logs out, the state of the session is saved to the `/$HOME/.vue/sessions/current` directory. This session is retrieved the next time the user logs in. You can customize the appearance of the first session by modifying `sys.session` and `sys.resources`.

Appendix A — Introduction to HP VUE Administration

Session Manager Files

Default Session Files

`/user/vue/config/sys.session`	The default session initiation file for HP VUE. This file is processed once for new HP VUE users.
`/etc/vue/config/sys.resources`	The default session resource file.

Session Data

Session data is stored in:

`vue.session`	Names of active clients, their window geometries, workspace presence status and startup string.
`vue.resources`	Resources for active clients in the session.
`vue.settings`	Server and session manager settings, such as screen saver timeout, audio and keyboard repeat settings.

Session Files

Session files are located in:

current	`/$HOME/.vue/sessions/current`
home	`/$HOME/.vue/sessions/home`
Back-Up	`/$HOME/.vue/sessions/current.old` and `/$HOME/.vue/sessions/home.old`

Customizing the Initial Session

The System Administrator can establish resources that will be in effect when a new user starts HP VUE for the first time. After the first time, the system-wide resources files are ignored in favor of session-specific files. The file you customize to set system-wide default resources depends on whether the session is HP VUE or HP VUE Lite. Each resource has a default value, but for most, the user can change that value.

HP VUE	`/etc/vue/config/sys.resources`
HP VUE Lite	`/etc/vue/config/sys.res.lite`

Execute Additional Commands at Startup

You can create a file called `sessionetc` and place it in each user's $HOME directory. If the file `sessionetc` exists, it is executed each time the user logs into a regular HP VUE session. Processes started by `sessionetc` must be run in the background. For example, the following line customizes the root cursor.

```
xsetroot -cursor /users/brian/point /usr/brian/mask -fg blue -bg red
```

Customizing the Fail-Safe Session

The fail-safe session consists of a window manager and a single terminal emulator. It is useful for correcting problems in Login Manager and Session Manager configuration files that prevent an HP VUE session from starting properly.

To customize the fail-safe session, edit /etc/vue/config/Xfailsafe. The commands you insert must run in the background. Add commands after the line that starts the window manager, and before the command that starts the hpterm.

```
$SDIR/mwm sleep $DELAY
new command
$XDIR/hpterm -ls
```

Who Can Unlock the Display

If the display is locked, only root or the user who locked it can unlock the display. You can set the resource vuesession*keys to allow other users to unlock the display.

```
vuesession*keys:    user [user, ... ]
```

Regardless of how you specify users in the keys resource, root will always be able to unlock the display.

A-10. TEXT PAGE: Example of Default Session Customization

Example 1: Disable VUE File Manager

Suppose you decide to disable the VUE file manager for all users. The `sys.session` file is the file that initiates sessions. Remember that it is read once for each user, and then a copy of it is made in the user's `$HOME/.vue/sessions/current` directory. If you already have users who have logged on, you will have to modify not only the `sys.session` file, but each user's `$HOME/.vue/sessions/current/vue.session` file.

To prevent access to the file manager for all users you will have to make the following changes:

Edit the `/etc/vue/config/sys.session` file and comment out the line

```
vuesmcmd -cmd "vuefile -geometry -30+30"
```

which starts the File Manager.

Create or modify a `$HOME/.vue/session/current/vue.session` file (that does not start the file manager) for each user.

Next you will need to modify the front panel so that the File Manager control does not appear. The front panel is defined in the Workspace Manager configuration file. The system file is `/etc/vue/config/sys.vuewmrc`. If the user has modified his own front panel, he will have a file `$HOME/.vue/vuewmrc`. (vuewmrc contains sections for Regular Sessions and Lite Sessions). Both of these files must be edited.

Comment out the line

```
CONTROL      Home
```

in the front panel description for Regular Session: Top Row.

Also comment out the whole section for Control description: Home

```
CONTROL Home
{
  TYPE   button
  SUBPANEL HomeSubpanel
   IMAGE     home
  PUSH_ACTION  f.action "OpenHomeDir"
   HELP_TOPIC   FPHome
}
```

Example 2: Set the value of hpterm

You may either edit the

/etc/vue/config/sys.resources file	for HP VUE
/etc/vue/config/sys.res.lite file	for HP VUE Lite

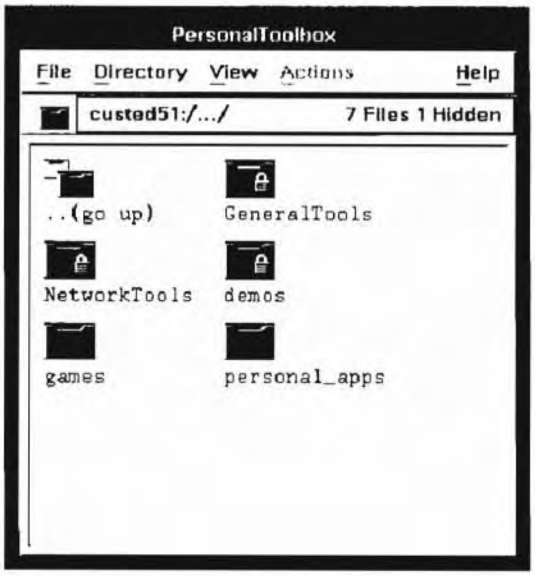
directly, or use the EditResources action in the General toolbox.

The following lines set the background for all hpterm windows to blue, but the background for the scrollbar part of the window to yellow.

```
hpterm*background:            blue
hpterm*scrollBar:             True
hpterm*scrollBar*background:  yellow
```

A-11. SLIDE: Toolboxes and Actions

Toolboxes and Actions



H6296 A-11 239

Student Notes

A **toolbox** is a container for action icons. These icons are a visual way of representing applications utilities and other commands.

An **action** provides a user interface for applications and other commands. Actions provide a visual representation of a command. The action definition specifies the command to be executed when the action is invoked, and connects the command with a visual representation (an icon). That icon becomes a graphical user interface for the command.

There are three toolboxes in HP VUE 3.0:

Personal Toolbox — A user's personal toolbox. You can create actions yourself or copy them to other toolboxes. Initially the personal toolbox contains actions that you personally

	place there, actions from the general toolbox that may be used frequently and finally actions that the System Administrator may have provided.
General Toolbox	This toolbox contains applications and utilities. The general toolbox contains actions built into HP VUE and actions that the System Administrator has placed there for system-wide usage. Only the System Administrator can add actions to the General Toolbox.
Network Toolbox	This toolbox lets you access actions on other systems. The top level of the network toolbox contains a sub-directory for each remote system that you can access. Only the System Administrator can add actions to the Network Toolbox.

Locations of Toolboxes

Toolboxes are special directories:

Personal Toolbox	`/HOME/.vue/types/tools`
General Toolbox	`/etc/vue/config/types/tools`
Network Toolbox	`/etc/vue/config/import`

A-12. SLIDE: Adding System-Wide Actions

Adding System-Wide Actions

- Only the System Administrator can create system-wide actions.
- The procedure varies slightly for HP VUE versus HP VUE Lite.
- For HP VUE:
  1. Create a Personal Action
  2. Copy the definition file to `/etc/vue./config/types`.
  3. Copy the Action from the Personal Toolbox to the General Toolbox.
- For HP VUE Lite:
  1. Create a Personal Action.
  2. Copy the definition file to `/etc/vue/config/types`.
  3. Add the action to the system-wide front panel control.

H6296 A-12 240

Student Notes

Actions make it easier to run applications by letting you represent the application as an icon that you can manipulate. When you create an action, you integrate the application in the HP VUE environment.

There are two ways to create actions:

- Use the Create Action utility
- Manually edit a database file. You add an action icon to a toolbox by creating an executable file with the action name in the toolbox directory.

Creating a System-Wide Action

To make a system-wide action in HP VUE, you login as root, and use the Create Action utility from the General Toolbox.

- When the action is applied, the system will save it in a file. Make a note of the file name to which the action definition was saved.
- Copy the definition file to `/etc/vue/config/types`.
- Create an action icon in the General Toolbox by either:
  - Copying the action icon from the Personal Toolbox to the General Toolbox
  - *OR*, Creating a new file in `/etc/vue/config/types/tools` with the same name as the action name. The file must be executable, and may be empty.

Creating a System-Wide Action in VUE Lite

To make a system-wide action in HP VUE Lite, you login as root, and use the Create Action utility in the Toolbox.

- When the action is applied, the system will save it in a file. Make a note of the file name to which the action definition was saved.
- Copy the definition file to `/etc/vue/config/types`.
- The action can now be used in a system-wide Front Panel control, by editing `/etc/vue/config/panels/fp.tool`.

A-13. TEXT PAGE: Review of Using The Create Action Utility

You use Create Action to define new actions. HP VUE uses actions to provide icons representing applications or other utility programs and scripts.

- Initially the action is placed in your Personal Toolbox. The System Administrator can create an action in the Personal Toolbox and then move it to the General Toolbox.
- Create Action will do the following:
  - Display the Create Action dialog to create the definition for the action.
  - Reread the database so the action takes effect immediately.
  - Create an icon for the action in your Personal Toolbox.

Using Create Action

1. Start Create Action by double-clicking CREATEAction in the Personal Toolbox
2. In the Name field, type a unique name for the action.
3. In the Command Line field, type the command that starts the application. Where you would type a filename, substitute *$n*, where *n* is an integer. For example:

```
bitmap $1
diff $1 $2
```

4. The Window Type selects how the action's output will be displayed. Your options are:

   X Windows
   No Output
   Terminal
   Terminal (auto-close)

5. If the command takes a data file, type the text of the prompt into the Filename Prompt field.
6. If you have special icons for the action, type the filenames in the Large Icon and Small Icon fields. Supply the full path.
7. In the Description field, type the item (f1) help for the action icon.
8. Choose Apply. A dialog box appears telling you the name of the database file created for the new action, and the system is busy while the database is reread.

Note

As long as the Create Action window is open, you can edit the action, apply the changes, and test the modified action. Once Create Action is closed you cannot use it to edit the action—you must manually edit the file created by the Create Action for the action definition.

9. Test and modify the action until it behaves properly, then Close the Create Action dialog.

A-14. TEXT PAGE: Troubleshooting HP VUE

There are several ways to troubleshoot HP VUE problems:

- Read the error messages recorded in the error-monitoring logfiles.
- Use one of several utilities to help diagnose errors.

Error-Monitoring Files:

`/var/adm/sw/swinstall.log`	Lists errors encountered during the swinstall process.
`/var/vue/Xerrors`	The system-wide error log. Records errors that occur during login.
`/$HOME/.vue/errorlog`	The user's error log. A new error log is created each time a user logs in. Errors from the previous two sessions are saved in `errorlog.old` and `errorlog.older`.
`/tmp/errorlog.`*login_name*	Created if `/$HOME/.vue/errorlog` cannot be created

Diagnostic Utilities

Utilities shipped with HP VUE let you constantly monitor system errors and diagnose common configuration problems.

- The WatchErrors action opens a terminal window that monitors `/$HOME/.vue/errorlog`. The window can be minimized, and will automatically normalize itself whenever anything is written to the error log.
- `/usr/contrib/bin/X11/dr_vue` checks many common sources of problems, particularly problems with network configuration.

Common Problem

Broadcast Message Server Fails to Start

If the Broadcast Message Server (BMS) fails to start, Session Manager cannot start a session. You will see an error similar to:

```
the session manager cannot start HP VUE messaging system
```

A common cause of this problem is that the network hostname is not the same in `/etc/vue/config/Xservers` as it is in `/var/adm/inetd.sec`. The hostname must be the same in both files.

If you previously used Softbench and have a personal `$HOME/.softinit` file, you may need to remove the file or edit it to include the HP VUE tools. You can also copy a "known good" file from `/usr/softbench/config/softinit`.

You should also check the permissions/ownership of:

```
/etc/hosts                      bin   bin    -r--r--r--
/var/adm/inetd.sec              root  bin    -r--r--r--
/usr/bms/bin/softmsgsrv         bin   bin    -r-xr-xr-x
/usr/softbench/config/softinit  bin   bin    -r--r--r--
```

You may also need to check your network configuration.

Appendix B — Connecting an X Station

Objectives

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

- Connect an X station to your system.
- Sett up the X station user's environment.

Appendix B — Connecting an X Station

B-1. SLIDE: What is an X Terminal ?

What is an X Terminal ?

- An X terminal is a graphics terminal connected via LAN.
- An X terminal uses the industry-standard X Window System.
- An X terminal supports OSF/Motif^{TM}.

H6296 B-1 241

Student Notes

What is a X Terminal?

A graphics terminal connected via the LAN, an X terminal uses the industry-standard X Window System, Version 11, Release 4. An X terminal supports OSF/Motif and comes standard with a license to use HP's Visual User Environment (VUE). Sometimes called X stations, X terminals deliver the performance needed to display multiple complex graphic applications simultaneously. Pseudo device files are needed for each window used in a session. The login process is controlled through either `xdm` or `vuelogin`.

The HP 700/RX, the ENVIZEX station and the ENTRIA family of RISC-based X Window stations are high-performance X terminals that use the industry-standard X Window System, Version 11, Release 5 (HP-UX 10.0). HP 700/RX, the ENVIZEX and the ENTRIA stations support OSF/Motif, and come standard with a license to use HP VUE. All models use an Intel i960CA RISC processor to execute X server instructions at workstation performance levels.

Appendix B — Connecting an X Station

The X station communicates to the server via a LAN connection.

B-2. SLIDE: Adding an X Station

Adding an X Station

1. Configure the X station.
2. Configure the file server (host):
   a. Install the X station software.
   b. Execute `/usr/bin/X11/700X/admin/xtadm`.
3. Power on the X station.

H6296 B-2 242

Student Notes

When the X station is powered on, it sends out a request on the LAN. A system that has been configured as a file server (host) responds to the request. The host downloads configuration, fonts and a login screen to the X terminal.

In the next few slides you will see how to add an X station to your system. There are three tasks that must be successfully completed before you can use your X terminal: configure the X station, configure the file server and start the X station.

Appendix B — Connecting an X Station

Before you can begin you will need to know the following: (Your network administrator should be able to help)

- file servers name (host)
- IP address of the file server
- Internet Protocol (IP) address of the X station
- X terminal network name
- hardware address of X station

The IP address and system name for the host can be seen in the `/etc/host` file. The IP address and X terminal name are unique names assigned by the network administrator. The hardware address of the X station can be seen from the X terminal's configuration screen.

B-3. SLIDE: BOOTP — TFTP/NFS Overview

BOOTP — TFTP/NFS Overview

BOOTP	allows certain systems to load configuration parameters and host information from a server
TFTP	transfers files from a remote system
NFS	provides a faster alternative to TFTP for downloading files from a remote system

H6296 B-3 243

Student Notes

Two main bootstrap protocols are used to download code on a diskless host. One of them, `/etc/bootpd`, is used with X terminals.

The X station is a diskless workstation (like a HP 9000 workstation). In order for these diskless terminals to boot and to load graphic information such as fonts into RAM, they communicate via the bootstrap protocol BOOTP as a client with their server.

- The client broadcasts a *bootrequest* packet, containing at least its hardware address.
- The server answers with a *bootreply* packet, containing all information the client needs to know.
  - Client's IP address
  - Name of boot file which the client should load in the second phase of the protocol for the file transfer via Trivial File Transfer Protocol (TFTP) or Network File System (NFS), as well as other information, such as subnet mask, addresses of name servers, or gateways.

B-4. SLIDE: Configure the X Station

Configure the X Station

1. Connect the X terminal.
2. Turn on the X terminal.
3. Display the configuration screen:
   a. Modify the terminal screen.
   b. Modify the network screen.
   c. Modify the startup screen.

H6296 B-4 244

Student Notes

Connect the X terminal

1. Connect the mouse, keyboard and monitor. When connecting the X terminal to the network you may use either the ThinLan BNC connector or the LAN AUI connector for a ThickLan MAU, EtherTwist MAU or ThinLAN MAU.

2. Connect the power cord and turn on the X terminal.
   a. If you have an HP 700/RX: If the screen is nothing but "snow", then you will need to set up the monitor. To do this, cycle power and wait for the beep. Press and release the numeric keypad key as indicated in the following table.

Table B-1.

Keypad Key	Display	Resolution	Frequency	Monitor Number
0	color	1280x1024	72Hz	A1097A/B/C/D, A2094A
1	color	1280x1024	60Hz	98754A, 98789A, D1187A
2	color	1024x768	75Hz	A1497A/B
3	color	1024x768	60Hz	98785A D1188A, 98753A
4	color	1024x768	60Hz	D1195A/B (SVGA)
5	color	800x600	72Hz	D1195A/B (VGA+)
6	color	640x480	60Hz	D1195A/B (VGA),D1182A/B
7	monochrome	1280x1024	72Hz	98774A/B
8	monochrome	1024x768	60Hz	98778A/X
9	color	1024x768	70Hz	D1196A

   b. If you have an HP ENVIZEX: at the sound of the second beep, press and release the space bar. To set up your monitor, hold down the left (Alt) (or (Extend Char)) key and, using the numeric keypad keys, type the numbers from the monitor's model number (for example type 2745 for a monitor type of C2745A).

Display the Configuration Screen

- Press the (To Configuration) key on the X terminal to display the Configuration screen.
- The Configuration screen will pop-up. You will see the following six buttons at the top of the screen:

```
Preferences    Network    Statistics
Terminal       Startup    Self-Test
```

Terminal Screen

The following instructions assume that the screens are all at the factory default settings. You may, before continuing, wish to record the previous settings, if any, and press factory defaults.

- If you have a D1195A monitor, you may need to set the monitor's resolution:
  - Click on `Terminal` to go to the terminal configuration screen
  - Click on the monitor resolution you prefer : `HP D1195A (SVGA)` , `HP D1195A (VGA+)` , or `HP D1195A (VGA)` .

— Click on OK .

- If there are no other parameters to configure, click on OK
- If you have a non-USASCII PS/2 keyboard, the default keyboard language is USASCII. If you have to set up the keyboard language yourself, use the configuration screens as follows:
  — On the terminal screen, click on the PS/2 keyboard language button.

  Select your language from the table that appears.

  — Click on OK

Network Configuration

- From the top of the configuration screen use your mouse and click on **`Network`** .

1. Find the button labeled `Network Parms from` and click on it until it states BOOTP .
   a. Click on the `File Server` field button until it states TFTP or NFS depending of the file server configuration.
   b. All the other fields are left blank.
2. Select `Enter below` for the `Network Parms from` label. In this case you have to:
   a. In the "IP address" field enter the X terminal's IP address.
   b. In the "File Server" field (This is the machine where you will load the X terminal software.) enter in the IP address of the host.
   c. Click on the `File Server` field button until it states TFTP or NFS depending of the File Server.
   d. If you are using Domain Name Service (DNS) enter in the IP address of the server in the field labeled "Name Server".

   If you don't know if you are using DNS, use the `nslookup` command to find out. On the host system type:

   ```
   # nslookup
   ```

   Above the > prompt it states either:

   ```
   Using /etc/hosts on:  hped827
   ```

   or

   ```
   Default Name Server: hped827.hp.com
   Address: 15.18.188.3 (IP address)
   ```

If you get back the second output above, then you need to enter the IP address in the "Name Server" field.

e. Click on SAVE

Startup Configuration

- From the top of the configuration screen press Startup .

1. Find the button labeled "X Server from" and click on it until it states Network .
2. Find the two buttons labeled "XDMCP" and click on the first button to enable XDM (the indicator of the button either will turn red or light gray on monochrome monitors if XDMCP is enabled). The second button explains how the X terminal will request the download. The common choices are either *broadcast* if there are many file servers or *direct* if there is only one file server. Click on the second button until it states either Broadcast or Direct .
3. Click on SAVE .

Note You can find the hardware (H/W) address of your X station by choosing Preferences from the configuration screen. The number should start with 080009. This number will be needed when executing xtadm on the server.

The X station should now be configured. After configuring the file server, the X station should start.

B-5. SLIDE: Configure the File Server

Configure the File Server

- Install the X station software.
- Execute /usr/bin/X11/700X/admin/xtadm.

```
              HP X STATION ADMINISTRATIVE TASKS

                             MAIN MENU

              1) Add an X station
              2) Remove an X station
              3) Printers, plotters
              4) Installation testing
              5) XDM Administration

                   ?) Help          x) exit

         Please enter selection (default=1): 1
```

H6296 B-5 245

Student Notes

Now that the X station is configured, you will need to configure the host that will control the X station. The host must be configured and operational on the network.

Install the X station software using `swinstall`

The `swinstall` program installs executables, configuration files, fonts, man pages, and other files that support the X station in the appropriate directories on the system. `swinstall` also makes necessary modifications to system configuration files such as `/etc/rc` and various network configuration files such as `/etc/services` and `/etc/inetd.conf`.

Appendix B — Connecting an X Station

If you want to use TFTP

Check to see if you have a /var/adm/inetd.sec file. If you do, then add the entry (if it does not already exist) that looks like the following:

```
tftp allow *
```

TFTP is one way of transferring data on the network. It is recommended that this entry be the last line in the file. Later you might want to edit the file to limit TFTP access as discussed in the manual inetd.sec(4).

If you want to use NFS

NFS is a faster alternative to TFTP for downloading files to the X station. It also allows easy access to the workstation fonts, once your X station's font path has been changed to include those fonts.

If NFS is not already running on your system, you need to install it by using sam:

1. As superuser, type sam
2. Select Networking/Communications
3. Select Services: Enable/Disable
4. Select NFS Server Disable and Selected on the Menu Bar to enable the service.
5. Exit sam. The /etc/nfsd daemons (usually 4) should be running on your system now.

Using xtadm

Now that the X station software has been loaded you should have a program called /usr/bin/X11/700X/admin/xtadm. This program is a user friendly way for you to configure your X stations. It will modify the files needed to control the X stations. Execute xtadm.

```
# /usr/bin/X11/700X/admin/xtadm
```

1. The first thing you should do is Add an X station .

```
          HP X STATION ADMINISTRATIVE TASKS

               MAIN MENU

     1) Add an X station
     2) Remove an X station
     3) Printers, plotters
     4) Installation testing
     5) XDM Administration

          ?) Help         x) exit
```

```
Please enter selection (default=1): 1

           HP X STATION ADMINISTRATIVE TASKS

                ADD AN X STATION

To add an X station to this computer you will need to obtain
the following information from your local network administrator:

     * The name you want to call the X station (network hostname).

     * The internet protocol (IP) address of the X station.

     * The LAN hardware (link level) address of the X station.
       (Obtain from the boot screen or shipping carton of the X station)

If the network to which you are adding the X station is
connected to other networks you will also need to know:

     * The subnet mask of the X station.

     * The internet protocol (IP) address of the default gateway.

Continue adding an X station [y|n] (y): y


           HP X STATION ADMINISTRATIVE TASKS

                ADD AN X STATION


X station series: 700/RX

Please enter name of X station to add [q|?]:myxterm1

Enter the IP address of the X station in dot notation, [q|?]:15.17.135.72

The following line was added to /etc/hosts:

15.17.135.72   myxterm1  # X station [no SMTP]

The LAN hardware address of an HP X station is shown on
its boot screen and its shipping carton label.

Enter LAN hardware address of X station [q|?]+0800091448bd

Enter subnet mask in dot notation [q|?] (255.255.255):(Return)

Enter gateway IP address in dot notation [q|?] (none):(Return)
```

```
You have entered the following network parameters for this X
station:

Name:                      myxterm1

IP address:                15.17.135.72
LAN hardware address:          0800091448bd
Subnet mask:               255.255.255.5
Default gateway IP address:    none

Are these correct? [y|n|?] (y): y

myxterm1 added to /etc/bootptab.

nfsd is not running or there are no local file systems...
NFS cannot be used.

Configuration file
/usr/tftpdir/usr/lib/X11/700X/config/myxterm1.cfg created.

xdm is not running. Do you want to start xdm? [y|n|?] (y):n

OK  .. no changes to xdm.

Add another X station? [y|n] (y): n
```

Note

You may select NFS, if NFS is running and your system. The message will be :

```
NFS is running on this computer, use hostname as NFS
Server for this X station? [y|n](y):y

a copy of /etc/exports file has been saved in
/tmp/exports.3749/ -access=myxterm1

Append the above line to /etc/exports? [y|n](y): y

line appended to /etc/exports
```

Make sure the X terminal's `network` field "File Server" matches what you select here, either TFTP or NFS.

2. Then, check the installation of the software by using the XDM Administration Menu:

```
HP X STATION ADMINISTRATIVE TASKS
```

```
              MAIN MENU

       1) Add an X station
       2) Remove an X station
       3) Printers, plotters
       4) Installation testing
       5) XDM Administration

            ?) Help          x) exit

  Please enter selection (default=1): 5


            HP X STATION ADMINISTRATIVE TASKS

                XDM ADMINISTRATION MENU

       1) Check operation of xdm
       2) Start xdm
       3) Stop xdm
       4) Enable/disable xdm password aging
       5) Enable/disable xdm root login
       6) Trim xdm logfile

            ?) Help          p) previous menu

  Please enter selection (default=1)1

  This test will :

       1) Check if XDM is present in /etc/inittab.

       2) Check if the XDM main daemon is currently running.

  Proceed with the xdm test [y|n] (y): (Return)

  An entry for xdm is present in /etc/inittab.

  No xdm daemon is running.

  To start xdm select menu item "start xdm".

  Please type (Enter) to continue
```

This will verify the operation of tftp. If all the tests pass continue on to the next step, otherwise check `/var/adm/sw/swinstall.log` for installation errors.

If tftp works but the X terminal does not boot, make sure you did the configuration screens on the X terminal correctly and that you installed the proper X terminal software.

3. If you are not going to run VUE, from the XDM Administration menu select: `Start xdm`

```
Please enter selection: 2

Please enter selection (default=1): 2

This selection will :

      1) Check if xdm is being started up from /etc/inittab.

         If this is not the case, /etc/inittab will be edited in
order to start up xdm automatically whenever your host
computer boots.

                      and

      2) After prompting you, xdm will be started up
immediately.

Add xdm to /etc/inittab ? [y|n] (y): Return

There is an entry for xdm in /etc/inittab already.

No changes were made to /etc/inittab.

Start xdm running now? [y|n|?] (y): Return
```

4. To verify xdm is operational, from the XDM Administration menu select: `Check operation of xdm`

```
Please enter selection: 1
```

5. You may now exit out of `xtadm`, from the XDM Administration menu select: `Previous Menu`

```
Please enter selection: p
```

From the Main Menu select `Exit`

```
Please enter selection: x
```

The host is now configured for the X station. Let's take a look at the power on sequence of the X station.

B-6. SLIDE: Power on the X Station

Power on the X Station

H6296 B-6 246

Student Notes

After the X station is assembled, connected to the LAN, and powered on, the X station should be able to automatically begin an X session without any operator intervention. Following is a list of what occurs on starting the X station.

Powering On

1. The X station sends out a boot request on the LAN. The X station uses the boot request to ask any computer on the network to send the X station's network configuration parameters.
2. A host that has the `bootp` daemon and that has the file `/etc/bootptab` configured for this station responds to the X station. The host sends the X station the network parameters information and the name of its remote configuration file.

3. Once the X station has its network parameters, it connects with the file server (either using tftp or nfs) and downloads the X server code and fonts.
4. The X station then directs a request using the XDMCP protocol to the file server asking for a login screen.
5. After about 30 seconds, a login window appears. If you have an account on the file server, you can now log in.

Changing the login screen

Xdm itself and Vuelogin are both xdm processes. Only one is allowed to post the login to the X terminal. Select whether you wish to have a VUE or xdm login by modifying or adding the an entry to the host's `/etc/inittab` file. Note what the default init state of your machine is out of the `/etc/inittab` file, as the example bellow shows:

```
init:2:initdefault
```

Then edit `/etc/inittab` to run either VUE or xdm (not both) at the default init state, which is again 2 in this case.

For xdm:

```
xd:2:respawn:/usr/bin/X11/xdm -nodaemon < /dev/null > /dev/null 2>&1
... VERIFY PATH.....: /usr/bin/X11/adm
```

For VUE:

```
vue:2:respawn:/sbin/init.d/vuerc
```

Cycling power on the X terminal at this point should produce a working X terminal. You will receive either a login screen, or if you boot your X terminal in *broadcast* mode, you will see a list of hosts displayed. Selecting one of the listed machines should produce either a xdm or VUE login screen.

Setting Up the User's Environment

xdm automatically runs the `.xsession` script in the user's home directory to start the window manager and clients. The file `.profile` is not be used. If `.xsession` is not found, xdm defaults to `/usr/lib/X11/xdm/sys.xsession` to the users home directories and calls it `.xsession`. The users can then modify their own environment and control which clients are run at login.

When you are ready to end your X terminal session, logout by closing your console window. This can be done with an `exit` command.

Appendix C — DTC Management

Objectives

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

- Explain the function of a DTC.
- Configure a terminal to a DTC terminal port.
- Configure a printer to a DTC terminal port.
- Describe the functions of DTC Device File Access Utilities (DDFA).
- Configure a port using DTC Device File Access Utilities (DDFA).

Appendix C — DTC Management

C-1. SLIDE: DTC Introduction

DTC Introduction

- LAN-based communication controller
- controlled from personal computer (PC-based) or HP 9000 (host-based)
- five types of DTCs
- three types of connector cards
- DTC self-test
- software downloaded from host
- DTC Device File Access (DDFA) Utilities

H6296 C-1 247

Student Notes

The **Datacommunications and Terminal Controller (DTC)** connects to the Local Area Network (LAN). In an HP-UX environment, a DTC can be controlled by either an HP 9000 server or personal computer. We will discuss the configuration for an HP 9000. For more detailed information, refer to "Using the HP OpenView DTC Entry Level Manager/UX".

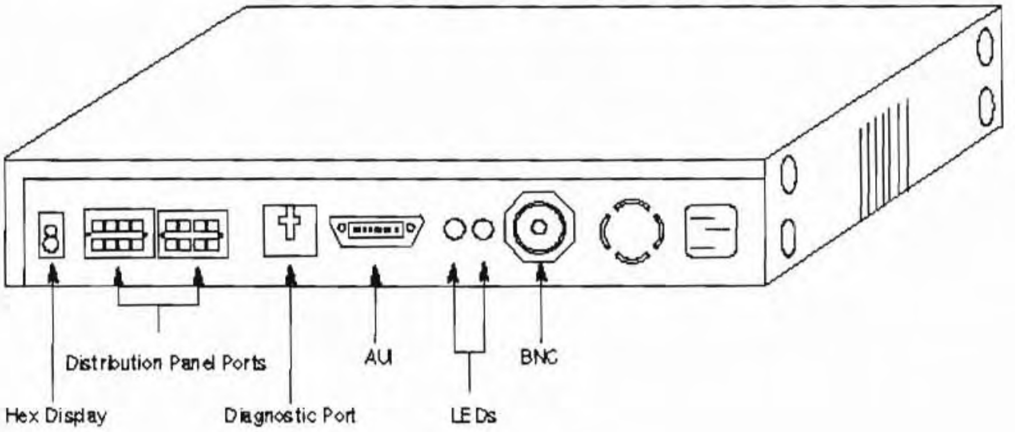
Types of DTCs

DTC16TN — is a LAN-based terminal server for systems that support the ARPA Telnet protocol. It has one 8-port RJ-45 panel and one modem (terminals) distribution panel, or two 8-port RJ-45 panels.

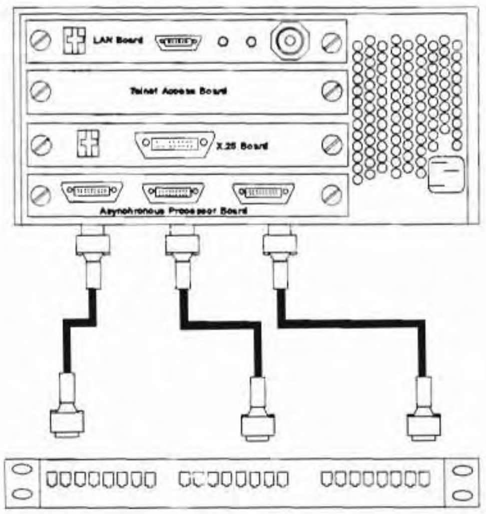
DTC72MX — has four card slots. One slot must contain a LAN card. It supports up to three asynchronous processor boards or up to three X.25 boards. Also, it supports one telnet board.

Appendix C — DTC Management

Each DTC has a processor card that handles the overall management of the DTC. The processor card is the central processing unit (CPU) of the DTC and contains the firmware and download software responsible for much of the data communications processing done by the DTC. Communication processing is off-loaded from the Server to the DTC.



DTC 16TN



DTC72MX

Appendix C — DTC Management

DTC Self-Test

The DTC self-test is automatically started whenever the DTC receives power. It may also be initiated through the diagnostic command `dtcdiag(1M)`.

DTC Download from Host

Once the DTC has passed the self-test, it initiates a software download by sending a boot request out on the LAN. The boot request contains the LAN address of the DTC. Each system receiving the boot request checks the LAN address against those contained in its own configuration files. When a match is found, the management system (Host) downloads the DTC operating code and the DTC configuration code to the DTC.

The dot on the front display which will blink during the download. When both the self-test and download have completed successfully, a sequence code of F2 will appear in the DTC display window with a steady dot.

Note The DTC will not be downloaded until all installed connector cards are configured at the Host.

DTC Device File Access Utilities

DTC Device File Access Utilities (DDFA) lets the HP 9000 system administrator pre-define a set of pseudoterminals (ptys) to communicate with asynchronous devices connected to the DTC. DDFA makes accessing DTC devices like accessing MUX devices. There is a correspondence defined between the DTC ports and HP-UX device files. This is especially important when configuring a printer on the DTC.

Prior to DDFA the pty device files were assigned randomly. When a user logged in they where randomly assigned a pty device file and were not guaranteed the same device file as before. DDFA facilitates the use of `/etc/ttytype` when setting the TERM variable.

C-2. SLIDE: Adding a Terminal to the DTC

Adding a Terminal to the DTC

1. First, set up the operating environment by doing the following:
   a. Updating the host with: HP-UX OpenView DTC Entry-Level Manager/UX
   b. Configuring the host with: LAN/9000 Link, Internet Service Product
   c. Starting DTC Manager daemons

H6296 C-2 248

Student Notes

We are going to take a look at adding a terminal to a DTC port in two steps. The first step is to set up the host's operating environment. Step two involves configuring the DTC using `/usr/sbin/dtcconfig` and adding the terminal.

Step 1: Set Up the Operating Environment

The software requirements for DTC Manager/UX are HP-UX 10.0, LAN/9000 Link and Internet Services Product. The network needs to be installed and operating. LAN/9000 Link and Internet Services Product are included in the operating system. The DTC uses the Telnet and TCP/IP protocols to establish terminal and device connections to the HP 9000 host.

Appendix C — DTC Management

You will need to configure the host on the network. Details of Internet Services Product operations and troubleshooting are discussed fully in *HP-UX Network Administration with HP-UX Clusters* (Course Number H6284S).

Use the `swinstall` process to load HP OpenView DTC Entry- Level Manager/UX.

The following two daemons must be running on the host computer in order for the configuration information to be downloaded to the DTC. They are automatically started when installed.

`/usr/sbin/rbootd`	The rbootd daemon waits for a boot request packet from the DTC. When it receives the packet, the host verifies the LAN address from the DTC packet. If there is a match, the host downloads the configuration information to the DTC. The configuration information is generated using the DTC manager's, `/usr/sbin/dtcconfig` command.
`/usr/sbin/dtcnmd`	The dtcnmd daemon enables `/usr/sbin/dtcdiag` to perform diagnostics on the DTC. Also, this daemon reports logged events and uploads from the DTC.

If you need to start these daemons manually, use the `rbootd` and `dtcnmd` scripts in `/sbin/init.d`.

Below is an example of how to determine the correct */dev/lan_device* needed for the execution of the `rbootd` and `dtcnmd` commands. The `lanscan` command provides the needed information.

```
# lanscan

Hardware Station        Crd Hardware Net-Interface  NM  MAC    HP DLPI Mjr
Path     Address        In# State    NameUnit State ID Type   Support Num
2.0.2    0x0800097843FB 0   UP       lan0     UP    4 802.3  Yes     52
```

If you have more than one LAN interface, run the `ioscan - f` command to find the correct one.

Now that the operating environment is configured, you are ready to configure the DTC.

C-2. SLIDE: Adding a Terminal to the DTC (Continued)

Adding a Terminal to the DTC (Continued)

1. Configure the DTC ports by doing the following:
   a. Executing `/usr/sbin/dtcconfig`
   b. Powering on DTC to initiate download
2. Connect to the terminal
3. Press (Return) on the terminal

H6296 C-2 249

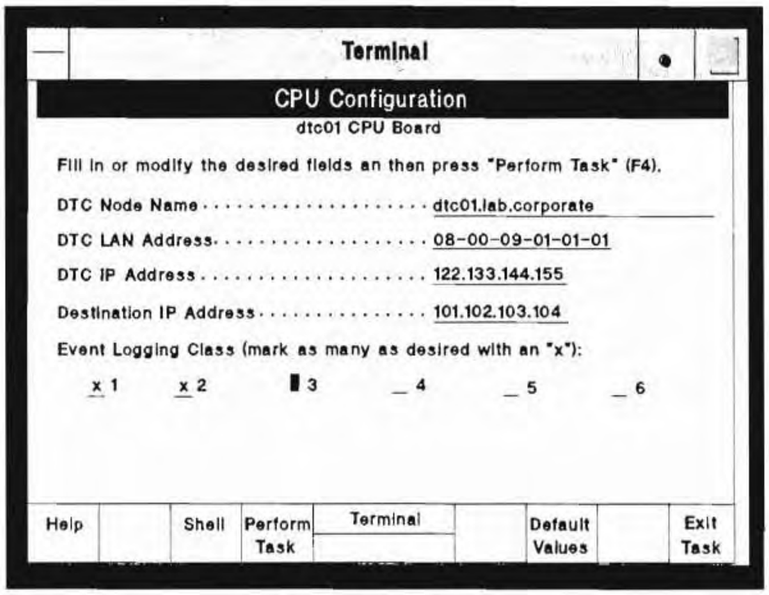
Student Notes

Now that the operating environment is set up and operational, we can configure each of the connector cards installed in the DTC. We will need to identify how each of the ports will be used.

Step 2: Configure the DTC Terminal Ports

Execute `/usr/sbin/dtcconfig` on the host computer to configure your DTC. This DTC configuration tool provides a user friendly interface of menus, fields and function keys.

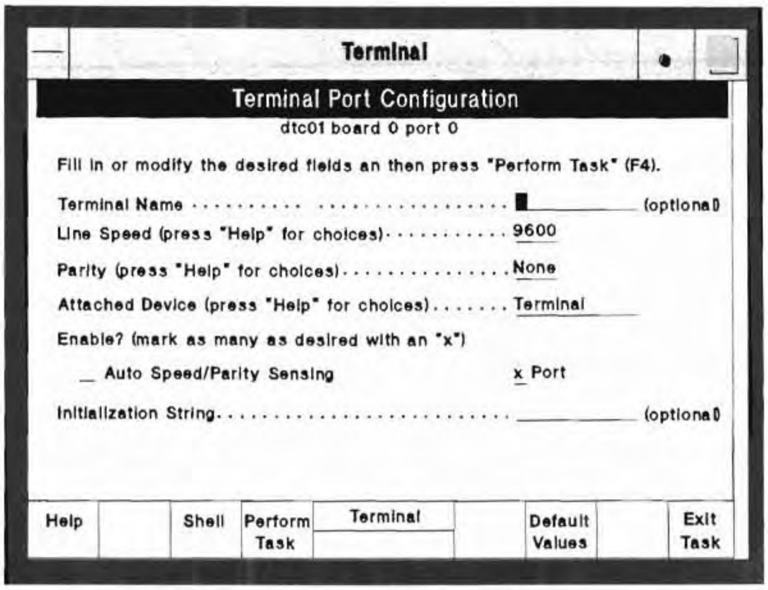
From the main menu, you may add, delete or modify a DTC. The first step in adding a DTC is to choose `Add a DTC`, which will ask you to identify the DTC's name. After providing the name of the DTC, the configuration screen will be displayed. On the configuration screen, you should configure the CPU board first. During the CPU board configuration you will identify the DTC's name, LAN address, IP address and node name.



CPU Configuration

After configuring the CPU card, configure the connector cards. From the main menu choose `Modify DTC Configuration` Action Menu. The Action Menu allows you to configure each port on a board or to change the LAN characteristics. Once you configure one board you may copy and paste it to other boards that are not yet configured.

For a particular port, a terminal, modem or printer may be configured. We will discuss printers later in this module. On the terminal configuration input screen, you will specify baud rate, modem characteristics, parity, and so forth.



Terminal Port Configuration

To initiate download, you must either power on the DTC or type `dtcreset`*DTC Name* from the host. As soon as the DTC receives power, it performs a self-test and then sends out a download request. The host you configured should respond and download the DTC's configuration to it.

Step 3: Connect the Terminal

After the DTC has downloaded, connect the terminal to the desired port.

Step 4: Press (Return)

Press return on the terminal. You should receive a logon prompt from either the DTC or from the `telnetd` daemon if you validate a default configuration (without switch).

To verify that the daemon is running, use

```
# ps -ef | grep telnetd
```

C-3. SLIDE: DTC Device File Access Utilities (DDFA)

DTC Device File Access Utilities (DDFA)

The DDFA — An Overview

- sets up correspondence between pty device special files and physical DTC ports
- configures DTC printers via SAM
- specifies DTC devices in applications that require configuration using device file names
- uses the Telnet port identification to determine identity of caller
- uses the dedicated port configuration file, `/etc/ddfa/dp`

H6296 C-3 250

Student Notes

What is DDFA?

The DTC Device File Access Utilities allow the system administrator to set up a correspondence between pty device special files and physical DTC ports. With this correspondence in place, the DTC's devices can be used in the same way that MUX-connected tty devices can. This is because the DDFA software eliminates the random assignment of ptys to DTC devices allowing the user or administrator to reference a specific DTC device by its predetermined pty name.

Appendix C — DTC Management

Benefits and Features of DDFA

The DTC Device File Access Utilities are available as a free utility supplied by Hewlett-Packard. These routines provide several benefits and capabilities to the HP 9000 on which they are installed, including the following:

- *make accessing DTC devices like accessing MUX devices.* The Device File Access Utilities allow the system administrator to set up a correspondence between DTC ports and HP-UX device special files. With this correspondence defined, the system spooler or a user application can manipulate predetermined device special files to read and write to specific DTC ports.
- *allow user applications to access DTC devices using standard HP-UX system calls.* After the correspondence between DTC devices and HP-UX device special files has been set up, user applications can use the standard HP-UX `read()`, `write()`, `open()`, `close()`, and `ioctl()` calls. These calls access DTC devices by manipulating their corresponding device special files.
- *allow DTC printers to be configured via SAM.* After the correspondence between DTC printers and HP-UX device special files has been set-up, using SAM to configure the spooler using DTC printers is identical to using MUX- connected printers, except that the pty name assigned to the DTC printer must be used in place of a tty name for a MUX printer. In fact, the standard HP-UX spooler model scripts can be used, even with DTC printers.
- *permit the system administrator specify DTC devices in applications which require configuration using device file names.* When using DDFA Utilities, most applications designed to read or write to MUX-connected devices can read and write to DTC-connected devices without the need to modify the application. It is only necessary to use the name of a DTC device's device special file instead of the MUX device special file when configuring the application.

Note

All of the above features depend on the host accessing a DTC port based upon a predefined pty device file name. Note that what is really being provided here is a means of opening an outgoing connection from the host to the DTC port by using a pty device file.

Telnet Port Identification

Telnet port identification is a related activity under DDFA that refers to the ability of the DTC to pass port identification information (IP address, board and port) to the host when connection is established. This information can be used by the host to map incoming connections to specific pty device files. In other words, Telnet port identification is a means of assigning incoming Telnet connections to predefined pty device files, usually so the identity of the caller (for example, a terminal) can be determined.

Port identification is possible through a set of enhancements made to the Internet Services Product Telnet daemon (telnetd) at HP-UX version 10.0, and version 12.1 of the DTC download code. With earlier versions of the Internet Services Product Telnet service, incoming Telnet connections were always assigned pty device files on a random basis. Therefore, the common practice of using the device file name to identify the port belonging to a login session had been impossible with DTC-based Telnet connections.

At HP-UX 10.0, the ARPA Telnet service was enhanced to allow the system administrator to set up predetermined pty names. In addition, the 12.1 version of the DTC download code was enhanced so that it passes board and port information to the host (via Telnet) at connection establishment time. Like DDFA Utilities, the newer version of Telnet uses the `/etc/ddfa/dp` file to obtain pty mappings.

DDFA Files and Daemons

Table C-1.

DDFA Utilities Product Contents	
`/etc/ddfa/dp`	Dedicated port configuration file. This ASCII file, created by the administrator, contains mappings between physical DTC boards and ports, and pty device file names.
`/usr/sbin/ocd`	Outgoing connection daemon. This daemon manages an outgoing connection from the HP-UX system to the DTC port.
`/usr/sbin/dpp`	Dedicated port parser. This program parses the dp file and executes an ocd process for each outgoing connection defined in the dp file.
`/etc/ddfa/pcf`	Port configuration file. This is an ASCII file that contains configuration parameters used by the ocd process.

DDFA Dedicated Port Configuration File

The `/etc/ddfa/dp` file defines the association of physical DTC ports to logical HP-UX pty device special files. A DTC port associated in this manner is referred to as a dedicated port, because of its fixed correspondence to a specific device special file.

Incoming and Outgoing Dedicated Ports

Dedicated ports can be defined as either incoming or outgoing ports. Incoming ports typically have user terminals on the DTC port. The system accepts incoming login sessions from incoming dedicated ports.

Outgoing dedicated ports typically have output devices such as printers, plotters, or terminals used as output-only devices attached to the DTC port. However, an outgoing dedicated port may also have an input/output device at the DTC end. The system, not the DTC, initiates connections to outgoing dedicated ports.

Note You must be root or a superuser to follow these configuration steps.

Appendix C — DTC Management

Configuration of DDFA Utilities and Telnet port identification centers around the `/etc/ddfa/dp`, or dedicated port configuration file. The following steps summarize the DDFA configuration:

1. Check to see if a DDFA working directory already exists:

   ```
   # ls /etc/ddfa
   ```

   If not, create a DDFA working directory:

   ```
   # mkdir /etc/ddfa
   ```

2. Check to see if a dedicated port configuration file already exists:

   ```
   # ls /etc/ddfa/dp
   ```

   If not, copy the default master dedicated port configuration file to the DDFA working directory:

   ```
   # cp /usr/examples/ddfa/dp /etc/ddfa/dp
   ```

Note Always use the dp files in the DDFA working directory (`/etc/ddfa`) and keep the default master files in `/usr/examples/ddfa` unchanged. This insures that a copy of the original files will always be available for reference.

---

3. For each DTC input device (such as a terminal) that you wish to permanently associate with a predetermined device file, perform the following steps.

   a. Determine the IP Address of the DTC, the board number and port number on the DTC to which the device is connected.

   b. Define a device file name that you will use for this input device. It is helpful to select a name that suggests the DTC port (i.e., `/dev/dtc1b2p4` for a device on DTC1, board 2, port 4).

   c. Create an entry in the `/etc/ddfa/dp` file of the following form:

   ```
   <DTC IP Address>     <board>/<Port>     <device file name>
   ```

   The slash (/) must separate the board and port parameters.

   For example, a terminal is on board 3 port 5 for a DTC whose IP address is 192.101.23.45 and whose name is dtc1. You want this terminal to be assigned `/dev/dtc1b3p5` each time its user logs in to the system. The entry in the dp file would be:

   ```
   192.101.23.45  3/5      /dev/dtc1b3p5
   ```

4. Execute the command who will reread the `ddfa` file:

   ```
   /usr/sbin/dpp /etc/ddfa/dp -k
   ```

DDFA Manual Information

For more information concerning DDFA software, please consult the on-line manual pages for the most up-to-date information. The following manual pages will provided more detailed information.

`ddfa(7)`	DTC Device File Access utilities description
`dp(4)`	Dedicated port file
`dpp(1M)`	Dedicated port file parser
`ocd(1M)`	Outbound connection daemon
`ocdebug(1M)`	Debug version of ocd
`pdf(4)`	Port configuration file

C-4. SLIDE: Adding a Printer to the DTC

Adding a Printer to the DTC

1. Configure the DTC ports:
   a. Execute `/usr/sbin/dtcconfig`.
   b. Power on DTC to initiate download.
2. Connect the printer.
3. Modify the `/etc/ddfa/dp` file.
4. Execute the `/usr/sbin/dpp` command to reread the `dp` file.
5. Create the printer.

H6296 C-4 251

Student Notes

First, set up and ensure that the operating environment is operational. Configure each of the connector cards that are installed in the DTC. Specify how each of the ports will be used.

Configure the DTC Printer Ports

Execute `/usr/sbin/dtcconfig` on the host computer to configure your DTC. This DTC configuration tool provides a user friendly interface of menus, fields and function keys.

From the main menu, you choose `Modify a DTC` .

Choose the port where you want to configure the printer, then ask for **`Change for type`** and choose `Printer type`. Exit the `dtcconfig` program.

You must power on DTC or execute `dtcreset`*DTC_name* to initiate download. As soon as the DTC receives power or reset, it performs a self-test and then sends out a download request. The host you configured should respond and download the DTC's configuration to it.

Connect the Printer

After the DTC has downloaded, connect the printer to the desired port.

Modify the /etc/ddfa/dp File

The `/etc/ddfa/dp` file defines the association of physical DTC ports to logical HP-UX pty device special files. A DTC port associated in this manner is referred to as a dedicated port, because of its fixed correspondence to a specific device special file.

Incoming and Outgoing Dedicated Ports

Outgoing dedicated ports typically have output devices such as printers, plotters, or terminals used as output-only devices attached to the DTC port. However, an outgoing dedicated port may also have an input/output device at the DTC end. The important part is that the system, not the DTC, initiates connections to outgoing dedicated ports.

Note You must be root or a superuser to follow these configuration steps.

Configuration of DDFA Utilities and Telnet port identification centers around the `/etc/ddfa/dp`, or dedicated port configuration file. The following steps summarize the DDFA configuration:

1. Check to see if a DDFA working directory already exists:

```
# ls /etc/ddfa
```

If not, create a DDFA working directory:

```
# mkdir /etc/ddfa
```

2. Check to see if a dedicated port configuration file already exists:

```
# ls /etc/ddfa/dp
```

If not, copy the default master dedicated port configuration file to the DDFA working directory:

```
# cp /usr/examples/ddfa/dp /etc/ddfa/dp
```

Note

Always use the dp files in the DDFA working directory (`/etc/ddfa`) and keep the default master files in `/usr/examples/ddfa` unchanged. This insures that a copy of the original files will always be available for reference.

3. For each DTC printer that you wish to permanently associate with a predetermined device file, perform the following steps.

   a. Determine the IP Address of the DTC, the board number and port number on the DTC to which the printer is connected.

   b. Define a device file name that you will use for this input device. It is helpful to select a name that suggests the DTC port (i.e., `/dev/dtc1b2p4_lp1` for a device on DTC1, board 2, port 4).

   c. Create an entry in the `/etc/ddfa/dp` file of the following form:

```
<DTC IP Address>   <board>/<Port>   <device file name> <port_config_file>
```

The slash (/) must separate the board and port parameters.

The *port_config_file* could be a copy of the `/usr/examles/ddfa/pcf` file.

For example, a printer is on board 3 port 5 for a DTC whose IP address is 192.101.23.45 and whose name is dtc1. You want this printer to be assigned `/dev/dtc1b3p5_lp1`. The entry in the dp file would be:

```
192.101.23.45  3/5      /dev/dtc1b3p5_lp1     /etc/ddfa/pcf
```

Execute `/usr/sbin/dpp` command

Execute the command that will reread the `ddfa` file:

```
/usr/sbin/dpp /etc/ddfa/dp -k
```

Create the Printer

Use the on-line command or SAM to create the printer.

```
#lpshut
#lpadmin -pimp_dtc1 -v/dev/dtc1b3p5_lp1 -mlaserjet
#accept imp_dtc1
#enable imp_dtc1
#lpsched
```

C-5. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. Describe how a DTC is controlled. What occurs at bootup?

2. Describe some examples of when DDFA can be used.

Appendix D — Disk Quotas

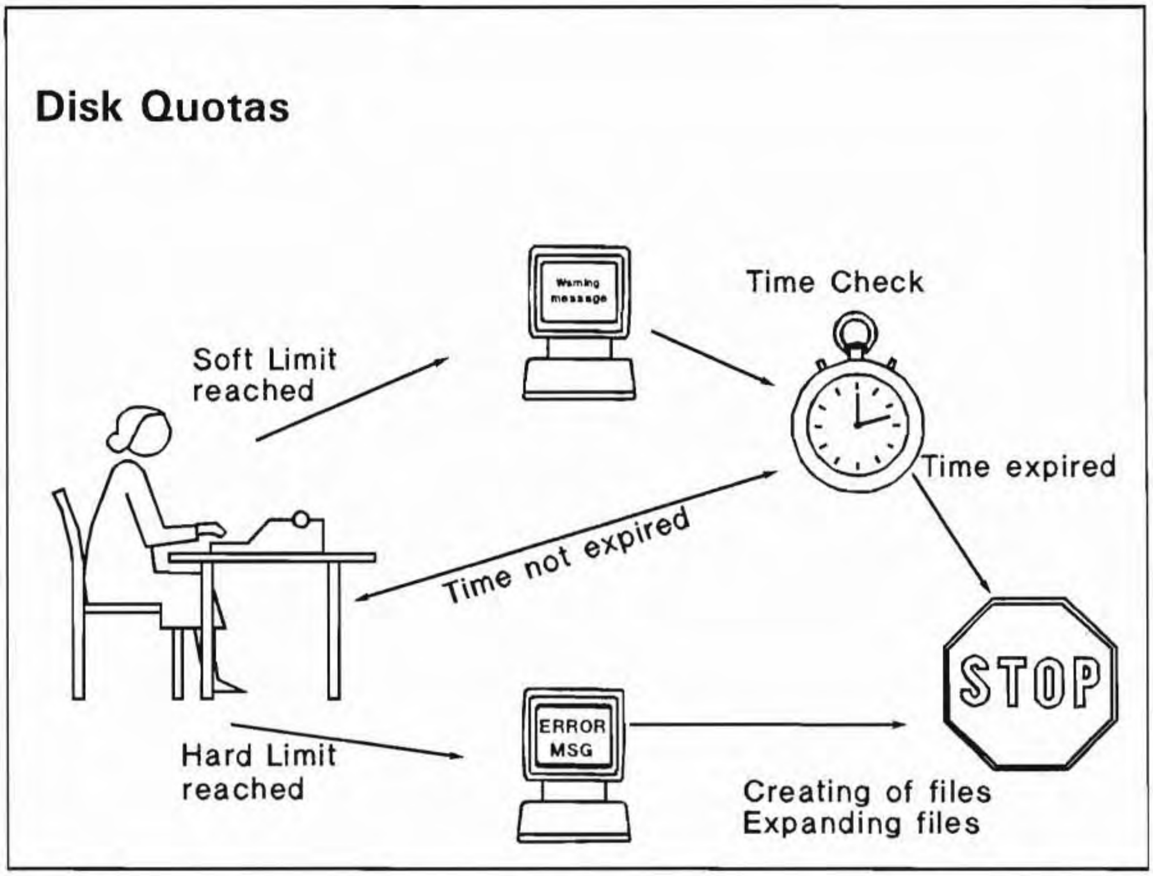
Objectives

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

- Implement disk quotas.

Appendix D — Disk Quotas

D-1. SLIDE: Disk Quotas



H6296 D-1 252

Student Notes

Disk quotas are used to limit the number of files and file blocks a user can own per file system. You can assign file and block quotas to users on any writable file system.

Disk quotas can be set up as limits for each user individually or for all users. Quotas can enforce a **soft limit** and a **hard limit** for each user's file system usage. Users who exceed their soft limits are notified and have a certain amount of time to correct the problem. The time the user is given to correct the problem is also configured by the system administrator.

Users who exceed their hard limits or exceed their soft limits beyond the specified time interval are prohibited from creating files or increasing file system block usage. The following messages appears if a user reached the block limit:

```
DISK LIMIT REACHED -- WRITE FAILED
```

Appendix D — Disk Quotas

The following messages appears if a user reached the file limit:

```
FILE LIMIT REACHED -- CREATE FAILED
```

A system administrator must intervene to increase or decrease the set limits or the user may temporarily save his data to a file system, `/tmp` for example, where he has not reached quota limits.

Statistics on user file system limits and usage are kept by the HP-UX operating system using data structures that reside in memory rather than on disk. Only a superuser can set up and manage disk quotas.

Pros and Cons of Disk Quotas

Advantages:

- When using disk quotas you control the way the file system is used.
- Impact on file system performance is rarely a problem.
- You can set a **soft limit** and a **hard limit** for each individual user, or groups of users, or for all users.
- You can set a **time limit** for each file system that uses quotas.
- You can set quotas running on any HFS and/or JFS file system. Not every file system needs to have quotas enabled.
- A user can check their quota status at any time.
- Quotas are optional, you do not have to use them if you do not need to use them.

Disadvantages:

- When using disk quotas the reboot time of a crashed system may take longer.
- A system administrator must intervene to increase or decrease the set limits for any user.
- A system administrator must reset quotas to 0 before removing a userid from the system.

D-2. SLIDE: Planning for Disk Quotas

Planning for Disk Quotas

You Must Decide:

- Which file systems require disk quotas
- What limits to set:
  - Uniform soft and hard limits?
  - Individual limits for each user?
  - How long (time) can a user exceed soft limits?

H6296 D-2 253

Student Notes

Deciding Which File Systems Require Disk Quotas

Typically, you would want to set disk quotas on file systems that would otherwise become full without limitations on their use. This usually means that only file systems containing users' home directories and possibly the `/usr` file system will require disk quotas. Do not assign quotas to the `/tmp` directory, which is historically a less restricted environment that is periodically purged of excess files by other means.

You can set disk quotas on all users on a file system, or only on specific users. You can also establish separate limits for the number of files and the number of blocks.

Appendix D — Disk Quotas

Choosing Limits

Setting up disk quotas involves setting soft and hard limits on the amount of file space a user can consume. The soft limit (or quota) is the amount of file space allocated for a given user. The soft limit can be temporarily exceeded. The hard limit (or limit) is the maximum amount of file space allowed to a given user.

The grace period for exceeding the quota is specified by you, or can be left at the default of one week. Once the grace period expires, the quota becomes a hard limit.

Disk Space Requirements

When you set up disk quotas, a `quotas` file is created at the root of each file system. The `quotas` file stores limits and utilization statistics for each user for the entire file system. The `quotas` file grows based on the numeric value of the user ID. If you are setting up disk quotas, you might want to keep the numeric value of your user IDs to low numbers.

D-3. SLIDE: How to Set Up Disk Quotas

How to Set Up Disk Quotas

1. Mount the file system
2. Edit /etc/fstab to enable quota checking file at reboot
3. Create an empty file quotas in root level of the file system using cpset(1M)
4. Set user quotas using edquota
5. Enable quotas by unmounting and mounting the file system or rebooting the system

H6296 D-3 254

Student Notes

Disk quotas are set by executing the following steps:

1. Mount the file system.
2. Edit /etc/fstab to enable quota checking file. If the current entry exists it must be edited, otherwise a new entry must be created. To enable quota checking on a file system, this entry would need to be changed to:

```
/dev/vg01/lvol2 /home hfs rw,suid,quota 0 1
/dev/vg01/lvol3 /var/mail vxfs rw,suid,quota 0 1
```

The next time the system boots, disk quotas will be automatically enabled for these file systems. Since the file systems were mounted when this change occurred, disk quota checking is *not* currently enabled. You must reboot the system for it to take effect, or you can manually start up quotas as described later.

3. Create an empty file quotas in root level of the file system using cpset(1M).

```
# cpset /dev/null /home/quotas 600 root bin
# cpset /dev/null /var/mail/quotas 600 root bin
```

The /home (/var/mail) part of the file name specifies that the file is to be created at the top level or "root level" of the file system that will be mounted to the /home directory at boot time. The /dev/null specifies that the file is created empty. For further information on the cpset command, see the cpset(1M) manual page in the *HP-UX Reference Manual*

This file will contain the limits and usage statistics after quotas is enabled on this file system. The data will be in binary form and keyed to each users "uid" number.

4. Set user quotas using edquota.

Set quotas for a prototype user and apply these quotas to all users of that file system, or set individual quotas user by user.

5. Enable quotas by unmounting and mounting the file system

```
# umount /home
# mount /home
```

Note You must activate quotas for *each file system*, because the quotas are checked on a per file system basis.

D-4. SLIDE: `edquota`: The Quota Editor

`edquota`: The Quota Editor

- Set uniform limits for users of a file system:
  - Set up a "prototype limits" for a "prototypical user"

    `/usr/sbin/edquota` *proto_user*

  - Implement prototype user's limits for other users of the file system

    `/usr/sbin/edquota -p` *proto_user* `bob gerry alicia`

- Set quotas for an individual user
  - Use `edquota`

    `/usr/sbin/edquota ted`

- Set time limits:
  - Use `/usr/sbin/edquota -t`

H6296 D-4 255

Student Notes

`edquota` is the quota editor. The `edquota` command converts the binary data stored in the `quotas` file to a readable form so that the superuser can establish quota limits for users of the file system. `vi` is the default editor.

While in the editor, you specify the *limits* you wish to set. The lines have the general format of:

`fs` *filesystem* `blocks (soft =` *limit*`, hard =` *limit*`) inodes (soft =` *limit*`, hard =` *limit*`)`

for editing user block and inode quotas, or for setting the time specifications:

`fs` *filesystem* `blocks time limit =` *days*`, files time limit =` *days*

Note

One block is 1 kilobyte.

Set Uniform Limits for the File System

To do this, you set limits for a *prototypical* user (this is a normal user on the system), and then replicate those limits for other users owning files in the file system.

```
# /usr/sbin/edquota tom
```

While in the `vi` editor, you should type in a specification of the limits you desire. An example might be:

```
fs /home blocks (soft = 1000, hard = 1200) inodes (soft = 250, hard = 300)
```

Then you should write (save) the text file, and quit the editor. This entry sets block and inode limits for the file system `/home`.

If you want uniform limits for groups of users or for all users, apply the prototype limits to actual users. This is done with the `edquota -p` command. Given the example above we might type:

```
#  /usr/sbin/edquota -p tom kathy ivan gisela robert
```

This command assigns the limits of the *prototypical* user *tom* to the users `kathy`, `ivan`, `gisela`, and `robert` who were already defined on the system as valid users. Notice that you can have several user names on the command line. You can also have several prototypes defined, but you use only one at a time with the `edquota -p` command.

Note

If you plan to remove a user from the system, you must run `/usr/sbin/edquota` to set that user's limits to **0** before actually removing the user from the system.

Set Quotas for an Individual User

Use the `/usr/sbin/edquota` command to set quotas for individuals. If you want to set quotas for user `mikkel`, that would allow him to have higher limits than the prototypical user, type:

```
/usr/sbin/edquota mikkel
```

When you are in the editor, enter a line such as:

```
fs /home blocks (soft = 1200, hard = 1500) inodes (soft = 300, hard = 350)
```

Time Limit Quotas

To set a **time limit** for users, use the `edquota -t` command. Time limits allow the users a specific amount of time to reduce their usage when they exceed the soft limits. Time limits apply file-system- wide and to *all* users in a given file system.

You can set one time limit for correcting a violation of the soft limit on the number of files created and a different time limit on correcting violation of the soft limit on the number of blocks used.

To set the time limits type:

```
# /usr/sbin/edquota -t
```

While in the `vi` editor, you should type in a specification of the time limits you desire. An example might be:

```
fs /home blocks time limit = 10.00 days, inodes time limit = 15.00 days
```

Then you should write (save) the text file, and quit the editor.

If you do not set these limits, the quota system defaults to seven days. If a user has exceeded his soft limit for more than the period specified by the time limit, the soft limit becomes the hard limit.

D-5. SLIDE: Changing Quotas

Changing Quotas

- Change the quotas using edquota

```
# edquota chris
```

- Bring the quotas file up to date, if the user has had no quotas before.

```
hfs usage: quotacheck [-F hfs] [-V] [-pPv] file system ...
       or  quotacheck [-F hfs] [-V] [-pPv] -a
vxfs usage: quotacheck [-F vxfs] [-V] [-pv] file system ...
        or  quotacheck [-F vxfs] [-V] [-pv]  -a
```

H6296 D-5 256

Student Notes

To change the quotas for a user you just have to change the quotas with the `edquota` command. But if the user has had no quotas before you have to use the `quotacheck` command to bring the `quotas` files up to date. Quotacheck is similar to `fsck`. It checks all the files in the file system for disk usage and the number of files for each user which has quotas.

Checking Consistency of Data

`quotacheck` examines each specified file system and builds a disk-usage table of user IDs and blocks used, which it compares to a table stored in that file system's `quotas` file. If inconsistencies are detected, both the `quotas` file and the kernel quotas table are updated. Inconsistencies typically develop if a system mounted with disk quotas has been used for a period of time with quotas enforcement suspended. Since `quotacheck` accesses the raw device in calculating the actual disk usage for each user, the file systems checked should be inactive when quotacheck is executed. This is most easily done in single-user mode. The options are explained below:

-a Obtain list of file systems to check from /etc/fstab. Only mounted rw and quota type file systems with the quota option are checked.

-v Indicate the calculated disk quotas for each user on a particular file system.

-p Check file systems in parallel as allowed by equal values in the pass number field in /etc/fstab.

-P (HFS Only). Preen file systems, checking only those with invalid quota statistics

During bootup the command is normally executed with the *-aP* options.

Example of the quotacheck command:

```
# quotacheck -v /
*** Checking quotas for /dev/vg00/rlvol1 (/)
chris       fixed:  files 1075 -> 1066  blocks 12508 -> 12405
```

D-6. SLIDE: Quota Status

Quota Status

Summarize quotas for a file system

```
# repquota /home

                 Block limits                    File limits
User       used  soft  hard   timeleft  used  soft  hard  timeleft

bill  --    59   100   200               24    30    40
fred  +-   199   100   200  1.7 weeks    10    30    40
dan   ++   173   100   200  1.4 weeks    32    30    40  1.4 weeks
```

Report Individual Usage

```
# /usr/bin/quota -v robert

Disk quotas for robert (uid 303):
File system usage quota limit timeleft files quota limit timeleft
/           12501 18000 40000            1077  1100  1500
```

H6296 D-6 257

Student Notes

Reporting File System Usage

The `repquota` command provides the system administrator with detailed quota information about file systems.

Reporting Individual Usage

A user can display his or her usage by using `quota`. Only the superuser can use the *user* option to view specific usage and quota information about other users.

Reporting Summary of Ownership

You can use the quot command to display the number of 1024-byte blocks in a file system (the quota are not necessary enable on the file system) that are currently owned by each user:

```
# /usr/sbin/quot -v /
/dev/vg00/lvol0:
17485   bin          2341    2341    2341
 5906   knud         2456    2454       2
 4775   root          206     203     199
 1922   dirk         1921    1921    1921
   16   daemon          0       0       0
    5   jg              5       5       5
    4   adm             0       0       0
```

The option -v show the three columns on the right. They contain the number of blocks not accessed in the last 30, 60, and 90 days.

D-7. LAB: Hands-On Disk Quotas

1. Set up a disk quota for the file system you added earlier in the class. Set quotas of 5 blocks and 3 inodes. Set limits of 10 blocks and 5 inodes.

D-8. REVIEW: Check Your Understanding

Directions

Write the answers to the following questions.

1. If you want to set quotas for a file system named `/home` where have you create the quotas file.

2. Why can't you directly edit the `quotas` file?

Solutions

1-8. REVIEW: Check Your Understanding

1. Describe the role of the System Administrator.

Answer:

The system administrator is responsible for setting up and maintaining the system.

2. What must the System Administrator understand in order to perform her/his duties appropriately?

Answer:

Not only must the administrator understand both hardware and software, but she/he must also understand the needs of the user community.

3. What are the three main categories of System Administration responsibilities?

Answer:

- Hardware responsibilities
- Software responsibilities
- Responsibilities to the Users

4. What are two items in the System Administrator's toolkit?

Answer:

- SAM
- Support Contract
- LaserRom
- Shell Scripts

Solutions

2-17. LAB: Hands-On with SAM

1. Log on and run SAM.

Answer:

/usr/sbin/sam

2. Select the Users and Groups Functional Area from the SAM Control window.

   Display a list of the users currently set up on your system.

   Use the Column editor to right-justify the Login Name column.

   Next, put all the entries in order by user ID.

   Now, display only the entries for users with user IDs of less than 100.

Answer:

Select Users and Groups

   Select Users.

   Select Columns from the View item on the menubar, then change the Justify column to Right for the first attribute, which is Login Name.

   Select Sort from the View menubar item, then choose Priority 1 for the User ID attribute.

   Select Filter from the View menubar item, then choose the operator Less Than and enter a Value of 100 for the attribute UID.

3. How much swap space is configured on the system you are using in class? (Hint: swap space is on disks)

Answer:

Choose Disks and File Systems , then choose Swap . Actual answers will vary.

4. How many local file systems are on the system you are using in class? (Hint: file systems are on disks).

Answer:

Choose Disks and File Systems , then choose Local File Systems . Actual answers will vary.

5. How many backup devices are configured on the system you are using in class?

Answer:

Choose Backup and Recovery , then choose Backup Devices . Actual answers will vary.

Solutions

6. Is there currently an Automated Backup schedule for this system? If so, what is the schedule?

Answer:

Choose `Backup and Recovery` , then choose `Automated Backups` . Actual answers will vary.

7. List three drivers that are configured in the kernel.

Answer:

Choose `Kernel Configuration` , then choose `Drivers` . Actual answers will vary.

8. Create an executable script in `/etc/sam/custom` to add a functionality to SAM. The script should show you who is doing what (Hint : use `/usr/sbin/whodo`). Then add your own script to the "Users and Groups" submenu. Verify the functionality.

Answer:

Create the `/etc/sam/custom/whodo` (and optionally a help file) containing:

```
/usr/sbin/whodo | /usr/bin/more
```

Start SAM, choose `Users and Groups`, then choose `Add Custom Menu Item` from the `Actions` menu. Fill necessary items in the task dialog.

Then try your new menu item. Verify the result of the command.

3-13. LAB: Hands-On Adding Users

1. Invoke `sam` and add a user to your system. (You must be superuser to invoke SAM.) Use your name as a user name. Assign the user to a group called `class` and give him or her the Posix shell.

Now, exit SAM and look at the `/etc/passwd` and `/etc/group` files. Do you see the user you added?

Answer:

Use `sam` to add a user.

2. Add a user to the system using HP-UX commands. This time, use your partner's name as the user name. (If you don't have a partner, pick any name.) Use a group called `class` and give the new user the C shell. The steps are provided below.

- Add entries to `/etc/passwd` and `/etc/group`.
- Create a home directory for the user and set the permissions and ownership correctly.
- Copy default set-up files to the home directory and set correct permissions and ownership.

Now, use the `login` command to log in as the user you created. Type `id` to verify your user ID.

Answer:

- Add entries to `/etc/passwd` and `/etc/group`.
- Create a home directory for the user and set the permissions and ownership correctly.
- Copy default set up files (`/etc/skel/.*`) to the home directory and set correct permissions and ownership.

3. Run the commands to check the integrity of the `/etc/passwd` and the `/etc/group` files. Discuss your findings with the instructor.

Answer:

Use `pwck` and `grpck` commands. You may want to modify the `/etc/passwd` and `/etc/group` files to cause `pwck` and `grpck` commands to show some output.

4. Add a user called `date` that executes the `date` command. What would happen if you tried to log in using the user name `date`?

Answer:

Entry in `/etc/passwd`:

```
date::92:1::/:/usr/bin/date
```

The command `date` is a valid login user name as indicated in the `/etc/passwd` file. `more` the `/etc/passwd` file and look at the *command* field (last entry) for these user names. For the user `who` the *command* is `who`. Therefore, the `who` command is executed when user `who` logs in. When the `who` command terminates, the number of processes attached to that session goes from one to zero causing logout.

5. Try changing the password on your own student account (not the one you just created) to your first name. Does this work? Why or why not?

Answer:

The command to change your password is: `passwd`. You will not be able to change it to your name because of the password naming restrictions insist on at least one non-alphabetic character and/or doesn't allow your login name as your password.

6. Create a news file that everyone on the system will be able to see, which contains your name and the name of your company. Use your name as the name of the file.

Answer:

```
cd /var/news
vi yourname
```

Solutions

3-14. REVIEW: Check Your Understanding

1. What steps is SAM taking when performing the following tasks.

   Adding a new user

   De-activating a user

   Modifying a user's information

   Adding a new group

Answer:

SAM is doing the following:

Adding a new user - Adding an entry to `/etc/passwd` and `/etc/group`, making a home directory, copying default setup files, and changing the owner and group of the login directory and default files.

De-activating a user - Replacing the user's password field (field two in the `/etc/passwd` file) with an asterisk .

Modifying a user's information - Modifying the appropriate information in the `/etc/passwd` file. Possibly changing the UID of the user's files.

Adding a new group - Adding an entry in the `/etc/group` file.

2. Describe the 7 fields of the `/etc/passwd` file

Answer:

- *user_name*:
- *encrypted password*:
- *user_id*:
- *group_id*:
- *comment_field*:
- *login_directory*:
- *startup_program*

3. Describe the fields of the `/etc/group` file.

Answer:

- *group_name*:
- *password*:
- *group_id*:
- *group_list*

4. What does it mean to set up groups? Explain.

Answer:

Groups possibly allow users access to files that other members of their group own. It allows sharing of data amongst group members.

5. List the steps to manually add a user.

Answer:

a. Add an entry to `/etc/passwd`

b. Add an entry to `/etc/group`

c. Create a home directory and set permissions to 755

```
# mkdir /home/bugs
# chmod 755 /home/bugs
```

d. Copy default setup files

```
# cp /etc/skel/profile /home/bugs/.profile
```

e. Change the owner and group of the home directory and default files to that of the new user

```
# chown bugs /home/bugs
# chgrp users /home/bugs /home/bugs/.profile
```

6. Describe two ways to create a user and send him straight into an application program.

Answer:

In the `/etc/passwd` file, you can specify a startup program in the seventh field, thereby by-passing a shell. Or you could give him a shell and in his `.profile` file you could run the application program and then use the `exit` command to terminate the shell immediately after he finishes with the application program, not allowing him shell access.

7. What is the difference between `/etc/profile` and `.profile`?

Answer:

`/etc/profile` is the system administrators profile file. Anything that the system administrator puts in this file will have an effect on all users, when they log in. The `.profile` file is the user's own profile file. Any command that he adds to this file will be executed when he logs in and *only* when *this* user logs in.

8. What is the `wall` command and how could it be useful?

Answer:

`wall` is used to write to all users at the same time and is useful when you have to warn all users that the system is going down in 5 minutes, and so forth.

9. If you were going to shutdown the system tonight at midnight, how might you communicate this to all of your users?

Answer:

Using the `/etc/motd` file. Anything that you put into the `/etc/motd` file will be displayed whenever a user logs into the system.

10. Explain why it is dangerous to put `.` ("dot") in root's `PATH` variable.

Answer:

Because it is a potential security breach.

4-21. REVIEW: Check Your Understanding

1. What is the function of an I/O interface (or device adapter)?

Answer:

Handles communication between external devices, such as disks and printers, and the CIO bus on CIO systems, or between external devices and the Precision bus on HP-PB systems.

2. What is an Access Port used for (Servers)?

Answer:

Allows a remote system console to be connected so that system administration functions can be performed by someone at a remote site. An Access Port is useful for customers who use HP Response Center support services.

3. How does a workstation extend I/O connectivity?

Answer:

A workstation extends the I/O beyond the core interfaces via an EISA card. Depending upon the model, a workstation can have 0, 1, or 4 EISA slots available. These slots can then be used for additional cards such as SCSI, LAN, Parallel, etc.

4. Use `ioscan` and/or `dmesg` to draw a diagram of your class system. Draw a line diagram similar to those you saw on the slides. Identify all interface cards, peripheral devices, and all addresses.

Answer:

Answers will vary in different Education Centers.

5. Draw a diagram showing the I/O interfaces and peripheral devices on the system that you administer or use at your own work site.

Answer:

Answers will vary.

5-18. LAB: Hands-On Creating Device Files

1. Use `lssf` to determine the options specified for all devices in the `/dev/rmt` directory.

Answer:

```
lssf /dev/rmt/*
```

2. Use the `ioscan` command to determine the current configuration of the system.

Answer:

```
# ioscan -f
```

3. Run the `ioscan -f` command to determine the current system configuration.

Answer:

```
# ioscan -f
```

4. Run the `lssf` command on all device files found in `/dev/rmt`. The `lssf` command can help you interpret the meaning of device file names.

Answer:

```
# lssf /dev/rmt/*
```

5. Use `ioscan` to help determine the device file name corresponding to the first tape drive on the system. (If the first tape drive is a 9-track, assume 1600 bpi. Assume no-rewind on close.)

Answer:

```
# ioscan -fn -I 0 -C tape
```

The correct device file would be `/dev/rmt/0mn`.

6. Create the device files for a dial-in modem for an available port on your system. (Assume U.S. modem protocol.)

Answer:

This example is for the fifth port on the first MUX

```
# mksf -d mux2 -I 0 -p 5 -i        creates /dev/cua0p5
# ln /dev/cua0p5 /dev/cul0p5      additional device file needed for uucp only
```

7. Remove all device files for a non-root disk drive. Recreate the device files without rebooting the system.

Answer:

```
# rmsf -H 52.6.0
# insf -e -H 52.6.0
```

or

```
# insf -e -I ? -d disc3
```

5-19. REVIEW: Check Your Understanding

1. What is a device file?

Answer:

A device file is used to associate a physical device with a logical file name.

2. What is the difference between a block and a character device file?

Answer:

The smallest unit that is exchanged during an I/O operation with a block device is a block (defined by `f_blksize` as returned by *ustat*(2)). The smallest unit that is exchanged during an I/O operation with a character device is a byte (eight bits).

3. Why do you need both block and character device files for disks?

Answer:

Because some commands like `mount` expect a block device and other commands like `newfs` expect a character device.

4. Describe the following device files:

a. `/dev/rdsk/c1t3d0s8`

b. `/dev/lp0`

c. /dev/mt/3mn

d. /dev/tty1p3

Answer:

a. A character device file for a disk with bus address 3 at card instance 2, section 8.
b. A character device file for a line printer at card instance 0.
c. A magnetic tape drive. It is a block device file at card instance 3, for a medium speed (1600 BPI) drive and no rewind.
d. A character device file for a terminal at card instance 1, port 3.

5. What are major and minor numbers?

Answer:

Major numbers describe the kind of device and the device driver. Minor numbers describe the location of the device and device dependent specialties.

6. What do the insf and mksf commands do?

Answer:

They create device files based on the information in /etc/ioconfig.

7. What is the difference between the ll and the lssf commands related to device files?

Answer:

ll lists major and minor numbers in their "encrypted" form, whereas lssf decrypts them into a readable form.

8. What does the lsdev command do? What does the dmesg command do?

Answer:

The lsdev command gives a listing of all device drivers and the corresponding block and character major numbers. The dmesg commands gives us some information from the boot process such as the amount of memory and some device address information.

9. How can the ln command be useful in terms of device files?

Answer:

The ln command can be used to create more meaningful device file names (such as data_disk or printer) in addition to conventional names like c3t4d0 or lp2. The use of ln (rather than mv) leaves the original device file names intact.

6-13. REVIEW: Check Your Understanding

1. What is Stable Storage and what is stored there?

Answer:

Stable Storage is static RAM, a non-volatile type of memory that retains data even if power to the computer is shut off. It stores information about the Primary Boot Path, the Alternate Boot Path and the Console Path. The contents of Stable Storage can be changed at the ISL prompt.

2. Describe two commands that can be executed at the ISL prompt, besides hpux.

Answer:

`primpath` allows you to modify the primary boot path.

`conspath` changes the console path.

`altpath` changes the alternate boot path.

`autoboot` allows you to toggle autoboot on and off.

3. How would you boot your system if your regular `/stand/vmunix` kernel will not boot? Write the command to boot your system.

Server: Assume your regular kernel is located on the root disk with hardware path 52.6.0, which is also the primary boot path.

Workstation: Assume your regular kernel is located on the root disk on the scsi disk at address 6, which is also the primary boot path.

Answer:

Server:

```
   ISL> hpux (52.6.0;0)/stand/vmunix.prev
or ISL> hpux (;0)/stand/vmunix.prev
or ISL> hpux (52.6.0;0)stand/vmunix.prev
or ISL> hpux (;0)stand/vmunix.prev
```

Workstation:

```
   ISL> hpux(scsi.6.0;0)/stand/vmunix.prev
or ISL> hpux (;0)/stand/vmunix.prev
or ISL> hpux (scsi.6.0;0)stand/vmunix.prev
or ISL> hpux (;0)stand/vmunix.prev
```

4. What would you do if you forgot the root password?

Answer:

Boot the system to single-user mode by:

- Cycle power (or reset). Would you do this when the system is busy, or would you want to have all users logged off?
- Manually boot the system to ISL. Type hpux `-is` *bootpath*. This will boot the system to single-user mode.

- Then change the root password, and bring the system up to multi-user mode by typing `init` *n*, where *n* is the run level you wish to bring the system up to.

7-21. REVIEW: Check Your Understanding

1. How is the run-level of the system controlled?

Answer:

During the bootup procedure by the `initdefault` entry in `/etc/inittab`, later by the `init` command.

2. What is the purpose of the `/etc/inittab` file?

Answer:

`/etc/inittab` serves as a data base for the `init` command. Each line corresponds to a process which is dispatched by `init` during the system bootup or at run-level changes.

3. What is the difference between `init s` and `init S`?

Answer:

`init s` restricts the system access to the system console, whereas `init S` restricts to the terminal this command was invoked from.

4. Describe what the `/sbin/rc` file does.

Answer:

This file is an initialization program that is scheduled by `init` to initialize many system features such as networking, turning on swapping, disk quotas, and so forth.

5. Determine the default run-level for the system you are using.

Answer:

```
# grep initdefault /etc/inittab
```

6. Determine the current run-level for the system being used. How many times has the current state been entered previously? What was the previous state of your system?

Answer:

```
# who -r
```

7. During the normal system startup process, which run-levels does the `/sbin/rc` file run through?

Solutions

Answer:

The `init` process reads the `initdefault` value in `/etc/inittab` and starts with run-level `S`. Then the system is set to run-level 2 (the `initdefault` value) by `/sbin/init`. `/sbin/rc` is run by the `/etc/inittab` entry and first performs all startup (S*xxx*) tasks of run-level `0`, then all of run-level `1`, and last all startup tasks of run-level `2`.

8-7. LAB: Hands-On with `shutdown`

1. Shut down your system immediately (0 seconds) to single-user mode using the `shutdown` command. Look at what processes are still running. Then reboot your system with the `reboot` command.

Answer:

```
# cd /; shutdown 0
# ps -ef
# reboot
```

2. After rebooting in the previous exercise, use `init s` to shut down your system. Look at what processes are still running. How does this compare to the processes that were running after using the `shutdown` command in the previous exercise? Reboot your system. (`reboot` or `shutdown -r 0` will work.)

Answer:

Nearly all system processes, like `inetd`, are still running. The system is still in the network and therefore not in a single user state, where the only access to the system is via root's terminal or console.

3. Customize the shutdown procedure by modifying the lab scripts and putting them in the appropriate directories. A good idea could be to `touch` a file in /tmp, showing that startup/shutdown worked correctly.

Answer:

Modify the scripts and put them in `/sbin/init.d` and in `/etc/rc.config.d`. Create a `K` (Kill) link in `/sbin/rc1.d`.

4. Set the system up so that your user name can execute the `shutdown` command.

Answer:

Add your user name to the file `/etc/shutdown.allow`.

1. Using the `simulate` command provided by the instructor, run the following simulation:

```
3. Shutting Down Your System
```

Answer:

Complete the simulation as directed.

8-8. REVIEW: Check Your Understanding

1. While in multi-user mode, why should the `shutdown -h` command be used, rather than `reboot -h` for halting the system?

Answer:

`shutdown` allows you to warn users. Also, `shutdown` shuts down more slowly and more gracefully than `reboot`. `shutdown` uses `kill -15` to kill processes, which lets processes terminate naturally within a grace period, whereas `reboot` uses `kill -9` which kills all processes very quickly allowing them no grace period.

2. What does `reboot -n` do, and why would you want to do this?

Answer:

`reboot -n` does *not* do a `sync` before halting or rebooting the system. After running `fsck`, you usually want to avoid the copying the corrupt buffer cache contents onto the file system you just repaired.

3. Once the system is in an `s` (or `S`) run-level (after execution of `shutdown`) , what is the difference between `init 2` and `reboot`? Assume an `initdefault` entry of 2 in `/etc/inittab`.

Answer:

`reboot` allows you to boot an alternate kernel and to interact with the ISL before the system comes up in run-level 2; `init 2` doesn't. Usually, the effects of `init 2` can be seen as a subset of the effects of `reboot`.

4. If the System Administrator wants to customize the shutdown process, what should be done?

Answer:

Add additional shutdown scripts into the directory `/sbin/init.d` and create links to it in the run-level's directory in `/sbin/rc`*R*`.d/`. The scripts in this directory will be executed each time the `shutdown` command is executed. These scripts may be scripts used to safely close data base files or smoothly shut down other user applications.

5. Is the system administrator the only person who can shut down the system? If not how can this capability be given to another user?

Answer:

No, the system administrator can give shutdown capability to other users by adding their names into the `/etc/shutdown.allow` file.

9-17. REVIEW: Check Your Understanding

1. List the types of file systems available on an HP-UX system.

Answer:

- HFS file system
- JFS file system
- CD-ROMfile system
- NFS file system

2. List the dynamic information in an HFS superblock.

Answer:

- Total number of free data blocks
- Total number of free inodes
- File system clean flag

3. Describe a cylinder and a cylinder group.

Answer:

A cylinder is a collection of tracks formed as the head-disk-assembly positions all the heads on multiple platters at the same distance from the edge of the disk surface.

A cylinder group is a group of one or more disk cylinders.

4. List the contents of the cylinder group information.

Answer:

- Number of inodes and data blocks
- Pointers to the last used block, fragment, and inode
- Number of available fragments
- Used inode map
- Free block map

5. Describe the inode table.

Answer:

The inode table is physically located immediately after the cylinter group information structure on the same cylinder group. Each inode table entry is 128 bytes long.

6. List the parts of an inode.

Answer:

- mode or permissions of the file
- type of file (that is, regular, directory, special)
- number of hard links to the file
- current owner of the file
- group associated with the file
- actual file size in bytes (more may be allocated as the file grows)
- time stamps relating to file activity
  - time/date of last file data change
  - time/date of last file access
  - time/date of last inode modification
- disk addresses, or pointers to disk addresses, where the file's data is stored

7. Explain three different ways that an inode references a data block.

Answer:

8. Explain fragment and block allocation.

Answer:

A data block is divided up into fragments, where the fragment size is 1/8, 1/4, 1/2 or 1/1 of the block size. The individual fragments within one block form the smallest logical unit that can be accessed by HP-UX. When allocating blocks and fragments for a file:

- A fragment will only be allocated at the end of a file (in other words, full blocks are allocated until less than a full block is left).
- A file will use fragments contained within one block.
- A block may contain fragments from more than one file.
- Multiple fragments can be used by a file, but only at the end of the file, and they must be contiguous within one block.

9. Explain the link process.

Answer:

When the `ln` command is executed it creates a directory entry that has the new file name (the target name) associated with the same i-number as the original file. Both directory entries are then said to "point to the same file". A field in the inode is incremented to indicate the presence of the link. Only after the link count is reduced to zero does the actual data and inode get recycled.

10-9. LAB: Hands-On with Logical Volume Manager

1. Assume that you have added a new SCSI drive to your system. The Instance is 1 and the Target is 2. You already have one volume group, and you now plan to set up a second volume group consisting of this disk. Write the steps you would follow and commands you would use to do this.

Answer:

1. Make the disk a LVM disk (physical volume):

```
pvcreate /dev/rdsk/c0t2d0
```

2. Create the volume group directory:

```
mkdir /dev/vg01
```

3. Create the control file group:

```
mknod /dev/vg01/group c 64 0x010000
```

4. Create the volume group:

```
vgcreate /dev/vg01 /dev/dsk/c0t2d0
```

5. Verify the creation of the volume group:

```
vgdisplay -v /dev/vg01
```

2. Create a logical volume of 50 megabytes in the volume group `/dev/vg01` and name it *yourname*`fs`.

Answer:

```
lvcreate -L 50 -n yournamefs /dev/vg01
```

3. What command would you use to display all the volume groups on your system?

Answer:

```
vgdisplay -v
```

4. What command would you use to display all the logical volumes on your system?

Answer:

`lvdisplay` *path*

10-10. REVIEW: Check Your Understanding

1. List two benefits of using LVM.

Answer:

a. You can specify the size of a file system based on any need, or expand a file system by adding disk space anywhere on your system.

b. Data can exceed the capacity of a single disk.

c. Disk mirroring is available with LVM

2. Differentiate between a volume group and a logical volume.

Answer:

A volume group is a group of physical disks. A logical volume is a "virtual disk", a distribution of space within a volume group. A logical volume can span more than one disk, or represent only a portion of a disk.

3. What are the two reserved areas on a non-bootable disk?

Answer:

PVRA (Physical Volume Reserved Area) and VGRA (Volume Group Reserved Area). The VGRA is organized into VGDA (Volume Group Descriptor Area, VGSA (Volume Group Status Area, and MCR (Mirror Consistency Record).

4. What two areas are unique to bootable disks?

Answer:

BDRA (Boot Data Reserved Area) and LIF (Logical Interchange Format).

5. What command do you use to initialize a disk as an LVM physical disk?

Answer:

`pvcreate`

6. What are the steps for creating a volume group?

Solutions

Answer:

a. Select disks for the volume group
b. Make the disks physical volumes (`pvcreate`)
c. Create a control directory for the volume group (`mkdir`)
d. Create a "group" file in the volume group directory (`mknod`)
e. Create the volume group (`vgcreate`)

7. Write the commands you would use to perform these tasks:

a. Make a disk an LVM disk
b. Create a volume group
c. Create a logical volume

Answer:

a. `pvcreate`
b. `vgcreate`
c. `lvcreate`

11-8. LAB: Hands-On, Creating a File System

1. Run SAM. Choose **`Disks and File Systems-->`** on the Control Box. Choose **`File Systems`** from the Functional Area list. Then, choose the **`Add Local File Systems...`** Action from the menu bar and **`Using the Logical Volume Manager`** from the submenu.

Within SAM, perform the following sequence of steps. As you carry out each step, SAM asks you to supply certain information. From the nature of this information, can you deduce which command SAM is invoking at each step?

1. Enter a mount directory.
2. Choose `HFS File System` and `Modify File System Defaults`.
   a. Choose the **`Now`** option under **`When to Mount`**.
   b. Choose the **`Every System Boot`** option under **`When to Mount`**.
   c. Turn on the `Create new file system` check box and activate (OK).

Answer:

1. SAM makes the specified mount directory if it does not already exist:

   `mkdir` *mountdir*

2. Uses the `mount` command when the screen is complete
3. Adds an entry for this file system to `/etc/fstab`

4. SAM uses newfs to create the file system in the logical volume or disk partition specified, mounts it if you specified "Now", and adds an entry to /etc/fstab if you specified to mount at every bootup.

2. Create a file system using the newfs command on the partition assigned to you by your instructor.

Answer:

```
# newfs /dev/vg01/rlvol1
```

3. Create a directory called /disk*n*, if it does not already exists, where *n* is the number of your group. Mount the newly created file system to that directory. Then copy the file /etc/passwd to the "root" directory of your new file system.

Answer:

```
# mkdir /disk1
# mount /dev/vg01/lvol1 /disk1
# cp /etc/passwd /disk1
```

4. Unmount your file system. Can you access your copy of /etc/passwd? Why or why not?

Answer:

```
# umount /disk1
```

5. Edit the /etc/fstab file and add a line that will automatically mount your newly created file system at boot time. If you are the working together with other groups on the same system, make sure that you are the only one who edit the fstab file.

Answer:

```
# ex /etc/fstab
:a
:/dev/vg01/lvol1   /disk1   hfs  defaults 0 5
:.
:wq!
```

6. What happens if you mount your new file system to a directory with files in it? How can you recover?

Answer:

The files that were there are not accessible, in fact, they don't even appear when doing an ls -a. This can be corrected by unmounting the file system.

7. You accidentally destroyed an important file /home/susan/goodfile. Although a backup tape is *not* available, you do have another disk which is an exact replica of your system disk (before /home/susan/goodfile was removed). List out the steps to restore this file from the HFS file system on your auxiliary disk.

Answer:

The steps are as follows:

```
# mkdir /newdisk
# mount /dev/vg01/lvol1 /newdisk
# cp /newdisk/home/susan/goodfile /home/susan/goodfile
# umount /newdisk
```

12-9. LAB: Hands-On Maintaining the File System

1. For the system you are using in class, on which file systems would the command `fsck -F hfs -p` perform a file system check? In what order would these file systems be checked? Why is the order potentially significant ?

Answer:

The command `fsck -F hfs -p` would search the `/etc/checklist` file for the file systems to be checked. The last column in the `/etc/checklist` file indicates the *pass number* or the order in which to check the file systems. The root file system should have a pass number of 1. All other file systems should have a higher pass number. The significance of this is that file systems on different disk drives can be checked in parallel if their pass numbers are equivalent in the `/etc/checklist` file. This can save quite a bit of time depending on the configuration.

2. Run `fsck` on the file system you created and mounted earlier.

Answer:

```
# umount /dev/vg01/yourfilesystem
# fsck /dev/vg01/ryourfilesystem
```

12-10. REVIEW: Check Your Understanding

1. Describe the buffer cache

Answer:

The buffer cache is an area of physical memory that is used as a buffer between user memory space and disk space. This buffer cache needs to be flushed before shutting down the system.

2. What are the advantages of the buffer cache

Answer:

Speed, allowing processes to execute faster when transferring data to and from the disk. Re-use of data files that are stored in the buffer cache can save time when accessing the file's data.

3. What one disadvantage of the buffer cache does the System Administrator have to worry about?

Answer:

The fact that the buffers need to be flushed to the disks before shutting down the system. If not, `fsck` needs to be run to fix the file system that may have been corrupted.

4. What does `fsck` do with a HFS File System?

Answer:

`fsck` checks the HFS file system. That is it will attempt to fix all of the problems that it finds on each file system that it is run on. Most of the time `fsck` does a complete job of fixing the disk. Every once in a while the System Administrator may have to run `fsck` in the interactive mode to fix certain problems that it would not fix in preening mode.

5. What does `fsck` do with a JFS File System?

Answer:

Instead of read all the file system (like with an HFS file system) `fsck` reads the intent log and repairs the file system from it. The JFS is a Fast File System Recovery.

6. What is the `lost+found` directory used for?

Answer:

It is used by `fsck` to store files or pieces of files that it cannot find a home for. That means no directory entry points to the inode.

Solutions

13-13. LAB: Hands-On File System Management

1. You have just run out of space on one of the disks. Use SAM to search for files larger than 50000 bytes, with a "modification since" time of 1 day, in the `/usr/bin` directory. Also search the `/etc` directory for core files.

Answer:

Run SAM. From the Control Box choose **`Routine Tasks`**. From the Routine Tasks Functional Area, choose **`Disk Space Recovery`**. Choose **`Selective File Removal`**. Skip the initial search. Choose "Actions" from the menubar, and choose "Modify Search Criteria." Set Beginning of Search Path to the `/usr/bin` directory; Size to 50000, and Time Since Last Modification to 1. Then activate (OK).

Choose "Actions" from the menubar, and choose "Modify Search Criteria." Choose (Search For) and select (Core File Only), Set "Beginning of Search Path" to the `/etc` directory; Then activate (OK).

2. Copy the file `/var/adm/sw/swinstall.log` to your home directory and add it to SAM's list of ASCII log files to check. Have SAM check the logfiles on the system, and trim *your* `swinstall.log` file. Keep 25% of the file.

Answer:

Run SAM. From the Control Box choose `Routine Tasks`. From the Routine Tasks Functional Area, choose **`Disk Space Recovery`**. From the **`Disk Space Recovery`** functional subarea, choose **`System Log File Trimming`**. Choose "Actions" from the menubar, and choose "Add To List". Add your filename, and a recommended size of 10000 bytes. Now, highlight your `swinstall.log` filename, and then choose "Actions" from the menubar, and choose "Trim/To a Percentage." Specify 25% and activate (OK).

3. Add one disk, if it is available. And extend the volume group specified by your instructor.

Answer:

- Halt system

```
# shutdown -hy
```

- Connect disk
- Check Bus address!
- Power on disk(s) and system
- Verify the device instance and the special files

```
# ioscan -f -C disk      example with instance=1 and disk address=3
# ll /dev/dsk/c1t3d0
```

- Make it a physical volume

```
# pvcreate /dev/rdsk/c1t3d0      Create physical volume
```

- Unmount the logical volumes included in the volume group

```
# umount /dev/vg01/lvolm
```
m = 1,2,3 … = logical volumes number

- Extend the volume group

```
# vgextend /dev/vg01 /dev/dsk/c1t3d0
```

- Mount all the logical volumes

```
# mount /dev/vg01/lvolm
```
m = 1,2,3, … = logical volumes number

4. Extend one logical volume specified by your instructor. Add 50 Mb.

Answer:

Let's assume the logical had 50 Mbytes and we want to extend it to 100 Mbytes.

```
# umount /dev/vgnn/lvolm
# lvextend -L 100 /dev/vgnn/lvolm
# mount /dev/vgnn/lvolm
#extendfs /dev/vgnn/yourfilesystem
```

13-14. REVIEW: Check Your Understanding

1. List and define two commands to monitoring the free disk space on the system

Answer:

`bdf` and `du` are easy ways to monitor the disk file system loading. You can also use the `df` or the `diskusg` commands.

2. What are some different solutions to recover space on your file system ?

Answer:

- Moving data from one disk to another
- Extend file systems
- Move a volume group from one system to another
- Remove unused software

3. Give two HP-UX commands which enable you to remove some file sets?

Answer:

`swremove` and `diskfree` can help you to remove some file sets.

Solutions

14-11. LAB: Hands-On, Enabling Swap

1. How much memory does your training system have? How much is lockable, avail, and real?

Answer:

Use the `dmesg` command to see `real mem`, `avail mem`, and `lockable mem`.

2. Add your logical volume of 50mb as swap space.

Answer:

```
lvcreate -L50 -n lswap /dev/vg01
```

3. Add a free disk partition as swap. Use either SAM or HP-UX commands. Also, make sure that this partition will be enabled as swap each time the system is rebooted.

Answer:

a. Add something like the following to `/etc/fstab`:

```
/dev/vg01/rlswap defaults   swap   defaults  0 0
```

b. Execute `swapon -a` to enable this swap area.

c. Execute `swapinfo` to see the new swap space.

4. What is the command to add a file system swap in a mounted file system?. How would you make sure that this swap will be enabled each time the system is rebooted?

Answer:

1. Add the something like the following to `/etc/fstab`:

```
/dev/vg01/lvol3 /var/paging   swapfs   min=0,lim=1000,pri=2 0 0
```

2. Execute `swapon -a` to enable this swap area.

14-12. REVIEW: Check Your Understanding

1. Define physical memory and virtual memory.

Answer:

Virtual address space is all valid memory addresses; whereas **physical address space** is the actual RAM purchased.

2. Define physical memory, available memory, and lockable memory.

Answer:

Physical memory is the actual RAM purchased. That physical memory is divided up for different uses by the MMU. Some memory is reserved for kernel code and data structures. The amount of memory remaining is referred to as available memory, and is used by the system for demand paging. All or part of available memory can be locked by a subsystem or by user processes. Lockable memory cannot be swapped out to disk. Typically, locked memory holds frequently accessed programs or data structures.

3. Explain the paging process?.

Answer:

paging involves moving smaller units (called pages) between RAM and mass storage.

4. What are the two types of swap space that a system administrator can set up?

Answer:

Device swap and file system swap.

5. What are the two types of device swap?

Answer:

Primary and secondary swap.

6. What are the advantages of device swap?

Answer:

Faster.

7. What are the advantages of file system swap?

Answer:

Less disc space totally committed to swap space.

Solutions

8. Name the two methods for allocating swap space?

Answer:

Manual method, or SAM.

9. Describe two guidelines for setting up device swap.

Answer:

1. Two swap areas on different discs better than one larger swap space.
2. Swap areas on different disc should be of the same size.

10. Describe how to set priorities for swap.

Answer:

Select device swap over file system swap.

Select faster devices over slower devices.

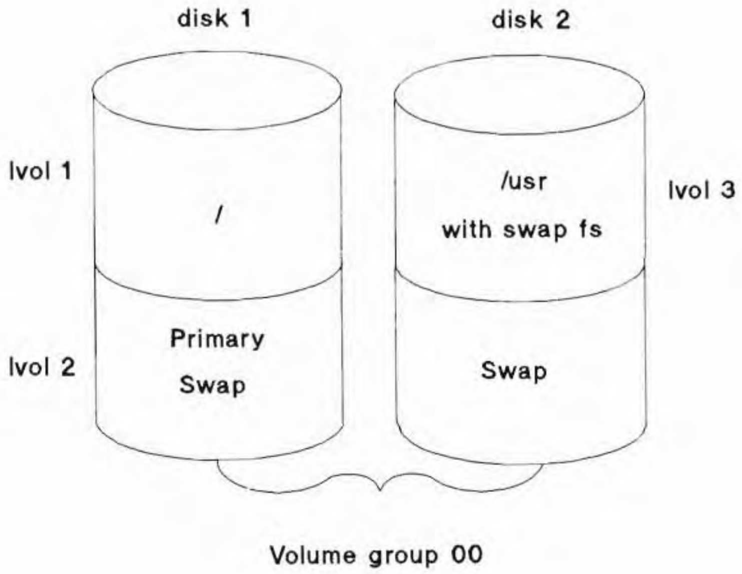
Select infrequently used file systems over frequently used file systems.

11. What is the name of the file used to automatically setup either device or file system swap?

Answer:

`/etc/fstab`.

12. What would the `/etc/fstab` file look like for these disks?



Answer:

```
/dev/vg00/lvol1  /      hfs    defaults  0 1 #root fs
/dev/vg00/lvol2  ....   swap   0         0 0 #primary swap
(optional)
/dev/vg00/lvol3  /usr   hfs    defaults  0 2 #/usr fs
...............  /usr   swapfs res=12500 0 0 #/usr fs, file
system swap
/dev/vg00/lvol4  ....   swap   0         0 0 #device swap
```

A zero in the field indicates it doesn't matter or is unused.

15-10. LAB: Hands-On, Reconfiguring the Kernel

1. Move to the `/stand` directory. Move the current `/stand/system` to `/stand/system.`*your_name*. Then, create a new `/stand/system` file that shows the current system configuration.

Answer:

```
# cd /stand
# mv /stand/system /stand/system.your_name
# /usr/lbin/sysadm/system_prep
```

2. Now, edit `/stand/system` and add the appropriate entry for a CD-ROM file system if one is not already present. Also you should make sure your system can access the CD-ROM disk interface.

Answer:

```
# vi /stand/system
```

Add the following to `/stand/system`:

```
cdfs
```

Then write the file and exit the "vi" editor.

3. Now, perform the commands to build a new kernel. Once again, assume the hardware has already been added to the system.

Answer:

```
# cd /stand
# mk_kernel
# mv /stand/vmunix /stand/vmunix.prev
# mv /build/vmunix_test /stand/vmunix
# cd /
```

```
# shutdown -r 0
```

4. Invoke SAM. Choose the `Kernel Configuration` Functional Area from the Control Window. Look through SAM's options on kernel configuration, but don't reconfigure your kernel with SAM.

Answer:

Invoke `sam` and then choose `Kernel Configuration`.

5. Add the appropriate entry for the parallel interface.

Answer:

`CentIf`

16-14. LAB: Hands-On, Backing Up the System

1. Create a graph file in `/var/adm/fbackupfiles` specifying the inclusion of the following files in your backup:

   include all files and dirs underneath `/var/spool/lp/model` EXCEPT `thinkjet` and `remote`

   exclude all files and dirs underneath `/etc`, except `/etc/passwd`

   exclude the tape device file in `/dev/rmt`

Answer:

```
/var/adm/fbackupfiles/graphfile:

i /var/spool/lp/model
e /var/spool/lp/model/thinkjet
e /var/spool/lp/model/thinkjet
e /etc
i /etc/passwd
e /dev/rmt/5m
```

2. Perform a level 0 backup of the designated files to your tape drive. (Remember to use the -u option of `fbackup` to record the timestamp of the backup.)

Answer:

```
#fbackup -f /dev/rmt/5m -0vug /var/adm/fbackupfiles/graphfile
```

3. Make a modification to `/etc/passwd`.

Answer:

```
#vi /etc/passwd
```

4. Recover a copy of the `/etc/passwd` file into the `/tmp` directory. Make sure that you DON'T overwrite the newer copy of `/etc/passwd` which you modified in the step above.

Answer:

```
#cd /tmp
#frecover -f /dev/rmt/5m -xFv -i /etc/passwd      will restore as /tmp/passwd
#frecover -f /dev/rmt/5m -xXv -i /etc/passwd      will restore as /tmp/etc/passwd
```

5. Perform a level 1 backup of the files included in your graph file to your simulated tape drive. How many files were backed-up this time?

Answer:

```
#fbackup -f /dev/rmt/5m -1vug /var/adm/fbackupfiles/graphfile
```

At least two files will be backed-up, the modified `/etc/passwd` files and the "new" restored version of `/etc/passwd` to the `/tmp` directory.

6. Use the `find` and `cpio` commands to backup the directory `/usr/share/lib/term` to your simulated magtape. Now look at the "tape" table of contents. Restore these files from magtape to the directory `/cpio_files`. Verify that the restoration was successful. Then remove these files. Remember to use the relative form of the `find` command when performing the copy.

Answer:

```
#mkdir /cpio_files
#cd /usr/share/lib/term
#find . | cpio -ocv > /dev/rmt/5m  # copy files to tape
#cpio -itc < /tmp/0m               # tape table of contents
#cd /cpio_files
#cpio -icdumv < /tmp/0m            # restore files
#ls
#cd
#rm -r /cpio_files
```

7. Use the `tar` command to backup the directory `/usr/share/lib/term` to your simulated magtape. Now look at the "tape" table of contents. Restore these files from magtape to the directory `/tar_files`. Remember once again to use the relative form of the `tar` command and to remove the restored files when you are done.

Answer:

```
#mkdir /tar_files
#cd /usr/share/lib/term
```

```
#tar cvf /tmp/0m .
#tar tvf /tmp/0m
#cd /tar_files
#tar xvf /tmp/0m
#ls
#cd
#rm -r /tar_files
```

17-7. LAB: Hands-On, Scheduling Jobs

1. Time schedule a command which runs at lunch time and echoes out to your terminal - "Lunch time - Where are we going?"

Answer:

```
at 12:00 Return
/usr/bin/echo "Lunch time - Where are we going?" > /dev/tty0p1
Ctrl+d
```

2. Time schedule the date program to execute every minute with the display coming to your screen. (Hint: Make sure you redirect the standard output to your screen device.)

Answer:

The crontab file should look something like this:

```
* * * * *   /usr/bin/date > /dev/tty0p1
```

3. Now add another process to your time scheduled list. Have this one execute a program to display to your screen who is logged in. This program should run every 10 minutes.

Answer:

The crontab file should now look something like this:

```
 * * * * * /usr/bin/date > /dev/tty0p1

0,10,20,30,40,50  * * * *  /usr/bin/who > /dev/tty0p1
```

17-8. REVIEW: Check Your Understanding

1. What type of activities might be useful to schedule with `cron`?

Answer:

Backups, cleanup type activities, regularly run large CPU intensive programs.

2. When will the following `crontab` entry be run?

```
2 * * * * /usr/bin/who >> /var/adm/who_log
```

Answer:

Every day, every hour at two minutes after the full hour.

18-14. LAB: Hands-On Adding Printers

1. If you have a printer available, use SAM to configure the printer and enable it. Then, exit SAM and print a file like `/etc/passwd`. Finally, use SAM to remove the printer.

Answer:

2. Determine the status of the LP scheduler. Shut it down if it is running.

Answer:

```
# lpstat -t
# lpshut
```

3. Add the printer with the `lpadmin` command by specifying the name, device file, and model.

Answer:

```
# lpadmin -pmy_printer_name -v/dev/tty0p1 -mmodel_name
```

4. Assign the default destination to the printer.

Answer:

```
# lpadmin -ddefault_printer
# lpstat -t    # to check default printer
```

5. Allow the scheduler to accept requests for your new printer.

Answer:

```
# accept my_printer_name
```

6. Enable the scheduler to print to your new "printer."

Answer:

```
# enable my_printer_name
```

7. Disable your printer, then send a file to the printer. Determine if the files are queued. Now cancel one of the requests from the queue. Enable the printer. What happens?

Answer:

```
# disable -r reason printername
# lp /etc/passwd
# lp /etc/group
# cancel print_request
# enable printername
```

8. Remove your printer.

Answer:

```
$
#lpshut
# lpadmin -x printername
```

18-15. REVIEW: Check Your Understanding

1. What functions does the spooling system provide, and why are they required?

Answer:

The spooling system manages print requests. They are required for optimizing the usage of limited system resources.

2. Which of these functions are available to the administrator and which to normal users?

Answer:

Functions like `lp`, `lpstat`, `enable`, `disable` and `cancel` can be executed by a normal user, the other commands by the LP administrator only.

3. What is the difference between a device, a printer and a destination?

Answer:

A device refers to the physical device and its related device file. A printer is a logical name to represent a physical device within the spooling system. A destination is the name of a spooling queue. Printers, but also classes are destinations.

4. Is it possible, to install two printers for one device?

Answer:

Yes. And it might be useful, too. When you regularly intend to print on two different types of paper (for example, preprinted form sheets and normal blank paper), having only one print device, address each print request to the appropriate printer (for example, "prform" and "prnorm"). Both printers are related to the same device, but only one of them can be `enabled` at the same time. The other collects its print requests until you change the type of paper. Prior to the change `disable` both, after the change `enable` the one waiting so far.

5. How would you cancel your own print request?

Answer:

`cancel` *request_id_number*

6. How would you cancel a print request owned by someone else?

Answer:

`cancel` *request_id_number*

7. How would the owner of a canceled request know that you had canceled it?

Answer:

By a mail message, automatically generated by `cancel`.

8. What is an interface program?

Answer:

A link between the printer and its device file.

9. If you have stopped the scheduler (by `lpshut`), does print-out continues?

Answer:

No, seen apart from the actual contents of the internal printer device buffers.

10. If you have stopped the scheduler (by `lpshut`), can you still use the `lp` command to add print requests to the queues?

Answer:

Yes.

11. How can you tell other users that a printer is "broken"?

Answer:

By `disable -r"Printer is broken"` *printer*

12. How can you redirect print requests from the "printer1" queue to the "printer2" queue?

Answer:

By `lpmove printer1 printer2`.

19-12. LAB: Hands-On Terminal Configuration

1. Use SAM to answer this question. Invoke SAM, then select `Peripheral Devices ->` from the Control Box, then select `Terminals and Modems` from the Peripheral Devices functional area. Choose the "Add Terminal" action from the Terminals and Modems list screen. If you were adding a terminal, what information would you need to provide so that SAM could successfully add the terminal?

Answer:

(servers) : Port number, LU, and baud rate.
(workstations) : Hardware path, possibly Port number and baud rate.

2. Perform all the necessary steps to configure a terminal using HP-UX commands. If you are using a Server, assume you are adding a terminal to port 5 on logical unit 0. The only step you will not be able to perform is rerunning `init`. If you are using a workstation, assume you are adding a terminal to port 0 on Card Slot 204.

Answer:

For Servers:

```
# lssf /dev/tty0p5
# more /etc/gettydefs     (verify label for terminal)
# vi /etc/inittab         tt05:2:respawn:/usr/sbin/getty -h tty0p5 9600
```

For Workstations:

```
# lssf /dev/tty0p0
# more /etc/gettydefs     verify label for terminal
```

```
# vi /etc/inittab      tt05:2:respawn:/usr/sbin/getty -h tty0p0 9600
# vi /etc/ttytype      2392 tty0p0 myterm
```

19-13. REVIEW: Check Your Understanding

1. When you add a terminal to your system, what file should you modify to start the `getty`?

Answer:

`/etc/inittab`

2. When adding a dial-in and dial-out modem, which files need to be modified?

Answer:

For dial-in edit `/etc/inittab` and confirm speed entry exists in `/etc/gettydefs`. For dial-out edit `/etc/uucp/Devices` and confirm entry exists in `/etc/uucp/Dialers`.

21-9. LAB: Hands-On, Installing HP-UX

1. If there is time enough, your instructor will provide installation materials and you can perform an actual installation

Answer:

22-10. LAB: Hands-On, Update Using `swinstall`

1. Perform an update using the media from your instructor. Update only those file sets she/he named. Installing other file sets may disrupt normal system operation or require a system reboot.

Answer:

- If possible, shutdown your system to single user mode
- Prepare for the update:
  - Read the *READ ME FIRST* document
  - Read the *HP-UX Release Notes*
  - Note the device file name for your update device

- Ensure that the TERM environment variable is set correctly
- Clean up the file system, if the system is in Single User Mode and this is possible
- Check free disk space
- Save a copy of the Kernel in `/stand/vmunix`

- Check whether `swagentd` is running. If not, start it with `/sbin/init.d/swagentd start`.
- Insert media; mount CD-ROM, write protect DDS/DAT media
- run `swinstall`
- Select products and file sets
- Check the `Options` if you are updating software with the same revision
- Start the update
- View Logfile during update to see what's going on
- Wait until `swinstall` has finished.
- Check the logfile in `/var/adm/sw`
- If possible, remove the files listed in the `cleanup` file.

C-5. REVIEW: Check Your Understanding

1. Describe how a DTC is controlled. What occurs at bootup?

Answer:

The DTC is controlled by either a PC or HP 9000. If controlled by and HP 9000, there are two daemons needed, `/usr/sbin/rbootd` for download and `/usr/sbin/dtcnmd` for diagnostics.

When a DTC receives power it performs a self-test and then sends a download request out on the LAN. The download occurs from the host and the terminal can now be used.

2. Describe some examples of when DDFA can be used.

Answer:

To "hardwire" or associate a specific physical DTC port to a specific logical HP-UX pty device special file. Especially useful when setting up printers on the DTC.

D-7. LAB: Hands-On Disk Quotas

1. Set up a disk quota for the file system you added earlier in the class. Set quotas of 5 blocks and 3 inodes. Set limits of 10 blocks and 5 inodes.

Answer:

a. Edit `fstab`. Add "quota" after "defaults". Unmount and remount the file system.

b. Add a new user whose home directory is in the file system.

c. `# cpset /dev/null` */file system/*`quotas 600 root bin`

d. `# /usr/sbin/edquota` *username* `fs` */file system* `blocks (soft = 5, hard = 10) inodes (soft = 3, hard = 5)`

e. `quotacheck -a`

f. `quotaon -a`

g. Use `login` to log in as the new user, and use `touch` to create files up to the limit.

D-8. REVIEW: Check Your Understanding

1. If you want to set quotas for a file system named `/home` where have you create the quotas file.

Answer:

The `quotas` file has to be in the root of the file system and therefore in `/home`.

2. Why can't you directly edit the `quotas` file?

Answer:

Because it has a binary format.

Glossary

action
Provides a user interface for applications and other commands. Actions provide a visual representation of a command.

Activate
Select. Usually activated items will be highlighted.

auxiliary file systems
Mountable file systems referred to by the name of the device file associated with the particular partition that contains the file system.

Bad Block Relocation Area
Contains information specific to the bad block recovery mechanism.

Block device files
Block device files transfer data using the system buffers.

boot
Load a system kernel from a disk or tape device, thereby starting the UNIX operating system. To "boot a system" is often used as a term meaning "Load the kernel, start all system processes and enable login."

bundle
A useful set of file sets put together to add all file sets necessary for using a product.

Bus Converter
A board that serves as an internal interface between higher- and lower-speed buses.

CD-ROM File System
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.

Channel I/O Adapter
A module that serves as an internal interface between the mid-bus and the lower-speed CIO bus.

Character device files
Character device files transfer data one character at a time.

Glossary

class
A group of printers handled together as one logical printer. Print jobs are queued into the class, the output will be made on the first available printer.

clean byte
A one byte information in the file system's super block, showing whether a disk is in use (mounted) or not in use.

CPU
Central Processing Unit. The instruction-processing module inside the computer, not to the computer itself. The CPU processes data supplied to it by the I/O system.

current session
A VUE session which stored at logout.

cylinder group
A group of one or more cylinders.

cylinder group information
The dynamic parameters of the cylinder group.

data blocks
The actual data in a file. These are referenced by an addressing scheme in the inode.

deactivation scheme
Processes are deactivated when the system detects thrashing as well as when the system experiences memory pressure.

demand-paged virtual memory
The MMU allows the total size of user processes to exceed physical memory. By using the rules associated with the concept of virtual memory, when a process executes, parts of the process are brought into main memory only as needed, that is, on demand.

Device Adapter
Interface cards that make the link that enables the system to communicate with external peripherals. Device adapters are also called I/O cards and host adapter cards.

device file
A file referencing hardware connected to the system

Glossary

directory
Like a file, a directory is a container. But instead of text or other data, directories contain files and other directories. Directories are used to impose an organizational structure on a given group of files and sub-directories, making them easier to locate, reference, and access.

disk partition
A special area on the disk where space for a file system is created.

Dynamically allocatable swap
disk space, device swap or file system swap, that can be allocated while the system is running.

effective group
The group ID effectively used during an operation. It may vary from the users group ID due to Set-Group-ID-Bits set on executable files.

extent
One or more adjacent blocks of data within the file system

External Bus
A mechanism (like a cable) that can connect many like devices to one interface card. This is how devices like disks and tape drives are attached to the system.

file set
The smallest selectable subset of a product, adding only ONE necessary functionality to the system.

file system
The entire hierarchical HP-UX file system tree or directory structure, or the specific collection of files on a partition or storage device such as a whole disk or logical volume.

fileset
The smallest selectable subset of a product, adding only ONE necessary functionality to the system.

final flags
Parameters set on a serial line AFTER the user has logged in.

Halting
Brings the system to a complete stop; in this state, the only way to restart the system is to cycle the power or reset the hardware.

hard limit
Limit of blocks and/or inodes that an user is not allowed to exceeded

HFS File System
HFS is an acronym for High-performance File System. HFS file systems physically reside on mass storage devices, usually hard disk drives.

home session
A Vue session which is stored by the user at some time during the session.

I/O System
The physical hardware in the SPU that allows both built-in and add-on interface cards to be connected.

incremental backup
Back up only those files modified since the last backup. The last backup may have been either a FULL backup OR an INCREMENTAL backup.

initdefault
The default system run level which is set when the normal (uninterrupted) boot process is performed.

initial flags
Parameters set on a serial line BEFORE the user has logged in.

Initial Program Loader
A synonym for ISL.

Initial System Loader
A program located on a disk or tape which is used to perform basic operations like testing the hardware, set boot parameters and loading the kernel.

inode table
Contains entries for a set of inodes. Inodes contain information about individual files.

installation
Install a new operating system on a disk. This deletes ALL DATA on the target disk.

Interface Card
An accessory card that is either built into the SPU or plugs into slots in the I/O system (typically on the back of the SPU case). This card will provide connections for and access to peripheral devices.

interface script
The link between the scheduler and the device (that is, its device file).

Glossary

Interleaved swapping
space from one swap device is used and then space from another space device.

Internal Bus
The electronic path that connects the various areas of the SPU and allows data to flow throughout.

JFS File System
The HP-UX Journaled File System (JFS) 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. JFS is not available for root or bootable file systems.

kernel driver
A software program that controls I/O for a particular device (or class of devices).

LAN Converter
A board that serves as an internal interface between the system bus and the 802.3 LAN interface.

LIF Directory Area
Contains HPUXboot, `LABEL` files and other LIF files. Created by `mkboot(1M)`. `LABEL` files are created by `lvlnboot(1M)`.

LIF header
contains the LIF header information. This area will also be present on Physical Volumes containing dump or swap devices. Created by "mkboot(1M)".

local printer
A printer that is physically connected to your system.

Logical address space
All valid memory addresses; also called virtual address space.

Logical Extents
A logically contiguous set of logical blocks within a single logical volume. Corresponds to one physical extent unless the volume is mirrored.

logical volume
A logical volume is a volume that is implemented by the LVM. A logical volume is a block device composed of an integral number of logical extents. The number of logical extents within a logical volume vary, and is configurable by the LVM.

LP spooler
a collection of utilities and commands that controls the print requests of users.

major number
A pointer to the kernel driver HP-UX is to use when communicating with the peripheral.

manpages
Colloquial name for the on-line version of *HP-UX Reference*. Manpages are viewed by means of the `man(1)` command.

Memory
Physical Random Access Memory (RAM) located in the SPU and available for use by the CPU and I/O system. This is where all data is operated on.

memory management
The rules that govern physical memory and allow for efficient sharing of the system's resources by user and system processes.

Memory Management System
The hardware and software that enforces memory management rules.

Memory Management Unit
The hardware and software that enforces memory management rules.

minor number
A compact representation of the device address and certain characteristics of the device.

mount point
The directory to which you attach a file system.

network source install
Perform a complete update or installation not from a CD-ROM, DAT/DDS or other media, but via the local area network (LAN).

NFS File System
NFS is an acronym for Network File Services. NFS file systems are remote HFS file systems, accessible over a network, that can be used in a local file system.

object
A device, a parameter, a logical entity, or some other system element.

object-list screen
A list of all available information about the selected topic. Some information may be hidden and will be visible only if the user specifies this using the *View* menu.

page
In HP-UX, the basic virtual memory unit. A page is 4 Kbytes in size.

paging
Moving smaller units (called pages) between RAM and mass storage when needed or on demand.

partitioning
The creation and management of disk partitions by the Logical Volume manager or by the use of whole disk.

partitions
A logical part of a disk. The partition size is defined during its creation.

physical address space
The actual RAM purchased.

Physical Extents
The basic unit of allocation used to assign space to logical volumes. Contiguous sets of addressable disk blocks contained in a single physical volume. Always the same size within a volume group. Default is 4 Mb.

Physical memory
The RAM (random access memory) installed in your computer.

Physical Volume Reserved Area
Contains LVM information specific to that Physical Volume. Created by the `pvcreate(1M)`.

physical volumes
The term physical volume is used to distinguish a volume that the LVM uses for its storage from a volume that a LVM implements. A given physical volume must be assigned to a volume group before the physical volume may be used by the LVM. A physical volume is also referred to as an "LVM disk".

product
A useful set of filesets put together to add all filesets necessary for using a product.

pseudo-swap reservation
The pseudo-swap is space in system memory considered as available virtual memory space in addition to device swap space on disk. By default, pseudo-swap is enabled.

raw devices
Character devices are often referred to as raw devices.

raw I/O
Character I/O is also called raw I/O.

Rebooting
Brings the system to a complete stop, but then restarts the system as if you had booted it.

remote printer
A printer that is not physically connected to your system.

resource
A mechanism of the X Window System used to describe, or specify, certain attributes (appearance or behavior) of a window or application.

root access
The superuser has the right to access all files and directories, no matter what permissions are set by the owner. Most administrative commands and files are owned by "root", who is the only user allowed to use / modify them.

root disk
The disk which contains the boot area, system files, primary swap, and dump.

root file system
Is volume used to boot the system (find the kernel:/stand/vmunix). It will be the root (/) of the file system when the system will be booting.

root volume group
Is volume used to boot the system (find the kernel:/stand/vmunix). It will be the root (/) of the file system when the system will be booting.

run-level
A system state in which a specific set of processes is allowed to run.

secondary loader
A synonym for ISL.

sections
In HP-UX, a disk is divided into portions (sometimes overlapping) called sections. The amount of space of each section is a percentage of total disk space. For example, section 0 references 100% (the whole disk). Disk sections on a specific disk are defined by a starting cylinder and an ending cylinder.

session
In the VUE context, the lifetime of the HP VUE Session Manager - the time between logging in and logging out.

soft limit
Limit of blocks and/or inodes that an user is allowed to use. When a user exceeds his/her soft limit, a warning is emitted on /dev/tty. The user can continue to increase utilization over the soft limit until he/she either exceeds the hard limit or the established time limit.

Software Disk Striping
The capability to spread data across multiple disks to increase performance.

special file
A synonym for "device file"

SPU
System Processing Unit. The SPU cabinet contains the CPU, system memory, and the I/O system.

Stable Storage
Static RAM, a non-volatile type of memory that retains data even if power to the computer is shut off.

Step menus
Dialogs which must be filled in step by step. The task can be performed only when all required steps have been completed.

superblock
A contiguous 8 K block of disk space, which HP-UX uses to keep track of the current state of the file system.

superuser
The administrator of the system. Her/His login name is "root" and (s)he has all permissions on all files and directories.

time limit
When a user exceeds the soft limit for blocks or inodes on a file system, a countdown timer is started and the user has an amount of time equal to the time limit in which to reduce usage to below the soft limit.

toolbox
A container for action icons. These icons are a visual way of representing applications utilities and other commands.

Glossary

update
Add functionality to a running system

User Data
The user data area is where all user data is stored. This includes file systems, virtual memory (swap), or user applications.

Virtual address space
All valid memory addresses; also called logical address space.

Volume Group Reserved Area
Contains LVM information specific to the entire Volume Group. Created by `vgcreate(1M)`.

volume groups
A volume group is a collection of volume that are managed by the LVM. It may be viewed both as a collection of logical volumes (implemented by the LVM) or as a collection of physical volumes (used by the LVM).

VxFS File System
The HP-UX Journaled File System (JFS) 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. JFS is not available for root or bootable file systems.

XDM
An MIT client upon which `vuelogin` was based.

XDMCP
A standard developed by the X Consortium, at least in part to support X Terminals.