

**Change Pages for CLIPPER System
Administrator's Guide**

DSA029912

September 1990

INTERGRAPH

**Change Pages for CLIPPER System
Administrator's Guide**

DSA029912

September 1990

Changing Your Pages

Replace the pages in the *CLIPPER System Administrator's Guide* with the following change pages. Note that a date appears at the bottom of each change page to help you identify the most recent version of a manual page when you have more than one release.

Section/ Chapter	Remove Pages	Insert Pages
Contents	ix to xvi	ix to xvi
9	9-1 to 9-56	9-1 to 9-32
10	10-29 to 10-32	10-29 to 10-32
11	11-1 to 11-42	11-1 to 11-54
Index	In-1 to In-32	In-1 to In-32

Contents

Chapter 1: Setting Up a New System

1.1	Setting Up and Adjusting the Hardware	1-3
1.2	Evaluating the Default Partitions	1-4
1.2.1	Default Partitions	1-4
1.2.2	Determining Partition Sizes	1-5
1.3	Configuring the Workstation Using the Utility Pages	1-6
1.4	Setting Up User Accounts	1-8
1.5	Assigning an Internet Address	1-9
1.6	Delivering Software	1-10
1.7	Determining a Backup Schedule	1-11
1.8	Using Intergraph System Documentation	1-12

Chapter 2: System Startup and Initialization

2.1	The /etc/init Process.....	2-3
2.2	The /etc/getty Process	2-5
2.3	Modifying the inittab and gettydefs Files for a 32-Channel RS-232 Interface Assembly.....	2-6
2.3.1	Modifying the /etc/inittab File.....	2-6
2.3.2	Modifying the /etc/gettydefs.....	2-8
2.4	The /etc/rc2 Process	2-9
2.4.1	The /etc/init.d Directory	2-10
2.4.2	Initializing Additional Processes from Multiuser Mode	2-10

Chapter 3: Creating and Maintaining User Accounts

3.1	System and Administrative Accounts	3-3
3.1.1	The root Account	3-3
3.1.2	The sysadm Account	3-3
3.1.3	The setup Account	3-4
3.1.4	The sys Account	3-4
3.1.5	The nuucp Account	3-4
3.2	Adding User Accounts	3-5
3.2.1	Using addusers to Add User Accounts	3-6
3.2.2	Using sysadm to Add User Accounts	3-8
3.3	Removing User Accounts	3-10
3.3.1	Using rmusers to Remove User Accounts	3-10
3.3.2	Using sysadm to Remove User Accounts	3-11
3.4	Assigning, Changing, and Removing Passwords	3-13
3.4.1	Restrictions on Passwords	3-13
3.4.2	Creating a Password	3-13
3.4.3	Forcing a Password Change	3-14
3.4.4	Removing a Password	3-15
3.4.4.1	Removing a Password from User Accounts	3-15
3.4.4.2	Removing the root Password	3-15
3.4.4.3	Removing the Utility Pages Password	3-18

3.5	Using Protection Codes to Manage Files	3-19
3.5.1	File Protection Codes	3-19
3.5.1.1	Directory Files	3-20
3.5.1.2	Ordinary Files	3-21
3.5.2	Modifying File Permissions	3-21
3.5.3	Modifying File Ownership	3-22
3.5.4	Changing Group Access	3-23
3.5.4.1	Changing Your Active Group	3-23
3.5.4.2	Changing the Group of a File	3-24

Chapter 4: Modifying the Login Environment

4.1	The <code>getty</code> and <code>login</code> Processes	4-3
4.1.1	The <code>/etc/passwd</code> File	4-4
4.1.2	Customizing the <code>/etc/passwd</code> File	4-5
4.2	The <code>/etc/profile</code> File	4-6
4.3	The <code>.profile</code> file	4-8
4.4	The <code>.env</code> File	4-10
4.4.1	Command Recall (The <code>.history</code> File)	4-11
4.4.1.1	Command Recall Using the <code>vi</code> Editor	4-12
4.4.1.2	Command Recall Using the <code>emacs</code> Editor	4-12
4.4.1.3	Temporary Editors for Command Recall	4-12
4.5	The <code>.exrc</code> File	4-13
4.6	Creating a Reminder Service	4-14
4.7	Using <code>cron</code>	4-15
4.7.1	The <code>crontab</code> Command	4-16
4.7.2	Executing One Command at a Specified Time	4-17

Chapter 5: Window and Screen Configuration

5.1	Setting the Workstation User	5-3
5.2	Creating VT220 Windows	5-4
5.2.1	Using <code>vterm</code>	5-4
5.2.2	Creating Local Windows Through the Workstation Menu	5-8
5.2.3	Creating a Terminal Window	5-8
5.3	Using the General Setup Menu to Define VT220 Characteristics	5-10
5.4	Using the Display Setup Menu to Define Display Characteristics	5-12
5.5	Using the Keyboard Setup Menu to Define Keyboard Characteristics	5-13
5.6	Using the Tab Setup Menu to Define Tabs	5-14
5.7	Using the Printer Setup Menu to Define a Window Attached to a Dedicated Printer	5-15
5.8	Using the Macro Setup Menu to Assign Macros to Keys	5-16

5.9	Creating Color Windows	5-18
5.9.1	The Intergraph-Developed regis Color Command	5-18
5.9.2	The ReGIS Color Command	5-19
5.9.3	The ANSI Color Escape Sequences	5-22
5.10	Using the Catalog Function to Save Customized Windows	5-24
5.11	Altering the Workstation Pulldown Menu	5-26
5.11.1	Adding Processes to the Workstation Pulldown Menu	5-26
5.11.2	Modifying the Workstation Pulldown Menu	5-28
5.11.3	Deleting Processes from the Workstation Pulldown Menu	5-28
5.12	Changing Screen Configuration	5-30
5.12.1	Changing the Number of Virtual Screens (Single-Monitor Workstations Only)	5-30
5.12.2	Changing the Number of Message Strips (Dual-Monitor Workstations Only)	5-31
5.12.3	Changing the Message Strip(s) Location	5-31
5.13	Changing the Background Pattern	5-32
5.14	Changing the Screen Color	5-33
5.15	Invoking Processes Automatically	5-34
5.15.1	The def_proc File	5-34
5.15.2	The .def_proc File	5-35
5.15.3	The sys_proc File	5-36

Chapter 6: Setting Up a Line Printer and Printing

6.1	Configuring a Line Printer	6-3
6.2	Using the Line Printer	6-4
6.2.1	Submitting a Print Request	6-4
6.2.2	Monitoring a Print Request	6-4
6.2.3	Canceling a Print Request	6-5

Chapter 7: Creating and Maintaining File Systems

7.1	Creating a File System	7-3
7.1.1	Creating a Device File	7-3
7.1.2	Creating a File System on a Partition	7-9
7.1.2.1	Creating a Standard File System	7-10
7.1.2.2	Creating a Fast File System	7-11
7.1.3	Mounting a Partition	7-12
7.2	Monitoring Disk Space	7-13
7.2.1	Displaying the Amount of Free Disk Space	7-13
7.2.2	Displaying the Amount of Disk Space Used	7-14
7.2.3	Freeing Disk Space	7-14
7.3	Checking the Integrity of a File System.....	7-17

Chapter 8: Monitoring Processes with CRM

8.1	Installing CRM	8-3
8.2	Invoking the Curses-Based FMLI Interface for CRM	8-4
8.2.1	Graphics Monitors.....	8-4
8.2.2	Alphanumeric Terminals.....	8-4
8.3	Using the FMLI Interface	8-5
8.4	System Monitors	8-7
8.4.1	Running System Monitors from the Command Line	8-7
8.4.2	Running Graphics-Based System Monitors	8-9
8.4.3	Using the System Monitors from the Curses-Based FMLI Menus	8-10
8.4.3.1	Change Defaults Form	8-10
8.4.3.2	Monitor Parameters (monparam)	8-11
8.4.3.3	Top Fault Monitor (topfault)	8-12
8.4.3.4	Top Memory Monitor (topmem)	8-14
8.4.3.5	Top CPU Monitor (topcpu)	8-15
8.4.3.6	Top I/O Monitor (topio)	8-15
8.4.3.7	Top System Monitor (topsys)	8-16
8.4.3.8	Show Open Files (showfiles)	8-17
8.4.3.9	Show Memory Usage (showmemory)	8-17
8.5	Process Monitors	8-19
8.5.1	Running Process Monitors from the Command Line	8-19
8.5.2	Using the Process Monitors Interactively from Curses-Based FMLI Menus	8-24
8.5.2.1	Select Process to Monitor	8-24
8.5.2.2	Select Profiler Options	8-25
8.5.2.3	Enable Page Faults	8-26
8.5.2.4	Enable System Calls	8-26
8.5.2.5	Run Profiler	8-27
8.5.2.6	Run Memory Monitor	8-28
8.5.2.7	Run Process Monitor	8-29

Chapter 9: Backing Up and Restoring Files

9.1	Determining a Backup Schedule	9-3
9.2	Devising a Backup Plan According to System Configuration	9-4
9.2.1	Full Network Configuration	9-4
9.2.2	Standalone Configuration	9-5
9.3	Backup Commands and Utilities	9-6
9.3.1	The dd Command	9-10
9.3.2	The tar Command	9-11
9.3.3	The cpio/scpio Commands	9-13
9.3.3.1	Appending Files to ¼-inch Cartridge or 9-Track Tapes	9-15
9.3.3.2	Appending Files to 8 mm Helical Tapes	9-16
9.3.4	The backup and restore Commands	9-18
9.3.5	The ansitape Utility	9-20
9.3.6	The vmsbackup Utility	9-21

9.3.7	The rtc Utility	9-22
9.3.8	The compress Utility	9-23
9.3.9	Erasing Tapes	9-25
9.4	Additional Commands for Floppy Disks	9-26
9.4.1	The format Command	9-27
9.4.2	The mkfs Command	9-29
9.4.3	The cpflop Command	9-31
9.4.4	The to_flop Command	9-31
9.4.5	The fr_flop Command	9-31

Chapter 10: Using the Utility Pages

10.1	Workstation Utility Menus	10-3
10.1.1	Main Utility Page	10-4
10.1.2	Peripheral Configuration Page	10-7
10.1.3	Operating System Parameters Page	10-11
10.1.4	Disk Maintenance Utility Page	10-13
10.1.5	Hard Disk Flaw Data Utility Page	10-16
10.1.6	Verify Utility Page	10-19
10.1.7	Format Utility Page	10-21
10.1.8	Disk Partitioning Utility Page	10-23
10.1.9	Device Image Copy Utility	10-26
10.1.10	Rebuild Utility Page.....	10-30
10.2	Server Utility Menus	10-32
10.2.1	Startup Configuration Utility Menu	10-34
10.2.2	Main Utility Menu	10-35
10.2.3	Date and Time Menu	10-36
10.2.4	Peripheral Configuration Menu	10-39
10.2.5	Operating System Parameters Menu	10-42
10.2.6	Disk Maintenance Utilities Menu	10-43
10.2.7	Disk Partitioning Menu.....	10-46
10.2.8	Device Image Copy Utility Menu	10-48
10.2.9	Rebuild Utility Menu.....	10-54

Chapter 11: Rebuilding and Repartitioning the Hard Disk

11.1	Backing Up Files	11-3
11.2	Creating Current Rebuild Floppy Disks	11-4
11.3	Booting from the Rebuild Boot Floppy Disk	11-7
11.4	Entering the Utility Pages	11-8
11.4.1	Entering the Utility Pages on a Workstation	11-8
11.4.2	Entering the Utility Pages on a Server	11-9
11.5	Verifying Hard Disk Flaws	11-10
11.5.1	Verifying a Workstation Hard Disk	11-11
11.5.2	Verifying a Server Hard Disk	11-12
11.6	Formatting the Hard Disk	11-13
11.6.1	Formatting Workstation Hard Disks	11-13
11.6.2	Formatting Server Hard Disks	11-14
11.7	Selecting the Rebuild Utility Page Icon	11-16

11.8	Repartitioning the Hard Disk	11-17
11.8.1	Using the Current Partition Table	11-18
11.8.2	Repartitioning Using Default Partitions	11-19
11.8.3	Repartitioning Using Custom Partitions	11-20
11.8.3.1	Partitioning Overview	11-20
11.8.3.2	Creating Nonstandard Partitions	11-28
11.9	Loading the Rebuild Root Media	11-32
11.10	Entering the Rebuild Environment	11-33
11.11	Restoring the File Systems	11-34
11.12	Loading Nucleus Software with newprod	11-42
11.13	Preparing Nonstandard Partitions for Use	11-52
11.14	Loading Application Software with the newprod Utility	11-53
11.15	Restoring Files from Backups	11-54
Appendix A: Keyboard Character Codes.....		A-1
Appendix B: Read/Write Compatibility of Floppy Disks with Intergraph Workstations and IBM PCs.....		B-1
Appendix C: System V/PC-DOS Utilities.....		C-1
Glossary.....		gl-1
Index.....		in-1

Chapter 9: Backing Up and Restoring Files

This chapter contains the following sections:

9.1	Determining a Backup Schedule	9-3
9.2	Devising a Backup Plan According to System Configuration	9-4
9.2.1	Full Network Configuration	9-4
9.2.2	Standalone Configuration	9-5
9.3	Backup Commands and Utilities	9-6
9.3.1	The dd Command	9-10
9.3.2	The tar Command	9-11
9.3.3	The cpio/scpio Commands	9-13
9.3.3.1	Appending Files to ¼-inch Cartridge or 9-Track Tapes	9-15
9.3.3.2	Appending Files to 8 mm Helical Tapes	9-16
9.3.4	The backup and restore Commands	9-18
9.3.5	The ansitape Utility	9-20
9.3.6	The vmsbackup Utility	9-21
9.3.7	The rtc Utility	9-22
9.3.8	The compress Utility	9-23
9.3.9	Erasing Tapes	9-25
9.4	Additional Commands for Floppy Disks	9-26
9.4.1	The format Command	9-27
9.4.2	The mkfs Command	9-29
9.4.3	The cpflop Command	9-31
9.4.4	The to_flop Command	9-31
9.4.5	The fr_flop Command	9-31

9-2 Backing Up and Restoring Files

You should keep accurate and complete records of hard disk files for the following reasons:

- If the system crashes or if the hard disk must be rebuilt, backup files prevent data loss.
- If you have disk space problems, you may need to store less frequently used files on storage media and load them only when needed.
- Backing up files regularly retains a history of the file system. For example, you may need a file version from a day, a week, or a month ago.
- Certain files or directory structures may require tighter security. For example, if you do not wish for users to access certain files, you can back up those files on a tape or removable disk and remove them from the system.

Backing up files is a complex process that requires careful planning. The system administrator should establish a specific schedule and method of backing up workstation/server files. This chapter provides procedures for backing up and restoring files, as well as establishing an appropriate schedule and method.

Procedures for the following tasks are provided:

- Determining an appropriate backup schedule
- Devising a backup plan according to system configuration

Numerous backup commands and utilities are available under the CLIX operating system. This chapter describes and offers suggestions for using each of the following commands. However, the system administrator must determine which backup command is most appropriate for backing up a particular system.

- dd
- tar
- cpio/scpio
- backup and restore
- ansitape (magnetic tape only)
- vmsbackup (magnetic tape only)
- rtc (magnetic tape only)
- compress (used to compress archive files)

The final section in this chapter describes commands used only for floppy disks. The following floppy commands are discussed:

- format
- mkfs
- mf
- uf
- cpflop
- to_flop
- fr_flop

Updated 9/90

9.1 Determining a Backup Schedule

You should define a schedule for backing up files so you can maintain complete and accurate records of hard disk files. The backup schedule for each system will vary according to the importance of backups. For example, a system that stores critical data should be backed up more frequently than other systems should be. The following is an example of a schedule that can be used to back up files daily, weekly, and monthly.

Assume that the ACE Engineering Company has three design engineers, each equipped with an InterPro 340 and the EMS design package. This company also has an InterServe 3005 with a Kennedy tape drive mounted on it. ACE engineers realize the importance of quality system backups, so they devised the following plan to periodically back up files from their 340s to the tape drive on the InterServe 3005:

- The daily backup is planned as follows:
 - Each engineer has four tapes for daily backups, one a day for Monday through Thursday. (Friday will be included in the weekly backup.)
 - Each engineer backs up his/her user account daily.
 - Each engineer backs up files at a different predetermined hour to allow tape drive sharing.

- The weekly backup is planned as follows:
 - Each engineer has three tapes for weekly backups, one a week for the first three weeks in the month. (The fourth week will be included in the monthly backup.)
 - Each engineer backs up his/her user account, the */etc* directory, the */usr/ip32/smgr* directory, and the */usr/ip32/vt200/catalog* directory, as the files in these directories contain user-configurable data that may change frequently.

- The monthly backup is planned as follows:
 - Each engineer has one or more tapes for monthly backups, depending on how many monthly backups are needed.
 - Each engineer backs up his */usr* file system and possibly some directories under the root (*/*) directory (such as */etc*).

9.2 Devising a Backup Plan According to System Configuration

The most appropriate backup plan for your system depends on the configuration. Most systems generally fit in one of two configurations:

- The configuration includes a full network with workstations, servers, and VAXes.
- The configuration includes a standalone workstation and a local tape drive.

This section provides an example for devising a backup plan according to your system configuration. It uses the `cpio` command to back up and restore files. For a more complete description of `cpio` and other backup commands and utilities, refer to section 9.3, "Backup Commands and Utilities."

9.2.1 Full Network Configuration

In a full network configuration, possibly including workstations, servers, and VAXes, the backup plan is complex but allows many options. The following is an example of a backup method that uses a tape drive in the network (on a workstation, server, or VAX) to back up workstation/server files.

In the previous subsection, ACE Engineering devised a schedule for backing up files. Now, the company must determine the backup plan most suitable for its system configuration. Since the tape drive is mounted in the InterServe 3005, the ACE engineers will use it as their backup device. They will use the `rtc` (remote tape controller) utility that Intergraph developed so that each workstation can use the tape drive as if it were attached locally.

It is 3 p.m. on Monday and time for John Doe to back up his user account (as he does everyday at this time). First, he uses the `rtc` utility to allocate the tape drive to his workstation. The syntax of `rtc` is as follows:

```
$ rtc -ad -s<nodename> -r<rewdev> -n<norewdev> -t<timeout> <device>
```

- a Allocates a remote device.
- d Deallocates a remote device.
- s Specifies the nodename or address of the host where the remote device resides.
- r Specifies the rewindable device file that references the tape on the remote node.
- n Specifies the nonrewindable device file that references the tape on the remote node.
- t Specifies the length of time (in minutes) that the allocated drive is idle before the connection is terminated. This argument is optional.
- device Specifies the `rtc` device controller specification.

John issues the following command to allocate the remote tape drive to node ace_pn for 30 minutes, specifying an automatic-rewind device file:

```
$ rtc -a -sace_pn -r/dev/rmt/mt6 -t30 /dev/rmt/rt0ctl
```

The following message appears, confirming that the tape drive has been allocated to John's workstation:

```
allocate: /dev/rmt/rt0ctl ace_pn /dev/rmt/mt6 5
```

Now, John can back up files as if he has a local tape drive. He uses the `cpio` command (described in detail in section 9.3.3) to back up his user directory (`/usr/john`) as follows. Notice that the device file specified in the `cpio` command does not require the `.ctl` extension, since `/dev/rmt/rt0ctl` was specified when the drive was allocated.

```
$ find /usr/john -print | cpio -ov >/dev/rmt/rt0
```

Once John's user directory has been backed up, he can deallocate the tape drive as follows:

```
$ rtc -d /dev/rmt/rt0ctl
```

9.2.2 Standalone Configuration

If a system is a single workstation/server with a dedicated tape drive, John will not need to allocate the tape drive because the device is not remote. Instead, he can use the `cpio` command to back up the files in his user directory. The device file in the following example is `/dev/rmt/mt6` to reference a local tape drive at SCSI ID 6.

```
$ find /usr/john -print | cpio -ov >/dev/rmt/mt6
```

9.3 Backup Commands and Utilities

Numerous backup commands and utilities exist under the CLIX operating system. Use the following chart to help determine which utility is best for backing up your system. Each of these commands are described in detail later in this section.

Command	Advantages	Disadvantages	Recommended Uses
dd	Fast	<p>You must know the original filename to recover a file from a backup</p> <p>You cannot write to multiple volumes.</p>	<p>Copying raw binary data</p> <p>Copying to and from device files</p>
tar	Simple	<p>If a file in the saveset becomes corrupted, the remaining files cannot be recovered.</p> <p>You cannot back up device files.</p> <p>You cannot write to multiple volumes. (However, for floppies, you can use <code>to_flop</code> and <code>fr_flop</code>.)</p>	<p>Backing up noncritical data</p> <p>Backing up individual files</p>

Command	Advantages	Disadvantages	Recommended Uses
<p>cpio</p>	<p>You can back up any file, including device files</p> <p>You can direct output to any device, pipe, or file on the system.</p> <p>You can write to multiple volumes.</p>	<p>No verify mode (see scpio)</p>	<p>Performing any level of backup including large-scale backups of critical data</p>
<p>scpio</p>	<p>Includes all advantages of cpio.</p> <p>Error-handling capabilities</p> <p>Asynchronous output and multi-buffering (faster than cpio)</p> <p>Verify mode to confirm integrity of input and archived files</p>		<p>Performing any level of backup including large-scale backups of critical data</p>

Command	Advantages	Disadvantages	Recommended Uses
backup	<p>Good user interface</p> <p>Fair error recovery mechanisms</p> <p>Can write to multiple volume save sets</p>	<p>Cannot back up individual files (only entire file systems)</p>	<p>Writing one or more volumes of tapes</p>
restore	<p>Good user interface</p> <p>Fair error recovery mechanisms</p> <p>Can read multiple volume save sets</p>	<p>Can only recover backup save sets</p>	<p>Recovering backup save set</p>
ansitape	<p>Can read or write tapes</p> <p>Can transfer data between CLIX and VMS</p>		<p>Reading/writing tapes between CLIX and VMS</p> <p>Useful for transferring design files to remote locations with no network link</p>
vmsbackup	<p>Can transfer data between CLIX and VMS</p>	<p>Does not write tapes (only reads them)</p>	<p>Reading VMS-generated backup format tapes</p> <p>Useful for transferring design files to remote locations with no network link</p>

Command	Advantages	Disadvantages	Recommended Uses
rtc	Can access a remote tape drive as though it were local		Reading/writing to tape on a remote machine

9.3.1 The dd Command

The most basic command for transferring data is `dd`. You can use this utility to move raw binary data or files to and from the storage device (hard disk, tape, or floppy disk).

Although you can use `dd` as a backup utility, it lacks a user interface for recovering data. For example, if you use `dd` to write a file to a tape, you must know the original filename, as this information is not maintained in a header.

The `dd` utility is useful for manipulating raw data transferred to a tape from another operating system or for imaging a disk drive by reading input from a raw partition. For example, the following command creates an image of the `/usr` file system and sends it to the tape drive referenced by `/dev/rmt/mt6`:

```
# dd if=/dev/rdisk/s0u0p7.3 of=/dev/rmt/mt6
```

The string `"if=/dev/rdisk/s0u0p7.3"` specifies that `dd` will accept input from the raw device located at SCSI ID 0, LUN 0, partition 7, modifier 3. (The input file is the `/usr` file system.) The string `"of=/dev/rmt/mt6"` specifies that `dd` will send the output information to a tape device located at SCSI ID 6. (The output file is the rewindable tape device at SCSI ID 6.)

The following command restores the data backed up with the previous command:

```
# dd if=/dev/rmt/mt6 of=/dev/rdisk/s0u0p7.3
```

Although `dd` does not have the functionality of most other backup utilities, it is fairly fast. Using blocking factors increases performance. (The blocking factor signifies the amount of data buffered in one data block before it is written to the media.)

Caution:

Because data is treated as one large file, if any part of the data becomes corrupted, you probably will not be able to recover any of the backed up data.

Note:

To use `dd` with a cartridge tape, you must add the `conv=sync` option to the command line to preserve a 512 (or multiple) byte boundary as follows:

```
# dd if=/dev/rdisk/s0u0p7.3 of=/dev/rmt/mt6 conv=sync
```

Please refer to the `dd(1)` manual page from the System V Online Documentation product for more information on the `dd` command.

9.3.2 The tar Command

The tape file archiver utility (**tar**) is used to back up a file, directory, or entire file system. However, **tar** does not have any error handling capability. If a file in a **tar** saveset becomes corrupted, all files following that file cannot be recovered with **tar**. In addition, **tar** will not back up device files. When attempting to back up files, **tar** skips over device files.

The **tar** command is best used as a secondary method for performing major backups or as a utility to back up noncritical information.

The format for the **tar** command is as follows:

```
$ tar [key] [option] [file]
```

tar is the executable command.
key(s) designates a specific action of the command.
option(s) is a command supplement.
filename(s) designates the files to transfer.

Note:

The **tar** command format does not require a dash (-) in front of the key.

The following command backs up a user account to a tape drive at SCSI ID 6:

```
# tar cvfb /dev/rmt/mt6 20 /usr/john
```

c specifies a create operation.
v specifies verbose mode.
f specifies that a device file (/dev/rmt/mt6) will be supplied.
b specifies that a blocking factor (20) will be supplied. (The blocking factor signifies the amount of data buffered into one block before the data is written to the media.)

To list the files saved on a tape, use the **t** (table) option as follows:

```
$ tar tvf /dev/rmt/mt6
```

The following command recovers the data backed up previously:

```
$ tar xvfb /dev/rmt/mt6 20
```

`tar` output defaults to floppy disk but can be directed to any device, pipe, or file on the system. For example, to back up files to floppy, omit the `f` option and do not specify a device file parameter.

Caution:

The `tar` command is not compatible with floppy disk file systems. The `tar` command will destroy a floppy disk file system by overwriting it.

The following command backs up all files with the `.dgn` extension in the user's current directory:

```
$ tar cv *.dgn
```

To list files saved on a floppy, use the `t` (table) option as follows:

```
$ tar tv
```

To recover all files in this `tar` saveset, use the `x` (extract) option as follows:

```
$ tar xv
```

To recover a single file in this `tar` saveset, use the `x` option and specify the filename to extract as follows:

```
$ tar xv house.dgn
```

The following example appends files to the same floppy disk. The `r` (read) option reads to the end of the tape and appends new files. If the `c` (create) option were used, `tar` would begin writing at the beginning of the tape and thus overwrite existing data.

```
$ tar rv *.sh
```

The following example updates the `tar` saveset. The `u` (update) option copies any file with a more recent date and any new files.

```
$ tar uv *.sh
```

Please refer to the `tar(1)` manual page in the System V Online Documentation product for more information on the `tar` command.

9.3.3 The cpio/scpio Commands

The `cpio` (copy file archives in and out) command can back up any file, including device files. Output from `cpio` can be directed to any device, pipe, or file on the system.

The `scpio` command (streaming `cpio`) is an enhanced version of the `cpio` command for streaming tape drives. It provides two major additions to the features of the `cpio` command: asynchronous input/output and multibuffering, which allows faster processing.

The `cpio` and `scpio` commands are both useful and robust backup utilities. Both utilities are capable of reading each other's savesets, and both utilities can write multiple volumes of tapes. This capability allows more flexibility because you do not need to know how much tape will be needed to store the data.

The input for these utilities must be a list of filenames. Thus, to write a saveset to either of these utilities, you must use a command such as `ls` or (more preferably) `find` and pipe the output to `cpio`. For example, if you wanted to back up the `/usr2` file system to a tape drive at SCSI ID 5, you could use the `scpio` command as follows:

```
# find /usr2 -print | scpio -ov -C10240 -z20 -O/dev/rmt/mt5
```

find searches the hard disk for the specified directory or file (in this example, `/usr2`).

print writes a list of filenames in the `/usr2` directory in the pipe (`()`), which `scpio` reads from.

o copies output from the input list specified (in this case through the `find` command).

v specifies that output will display in verbose mode.

C specifies the block size that will be used. The block size must be a multiple of 512, between 512 and 63,488. In this example, a block size of 10,240, or 512 x 20, was used.

z specifies the number of buffers of the block size that will be allocated. In this example, 20 buffers are allocated, each holding 10,240 bytes. The value specified by `z` must be an integer from 2 to 25. The `z` option enables `scpio` to stream data because a new buffer of data is ready to be flushed before the prior buffer has been written to the media. This option is not available for `cpio`.

- O specifies the output device (in this example, */dev/rmt/mt5*). In addition, this option allows you to press <Return> when changing volumes, instead of supplying the device file again. This capability is possible because *scpio* received this information before beginning the backup.

To list the files saved on the tape, use the *scpio* command with the *t* (table) option as follows. The *i* option specifies that input will be received and the *I* option specifies where input will be read from (*/dev/rmt/mt5*).

```
# scpio -ivt -C10240 -z20 -I/dev/rmt/mt5
```

To recover the previously saved data, issue the following command:

```
# scpio -iv -C10240 -z20 -I/dev/rmt/mt5
```

Please refer to the *cpio(1)* and *scpio(1)* manual pages in the System V Online Documentation product for more information on the *cpio* and *scpio* commands.

9.3.3.1 Appending Files to ¼-inch Cartridge Tapes or 9-Track Tapes

In addition, you can use `cpio/scpio` to append files to the end of a cartridge or 9-track tape. The following procedure can be used on any ¼-inch cartridge or 9-track tape drive:

1. Write the original `cpio/scpio` saveset. This example writes the `/usr2` file system to a tape drive at SCSI ID 6:

```
# find /usr2 -print | scpio -ov -C63488 -z20 -0/dev/rmt/mt6
```

2. After the tape is rewound to the beginning, use the following command to skip forward to the end of the tape:

```
# mt -f /dev/rmt/mt6n fseot
```

`mt` passes commands to the tape drive.

`f` specifies that the `mt` command will use the `/dev/rmt/mt6n` tape drive.

`fseot` tells the drive to skip to the end of the recorded portion of the tape.

3. Use the `cpio/scpio` command to write the second saveset to the tape. Since you skipped to the end of the recorded media in step 2, you are essentially appending files to the end of the tape. The following example appends all files in the `/usr/john` directory to the tape:

```
# find /usr/john -print | scpio -ov -C63488 -z20 -0/dev/rmt/mt6
```

Note:

The following command is used to list files saved on tape. However, when you issue this command, only the first saveset is listed.

```
# scpio -it -I /dev/rmt/mt6n
```

To list the second saveset, reissue this command as follows:

```
# scpio -it -I /dev/rmt/mt6n
```

A message similar to the following displays:

```
Reached end of medium on input.  
Change to part 2 and press Return Key [q].  
  <press return> list rest
```

Issue this command as follows for a third time and the second saveset is listed:

```
# scpio -it -I /dev/rmt/mt6n
```

4. If you wish to append a third saveset to the tape, issue the `mt` command again as follows:

```
# mt -f /dev/rmt/mt6n fseot
```

5. Use the `cpio/scpio` command to write the third saveset to the tape. The following example appends all files in the `/usr/jane` directory to the tape:

```
# find /usr/jane -print | scpio -ov -C63488 -z20 -0/dev/rmt/mt6
```

9.3.3.2 Appending Files to 8 mm Helical Tapes

You can also use `cpio/scpio` to append files to the end of an 8 mm tape. The following example is one method for appending files to an 8 mm tape drive:

Note:

Any time you append files to a tape, make sure that you know the exact number of savesets currently on the tape.

1. The following command could be used to write the first saveset to an 8 mm tape:

```
# find /usr/fred -print | scpio -ov -C63488 -z20 -0/dev/rmt/mt6
```

2. List the first saveset by keying in the following command:

```
# scpio -it -I /dev/rmt/mt6n
```

3. Issue the following command to skip to the end of the recorded media:

```
# mt -f /dev/rmt/mt6n fsf 1
```

`mt` passes commands to the tape drive.

`f` specifies that the `mt` command will use the `/dev/rmt/mt6n` tape drive.

`fsf 1` tells the drive to skip forward one saveset.

4. Now that the tape has been forwarded to the end of the recorded media, use the `cpio/scpio` command (as shown in the following example) to write the next saveset to the tape. Since you skipped to the end of the last saveset, you are essentially appending files to the end of the tape.

```
# find /usr/john -print | scpio -ov -C63488 -z20 -0/dev/rmt/mt6
```

5. If you need to append another saveset, use the following command to list each saveset on the tape. (You will need to issue this command as many times as you have savesets on the tape.)

```
# scpio -it -I /dev/rmt/mt6n
```

6. Issue the `mt` command to skip to the end of the recorded media:

```
# mt -f /dev/rmt/mt6n fsf 1
```

7. Append the saveset to the tape by using a command similar to the following:

```
# find /usr/alice -print | scpio -ov -C63488 -z20 -0/dev/rmt/mt6
```

8. Repeat steps 5 through 7 to append any more savesets.

9.3.4 The backup and restore Commands

The **backup** and **restore** commands have been ported to CLIX from Berkeley UNIX. These commands are used in tandem; **backup** is used to write a tape and **restore** is used to recover a tape written with **backup**. These commands are used only for backing up and recovering entire file systems. These utilities have a helpful user interface and good error recovery mechanisms. In addition, **backup** and **restore** can write and read multiple-volume savesets.

The following example illustrates the use of **backup** and **restore**.

A system manager wishes to back up the */usr2* file system. The */usr2* file system resides on the disk drive located at SCSI ID 1, partition 7, modifier 4 (*/dev/dsk/s1u07.4*) and it is 100,000 blocks in length. The tape drive is located at SCSI ID 5 and the system manager mounted a 2,400-ft. tape on the drive. He issues the following command to back up the file system:

```
# backup Ofunds /dev/rmt/mt5 6250 2350 20 /usr2
```

The **backup** command lists all options first, followed by the arguments.

- 0** specifies the backup level. The level must be an integer between 0 and 9. A level 0 backup will back up all files, while a level 9 backup will back up up the least number of files. The level option has no accompanying argument.
- f** specifies the device that the information will be written to (in this example, */dev/rmt/mt5*).
- u** causes **backup** to write information about the backup level, the file system being backed up, and the time the backup was performed to a tracking file called */etc/dumpdates*.
- n** notifies the operator when **backup** has encountered an error requiring attention. This option has no accompanying argument.
- d** specifies the density the media is being written at (in this example, 6250).

- s** specifies the size of the media, or the length of the tape (in this example, 2,350). Although the physical length of the tape is 2,400 ft., the beginning of tape (BOT) and end of tape (EOT) markers must be accounted for. The size specified is the length of tape that can actually be accessed. You should specify an amount less than the physical length so that backup will not attempt to write past the EOT.
- b** specifies a blocking factor (in this example, 20 KB). The blocking factor must be an integer between 1 and 20.

/usr2 specifies the file system to be backed up.

Note:

The **X** command-line option specifies an 8mm tape drive.

You can recover the file system or any file in the backup saveset with the **restore** command. To recover the entire backup saveset, key in the following command, where **r** recovers all files from the saveset.

```
# restore -r
```

To recover a file, use the interactive mode of the **restore** command. Once invoked, **restore** simulates the directory structure of the file system and activates the **restore** environment, complete with prompt and basic command ability such as **cd**, **ls**, and **pwd**. You can use these commands to locate the file to recover from the saveset. When you locate the file to recover, use the **a** command to add the file to the list of files to recover and use the **x** command to extract the file from the saveset. The following example illustrates how the **/usr2/john/help** file can be recovered.

```
# restore -i
restore > cd /usr2/john
restore > ls
abcd  check
help  test1
restore > a help
restore > x
```

Please refer to the **backup(1)** and **restore(1)** manual pages in the System V Online Documentation product for more information on the **backup** and **restore** commands.

9.3.5 The ansitape Utility

The **ansitape** utility, an ANSI-standard magnetic tape label program, reads, writes, and creates magnetic tapes conforming to the ANSI standard for magnetic tape labeling. This utility's primary use is to enable you to exchange tapes with those made with VAX/VMS. **ansitape** is useful for transferring design files to remote locations that do not have a network link.

The following command uses **ansitape** to create a tape of an **ansitape** saveset on a tape drive located at SCSI ID 6:

```
$ ansitape -cv mt=/dev/rmt/mt6 rs=512 rf=f cc=e *.dgn
```

c specifies the create operation.

v specifies verbose mode.

mt specifies the output device (in this example, */dev/rmt/mt6*).

rs=512 sets the record size to 512 bytes. This qualifier must be specified exactly as shown in this example or **ansitape** will not transfer design files correctly.

rf=f specifies fixed record format. This qualifier must be specified exactly as it appears in this example or **ansitape** will not transfer design files correctly.

cc=e specifies that carriage control is embedded in the records. This qualifier must be specified exactly as it appears in this example or **ansitape** will not transfer design files correctly.

***.dgn** specifies which files to be transferred. The asterisk is a wildcard. All files with the *.dgn* extension will be transferred.

The following command uses the **t** (table) option to list the contents of an **ansitape** readable tape:

```
$ ansitape -tv
```

To recover an **ansitape** saveset created on a workstation/server, you must use the **x** option as follows to extract the files:

```
$ ansitape -xv
```

To recover an **ansitape** saveset created on a VAX, use the VMS **copy** command. Ensure that files are contiguous before graphically displaying them.

Please refer to the **ansitape(1)** manual page in the System V Online Documentation product for more information on the **ansitape** command.

9.3.6 The vmsbackup Utility

The `vmsbackup` utility reads VMS-generated backup format tapes. `vmsbackup` cannot write these types of savesets. Like `ansitape`, this utility is useful for transferring design files between the VMS and CLIX operating systems.

The following command reads a VMS backup format tape mounted on a tape drive located at SCSI ID 6:

```
$ vmsbackup -xvf /dev/rmt/mt6
```

- x** Specifies that files will be extracted.
- v** Specifies verbose mode.
- f** Specifies the device file (in this example, `/dev/rmt/mt6`).

The following command uses the `t` (table) option to list the contents of the VMS backup format tape:

```
$ vmsbackup -tv
```

Please refer to the `vmsbackup(1)` manual page in the System V Online Documentation product for more information on the `vmsbackup` command.

9.3.7 The rtc Utility

The rtc (Remote Tape Controller) allows you to use a tape drive on another machine as if it were connected directly to your machine. For example, you can use rtc to allocate the remote tape drive to your workstation/server. Then, you can use a backup utility (such as cpio) to transfer files to or from the tape drive. And finally, you can use rtc again to deallocate the remote tape drive.

For more information on rtc, refer to the *Intergraph Network Core User's Guide*.

9.3.8 The compress Utility

The **compress** command is not used to back up files. However, this command can be helpful for archiving files because it is used to compress a large amount of data into a small area of media space.

The **compress** command can be used with **tar** and **cpio** to compress the contents of a directory into one file. The **compress** command can also be used alone to compress one file to a smaller size. When you compress a directory or file, a new file (with a **.Z** extension) is created containing the compressed file(s).

Note:

To compress a directory, you must be in superuser mode.

The following examples compress the */usr/chris* directory into an archive file (called *archive_file*) on a storage media such as tape or floppy disk:

```
# tar cvf - /usr/chris | compress > archive_file
```

OR

```
# find /usr/chris -print | cpio -ov | compress > archive_file
```

To decompress a file, use the **compress -d** command. If you used **compress** with a backup utility, you must decompress the file with the same backup utility you used to compress it. The following examples decompress the archive file (*archive_file*) created in the previous examples:

```
# compress -d < archive_file | tar xvf -
```

OR

```
# compress -d < archive_file | cpio -ivmud
```

If you wish to compress a single file (possibly to be stored on storage media later), you can use the **compress** command by itself, as shown in the following example:

```
# compress filename
# ls
filename.Z
```

The following example decompresses this single, compressed file. When you decompress a single file, notice that you should not include the `.Z` extension.

```
# compress -d filename
# ls
filename
```

Please refer to the `compress(1)` manual page in the System V Online Documentation product for more information on the `compress` command.

9.3.9 Erasing Tapes

When used with the `erase` option, the `mt` command (magnetic tape manipulation program) will erase tapes. However, because all Intergraph tape drives perform a full-width erase automatically during write operations, issuing a separate command to erase tapes is unnecessary.

The following command erases a tape mounted in a drive at SCSI ID 6:

```
# mt -f /dev/rmt/mt6 erase
```

When this command is issued, the drive logically disconnects from the SCSI bus to perform the erase. The drive will erase to the end of the tape, rewind, and send *command complete* status back to the host, at which point the host is available to execute new commands.

Note:

Both ¼-inch and 8 mm tape drives perform the erase procedure as described above. However, 8 mm tapes require between 2 and 2.5 hours to complete an erase operation. Thus, if you wish to initiate an erase operation on an 8 mm tape drive, issue the command as a background process or create another window so that you can continue working while the operation is being executed. Keep in mind, however, that you will not be able to access the drive that is erasing the tape until the operation has completed. The only way to interrupt an erase operation is to reboot the system.

9.4 Additional Commands for Floppy Disks

This section discusses additional commands that can be used to read from and write to floppy disks. These commands include the following:

- format** (to prepare the disk for storing data)
- mkfs** (to create a floppy disk file system)
- mf** (to mount a floppy disk file system)
- uf** (to unmount a floppy disk file system)
- cpflop** (to copy the contents of one floppy to another floppy)
- to_flop** (to write to multiple volumes of floppies)
- fr_flop** (to read data from multiple volumes of floppies)

However, before these commands are presented, a discussion on how the CLIX operating system accesses floppy disk drives is appropriate. First of all, before a floppy disk can be accessed, it must be formatted for CLIX (using the **format** command). After a floppy disk has been formatted, it can be written using one of two methods. You can create a *file system* on a floppy disk and use it like a CLIX file system (accessing it through UNIX commands like **mv** and **cp**). Or, you can use the floppy disk as a *raw device* (accessing it through backup utilities like **cpio** and **tar**). The raw device method is usually used for archiving data.

The floppy disk device file names the disk type. All disk device files reside in the **/dev/dsk** (for mountable disk types) and **/dev/rdisk** (for raw disk types). Intergraph offers 5¼-inch and 3½-inch floppy disk drives. The device files that represent the 5¼-inch floppy disk drive are **/dev/dsk/floppy** and **/dev/rdisk/floppy**, while the device files that represent the 3½-inch floppy disk drive are **/dev/dsk/ufloppy** and **/dev/rdisk/ufloppy**.

Most systems have only one floppy disk drive. However, some 6000-series systems may have both a 3½-inch and a 5¼-inch floppy disk drive. If your system has only one floppy disk drive, the device files representing that drive will be linked to a file called **/dev/(r)dsk/fl**. This file is called by the various backup utilities when the floppy disk drive is being accessed. If your system has both a 5¼-inch and 3½-inch floppy disk drive, the 3½-inch drive will automatically be assigned as the default (to be linked to **/dev/(r)dsk/fl**). When you wish to access the 5¼-inch drive through a backup utility, you will need to link **/dev/(r)dsk/fl** to **(/dev/(r)dsk/floppy**.

9.4.1 The format Command

Initially, floppy disks require formatting before they can be used as storage media. However, once the storage device has been formatted, you do not need to reformat. Formatting a floppy disk erases all data and rewrites the tracks and sectors to prepare the disk for holding data. Diskettes can be high- or low-density. High-density on Intergraph systems is quad-density, double-sided for floppy disks. Intergraph recommends using high-density diskettes.

	High-Density	Low-Density
5¼-inch	1.2 MB (2400 blocks)	.36 MB (720 blocks)
3½-inch	1.44 MB (28880 blocks)	.72 MB (1440 blocks)

Cautions:

- Formatting destroys all data on a floppy disk.
- A floppy disk formatted for System V must be reformatted to be used in PC-DOS. Reformatting is necessary because System V formats structure the disk differently than PC-DOS disks do.
- A high-density floppy disk should not be formatted as a low-density floppy disk and a low-density disk should not be formatted as a high-density disk.

Follow these steps to format a high-density floppy disk for System V using the format command:

1. Insert the high-density floppy disk in the floppy disk drive.
2. At the system prompt, key in the following:

```
$ /etc/format
```

The following messages appear:

```
Formatting high density floppy,  
total 512 blocks -2400  
Warning: All data on this floppy will be lost  
Insert floppy and hit return when ready
```

When the disk activity light goes out and the system prompt appears, the format procedure is complete.

Follow these steps to format a low-density floppy disk for System V using the **format** command:

1. Insert the low-density floppy disk in the floppy disk drive.
2. At the System V prompt, key in the following:

```
$ /etc/format -l
```

The following messages appear:

```
Formatting low density floppy,  
total 512 blocks -720  
Warning: All data on this floppy will be lost  
Insert floppy and hit return when ready
```

When the system prompt appears, the format procedure is complete.

9.4.2 The mkfs Command

The `mkfs` command (make file system) creates a file system on a floppy disk. Creating a file system allows a floppy disk to be used as a CLIX file system (such as `/usr`). If you do this, standard System V commands can be used to create and manipulate directories and files on the floppy disk.

Follow these steps to create a file system on a floppy disk:

1. Insert a formatted floppy disk in the floppy disk drive.
2. For a high-density floppy disk, key in the following (specifying 2,400 as the number of blocks that a high-density, 5¼-inch floppy disk stores or 2,880 as the number of blocks that a high-density, 3½-inch floppy disk stores):

```
# /etc/mkfs /dev/dsk/fl 2400
```

OR

```
# /etc/mkfs /dev/dsk/fl 2880
```

For a low-density floppy, key in the following (specifying 720 as the number of blocks that a low-density, 5¼-inch floppy disk stores or 1,440 as the number of blocks that a low-density, 3½-inch floppy disk stores):

```
# /etc/mkfs /dev/dsk/fl 2400
```

OR

```
# /etc/mkfs /dev/dsk/fl 2880
```

3. After creating a file system on a floppy, run the `labelit` program to attach a base directory name to the file system. For instance, to run `labelit` on the floppy disk, key in the following, where `mnt1` represents the mounted name of the file system and `mnt2` represents the (user definable) volume name:

```
# /etc/labelit /dev/dsk/floppy mnt1 mnt2
```

4. Like any file system, the floppy disk file system must be mounted before it can be accessed. The `mf` (mount floppy) command can be used to mount the floppy disk file system to the `/mnt` directory (an empty directory used for temporarily mounting additional file systems). Key in the following to mount the floppy disk file system:

```
$ mf
```

5. Once the floppy disk file system has been mounted, change to the `/mnt` as follows to access the floppy disk file system:

```
$ cd /mnt
```

You can now create, manipulate, and delete files located in the floppy disk file system directory.

6. When you have finished accessing the floppy disk, you must use the `uf` (unmount floppy) command to unmount the floppy disk file system. To do this, first change to another directory besides `/mnt`. Then key in `uf` to unmount the floppy file system as follows:

```
$ cd
```

```
$ uf
```

For more information on the `mkfs`, `mf`, and `uf` commands, refer to the manual pages in the System V Online Documentation product.

9.4.3 The `cpflop` Command

The `cpflop` (copy floppy) command copies the entire contents of one floppy disk to another floppy disk, using the same drive. To use this command, key in `cpflop` at the system prompt, as shown in the following example:

```
$ cpflop
Insert source floppy and press <Return> when ready ...
Insert destination floppy #1 and press <Return> when ready ...
1228800 bytes read
1228800 bytes written
```

For more information on the `cpflop` command, refer to the manual page in the System V Online Documentation product.

9.4.4 The `to_flop` Command

The `to_flop` command copies input to the floppy disk drive, prompting you to insert sequential floppy disks as needed. This command can be used with backup utilities such as `tar` and `cpio` to write multiple volumes of floppy disks. The following examples write the `/usr/chris` directory to one or more floppy disks:

```
$ tar cvf - /usr/chris | to_flop

OR

$ find /usr/chris -print | cpio -ov | to_flop
```

To restore files backed-up using `to_flop`, you must use the `fr_flop` utility described in the next section.

For more information on `to_flop` refer to the manual page in the System V Online Documentation product.

9.4.5 The `fr_flop` Command

The `fr_flop` command reads data from one or more floppy disks. This command can be used with backup utilities such as `tar` and `cpio` to restore files located on multiple volumes of floppy disks. The following examples restore the `/usr/chris` directory to the hard disk:

```
$ fr_flop | tar xvf -

OR

fr_flop | cpio -ivmud
```

For more information on `fr_flop` refer to the manual page in the System V Online Documentation product.

Blocks Remaining

This field indicates how many 512-byte blocks remain to be transferred.

Compare

This field allows you to compare the data specified in the "From" descriptors to the media specified in the "To" descriptors.

Copy

This field allows you to copy the data specified in the "From" descriptors to the media specified in the "To" descriptors. The size of the transfer will either be the entire size of the smaller partition or the size specified in the Block Count field as long as no partition boundaries are crossed.

The remaining screen buttons are used only when copying from or to cartridge tap.

Skip File

This field allows you to initiate a forward skip to the next file on the tape at the specified SCSI address.

Skip to EOT

This field allows you to initiate a forward skip to the end of the tape on the device at the specified SCSI address.

Rewind

This field allows you to rewind the tape on the device at the specified SCSI address.

Retension

This field allows you to retension the tape on the device at the specified SCSI address.

10.1.10 Rebuild Utility Page

The Rebuild Utility Page assists you in rebuilding the hard disk. From the Rebuild Utility Page, you will create acceptable partitions (using the default partitions or defining your own), load the Rebuild media on the hard disk, and enter the Rebuild environment (System V single-user mode). In single-user mode, the Rebuild menu provides the mechanism for restoring the file systems, loading software, and rebooting the system. For step-by-step instructions on rebuilding, refer to Chapter 11, "Rebuilding and Repartitioning the Hard Disk."

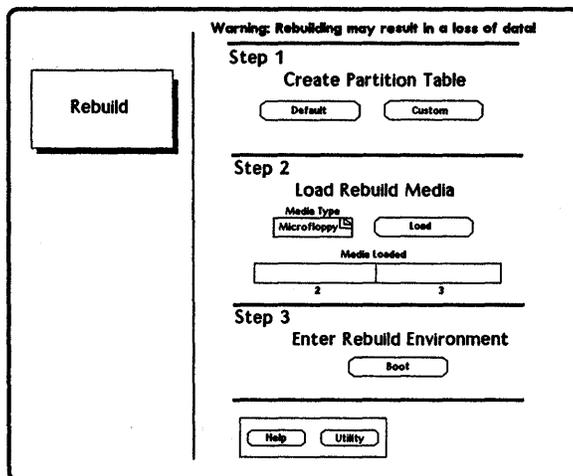


Figure 10-12. Rebuild Utility Page

The Rebuild Utility Page is designed to lead you through the process of rebuilding the hard disk. Notice that the page is separated into three steps (defining partitions, loading media, and entering the Rebuild environment). You will not be able to move on to the next step until you have completed the previous step satisfactorily. The left-hand portion of the screen contains important messages informing you of your current status and also identifying your next step.

Create Partition Table

Step 1 requires you to define acceptable hard disk partitions. (This documentation uses the terms *partition table* and *partitions* synonymously.) If your current partitions are not acceptable, you will not be allowed to proceed to step 2 until you redefine them. Messages on the left-hand portion of the screen will indicate specific partition requirements that must be met.

The Default icon uses the Intergraph-defined default partitions and the Custom icon allows you to define your own partitions. Selecting the Custom icon will send you to the Partition Utility Page to redefine hard disk partitions. After you have redefined the partitions, go to the Main Utility Page and then to the Rebuild Utility Page, to continue in the rebuild process.

Load Rebuild Media

Step 2 on the Rebuild Utility Page requires you to load the Rebuild media on the hard disk. You cannot begin this step until partitions acceptable for the hard disk have been defined.

To load the media, select your Rebuild media type (such as floppy) in the Media Type roll-through box. Then select the Load icon. You will be prompted to insert Rebuild Root floppy #2 and press <Return>. After you do so, the system begins loading the media onto the hard disk, and the Media Loaded bar chart indicates the percentage of data that has been loaded. When Rebuild Root floppy #2 finishes, you will be prompted to insert floppy #3 and press <Return>. Once again, the Media Loaded bar chart indicates the percentage of data that has been loaded from the media.

Enter Rebuild Environment

Step 3 on the Rebuild Utility Page requires you to enter the Rebuild environment. You cannot enter the Rebuild environment until you have loaded the Rebuild media on the hard disk.

Once you are ready to enter the Rebuild environment, select the Boot icon. This icon will cause the system to reboot into System V single-user mode and initiate the Rebuild menu. From this menu, you will be able to restore the file systems, invoke `newprod` to load software, and reboot the system. For more information on rebuilding, refer to Chapter 11, "Rebuilding and Repartitioning the Hard Disk."

10.2 Server Utility Menus

Follow these steps to access the Server Utility Pages:

1. Power up the InterServe.

OR

Reboot the InterServe by keying in /etc/reboot at the superuser (#) prompt or by pressing the reboot button.

2. Respond as directed at the following prompt:

```
Intergraph Corporation  
InterServe 200  
-----
```

Strike any key within 5 seconds to
access Startup Utility Menus.

The Startup Configuration Utility menu appears, as shown below.

```
-----  
Startup Configuration Utility  
-----
```

```
UTILITY      [go to main utility menu]  
UNIX        [boot unix operating system]  
Help
```

Key in your selection at the Option prompt.
Capitals indicate key-in abbreviations.

Option:

3. Key in the following at the Startup Configuration Utility menu to access the Main Utility Menu, from which the other utility menus can be accessed:

Option: **UT**

Chapter 11: Rebuilding and Repartitioning the Hard Disk

This chapter contains the following sections:

11.1	Backing Up Files	11-3
11.2	Creating Current Rebuild Floppy Disks	11-4
11.3	Booting from the Rebuild Boot Floppy Disk	11-7
11.4	Entering the Utility Pages	11-8
11.4.1	Entering the Utility Pages on a Workstation	11-8
11.4.2	Entering the Utility Pages on a Server	11-9
11.5	Verifying Hard Disk Flaws	11-10
11.5.1	Verifying a Workstation Hard Disk	11-11
11.5.2	Verifying a Server Hard Disk	11-12
11.6	Formatting the Hard Disk	11-13
11.6.1	Formatting Workstation Hard Disks	11-13
11.6.2	Formatting Server Hard Disks	11-14
11.7	Selecting the Rebuild Utility Page	11-16
11.8	Repartitioning the Hard Disk	11-17
11.8.1	Using the Current Partition Table	11-18
11.8.2	Repartitioning Using Default Partitions	11-19
11.8.3	Repartitioning Using Custom Partitions	11-20
11.8.3.1	Partitioning Overview	11-20
	Determining Partition Sizes	11-20
	Reallocating Space Between Partitions	11-23
	Understanding Partition Names	11-25
11.8.3.2	Creating Nonstandard Partitions	11-28
	Repartitioning Workstation Hard Disks	11-28
	Repartitioning an InterServe Hard Disk	11-29
11.9	Loading the Rebuild Root Media	11-32
11.10	Entering the Rebuild Environment	11-33
11.11	Restoring the File Systems	11-34
11.12	Loading Nucleus Software with newprod	11-42
11.13	Preparing Nonstandard Partitions for Use	11-52
11.14	Loading Application Software with the newprod Utility	11-53
11.15	Restoring Files from Backups	11-54

11-2 Rebuilding and Repartitioning the Hard Disk

Rebuilding the hard disk involves (at the minimum) recreating the file system(s). Existing data on a file system is overwritten; thus, rebuilding can be a destructive process. You must rebuild the hard disk under any of the following circumstances. Unless one of these circumstances is true, do not rebuild the hard disk.

- The hard disk is corrupted and you cannot boot your workstation.
- You cannot boot your workstation to single-user mode. (You never reach the blue introductory screen.)
- Bad block messages such as the following appear on the screen:

Disk failed: s0u0p7.1
read error at block 2933
- You want to change file system types. (For example, you want to convert */usr* from a standard file system to a Fast File System.)
- You want to alter the partition sizes on the hard disk.

This chapter describes how to rebuild the hard disk. Intergraph recommends that you complete the following steps in order to rebuild the hard disk. These steps will ensure that you lose as little data as possible when you rebuild. Each step is described in detail throughout the rest of this chapter.

1. Back up all personal files.
2. Create Rebuild floppy disks with the current software.
3. Boot from the Rebuild Boot floppy (#1)
4. Enter the Utility Pages.
5. Verify hard disk flaws.
6. Format the hard disk (optional).
7. Select the Rebuild Utility Page icon.
8. Repartition the hard disk (optional).
9. Load the Rebuild Root media.
10. Enter the Rebuild environment.
11. Restore the file systems.
12. Load nucleus software with the *newprod* utility.
13. Prepare nonstandard partitions for use.
14. Load application software with the *newprod* utility.
15. Restore files from backups.

11.1 Backing Up Files

Before you rebuild the hard disk you should first back up all personal files, including system files (such as */etc/passwd*, */etc/group*, and any *cron* tables), that are unique to the machine you are rebuilding. The rebuild process recreates the file systems on the hard disk and, in the process, overwrites all data on the hard disk. After the rebuild, you will not be able to access any files in the file systems that were recreated. Thus, before you begin rebuilding, back up all personal files that you want to keep.

For procedures for backing up files, refer to Chapter 9, "Backing Up and Restoring Files."

After you back up all personal files that you want to keep, continue to 11.2, "Creating Current Rebuild Floppy Disks."

11.2 Creating Current Rebuild Floppy Disks

After you have backed up your personal files, you should create or locate a current set of Rebuild floppy disks. The Rebuild floppy disk set is used to rebuild a corrupted System V hard disk to a bootable state.

New customers receive a set of Rebuild floppy disks with each workstation/server. Each Rebuild floppy disk set is specific to the machine it was shipped with. CLIPPER Rebuild floppy sets contain these diskettes:

- Rebuild Boot (floppy #1)
- Rebuild Root (floppy #2)
- Rebuild Root (floppy #3)

To rebuild, you will use the Rebuild Boot floppy disk appropriate for the machine you are rebuilding to boot and the Rebuild Root floppies to load the rebuild utility.

If your Rebuild floppy disk set is no longer current, you must create a current set. This section provides procedures for creating Rebuild floppy disks. You need three formatted, high-density floppy disks to create a Rebuild floppy disk set.

Note:

Do not attempt to make Rebuild floppy disks on the machine that you are rebuilding. Instead, use another machine that is completely operational.

Follow these steps to create a CLIPPER Rebuild floppy disk set:

1. You must be in superuser mode to perform this procedure. Log in and key in su at the system prompt as follows:

```
login: username
$ su
#
```

2. Invoke the newprod utility to load the rebuild product (ss01003) by keying in the following:

```
# newprod ss01003
```

3. Key in the correct entry at the following prompt:

Enter source of installation: n)etwork, f)loppy, t)ape, r)emote cdrom, or local c)drom →

If you key in n for network or r for remote cdrom, the following prompt appears:

Enter network connect nodename or address (08-00-36-XX-XX-XX) →

Key in the network address or the nodename of the delivery node where the new products reside and a valid username and password.

4. Messages similar to the following appear. Follow the instructions as prompted on the screen. To create a complete CLIPPER Rebuild floppy set, select "both" at the first prompt.

Installing: REBUILD (ss01003)
Installing...

Available Rebuild floppy sets:

1) 125/32C/220/225 Systems	(80186 IOP and ROP Graphics)
2) 200/240 Systems	(80186 IOP and optional IFB Graphics)
3) 300/400/3000/4000 Systems	(80386 IOP and optional IFB Graphics)
4) 2000 Systems	(Clipper IOP and optional MMG Graphics)
5) 6000 Systems	(Clipper IOP and optional EDGE Graphics)
6) None	

Which type of rebuild floppies do you want to make (1-6)? 1

Do you want to make a "root" set, a "boot" floppy, both or none? [both]

Downloading the Clipper Rebuild #1 (125/32C/200 Boot) floppy.
Insert a formatted floppy into the floppy drive and press <RETURN> →

You have a good "125/32C/200 Boot" floppy!
Label this floppy "Clipper Rebuild #1 (125/32C/200 Boot)"

The "125/32C/220/225 Root" floppy set will require 2 floppies.

Do you want to make Rebuild floppy #2 (y/n)? [y]
Do you want to make Rebuild floppy #3 (y/n)? [y]

Insert rebuild root floppy #2 and press <RETURN>
Writing the floppy
Verifying the floppy
Rebuild root floppy #2 created successfully

Insert rebuild root floppy #3 and press <RETURN>
Writing the floppy
Verifying the floppy
Rebuild root floppy #3 created successfully

Cleaning up...
Successful installation: REBUILD (ss01003)
Product installed in the /usr/ip32/rebuild directory

5. After these messages appear, press <Return> to return to the newprod menu. Key in q to exit the newprod utility.
6. Continue to 11.3, "Booting from the Rebuild Boot Floppy Disk."

11.3 Booting from the Rebuild Boot Floppy Disk

After you have created a set of Rebuild floppy disks with the current software, you must boot the workstation from the Rebuild Boot floppy disk. To rebuild the workstation, you may use the set of Rebuild floppy disks delivered with your workstation/server (if the set is not current) or you can use the Rebuild software product to create a set. The previous section describes how to create a set of Rebuild floppies.

To boot from the Rebuild Boot floppy, insert the Rebuild Boot floppy (appropriate for the machine that you will rebuild) in the floppy disk drive and boot the workstation/server. The machine will boot from the Rebuild floppy rather than from the hard disk.

Continue to 11.4, "Entering the Utility Pages."

11.4 Entering the Utility Pages

After you have booted from the Rebuild Boot floppy disk, you must enter the Utility Pages. You will use the Utility Pages to verify hard disk flaws, format the hard disk (if desired), repartition the hard disk (if desired), and load the rebuild media. After you have entered the Utility Pages, proceed to 11.5, "Verifying Hard Disk Flaws."

Because the interface is different for workstations and servers, this section separates the procedures for workstations and servers.

11.4.1 Entering the Utility Pages on a Workstation

When an Intergraph workstation boots from the Rebuild floppy disk, a blue introductory screen appears. You must move the mouse or cursor within five seconds if you wish to enter the Utility Pages. From the introductory screen, select the **Utility** icon to enter the Utility Pages. If the Utility Pages have been assigned a password, you will be prompted to enter the Utility Page password.

Note:

If the Utility Pages do not have a password, you may wish to assign one for security purposes. (Only the system manager should be allowed to access the Utility Pages because this environment allows you to format and repartition the hard disk.) To assign a password to the Utility Pages, select the **Workstation Password** icon on the Main Utility Page and key in the desired password when prompted.

The Utility Pages are a series of screens that allow you to configure the system. At the bottom of each Utility Page screen, notice the **Help** icon. To use the **Help** function, place the cursor on the item for which you want information and tap the reset button.

11.4.2 Entering the Utility Pages on a Server

When an Intergraph server boots from the Rebuild floppy disk, an introductory screen appears. You must press any key within five seconds to enter the Startup Utility Pages. From this screen, key in UT to enter the Utility Pages. If the Utility Pages have been assigned a password, you will be prompted to enter the Utility Page password.

Note:

If the Utility Pages do not have a password, you may wish to assign one for security purposes. (Only the system manager should be allowed to access the Utility Pages because this environment allows you to format and repartition the hard disk.) To assign a password to the Utility Pages, key in PW at the Main Utility Page and key in the desired password when prompted.

The Utility Pages are a series of screens that allow you to configure the system. At the bottom of each Utility Page screen, notice the Help option. To use the Help function, key in h followed by the string required to select the option. For example, to read Help information on the Disk Partitioning screen, key in h dp. Notice that the Help function is not case-sensitive.

11.5 Verifying Hard Disk Flaws

From the Utility Pages, you will be able to verify the locations of hard disk flaws by running the Verify procedure. This procedure reads the hard disk and records all flaws to ensure that data is not written over flawed areas. It does not erase or change data on the disk.

If you are rebuilding because you wish to repartition the hard disk or convert to a different file system type, you do not need to run the Verify procedure. However, if you are rebuilding the hard disk to correct bad block messages or to correct a corrupted disk, you must run the Verify procedure.

The following is an example of a bad block message:

```
Disk failed: s0u0p07.1 medium error:  
read error at block 2933
```

Bad blocks are flawed areas on the hard disk. As the disk drive writes data on the disk, it calculates an Error Correction Code (ECC) for each block and stores the code at the end of each block. When the drive reads data from the disk, it calculates another ECC and compares it to the original stored value. If the codes match, the data stored in that block is healthy and uncorrupted. If the codes do not match, the drive generates an ECC error and the host displays a medium error or a bad block message.

Most Intergraph disks have Automatic Read and Write Reallocation. This means that if the system finds an error that it cannot recover from (save the data, move it to a new location, and experience no data loss), the block will be automatically reallocated to another sector of the disk and the original data will be rewritten there (with no data loss).

However, if the system finds an unrecoverable data error while executing a read or write operation, it will report a medium error. If you receive one or more medium errors, run the Verify procedure on the disk to register and reallocate any other bad blocks that may have developed.

The Verify procedure attempts to read every location on the disk. It checks the data by comparing a computed value of ECC with a value that was computed when the data was written and stored on the disk at the end of each track. If the ECC code does not match, the procedure reads the disk again. One retry is categorized as a soft error. Two or more retries indicate a hard error. Hard errors are flagged as bad blocks. If the procedure finds a bad block, the host reassigns the bad block to a spare location on the disk.

After all bad blocks are found, the disk is clean and ready to go. A bad block indicates that a block of data has an error. As a result, the data in this block may not be recoverable.

Procedures differ for workstations and servers and are described in separate sections.

11.5.1 Verifying a Workstation Hard Disk

If the Verify procedure fails, you will need to reformat your disk, as described in 11.6, "Formatting the Hard Disk." Follow these steps to Verify the hard disk:

1. From the Main Utility Page, select the Disk Maintenance Utility screen button.
2. Select the Next Page screen button to access the Hard Disk Flaw Data Utility Page.
3. Select the Verify screen button to access the Verify page.
4. From the Verify page, check to see that the SCSI ID, Disk Capacity, and Logical Unit Number fields contain the correct settings for the disk you want to verify.

The SCSI ID and LUN for internal hard disks are 0. The disk capacity depends on the internal hard disk size.

5. Select the Verify screen button to begin the Verify process.
6. If the hard disk is corrupted with numerous bad blocks, the Verify process may not complete. If the process is not successful, you must format the hard disk to mark all bad blocks. Proceed to 11.6, "Formatting the Hard Disk."

7. If the verification process completes successfully, you will not be required to complete step six in the rebuild process, "Formatting the Hard Disk," but you may if you wish. Proceed to 11.6, "Formatting the Hard Disk," for more information.

11.5.2 Verifying a Server Hard Disk

If the Verify procedure fails, you will need to reformat your disk, as described in 11.6, "Formatting the Hard Disk." Follow these steps to Verify the hard disk:

1. From the Main Utility Page, key in the following to display the Disk Maintenance Utilities menu:

Option: DM

2. Check to see that the SCSI__id, Logical__Unit__Number, and Disk Capacity fields contain the correct settings for the disk you want to verify.
3. Key in the following to select the VeriFY option, which runs the verification procedure.

Option: VFY

4. If the hard disk is corrupted with numerous bad blocks, the Verify process may not complete. If the process is not successful, you must format the hard disk to mark all bad blocks. Proceed to 11.6, "Formatting the Hard Disk."
5. If the verification process completes successfully, you will not be required to complete step 6 in the rebuild process, "Formatting the Hard Disk," but you may if you wish. Proceed to 11.6, "Formatting the Hard Disk," for more information.

11.6 Formatting the Hard Disk

If the Verify procedure fails, you must format the hard disk. Otherwise, formatting is optional. Formatting overwrites all data on the hard disk; therefore, you should not format unless the Verify procedure failed.

Formatting involves structuring the disk so that hardware and software can communicate with the disk. This process writes a test pattern to the disk and reads it to verify the pattern, checks for any errors, and marks locations on the disk (flaws) that cause errors. These flaws are recorded to prevent any future writing to or reading from that location.

Rebuilding without formatting is possible. In many cases, only part of the hard disk is corrupted. You may be able to restore the corrupted file system(s) without formatting the hard disk. If you format, you will lose all data on the disk; if you do not format, you will lose only the data in file systems that you restore. If you do not wish to format the hard disk, proceed to 11.7, "Selecting the Rebuild Utility Page Icon."

Formatting takes approximately 45 minutes for 80-MB hard disks, 30 minutes for 156-MB hard disks, 25 minutes for 180-MB hard disks, 12 minutes for 200-MB hard disks, 30 minutes for 355-MB hard disks, and 50 minutes for 670-MB hard disks.

Caution:

Formatting destroys all data on the disk. Therefore, back up all personal files before formatting.

Formatting procedures for workstations and servers differ and are described separately.

11.6.1 Formatting Workstation Hard Disks

Follow these steps to format workstation hard disks:

1. From the Main Utility Page, select the Disk Maintenance Utility Page.
2. When the Disk Maintenance Utility Page appears, select the following settings for an internal hard disk:

SCSI ID = 0
Logical Unit Number = 0

For an external hard disk, you would set the SCSI ID to 1, 2, or 3, and the LUN would be 0.

3. Match the Intergraph Part Number on the Hard Disk Profile Sheet with the part number in the Disk Type roll-through box.
4. Key in the serial number from the Hard Disk Profile Sheet if it does not display in the Serial Number data entry field.
5. Select the Next Page screen button to access the Hard Disk Flaw Data Utility Page. Select the following settings:

Format With Flaw Data - Yes

Selecting **yes** for the Format With Flaw data option prevents flaws on the disk from being ignored during the format process.

Verify on Format - Yes

Selecting **yes** for the Verify on Format option runs the Verify procedure to record any new flaws found on the disk during the format.

6. Select the Format screen button to access the red Format Utility Page.
7. Select the Format screen button to start the procedure. While the format procedure is running, messages revealing the part of the disk being verified and identifying the bad blocks appear on the screen. You do not need to respond to any of these messages. The message "Format Complete" appears when formatting is finished. Select the Previous Page screen button to return to the Hard Disk Flaw Data Utility Page.
8. Proceed to 11.7, "Selecting the Rebuild Utility Page," for the next step in the rebuild process.

11.6.2 Formatting Server Hard Disks

Follow these steps to format an InterServe hard disk:

1. From the Main Utility Page, key in DM to display the Disk Maintenance Utilities menu.
2. Key in the following to set the SCSI ID to 0:

Option: SCSI 0

3. Key in the following to set the logical unit number to 0:

Option: **LUN 0**

4. Key in the following, where **<serial number>** is the correct serial number for the disk as the Hard Disk Profile Sheet indicates:

Option: **SN <serial-number>**

5. Key in the following so that flaw data will be considered during the format:

Option: **FWF Y**

6. Key in the following for the flaw data list to be verified during the format and for any additional bad blocks to be added to the list and registered:

Option: **FWV Y**

7. Key in the following to start the format procedure:

Option: **FORMAT**

While the format procedure is running, messages revealing the part of the disk being verified and identifying the bad blocks appear on the screen. The message "Format Complete" appears when formatting is finished. (Formatting takes approximately 45 minutes for 80-MB hard disks, 30 minutes for 156-MB hard disks, 25 minutes for 180-MB hard disks, 30 minutes for 355-MB hard disks, and 50 minutes for 670-MB hard disks.)

8. Proceed to 11.7, "Selecting the Rebuild Utility Page," for the next step in the rebuild process.

11.7 Selecting the Rebuild Utility Page

After verifying and formatting the hard disk, the next step in the rebuild process is selecting the Rebuild Utility Page from the Main Utility Page. Do so by selecting the Rebuild icon on the Main Utility Page (for workstations) or keying in REBUILD at the Main Utility Page (for servers).

The Rebuild Utility Page allows you to repartition, load Rebuild media, and enter the Rebuild environment. Before you can begin restoring the file systems, you must have specified an acceptable partition table, successfully loaded the Rebuild media, and selected the icon to enter the Rebuild environment. If any of these steps has not been completed, you will not be allowed to continue the rebuild process.

Notice that the left portion of the Rebuild Utility Page displays messages concerning your current status and your next step. While you are using the Rebuild Utility Page, refer to this message area for brief instructions.

Proceed to 11.8, "Repartitioning the Hard Disk."

11.8 Repartitioning the Hard Disk

From the Rebuild Utility Page, you may repartition the hard disk. This step is not required unless the current partitions are unacceptable for the rebuild procedure (as described in 11.8.1).

Partitioning logically divides the hard disk into separate sections or *devices* for specific areas of user and operating system functions. You must repartition the hard disk only if the current partitions are not acceptable for the Rebuild software. You may also repartition to increase or decrease the size of specific partitions or to add nonstandard partitions. For example, you may need to add a stash partition to produce large plots on raster output devices. See the *InterPlot User's Guide* for more information on creating a stash partition.

Note:

The Rebuild Utility Page refers to the partitions on the disk as the *partition table*. This documentation will use the terms *partitions* and *partition table* interchangeably.

During the rebuild process, you must choose one of the following options concerning hard disk partitions:

- Use the current partitions. (Refer to 11.8.1.)
- Repartition using Intergraph-defined default partitions. (Refer to 11.8.2.)
- Repartition using custom partitions. (Refer to 11.8.3.)

11.8.1 Using the Current Partition Table

If the current partition table is acceptable for the Rebuild software and you do not wish to alter it, you will not need to repartition the hard disk. Partitions must meet the following requirements before the Rebuild software will accept them:

- The root partition must have at least 25,000 blocks.
- The swap partition must have at least 15,000 blocks.
- The usr partition must exist.

Note:

Other software products may have additional partition requirements. The system manager should determine the partition sizes appropriate for the system and the software the system uses. Refer to 11.8.3.1, "Partitioning Overview," for information on factors that you should consider when you establish partition sizes.

Follow these steps if you wish to use the current disk partitions:

1. Access the Rebuild Utility Page.
2. If the messages on the left portion of the screen indicate that the current partitions are not acceptable, you must repartition the hard disk. Proceed to 11.8.2 to repartition using default partitions or 11.8.3 to repartition using custom partitions.
3. If the messages on the left portion of the screen indicate that the partitions are acceptable, proceed to 11.9, "Loading the Rebuild Media."

11.8.2 Repartitioning Using Default Partitions

If you are not currently using the (Intergraph-defined) default partitions but wish to, you must repartition the hard disk using the Default option supplied on the Rebuild Utility Page.

Caution:

You will lose all data on each partition that you alter.

The following are the default partition sizes that Intergraph has established:

	80 MB	156 MB	200 MB	355 MB	584 MB	670 MB
boot:	3988	7988	7988	7988	7988	7988
root:	25000	25000	25000	25000	25000	25000
swap:	27360	71000	71000	71000	71000	71000
usr:	100000	200600	288053	590298	1037988	1204900

Note:

The total number of blocks in the partition table and the disk capacity differ. They differ because a one-block header is associated with each partition. The header blocks are included in the disk capacity count and not in the partition table count.

Follow these steps to repartition using default partitions:

1. Access the Rebuild Utility Page.
2. Select the Default icon under the Create Partition Table heading. This menu option will automatically redefine all hard disk partitions according to the defaults.

Caution:

You will lose all data in any partition that is altered.

3. If the messages on the left portion of the screen indicate that the partitions are acceptable, proceed to 11.9, "Loading the Rebuild Media."

11.8.3 Repartitioning Using Custom Partitions

This section describes repartitioning the hard disk using custom partitions rather than using the Intergraph-defined default partitions described in the previous section. For example, you may want to create a `usr2` or `usr3` partition by taking space from `usr`.

Caution:

You will lose all data on each partition that you alter.

- If you are not familiar with the concept of repartitioning the hard disk, read 11.8.3.1, "Partitioning Overview," before you begin the process of actually repartitioning the hard disk.
- If you are familiar with the concept of repartitioning the hard disk, proceed to 11.8.3.2, "Creating Nonstandard Partitions."

11.8.3.1 Partitioning Overview

This section provides information concerning the following topics:

- Determining Partition Sizes
- Reallocating Space Between Partitions
- Understanding Partition Names

Determining Partition Sizes

Intergraph divides workstation/server hard disks into the following partitions:

- `boot` (system)
- `root`
- `swap`
- `DOS` (32C workstations only)
- `usr`

Note:

Intergraph does not automatically place a `stash` partition (for plotting) on hard disks. However, some plotters require `stash` on the plot server. A `stash` may be either a partition on the hard disk or physical memory. Refer to the *InterPlot User's Guide* for information on creating a `stash` partition.

The `boot` partition contains boot images, including the hardware diagnostics test, Utility Pages, communications processor, and System V kernel software. This partition contains 3,988 blocks for workstations with 80-MB hard disks and 7,988 blocks for all other hard disk sizes. You cannot access this partition on the Disk Partitioning Utility Page because you cannot alter its size.

The **root** partition contains a major portion of the System V operating system. It is used mainly for system administrative tasks. By default, this partition contains 25,000 blocks. You should not take space from this partition, and if you receive messages saying that the root (7.0) partition is out of space, you must add space to it.

The **swap** partition is used for swapping portions of memory to the hard disk. By default, it contains 27,360 blocks for 80-MB hard disks and 71,000 blocks for all other hard disk sizes. Space can be removed from the **swap** partition and added to another partition. However, some applications will not run without a minimum amount of swap space. This minimum amount varies among applications.

The **usr** partition contains user directories, most products, and any other directories and files that users access. By default, the **usr** partition contains 100,000 blocks for 80-MB hard disks; 200,600 blocks for 156-MB hard disks; 288,053 blocks for 200-MB hard disks; 590,298 blocks for 355-MB hard disks; 1,037,988 blocks for 584-MB hard disks; and 1,164,276 blocks for 670-MB hard disks. Because this partition is the user's *work* partition, it contains the most free space. You may wish to remove space from **usr** and form a **usr2** partition.

You must have a **stash** on your plot server to plot on most plotting devices. A **stash** may be either a partition on your hard disk or physical memory. (Physical memory may be used only if sufficient physical memory is available on your server. See section 1.13 in the *InterPlot User's Guide*.) The following information illustrates how to calculate the size for a **stash** partition on your hard disk.

Pen plotters and the following raster devices do not require a **stash**:

- Shinko 65, 635, and 645
- ILP811
- ILP2217
- Calcomp 5602 and 5602BW
- Versatec V80 (for B-size plots)
- Versatec 8524LR and 8536LR

Use the following guidelines to determine the size of your **stash** partition:

- On a 200-dpi plotter (such as the Versatec V7224), plots require 10 blocks per square inch for processing. Therefore, a 20- X 20-in plot requires 4000 blocks for processing. This conclusion is based on the following calculations:

$$\begin{aligned} 20 \text{ in X } 20 \text{ in} &= 400 \text{ sq in} \\ 400 \text{ sq in X } 10 \text{ blocks per sq in} &= 4000 \text{ blocks} \end{aligned}$$

- On a 400-dpi plotter (such as the Versatec 7436), plots require 40 blocks per square inch for processing. Therefore, a 15- X 15-in plot requires 9000 blocks for processing. This conclusion is based on the following calculations:

$$\begin{aligned} 15 \text{ in X } 15 \text{ in} &= 225 \text{ sq in} \\ 225 \text{ sq in X } 40 \text{ blocks per sq in} &= 9000 \text{ blocks} \end{aligned}$$

Note:

A 300-dpi plotter requires 22 blocks per square inch for processing. However, Intergraph's 300-dpi laser printer (ILP811) does not require a **stash** partition.

Stash must be allocated in 124-block increments. Therefore, the number of blocks in your **stash** partition must be a multiple of 124. If your **stash** calculation is not a multiple of 124, follow these steps:

1. Divide the number that you have calculated for **stash** by 124.
2. Round the quotient up to the next whole number.
3. Multiply the total by 124.

For example, if you have determined that you need 9000 blocks for **stash**, use this formula:

$$\begin{aligned} 9000/124 &= 72.5806 \quad (\text{Round up to } 73.) \\ 73 \text{ X } 124 &= 9052 \end{aligned}$$

Your **stash** will have 9052 blocks.

The following chart summarizes disk stash requirements:

<i>Device Type</i>	<i>Blocks Per Square Inch</i>
200-dpi device	10 blocks per square inch
400-dpi device	40 blocks per square inch
QCR camera (2K mode)	6400 blocks
QCR camera (4K mode)	24,700 blocks
Other device types	$(\text{dpi})^2/4096$ blocks per square inch

Follow these steps to determine the size of your stash partition:

1. Determine the size (in square inches) of the largest plot you will produce. This example assumes the largest plot is an E-size plot (36 X 44 in).

$$(36 \times 44 \text{ in}) = 1584 \text{ sq in}$$

2. Determine the number of blocks per square inch required for processing. For example, if your plotter is a 400-dpi device, it will require 40 blocks per square inch for processing. Multiply the size of the largest plot by 40 as follows:

$$1584 \text{ sq in} \times 40 \text{ sq in} = 63,360 \text{ blocks}$$

3. Round the number of blocks required for processing to the next highest multiple of 124.

$$63,360/124 = 510.9677$$
$$511 \times 124 = 63,364$$

Therefore, the stash partition for this example requires 63,364 blocks.

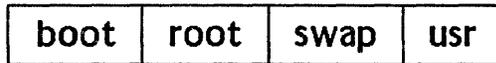
Note:

The partition number for stash is *f*, the hexadecimal representation of the decimal number 15. The modifier number for this partition is *0*. Thus, the device file for a stash partition on SCSI ID 0 would be *s0u0pf0*.

Reallocating Space Between Partitions

Before you can take space from one partition and add it to another, the two partitions must be adjacent on the disk.

The order for standard partitions on hard disks is as follows:



For example, you could remove the `usr` partition and add it directly to the `swap` partition (since these partitions are adjacent). However, you could not remove the `usr` partition and add it directly to `root`. You could add the `usr` space to `root` only by adding it to the `swap` partition and then removing it from `swap` and adding it to `root`.

In addition, when you add a partition, the system will search only forward on the disk for space to claim. That is, to remove the `swap` partition and add that space to the `usr` partition, you could not remove the `swap` partition and add a larger `usr` partition. Doing so would cause the system to search for and allocate space from the beginning of the `usr` partition and beyond. Instead, you would need to remove the `swap` and the `usr` partition and then add a larger `usr` partition. In this case, the system would start allocating space from the beginning of the (former) `swap` partition.

Caution: Repartitioning the hard disk loses any information in a moved, removed, or resized partition. Therefore, back up all files before repartitioning.

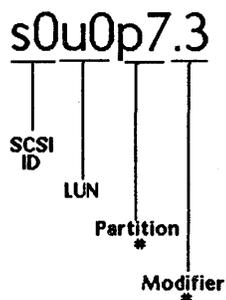
Understanding Partition Names

Partition names appear in the following form on the Disk Partitioning Utility Pages:

sAuBpC.D

The following table defines the components of this partition name:

<i>Labels</i>	<i>Variables</i>
s = SCSI ID	A: SCSI ID number
u = unit	B: logical unit (drive) number
p = partition	C: general-purpose division number (partition)
	D: modifier number (subpartition)



The chart on the following pages lists partition numbers for all partitions on Intergraph workstations.

Disk Partition	100/200 Series	300/3000 Series
2.0	Flaw Map	Flaw Map
4.0	Error Map	Error Logging
5.0	Configuration	Configuration
8.0	80186 IOP Code	80386 IOP Code
8.1	ROP Code #0	Not Used
8.2	Blue Screen Code	Blue Screen Code
8.3	UNIX Boot Leader	UNIX Boot Loader
8.4	80186 IOP Code	Not Used
8.5	ROP Bios Code	Digitizer Code
8.6	Coprocessor Code	FPE Exec Code
8.7	80186 CP Code	80386 CP Code
8.8	Bios Font Data	Not Used
8.9	ROP Code #1	Not Used
8.a	Downloaded Diagnostics	Not Used
8.b	ROP Code #2	Not Used
8.c	ROP Code #3	IFB Code (MAIN)
8.d	ROP Code #4	IFB Code (ROAM)
8.e	ROP Code #5	IFB Code (ROAM)
a.0	Not Used	Downloaded FDM
a.1	Not Used	IFB Diagnostics
a.2	Not Used	UNIX Engine II Diag
a.3	Not Used	UNIX Engine II Diag (IOP)
a.4	Not Used	IOP+ Diagnostics
a.5	Not Used	FPE Diagnostics
a.6	Not Used	HC/DIG Diagnostics
a.7	Not Used	HC/DIG Diagnostics (IOP)
a.8	Not Used	ISI Diagnostics
a.9	Not Used	VME Diagnostics
b.f	Disk Diagnostic Track	Disk Diagnostic Track
7.0	root File System	root File System
7.1	Swap Area	Swap Area
7.3	usr File System	usr File System
7.4	usr2 File System	usr2 File System
7.5	usr3 File System	usr3 File System
9.0	DOS File System	DOS File System
f.0	Plotting Stash	Plotting Stash
10.0	Unused Space	Unused Space
ff.0	End Of Disk	End Of Disk

Disk Partition	Series 2000	6000 Series	6100/6500 Series
2.0	Flaw Map	Flaw Map	Flaw Map
4.0	Error Map	Error Map	Error Map
5.0	Configuration	Configuration	Configuration
8.0	IO Code	IO Code	IO Code
8.1	ROP Code #0	Not Used	Not Used
8.2	Blue Screen Code	Blue Screen Code	Blue Screen
8.3	Not Used	Not Used	Not Used
8.4	Not Used	Not Used	Not Used
8.5	Not Used	Not Used	Not Used
8.6	Not Used	Not Used	Not Used
8.7	Not Used	Not Used	Not Used
8.8	Not Used	Not Used	Not Used
8.9	Not Used	Not Used	Not Used
8.a	Not Used	Not Used	Not Used
8.b	Not Used	Not Used	Not Used
8.c	Not Used	Not Used	Not Used
8.d	Not Used	EDGE I Code	EDGE I Code
8.e	Not Used	EDGE II Code	EDGE II Code
a.0	FDMDISK	FDMDISK	FDMDISK
a.1	System Board	EDGE I	EDGE I
a.2	Video	EDGE II	EDGE II
a.3	Digitizer	EDGE III	EDGE III
a.4	Not Used	System Board	IOI
a.5	Not Used	Not Used	IOI CE
a.6	Not Used	VME (Dianostics)	VME (Diagnostics)
a.7	Not Used	Digitizer	Digitizer
a.8	Not Used	ISI	ISI
a.9	Not Used	VME Monitor	VME Monitor
a.a	Not Used	SIP	SIP
a.b	Not Used	FIP	FIP
a.c	Not Used	FDDI	FDDI
a.d	Not Used	SR SCSI	SR SCSI
a.e	Not Used	Not Used	C3+
a.f	Not Used	32 Channel	32 Channel
b.0	Not Used	EDGE DSP (Mon.)	EDGE DSP (Monitor)
b.1	Not Used	Frame Grabber	Frame Grabber
b.2	Not Used	Not Used	Not Used
b.f	Disk Diag. Track	Disk Diag. Track	QWIK
7.0	root File System	root File System	root File System
7.1	Swap Area	Swap Area	swap Area
7.3	usr File System	usr File System	usr File System
7.4	usr2 File System	usr2 File System	usr2 File System
7.5	usr3 File System	usr3 File System	usr3 File System
9.0	Not Used	Not Used	Not Used
f.0	Plotting Stash	Plotting Stash	Plotting Stash
10.0	Unused Space	Unused Space	Unused Space
ff.0	End Of Disk	End Of Disk	End of Disk

11.8.3.2 Creating Nonstandard Partitions

This section supplies the general steps for creating new partitions. The system manager should decide how much space must be allocated for each partition. For example, to add a stash partition for plotting, refer to 11.8.3.1, "Determining Partition Sizes," to determine how large the stash partition must be and which partition(s) space should come from. Then, follow steps in this section for adding partitions and altering partition sizes. For more information on creating a stash partition, refer to the *InterPlot User's Guide*.

Caution: Repartitioning wipes out all data on any altered partition. Therefore, you should back up all files before repartitioning.

Repartitioning Workstation Hard Disks

Follow these steps to repartition workstation hard disks:

1. Select the Custom icon from the Rebuild Utility Page. This selection will transfer control to the Disk Partitioning Utility Page.
2. Notice the chart displaying the hard disk partitions at the top of the Disk Partitioning Utility Page. Study this chart to determine how you wish to alter the partitions. Keep in mind the following restrictions enforced by the Rebuild software:
 - The root partition must have at least 25,000 blocks.
 - The swap partition must have at least 15,000 blocks.
 - The usr partition must exist.

Note:

Other software products may have additional partition requirements. The system manager should determine the partition sizes appropriate for the system and the software the system uses. Refer to 11.8.3.1, "Determining Partition Sizes," for information on factors you should consider when you establish partition sizes.

3. From the Disk Partitioning Utility Page, you can add and remove partitions. Thus, to change partition sizes, you must remove all partitions being altered and then recreate them with new sizes.

To remove a partition, scroll through the Partition Names roll-through box until the name of the partition to remove appears. If the desired name is not in the roll-through box, key in the Partition Number and the Modifier Number (in their respective fields) for the partition to be removed. Select the Remove Partition button to remove the partition currently displaying in the Partition Names roll-through box.

Enter the partition number and modifier number of the new partition. Then, enter the Size in Blocks of the new partition. To add the specified partition, select the Add Partition button. The partition will be added in the first available slot that will accommodate that size.

Notes:

If you plan to make the size of the last partition equal to the amount of available free space listed in the partition table, you must subtract two blocks from the amount of free space. (The partition header and the "End of Disk" header require one block each.)

4. Notice that the bar chart at the top of the screen reflects adjustments to the disk partitions.
5. After you have defined all hard disk partitions, select the Utility icon to return to the Main Utility Page and then the Rebuild icon to return to the Rebuild Utility Page.
6. Proceed to 11.9, "Loading the Rebuild Media."
7. Keep in mind that you must create a device file and then create and mount a file system on any nonstandard partition that you add (such as `usr2` or `usr3`). These steps are described in 11.13.

Repartitioning an InterServe Hard Disk

Follow these steps to repartition an InterServe hard disk:

1. Select the Custom icon from the Rebuild Utility Page. This selection will transfer control to the Disk Partitioning Utility Page.
2. Key in the following to list the partitions currently on the hard disk:

Option: LP

Keep in mind the following restrictions enforced by the Rebuild software:

- The root partition must have at least 25,000 blocks.
- The swap partition must have at least 15,000 blocks.
- The usr partition must exist.

Note:

Other software products may have additional partition requirements. The system manager should determine the partition sizes appropriate for the system and the software the system uses. Refer to 11.8.3.1, "Determining Partition Sizes," for information on factors that you should consider when you establish partition sizes.

3. From the Disk Partitioning Utility Page, you can add and remove partitions. Thus, to change partition sizes, you must remove all partitions being altered and then read them with new sizes.

To remove a partition from an InterServe hard disk, first key in the following to list the existing partitions:

Option: LP

Next, write down the partition and modifier numbers for the partitions you will remove (such as 7 and 3 for usr). Refer to 11.8.3.3 for an explanation of partition and modifier numbers.

Then, key in the following command line to remove a partition, substituting the appropriate values in each field:

Option: RMP <partition number> <modifier number>

- RMP is the ReMove_Partition option, which removes the partition designated in the same line from the disk.
- <partition number> is the number of the partition to be removed.
- <modifier number> is the modifier number of the partition to be removed.

To add a partition to an InterServe hard disk, first key in the following to select the Add_SYStem option. This option reserves the first 7988 blocks of disk space for the bootable partitions and images that you must create later when rebuilding the hard disk with the Rebuild floppy disk set (described in 11.11, "Restoring the File Systems").

Option: **ASYS**

Key in the following to add each desired partition to the disk:

Option: **ADDP** <partition number> <modifier number> <size>

ADDP Selects the **ADD_Partition** option described in Chapter 8, "Utility Page Reference."

Note:

If you plan to make the size of the last partition equal to the amount of available free space listed in the partition table, you must subtract two blocks from the amount of free space. (The partition header and the "End of Disk" header require one block each.)

4. Key in the following to select the **List_Partitions** option, which lists existing partitions and verifies that the desired partitions have been added:

Option: **LP**

5. After you have defined all hard disk partitions, key in **UT** to return to the Main Utility Page and then **REBUILD** to return to the Rebuild Utility Page.
6. Proceed to 11.9, "Loading the Rebuild Media."
7. Keep in mind that you must create a device file and then create and mount a file system on any nonstandard partition that you add (such as **usr2** or **usr3**). These steps are described in 11.13.

11.9 Loading the Rebuild Root Media

The next step in the rebuild process is to load the Rebuild Root media. This step loads minimal file systems on the hard disk. The minimal file systems are stored temporarily in the swap space. When you actually begin restoring the file systems, these file systems will be copied from swap to the appropriate partitions.

Follow these steps to load the Rebuild Root media:

Note:

You can rebuild your workstation/server with different media types, including microfloppy (3½-inch diskettes) and floppy disk. The default is microfloppy for 2000/6000-series systems and floppy for all others.

1. Locate (or create) the Rebuild Root media. If your requested media type is floppy disk, locate or create current Rebuild Root floppy disks.
2. Select the appropriate media type (for example, floppy) from the roll-through box in the Load Rebuild Root Media portion of the Rebuild Utility Page.
3. Select the Load icon. When prompted, insert the first Rebuild Root floppy (#2) in the drive and press <Return>.
4. As the media is being loaded on the hard disk, notice the graphic display representing the portion of data that has been loaded from each floppy. This graphic is purely informational; you do not need to respond to it.
5. When floppy #2 has been loaded, remove it from the drive, insert the Rebuild Root floppy (#3) in the drive, and press <Return>.

Note:

If a "CRC error" is detected while you are loading the media, try again to load the media. If the media still does not load successfully, recreate the Rebuild Root disk.

6. Proceed to 11.10, "Entering the Rebuild Environment."

11.10 Entering the Rebuild Environment

After loading the Rebuild media, you may enter the Rebuild environment. After you have loaded the Rebuild media on the hard disk, select the Boot icon on the Rebuild Utility Page. This icon will boot the workstation into single-user mode and initiate the Rebuild menu. You do not need to boot from the Rebuild floppy disks because the Rebuild media has already been loaded on the hard disk (in swap space). Thus, you may remove Rebuild media from the disk drive before you boot into the Rebuild environment.

After the workstation has booted into single-user mode, proceed to 11.11, "Restoring the File Systems."

11.11 Restoring the File Systems

After you have entered the Rebuild environment, you may restore minimal file systems to the hard disk. Restoring minimal *root* and */usr* file systems prepares the hard disk so that software can be loaded through the *newprod* utility.

The Rebuild environment consists of a series of menus that allow you to perform tasks such as restore the *root* and */usr* file systems, create new file systems (such as */usr2*), install software using the *newprod* utility, reboot the system, and exit the menu.

The menu options you should choose will depend on the extent you need to rebuild. This section describes each option available.

After the workstation/server boots to System V single-user mode, the following screen appears:

```

Welcome to Rebuild

Use the arrow keys to navigate the menus, and then use the
RETURN key to select menu items.

Use the top two rows of the keypad to select the functions that
are shown at the bottom of the display.

TOP TWO ROWS OF THE KEYPAD:

CANCEL(pf1)  PREVPAGE(pf2)  NEXTPAGE(pf3)  (pf4)
(7)          HELP (8)      END-MENU (9)   CONTINUE (-)

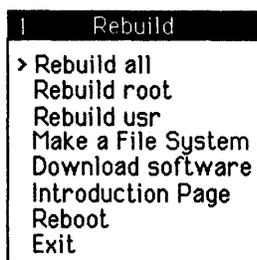
Press CONTINUE to go on to the Rebuild Menu

CANCEL  PREVPAGE  NEXTPAGE  (7)  (8)  (9)  CONTINUE

```

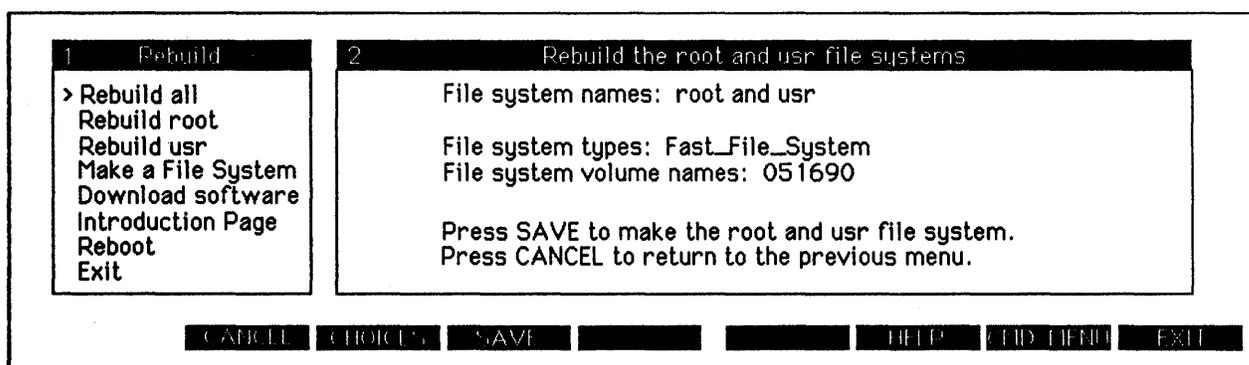
This screen provides an introduction to the rebuild utility. Notice that the menus require you to use the function keypad to move back and forth between menu levels. To continue to the main rebuild menu, press the CONTINUE (-) key on the function keypad.

The following menu appears. This menu is the main rebuild menu. From this menu, you can perform each step in the rebuild process from restoring (recreating) the *root* and/or */usr* file systems to loading software and then booting off the rebuilt hard disk. To select a menu option, press <Return> over the option.



Rebuild all

The following menu displays if you choose the "Rebuild all" option from the main rebuild menu:



This menu option creates minimal *root* and */usr* file systems. If you are performing a complete rebuild, you will need to restore both *root* and */usr*. This option creates *root* and */usr* as the same file system types. For example, if you choose this option, you cannot create *root* as a standard file system and */usr* as a Fast File System. (You can, however, choose as separate options "Rebuild root" and "Rebuild usr" if you want *root* and */usr* to be created as different types.)

To create new *root* and */usr* file systems, select the appropriate file system settings and then press the SAVE key (PF-3). Press the CANCEL key (PF-1) to exit the menu without creating *root* and */usr* file systems. The following settings are available on this menu:

- File system type for *root* and */usr*

Use the CHOICES key (PF-2) to toggle between the *Fast File System* (FFS) and *standard file system* (S51K) types. Fast File Systems contain 8K-byte units, whereas standard file systems contain 1K-byte units. For this reason, Fast File Systems read files more efficiently than standard file systems do. However, the RFS product does not support FFS. Thus, if you use RFS, do not create any file system as a FFS.

- File system volume names for *root* and */usr*

The *volume name* for a file system is the alphanumeric character string that the */etc/labelit* utility uses to label the partition and attach a logical base directory name to the file system. By default, the volume name is the current date. However, it can be any six-character string that you key in.

After the *root* and */usr* file systems have been created, return to the main rebuild menu and download software. Proceed to 11.12, "Loading Nucleus Software with newprod."

Rebuild root

The following menu displays if you choose the "Rebuild root" option from the main rebuild menu:

<pre>1 Rebuild Rebuild all > Rebuild root Rebuild usr Make a File System Download software Introduction Page Reboot Exit</pre>	<pre>2 Rebuild the root file system File system type: Fast_File_System File system volume name: 051690 Press SAVE to make the root file system. Press CANCEL to return to the previous menu.</pre>
<pre>CANCEL CHOICES SAVE [] [] HELP [] [] EXIT</pre>	

This menu option creates a minimal *root* file system. To create the new *root* file system, select the appropriate file system settings and then press the SAVE key (PF-3). Press the CANCEL key (PF-1) to exit the menu without creating the *root* file system. The following settings are available on this menu:

- File system type

Use the CHOICES key (PF-2) to toggle between the *Fast File System* (FFS) and *standard file system* (S51K) types. Fast File Systems contain 8K-byte units, whereas standard file systems contain 1K-byte units. For this reason, Fast File Systems read files more efficiently than standard file systems do. However, the RFS product does not support FFS. Thus, if you use RFS, do not create any file system as a FFS.

- File system volume name

The *volume name* for a file system is the alphanumeric character string that the */etc/labelit* utility uses to label the partition and attach a logical base directory name to the file system. By default, the volume name is the current date. However, it can be any six-character string that you key in.

After the *root* file system has been created, return to the main rebuild menu and rebuild the */usr* file system (if needed). Then, proceed to 11.12, "Loading Nucleus Software with newprod."

Rebuild usr

The following menu displays if you choose the "Rebuild usr" option from the main rebuild menu:

<pre>1 Rebuild Rebuild all Rebuild root > Rebuild usr Make a File System Download software Introduction Page Reboot Exit</pre>	<pre>2 Rebuild the usr file system File system type: Fast_File_System File system volume name: 051690 Press SAVE to make the usr file system. Press CANCEL to return to the previous menu.</pre>
<pre>CANCEL CHOICES SAVE [] [] [] HELP [MID-MENU] EXIT</pre>	

This menu option creates a file system other than *root* or */usr* (for example, */usr2*, */usr3*, etc.). To create the new file system, select the appropriate file system settings and then press the SAVE key (PF-3). Press the CANCEL (PF-1 function key) to exit the menu without creating a file system. The following settings are available on this menu:

- **SCSI ID**

Key in the SCSI ID of the disk on which to create the file system.

- **Logical unit number**

Key in the Logical unit number (LUN) of the disk on which to create the file system.

- **Partition number**

Key in the partition number of the partition on which to create the file system. This number should be a hexadecimal (0-9, a-f) digit. The partition number for all UNIX partitions is 7.

- **Partition modifier number**

Key in the modifier number of the partition on which to create the file system. This number should be a hexadecimal digit. The modifier numbers 0, 1, and 3 are reserved for *root*, *swap*, and */usr*, respectively.

- **File system type**

Use the CHOICES key (PF-2) to toggle between the *Fast File System* (FFS) and *standard file system* (S51K) types. Fast File Systems contain 8K-byte units, whereas standard file systems contain 1K-byte units. For this reason, Fast File Systems read files more efficiently than standard file systems do. However, the RFS product does not support FFS. Thus, if you use RFS, do not create any file system as a FFS.

- **File system name**

Key in the name of the file system (for example, *usr3* or *usr4*). The default file system name is *usr2*. Do not include a leading slash when you key in the file system name.

- File system volume name

The *volume name* for a file system is the alphanumeric character string that the `/etc/labelit` utility uses to label the partition and attach a logical base directory name to the file system. By default, the volume name is the current date. However, it can be any six-character string that you key in.

After the file system has been created, return to the main rebuild menu and download software. Proceed to 11.12, "Loading Nucleus Software with newprod."

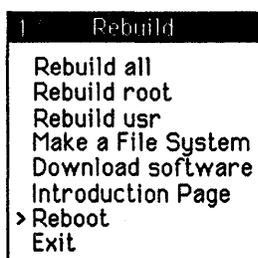
Introduction Page

The following menu (which also appears when you first enter the rebuild environment) displays if you choose the "Introduction Page" option from the main rebuild menu:

The screenshot shows a terminal window with a 'Rebuild' menu on the left and a 'Welcome to Rebuild' screen in the center. The 'Rebuild' menu lists options: Rebuild all, Rebuild root, Rebuild usr, Make a File System, Download software, > Introduction Page (highlighted), Reboot, and Exit. The 'Welcome to Rebuild' screen provides instructions on using arrow keys and the RETURN key to navigate and select menu items. It also shows the top two rows of the keypad with function keys: CANCEL (pf1), PREVPAGE (pf2), NEXTPAGE (pf5), (pf4), (7), HELP (8), END-MENU (9), and EXIT (5). At the bottom of the terminal window, there is a prompt 'Press CANCEL to go on to the Rebuild Menu' and a row of function keys: CANCEL, PREVPAGE, NEXTPAGE, (pf4), HELP, END-MENU, and EXIT.

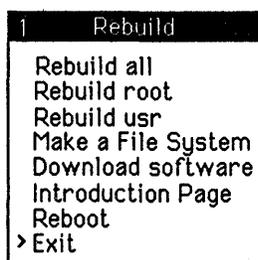
Reboot

If you choose the "Reboot" option from the main rebuild menu, the system will be rebooted. You may reboot after you have restored the file systems and before you load software (so that you will be booted off the *root* partition, rather than *swap*, where the rebuild environment is loaded.) In addition, you must reboot after you load software.



Exit

If you choose the "Exit" option from the main rebuild menu, you will exit the rebuild menus and be placed at the system prompt. To return to the rebuild menu, key in <Ctl-D>.



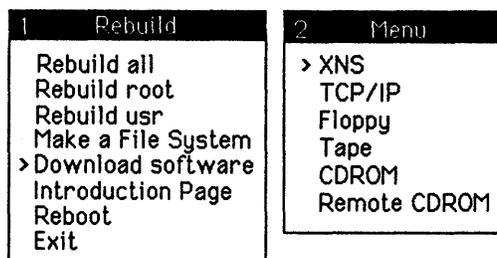
11.12 Loading Nucleus Software with newprod

After you have restored the file systems, you must load nucleus software using the `newprod` utility. Nucleus software includes the software products that must reside on the hard disk before you can run the System V operating system. You do not need to load application software until the system is completely functional.

Note:

You have the option of rebooting the system (off the Rebuild Boot floppy) before you load software, so that you will be booted off the `root` partition, rather than the `swap` partition. If you do so, you will need to, once again, boot into the rebuild environment. At that point, the main rebuild menu will not include the "Rebuild all," "Rebuild root," and "Rebuild usr" options.

After you have restored the file systems, you may choose the "Download software" option from the rebuild menu. The following menu appears:



The "Download software" menu invokes the `newprod` utility from the delivery media/source of your choice. Choices include an XNS-based network delivery source, a TCP/IP-based network delivery source, `newprod` floppy disk, `newprod` tape, local CDROM, or remote CDROM.

XNS network delivery source

The following menu displays if you choose the "XNS" option from the download software menu:

1 Rebuild	2 Menu
Rebuild all Rebuild root Rebuild usr Make a File System > Download software Introduction Page Reboot Exit	> XNS TCP/IP Floppy Tape CDROM Remote CDROM
3 Host Information	
Host Address (xxxxxxxx.xx-xx-xx-xx-xx-xx): _____ User Name: _____ Password: _____	
Press SAVE to load the software. Press CANCEL to return to the previous menu.	
[CANCEL] [CHOICES] [SAVE] [] [] [HELP] [F10-FMENU] [EXIT]	

This menu allows you to connect to an XNS-based network delivery source (a node that stores products in a deliverable format so that other nodes can connect and install software) and install software through the newprod utility. You must supply the following information at this menu:

- **Host Address**

The network address of the XNS-based host (delivery source) must be entered in the format "XXXXXXXX.XX-XX-XX-XX-XX."

- **User Name**

You must supply a valid username for the host.

- **Password**

You must supply a valid password for a user account on the host (if required on the host).

Press the SAVE key (PF-3) to connect to the delivery source and invoke newprod, or the CANCEL key (PF-1) to return to the previous menu.

Once you invoke `newprod`, you must update the following products. These products are *nucleus products*, or the minimum software products required to run the CLIX operating system.

- Delivery Tools (DELTOOLS)
- Workstation/Server Diagnostics (DIAG)
- Environ V Target Shared Library (ENVIRON_S)
- Gpipe Target Run-Time Shared Library (GPIPE_S)
- Workstation Network Software (INC)
- System V.3.1 File Systems (SYSTEMV)
- System V.3.1 Boot Images (UNIXBOOT)
- Workstation Graphic Resources (RESOURCES)
- I/Forms Run-Time Package (FORMS_S)
- RIS Client Support Package (RISCCU)
- Screen Manager (SCREENMGR) (not needed for InterServe systems)
- DEC VT220 Emulation (VT220) (not needed for InterServe systems)

Please refer to the *CLIPPER Software Delivery Guide* for instructions for using `newprod`.

After you have loaded software, reboot the system by selecting the Reboot option from the rebuild menu. Proceed to 11.13, "Preparing Nonstandard Partitions for Use."

TCP/IP network delivery source

The following menu displays if you choose the "TCP/IP" option from the download software menu:

1 Rebuild	2 Menu
Rebuild all Rebuild root Rebuild usr Make a File System > Download software Introduction Page Reboot Exit	XNS > TCP/IP Floppy Tape CDROM Remote CDROM
3 Network Address Information	
Host Node Name: _____ User Name: _____ Password: _____ Host Address (xxx.xxx.xxx.xxx): _____ Your Address (xxx.xxx.xxx.xxx): _____	
Press SAVE to load the software. Press CANCEL to return to the previous menu.	
[CANCEL] [CHOICES] [SAVE] [] [] [HELP] [CMD-MENU] [EXIT]	

This menu allows you to connect to a TCP/IP-based network delivery source (a node that stores products in a deliverable format so that other nodes can connect and install software) and install software through the newprod utility. You must supply the following information at this menu:

- **Host Node Name**
You must supply the nodename of the host (delivery source).
- **User Name**
You must supply a valid username for the host.
- **Password**
You must supply a valid password for a user account on the host (if required on the host).
- **Host Address**
You must supply the network address of the TCP/IP-based host (delivery source). Enter the address in the format "XXX.XXX.XXX.XXX."

- Your Address

You must supply the network address of the machine you are rebuilding. Enter the address in the format "XXX.XXX.XXX.XXX."

Press the SAVE key (PF-3) to connect to the delivery source and invoke *newprod*, or the CANCEL key (PF-1) to return to the previous menu.

Once you invoke *newprod*, you must update the following products. These products are *nucleus products*, or the minimum software products required to run the CLIX operating system.

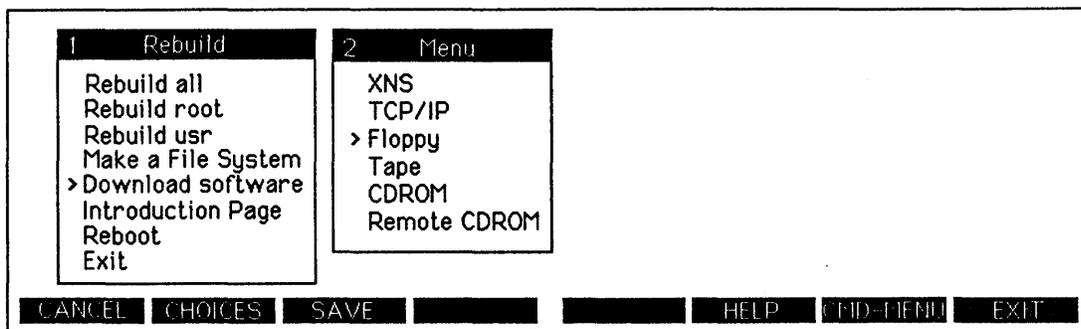
- Delivery Tools (DELTOOLS)
- Workstation/Server Diagnostics (DIAG)
- Environ V Target Shared Library (ENVIRON_S)
- Gpipe Target Run-Time Shared Library (GPIPE_S)
- Workstation Network Software (INC)
- System V.3.1 File Systems (SYSTEMV)
- System V.3.1 Boot Images (UNIXBOOT)
- Workstation Graphic Resources (RESOURCES)
- I/Forms Run-Time Package (FORMS_S)
- RIS Client Support Package (RISCCU)
- Screen Manager (SCREENMGR) (not needed for InterServe systems)
- DEC VT220 Emulation (VT220) (not needed for InterServe systems)

Please refer to the *CLIPPER Software Delivery Guide* for instructions for using *newprod*.

After you have loaded software, reboot the system by selecting the Reboot option from the rebuild menu. Proceed to 11.13, "Preparing Nonstandard Partitions for Use."

Floppy disk

The following messages display if you choose the "Floppy" option from the download software menu:



Working...
 Changing over to the hard disk root file system.
 Insert a newprod product floppy and press <RETURN> --->

This menu option allows you to load software from a local floppy disk drive. When prompted, insert a **newprod** floppy disk and press <Return>.

Once you invoke **newprod**, you must update the following products. These products are *nucleus products*, or the minimum software products required to run the CLIX operating system.

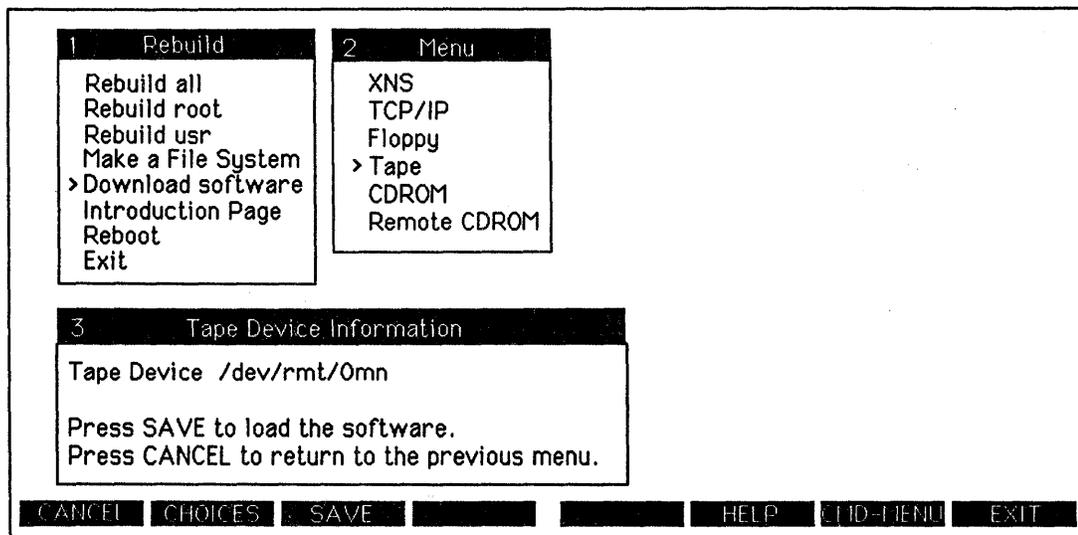
- Delivery Tools (DELTOOLS)
- Workstation/Server Diagnostics (DIAG)
- Environ V Target Shared Library (ENVIRON_S)
- Gpipe Target Run-Time Shared Library (GPIPE_S)
- Workstation Network Software (INC)
- System V.3.1 File Systems (SYSTEMV)
- System V.3.1 Boot Images (UNIXBOOT)
- Workstation Graphic Resources (RESOURCES)
- I/Forms Run-Time Package (FORMS_S)
- RIS Client Support Package (RISCCU)
- Screen Manager (SCREENMGR) (not needed for InterServe systems)
- DEC VT220 Emulation (VT220) (not needed for InterServe systems)

Please refer to the *CLIPPER Software Delivery Guide* for instructions for using **newprod**.

After you have loaded software, reboot the system by selecting the Reboot option from the rebuild menu. Proceed to 11.13, "Preparing Nonstandard Partitions for Use."

Tape

The following menu displays if you choose the "Tape" option from the download software menu:



This menu option allows you to load software from a local tape drive. This menu requires you to supply device information for the tape drive. After you have supplied the correct tape device information, press the SAVE key (PF-3) to invoke *newprod*, or the CANCEL key (PF-1) to return to the previous menu.

Once you invoke *newprod*, you must update the following products. These products are *nucleus products*, or the minimum software products required to run the CLIX operating system.

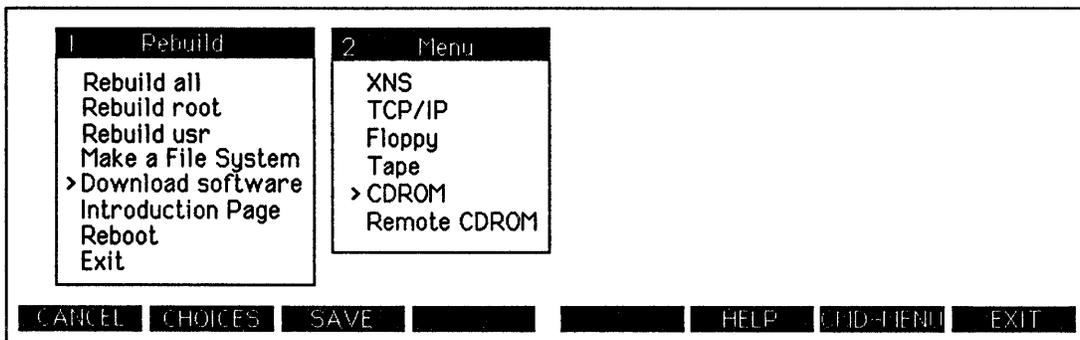
- Delivery Tools (DELTOOLS)
- Workstation/Server Diagnostics (DIAG)
- Environ V Target Shared Library (ENVIRON_S)
- Gpipe Target Run-Time Shared Library (GPIPE_S)
- Workstation Network Software (INC)
- System V.3.1 File Systems (SYSTEMV)
- System V.3.1 Boot Images (UNIXBOOT)
- Workstation Graphic Resources (RESOURCES)
- I/Forms Run-Time Package (FORMS_S)
- RIS Client Support Package (RISCCU)
- Screen Manager (SCREENMGR) (not needed for InterServe systems)
- DEC VT220 Emulation (VT220) (not needed for InterServe systems)

Please refer to the *CLIPPER Software Delivery Guide* for instructions for using *newprod*.

After you have loaded software, reboot the system by selecting the Reboot option from the rebuild menu. Proceed to 11.13, "Preparing Nonstandard Partitions for Use."

Local CDROM

The following messages display if you choose the "CDROM" option from the download software menu:



Working...
 Changing over to the hard disk root file system.
 Insert a newprod CDROM and press <RETURN>---->

This menu option allows you to load software from a local CDROM disk drive. When prompted, insert a newprod CDROM and press <Return>.

Once you invoke newprod, you must update the following products. These products are *nucleus products*, or the minimum software products required to run the CLIX operating system.

- Delivery Tools (DELTOOLS)
- Workstation/Server Diagnostics (DIAG)
- Environ V Target Shared Library (ENVIRON_S)
- Gpipe Target Run-Time Shared Library (GPIPE_S)
- Workstation Network Software (INC)
- System V.3.1 File Systems (SYSTEMV)
- System V.3.1 Boot Images (UNIXBOOT)
- Workstation Graphic Resources (RESOURCES)
- I/Forms Run-Time Package (FORMS_S)
- RIS Client Support Package (RISCCU)
- Screen Manager (SCREENMGR) (not needed for InterServe systems)
- DEC VT220 Emulation (VT220) (not needed for InterServe systems)

Please refer to the *CLIPPER Software Delivery Guide* for instructions for using newprod.

After you have loaded software, reboot the system by selecting the Reboot option from the rebuild menu. Proceed to 11.13, "Preparing Nonstandard Partitions for Use."

Remote CDROM

The following menu displays if you choose the "Remote CDROM" option from the download software menu:

1 Rebuild	2 Menu					
Rebuild all Rebuild root Rebuild usr Make a File System > Download software Introduction Page Reboot Exit	XNS TCP/IP Floppy Tape CDROM > Remote CDROM					
3 Host Information						
Host Address (xxxxxxxx.xx-xx-xx-xx-xx-xx): _____ User Name: _____ Password: _____						
Press SAVE to load the software. Press CANCEL to return to the previous menu.						
CANCEL	CHOICES	SAVE		HELP	CMD-MENU	EXIT

This menu allows you to connect to a remote CDROM drive and install software through the newprod utility. You must supply the following information at this menu:

- **Host Address**

The network address of the CDROM node (the workstation/server to which the CDROM drive is connected) must be entered in the format "XXXXXXXXXX.XX-XX-XX-XX-XX-XX."

- **User Name**

You must supply a valid username for the CDROM node.

- **Password**

You must supply a valid password for a user account on the CDROM node (if required on the CDROM node).

Press the SAVE key (PF-3) to connect to the CDROM node and invoke newprod, or the CANCEL key (PF-1) to return to the previous menu.

Once you invoke **newprod**, you must update the following products. These products are *nucleus products*, or the minimum software products required to run the CLIX operating system.

- Delivery Tools (DELTOOLS)
- Workstation/Server Diagnostics (DIAG)
- Environ V Target Shared Library (ENVIRON_S)
- Gpipe Target Run-Time Shared Library (GPIPE_S)
- Workstation Network Software (INC)
- System V.3.1 File Systems (SYSTEMV)
- System V.3.1 Boot Images (UNIXBOOT)
- Workstation Graphic Resources (RESOURCES)
- I/Forms Run-Time Package (FORMS_S)
- RIS Client Support Package (RISCCU)
- Screen Manager (SCREENMGR) (not needed for InterServe systems)
- DEC VT220 Emulation (VT220) (not needed for InterServe systems)

Please refer to the *CLIPPER Software Delivery Guide* for instructions for using **newprod**.

After you have loaded software, reboot the system by selecting the Reboot option from the rebuild menu. Proceed to 11.13, "Preparing Nonstandard Partitions for Use."

11.13 Preparing Nonstandard Partitions for Use

The Make a file system option on the rebuild utility now automatically creates a device file, creates a file system, labels the partition, and adds the partition to the */etc/fstab* file (so that the file system will be mounted and checked each time the system boots). Thus, you are no longer required to perform these steps manually.

Proceed to 11.14, "Loading Application Software with the newprod Utility."

11.14 Loading Application Software with the newprod Utility

After you have prepared any nonstandard partitions for use, you must load application software. Now that the system is functional and all partitions are mounted, you can load this software on the hard disk.

Follow these steps to invoke `newprod` and deliver software:

1. Log in to the system and access the superuser account as follows

```
login: username
$ su
#
```

Note:

If you restored the `/usr` file system(s), user accounts will no longer exist. Log in using the `sys` account. You will need to recreate all user accounts. Refer to 3.2, "Adding User Accounts."

2. Invoke the `newprod` utility as follows:

```
# newprod
```

```
Enter source of installation: n)etwork, f)loppy, t)ape or r)emote cdrom or local c)drom -->
```

If you keyed in `n` for network or `r` for remote cdrom, you will be prompted to enter the network connect string. After rebuilding, you cannot key in the nodename of the delivery node; you must key in the Ethernet address.

3. The `newprod` menu appears. From this menu, you may *select* (highlight) products to download. Select and load all software products that you wish to use on the system. For instructions on using the `newprod` utility, refer to the *CLIPPER Software Delivery Guide*.
4. After all products have been successfully downloaded, continue to 11.15, "Restoring Files from Backups."

11.15 Restoring Files from Backups

The final step in the process of rebuilding the hard disk is restoring files from backups.

After restoring the software products with the `newprod` utility, restore all user- and site-specific files that you backed up before beginning the rebuild procedure. For information on restoring files, refer to Chapter 9, "Backing Up and Restoring Files."

Index

.def_proc, 5-3, 5-35
.env file, 4-1, 4-9
.exrc file, 4-1, 4-13
.profile file, 4-1, 4-6, 4-8
smgr_pd, 5-26

/

/dev/dsk
 creating device file for, 7-3
/dev/rmt, 9-5, 9-10, 9-11, 9-13 to 9-18, 9-20, 9-21, 9-25
/etc/format, 9-27
/etc/gettydefs, 2-5, 2-6, 2-8
/etc/init, 2-3
/etc/init.d, 2-9
/etc/inittab, 2-3, 2-4, 2-6
/etc/labelit, 7-10
/etc/mkfs, 7-10
/etc/mknod, 7-8
/etc/motd, 4-6
/etc/mount, 7-12
/etc/newfs, 7-11
/etc/passwd, 3-5, 3-7, 3-13, 3-14, 3-16, 4-4
 fields in, 4-4
/etc/profile, 4-1, 4-6
/etc/rc2, 2-5
/usr/tmp, 7-14

3

32-channel RS-232 interface assembly

A

accessing

 directory files, 3-20
 ordinary files, 3-21

accounting

 initializing at bootup, 2-9

accounts

 adding passwords, 3-13
 adding user accounts, 3-5
 customizing variables in user accounts, 4-8
 removing passwords, 3-16
 removing user accounts, 3-10
 /etc/passwd file, 4-4

- active group
 - changing, 3-24
- adding cron command file, 4-15
- adding partition, 11-28
 - on server, 7-28
 - on workstation, 7-27
- adding user accounts, 3-5
 - comparison of addusers and sysadm, 3-5
 - manually, 3-5
 - using addusers, 3-6
 - using sysadm, 3-11
- adding processes to workstation menu, 5-26
- address
 - node or network, 1-7, 10-12, 10-42
- addusers
 - advantages, 3-5
 - using to add user accounts, 3-6
- administering
 - adding user accounts, 3-5
 - maintaining file systems, 7-1
 - removing user accounts, 3-10
 - setting up initial system, 1-1
- administrator
 - system administrator account, 3-3
- aliases
 - defining, 4-9
- ANSI color escape sequences, 5-22
- ansitape, 9-20
- assigning nodename, 1-6, 10-12, 10-42
- assigning passwords, 3-13
- asynchronous input/output
 - supported in scpio for backups, 9-13
- at, 4-17
- attaching
 - external devices, 2-6
 - window to printer, 5-15
- auto xon/xoff
 - setting on workstation, 10-10
- automatic dump, 10-12
- automatic reboot, 10-12
- auxiliary ports
 - setting RS-232 parameters
 - server, 10-39
 - workstation, 10-8
- available disk space, 7-14

B

background

changing screen color, 5-33

changing pattern, 5-32

backing up entire file system, 9-8, 9-11, 9-13

backing up files, 9-6

backups

ansitape example, 9-20

commands and utilities, 9-6

compatibility with VAX/VMS, 9-21

defining schedule, 9-3

examples using dd, 9-10

examples using tar, 9-11, 9-12

multiple volumes using scp/cpio, 9-13

raw binary data, 9-10

support of asynchronous input/output in, 9-13

dd command, 9-10

tar command, 9-11, 9-12

to local tape drive using cpio, 9-5

to remote tape drive using rtc, 9-4

using ansitape, 9-20

using backup and restore, 9-18, 9-19

using scp/cpio, 9-13, 9-14

using vmsbackup, 9-21

backup command, 9-18

backup plan

full network configuration, 9-4

standalone configuration, 9-5

bad blocks

explanation, 11-10

messages, 11-10

batch

run command at specific time, 4-17

run commands at regular intervals, 4-15

baud rate

setting on server, 10-40

setting on workstation, 10-9

bell

setting tone

workstation, 10-8

bgpattern, 5-32

binary data

moving using dd command, 9-10

- blocking factors with dd, 9-10
- boot
 - booting with rebuild disks, 11-7
- boot partition, 11-20
- booting single user
 - server, 11-41
 - workstation, 11-11
- bootup
 - files executed at, 2-1
 - starting accounting and lp, 2-9
- bourne shell, 3-7, 4-5
- buffered screens
 - setting up, 5-11

C

- calendar, 4-14
- canceling a print request, 6-5
- cartridge tapes
 - with dd command, 9-11
- catalog file
 - creating window using, 5-24
- catalog menu, 5-24
- changing
 - file ownership, 3-23
 - file protection codes, 3-20
 - group access to file, 3-24
 - background pattern, 5-32
 - number of virtual screens, 5-30
 - screen color, 5-33
 - your active group, 3-24
 - partition sizes, 11-28, 11-20
- character codes
 - keyboard, A-1
 - decimal and hexadecimal equivalents, A-3
 - received, A-6
 - transmitted, A-3
- checklist for initial setup tasks, 1-1
- chgrp, 3-25
- chmod, 3-22
- chown, 3-23
- clearinghouse database, 1-6

-
- CLIX Resource Monitor (CRM), 8-1
 - adding to def__proc, 8-3, 5-34
 - adding to pull-down menu, 8-3
 - changing defaults
 - system monitors, 8-9
 - forms, 8-6
 - installing, 8-3
 - invoking, 8-4
 - linking to /usr/bin, 8-3
 - menus, 8-10, 8-24
 - monitoring
 - all aspects of specific process, 8-29
 - all system processes, 8-16
 - CPU use, 8-15
 - input/output, 8-15
 - memory use, 8-14
 - memory use by process, 8-28
 - page faults, 8-26
 - system calls and faults, 8-26
 - Process Monitors, 8-19
 - command line, 8-19
 - exiting, 8-19
 - invoking, 8-19
 - memory monitor, 8-28
 - process monitor, 8-29
 - profiler, 8-26
 - selecting process to monitor, 8-24
 - selecting profiler options, 8-25
 - running curses-based
 - process monitors, 8-10
 - system monitors, 8-7
 - running graphics-based
 - system monitors, 3-9
 - System Monitors, 8-7
 - command-line options, 8-8
 - command-line syntax, 8-7
 - curses-based, 8-10
 - exiting, 8-7
 - graphics-based, 8-9
 - invoking from command line, 8-7
 - Monitor Parameters, 8-11
 - Show Memory Usage, 8-17
 - Show Open Files, 8-17

- Top CPU, 8-15
- Top Fault, 8-12
- Top I/O, 8-15
- Top Memory, 8-14
- Top System, 8-16
- topcpu, 8-15
- topfault, 8-12
- topio, 8-15
- topmem, 8-14
- topsys, 8-16
 - using FMLI interface, 8-7
- collapsing window to symbol, 5-7
- color
 - changing screen color, 5-33
- color window
 - creating with ANSI color escape sequence, 5-22
 - creating with regis, 5-18, 5-19
 - creating with vterm, 5-6, 5-18
- command recall, 4-10 to 4-12
 - defining, 4-11, 4-12
 - setting temporary editor, 4-12
- commands
 - addusers, 3-9
 - ansitape, 9-20
 - at, 4-17
 - backup, 9-18
 - calendar, 4-13
 - chgrp, 3-25
 - chmod, 3-23
 - chown, 3-23
 - conv=sync for dd, 9-11
 - cpflop, 9-31
 - cpio, 9-13
 - cron, 4-15
 - crontab, 4-16
 - dd, 9-10
 - df, 7-13
 - dls, C-6
 - dtu, C-3
 - du, 7-14
 - fr_flop, 9-31
 - format, 9-27
 - init, 2-3
 - labelit, 7-11
 - mkfs, 7-10, 9-29
 - mknod, 7-3 to 7-8

- mount, 7-12
- newfs, 7-11
- newgrp, 3-24
- passwd, 3-13
- recalling from command line, 4-10
- regis, 5-18
- restore, 9-19
- rmusers, 3-10
- rtc, 9-22
- showconfig, 2-6
- scpio, 9-13
- su, 3-3
- sysadm, 3-3, 3-8
- tar, 9-11
- to_flop, 9-31
- topcpu (CRM), 8-15
- topfault (CRM), 8-12
- topio (CRM), 8-16
- topmem (CRM), 8-14
- topsys (CRM), 8-16
- utd, C-4
- vmsbackup, 9-21
- vterm, 5-4
- compatibility
 - copying files from DOS to UNIX, C-3
 - copying files from UNIX to DOS, C-4
 - with IBM PC, B-1
- configure
 - background pattern, 5-32
 - number and location of message strips, 5-31
 - number of virtual screens, 5-30, 5-31
- configure option on workstation menu, 5-26
 - add, 5-26
 - delete, 5-28
 - modify, 5-28
 - screens, 5-30
- configure a line printer, 6-1, 6-3
- console login, 3-3
- console window
 - creating, 5-36
 - customizing, 5-36
- copy file archive in and out (cpio), 9-13
- copying files from DOS to UNIX, C-3
- copying files from UNIX to DOS, C-4

- core files
 - removing, 7-15
- cpio, 9-13
- creating color window
 - using ANSI, 5-22
 - using regis commands, 5-18, 5-19
- creating device file, 7-3 to 7-8, 9-31
- creating file system on new partition, 7-7
- creating rebuild floppy disks, 11-3
- CRM (see CLIX Resource Monitor), 8-1
 - adding to def_proc file, 8-3
 - adding to workstation menu, 8-3
- cron, 4-15
- crontab, 4-16
- csh, 3-7
- Curses
 - CRM interface, 8-10
- customizing
 - adding default processes to .def_proc, 5-34
 - adding default processes to def_proc, 5-35
 - adding default processes to sys_proc, 5-36
 - console window
 - through sys_proc, 5-36
 - creating window that logs you in, 5-7
 - creating windows, 5-4
 - display characteristics, 5-12
 - function key menu, 5-16
 - keyboard characteristics, 5-13
 - login prompt in gettydefs, 2-5
 - number and location of message strips, 5-30, 5-31
 - number of virtual screens, 5-30
 - saving customized window, 5-24
 - setting up reminder service, 4-14
 - terminal characteristics, 5-10
 - /etc/passwd file, 4-4, 3-16
 - /etc/profile file, 4-6
 - background pattern, 5-33
 - login environment, 4-1
 - screen color, 5-34
 - workstation menu
 - adding cataloged window, 5-27
 - adding processes to, 5-26
 - connecting to remote host, 5-27
 - deleting processes on, 5-28
 - modifying processes on, 5-28

D

- data capture
 - creating log file, 5-7
- dd, 9-10
- dedicated printer, 6-1
- default file permission modes, 3-20
 - setting with umask, 3-21
- default processes
 - setting with `.def__proc`, 5-35
 - setting with `def__proc`, 5-34
 - setting with `sys__proc`, 5-36
- defining aliases, 4-9
- defining macros, 5-16
- `def__proc`, 5-34
- deleting
 - suggestions for freeing disk space, 7-14
- delimiters
 - off, 2-6
 - respawn, 2-6
- design files
 - backing up using `ansitape`, 9-20
 - backing up using `vmsbackup`, 9-21
- device files
 - creating, 7-3
 - creating for scsi 0, lun 0, 7-4
 - creating for scsi 1, lun 0, 7-5
 - creating for scsi 2, lun 0, 7-6
 - creating for scsi 3, lun 0, 7-7
- Device Image Copy Utility Page
 - server, 10-48
 - workstation, 10-26
- device parameters, 2-8
- `df`, 7-13
- directory files, 3-22
 - accessing based on file permission modes, 3-22
- disk
 - creating device file, 7-3 to 7-8
 - floppy
 - file system, 9-29
 - formatting, 9-31
 - Format Utility Page
 - server, 10-43
 - workstation, 10-21
 - formatting, 11-13
 - formatting floppy disks for PC DOS, 9-27
 - freeing space, 7-14
 - monitoring space, 7-14

- partitioning, 11-17
 - Utility Page options on server, 10-46
 - Utility Page options on workstation, 10-26
- reading hard disk for errors, 11-10
- rebuilding using nonstandard partitions, 11-17
- SCSI ID on Utility Pages
 - server, 10-43
 - workstation, 10-14
- understanding partition names, 11-25
- verify flaws, 11-10
- verifying and adding flaw data
 - server, 10-11
 - workstation, 10-12
- Disk Maintenance Utility Page
 - server, 10-43
 - workstation, 10-13
- Disk Partitioning Utility Page
 - server, 10-46
 - workstation, 10-23
- disk space
 - amount free (df), 7-13
 - amount used (du), 7-14
 - freeing, 7-14
 - monitoring, 7-14, 7-17
- disk space used, 7-14
- display
 - amount of disk space used, 7-14
 - available disk space, 7-13
- display characteristics, 5-12
- display setup menu, 5-12
- dls, C-6
- DOS
 - copying files from DOS to UNIX, C-3
 - copying files from UNIX to DOS, C-4
 - listing contents of DOS directory, C-6
- dtu, C-3
- du, 7-14

- E
- editor
 - setting default editor, 4-10
- emulation
 - vt100, vt220, 5-4
- erasing
 - data on floppy disk, 9-27
- escape sequences
 - keyboard control characters, A-7

execute access, 3-21

external devices

 attaching, 2-6

F

fast backups

 dd, 9-10

Fast File Systems

 /etc/newfs, 7-11

 changing from standard to fast file, 7-11

 creating, 7-11

 creating from rebuild menu, 11-33

 mounting, 7-12

fields

 in file protection code (rwx), 3-20

 in /etc/passwd file, 4-5

file access, 3-20

 categories of users, 3-19

 directory files, 3-20

 modifying, 3-21

 ordinary files, 3-21

file owner, 3-22

 changing, 3-22

file protection codes (rwx)

 changing, 3-21

 fields, 3-21, 3-22

 meaning for directory files, 3-36

 meaning for ordinary files, 3-37

file size

 displaying with du, 7-14

file system

 backing up, 9-10, 9-11, 9-13

 changing from standard to Fast File, 7-11

 creating fast file system, 7-11

 commands and utilities for backing up, 9-10, 9-11, 9-13

 creating on new partition, 7-9

 Fast File System, 7-11

 standard file system, 7-10

 defining plan for backing up, 9-4

 difference between fast and standard, 7-9

 Fast File System

 changing to, 7-11

 creating Fast File System, 7-11

files

- backing up and restoring, 9-18
 - to local tape drive, 9-5
 - to network tape drive, 9-4
 - using backup and restore, 9-18
 - using dd, 9-10
 - using scp/cpio, 9-13
 - using tar, 9-11

labeling, 7-11

mounting, 7-12

restoring, 11-33

using ansitape to back up, 9-20

using vmsbackup to back up, 9-21

files

.env, 4-9

.exrc, 4-13

.profile, 4-8

/etc/gettydefs, 2-6, 2-7, 2-8

/etc/inittab, 2-6, 2-7

/etc/profile, 4-6

access, 3-20

changing ownership, 3-23

changing group, 3-25

copying from DOS to UNIX, C-3

copying from UNIX to DOS, C-4

core

- removing, 7-15

def_proc and CRM, 8-3

displaying size, 7-14

executed at login, 4-1

executed at system startup, 2-1

executed in multiuser mode, 2-9

getty, 2-5, 2-7, 2-8

gettydefs, 2-6, 2-7, 2-8

init.d, 2-9

inittab, 2-4, 2-6, 2-7

Korn shell environment file, 3-7

permission modes, 3-20

- changing, 3-22

profile file, 4-6

protection codes, 3-20

- changing, 3-22

screen manager default processes files, 5-34 to 5-36

startup processes, 2-1

terminal definition (gettydefs), 2-5

user's login environment file (.profile), 4-8

find

freeing disk space, 7-15

using with cpio/scpio, 9-13

- first
 - tasks for setting up system, 1-1
- flaws
 - adding on Utility Pages
 - server, 10-43
 - workstation, 10-16
 - adding to flaw list using verify
 - server, 10-43
 - workstation, 10-19
 - format with flaw data
 - server, 10-43
 - workstation, 10-18
 - verify disk flaws, 11-10
- floppy disks
 - backups with tar, 9-11
 - copy one floppy to another, 9-31
 - creating rebuild disks, 11-3
 - density, 9-27
 - file system, 9-29
 - formatting, 9-27
 - mkfs command, 9-29
 - mount file system, 9-29
 - preparing for use, 9-27
 - read/write compatibility with IBM PC, B-1
 - restore from multiple, 9-31
 - unmount file system, 9-30
 - write to multiple, 9-31
- FMLI
 - CRM interface, 8-5
- forcing password change, 3-14
- Format Utility Page
 - workstation, 10-21
 - server, 10-43
- formatting, 11-13
 - explanation, 11-13
 - floppy disks, 9-27
 - server, 11-14
 - Utility Page
 - server, 10-43
 - workstation, 10-21
 - workstation, 11-13
- free disk space
 - displaying, 7-13
- freeing, 7-14
- function keys
 - assigning macros, 5-16

G

- general setup menu, 5-10
 - creating log file, 5-7
- getting started, 1-1
- getty, 2-5, 2-6, 2-7, 2-8
- gettydefs, 2-6, 2-7, 2-8
- GID, 3-7, 3-9, 4-5
- global variables
 - setting up in /etc/profile, 4-6
- group, 3-7, 3-9, 4-5, 3-24
 - changing group access, 3-24
 - changing your active group, 3-24

H

- hard disk
 - adding flaws on Utility Page
 - server, 10-43
 - workstation, 10-16
 - adding flaws to flaw list
 - server, 10-43
 - workstation, 10-19
 - creating device file, 7-3 to 7-8
 - external
 - creating device file, 7-3 to 7-8
 - Format Utility Page
 - server, 10-44
 - workstation, 10-21
 - formatting, 11-13
 - internal
 - creating device file, 7-3 to 7-8
 - partitioning
 - Utility Page options on server, 10-44
 - Utility Page options on workstation, 10-23
 - reading for errors, 11-10
 - rebuilding using nonstandard partitions, 11-28
 - understanding partition names, 11-25
 - verify, 11-10
 - verifying
 - server, 10-43
 - workstation, 10-19
- Hard Disk Flaw Data Utility Page
 - workstation, 10-16
- history
 - setting up in Korn shell, 4-10
- home directory, 3-7, 3-9
- hue, saturation, and lightness chart, 5-21

I

icon

- creating symbol for collapsed window, 5-7

imaging

- using dd command to back up, 9-10

inactive window, 5-6

- creating, 5-6

init, 2-3, 2-10

init.d, 2-9

initial tasks, 1-1

initialization files, 4-1

initializing

- from multiuser mode, 2-9

inittab, 2-6, 2-7

interface

- CRM, 8-5

internal hard disk

- creating device file for, 7-3 to 7-8

invoking

- CRM, 8-4

iop

- setting on server, 10-37

K

key clicks

- setting, 10-7

keyboard

- assigning macros, 5-16

- character codes, A-1

 - decimal and hexadecimal equivalents, A-3

 - received, A-6

- control characters

 - escape sequences, A-7

- customizing, 5-13

keyboard

- setting key clicks, 10-7

keyboard characteristics setup menu, 5-13

Korn shell

- environment file, 3-7, 4-5

ksh, 3-7, 4-5

L

labeling file system, 7-11

labelit, 7-11

lightness, saturation, and hue chart, 5-21

line printer, 6-1

linking files to rc2.d, 2-5, 2-9

listing contents of DOS directory, C-6

- loading software
 - after rebuild, 11-41
- local backup, 9-5
- location of message strips
 - changing, 5-31
- log file
 - creating, 5-7
 - creating from setup menu, 5-11
- logical unit number, 10-14
- login
 - adding login accounts, 3-5
 - assigning passwords, 3-13
 - creating window to log you in, 5-7, 5-11
 - customizing global variables, 4-6
 - customizing .env file, 4-9
 - customizing users login environment, 4-1
 - environment, 4-1
 - files executed in, 4-9
 - file
 - /etc/passwd, 4-4
 - process, 4-3
 - removing login accounts, 3-10
 - removing passwords, 3-16
 - from root, 3-19
- login directory, 3-7, 3-9, 4-5
- logging in as root, 3-3
- lost+found directory
 - removing files, 7-16
- lp
 - initializing at boot-up, 2-9
- LUN, 10-14
 - setting on Utility Pages server, 10-42
 - workstation, 10-14
- M**
- macro setup menu, 5-16
- macros
 - defining, 5-16
- Main Utility Menu
 - workstation, 10-3
 - server, 10-33
- major and minor numbers
 - for SCSI 0, LUN 0, 7-4
 - for SCSI 1, LUN 0, 7-5
 - for SCSI 2, LUN 0, 7-6
 - for SCSI 3, LUN 0, 7-7
- manager
 - system manager account, 3-3

managing
 adding passwords, 3-13
 forcing password change, 3-14
 monitoring system with CRM, 8-1
 removing passwords, 3-16
 from root, 3-17
 from Utility Pages, 3-19
 setting up initial system, 1-1

manipulating directories
 based on file permission modes, 3-21

manipulating ordinary files
 based on file permission modes, 3-22

math coprocessor
 enabling, 10-11

menu
 deleting options, 5-28
 modifying options, 5-28
 adding options, 5-26

menus
 catalog, 5-24
 display setup, 5-12
 general setup, 5-10
 keyboard characteristics, 5-13
 macro menu, 5-16
 printer setup, 5-15
 tab setup menu, 5-14

message strips
 configuring number and location of, 5-31

mkfs, 7-9, 9-29

mknod, 7-3 to 7-8

modify option
 workstation menu, 5-26

modifying
 inittab file, 2-6, 2-7
 gettydefs file, 2-6, 2-8

monitoring processes, 8-1

monitoring a print request, 6-4

monitoring disk space, 7-13

mounting, 7-12
 a fast file system, 7-12
 a standard file system, 7-12

moving message strips, 5-31

multiuser initialization, 2-9

multiuser mode
 files executed in, 2-9

multivolume backup
 using backup and restore, 9-18
 using scp/cpio, 9-13

N

- naming file system, 7-10, 7-11
- network backup, 9-4
- newfs, 7-11
- newgrp, 3-24
- newprod
 - after rebuild, 11-41
- node address, 10-12
 - setting nodename from Utility Pages
 - server, 10-40
 - workstation, 10-12
- number of message strips
 - changing, 5-30

O

- off delimiter, 2-6
- operating system
 - files executed at startup, 2-1
 - monitoring, 8-1
 - restoring/rebuilding, 11-1
 - setting default from Utility Page, 10-6
 - setting parameters on Utility Page
 - server, 10-40
 - workstation, 10-11
- Operating System Parameters Utility Page
 - server, 10-40
 - workstation, 10-11
- ordinary files
 - accessing based on file permission modes, 3-22
- other user, 3-21
- owner, 3-21
 - changing, 3-23

P

- parameters
 - device, 2-8
- parity
 - setting on server, 10-38
 - setting on workstation, 10-9
- partition, 11-20
 - adding
 - on server, 11-29
 - on workstation, 11-28
 - changing default sizes, 11-28
 - connecting to existing file system, 7-12
 - copying
 - server, 10-46
 - workstation, 10-26

- creating device file, 7-3 to 7-8
 - for SCSI 0, LUN 0, 7-4
 - for SCSI 1, LUN 0, 7-5
 - for SCSI 2, LUN 0, 7-6
 - for SCSI 3, LUN 0, 7-7
- creating file system on, 7-10
 - Fast File System, 7-11
 - standard file system, 7-10
- creating nonstandard sizes
 - on server, 11-29
 - on workstation, 11-28
- default sizes, 11-19
- mounting, 7-12
- overview, 7-20
- raw
 - backing up, 9-10
- reallocating space between partitions, 11-23
- removing partition
 - on server, 11-29
 - on workstation, 11-28
- root, 11-21
- boot, 11-20
- standard naming conventions, 11-25
- stash
 - determining size of, 11-21
- swap, 11-21
- dos, 11-21
- usr, 11-21
- stash, 11-21
- types of, 11-20
- understanding names of, 11-25
- using nonstandard partition sizes, 11-28

partition table, 10-23

partitioning

- specifying from Utility Pages
 - server, 10-44
 - workstation, 10-23

partitioning hard disk, 11-17

- creating default, 11-19
- explanation, 11-20
- restoring file systems, 11-33

passwd, 3-13

password

- Utility Page
 - setting on server, 10-33
 - setting on workstation, 10-4

- passwords, 3-13
 - changing, 3-13
 - creating, 3-13
 - forcing, 3-14
 - removing, 3-15
 - from root, 3-15
 - from Utility Pages, 3-18
 - restrictions, 3-13
- PATH
 - setting search path, 4-8
- pattern
 - changing background pattern, 5-32
- PC
 - read/write compatibility of floppy disks, B-1
- PC-DOS
 - copying files from DOS to UNIX, C-3
 - copying files from UNIX to DOS, C-4
 - exchange between System V, C-1
 - formatting floppy disks, 9-27
 - listing contents of DOS directory, C-6
 - setting as default operating system, 10-6
- Peripheral Configuration Utility Page
 - server, 10-37
 - workstation, 10-7
- permission modes
 - changing, 3-22
 - description, 3-20
- pipe
 - into cpio/scpio, 9-13
- plotting
 - determining size of stash partition, 11-21
- preparing floppies to back up data, 9-27
- printer
 - attaching to window, 5-15
 - setting up a dedicated, 6-1
- printer setup menu, 5-15
- printing from a dedicated printer, 6-4
- privileges
 - system management, 3-3
- problem
 - bad block messages, 11-2
 - can't boot workstation, 11-2
 - which option to choose from Rebuild menu, 11-33 to 11-50
- procedures
 - adding password, 3-13
 - adding user accounts, 3-5
 - addusers, 3-6
 - sysadm, 3-8

-
- backup and restore commands, 9-6 to 9-9
 - command recall from command line, 4-10
 - creating device file, 7-3 to 7-8
 - creating file system, 7-9
 - Fast File System, 7-10
 - standard file system, 7-9
 - creating rebuild floppy disk set, 11-3
 - creating windows, 5-4
 - customizing login environment, 4-1
 - customizing login prompt, 2-5
 - displaying disk space used, 7-14
 - displaying free disk space, 7-13
 - forcing password change, 3-14
 - formatting, 11-13
 - formatting floppy disk, 9-27
 - maintaining system, 7-13, 7-17
 - rebuilding from rebuild floppy disks, 11-1
 - rebuilding using nonstandard partitions, 11-20
 - removing password, 3-16
 - from root, 3-17
 - from Utility Pages, 3-19
 - removing user accounts, 3-10
 - repartitioning after format, 11-20
 - setting global variables, 4-6
 - setting up initial system, 1-1
 - steps for creating nonstandard partition
 - on server, 11-29
 - on workstation, 11-28
 - verify hard disk, 11-10
- process ID
- moving message strip locations, 5-31
- Process Monitors (CRM), 8-19
- processes
- adding to `.def_proc` file, 5-35
 - adding to `def_proc` file, 5-34
 - adding to `sys_proc` file, 5-36
 - adding to workstation menu, 5-26
 - getty, 2-5
 - init, 2-3
 - login process, 4-3
 - login, 4-3
 - rc2, 2-9
- products
- removing to allocate disk space, 7-16
- profile, 4-6
- protection codes (rwx)
- changing, 3-22
 - description, 3-20

R

- raw data
 - backups, 9-10
- read access, 3-20
- read/write compatibility of floppy disks, B-1
- read, write, and execute access, 3-20
 - modifying, 3-22
- reading VMS backup tapes using vmsbackup, 9-21
- reallocating space between partitions, 11-23
- reboot
 - automatic after system crash, 10-12
- rebuild
 - floppy disks
 - booting from, 11-7
 - menu options, 11-34 to 11-50
 - restoring files, 11-42
- rebuild floppy disks, 11-3
 - options, 11-4, 11-5
- rebuilding, 11-1
 - conditions requiring, 11-2
 - creating rebuild floppy disks, 11-3
 - loading software using newprod, 11-41
 - overview of steps, 11-2
 - entire disk, 11-34
 - root file system, 11-35
 - usr file system, 11-36
 - to change to default partition sizes, 11-20
- recall
 - buffered screens, 5-11
 - command recall, 4-10
- recalling commands, 4-10
- reconfigure
 - number of virtual screens, 5-30
- redirection
 - to scp/cpio command for backups, 9-13
- regis, 5-15
- reminder service, 4-14
- remote
 - backups to remote tape drive, 9-4
- remote tape controller command, 9-4, 9-5, 9-22
- removing partition
 - on server, 11-29
 - on workstation, 11-28
- removing files to allocate disk space, 7-14 to 7-16
- removing passwords, 3-16
 - from root, 3-17
 - from Utility Pages, 3-19

- removing products, 7-16
- removing user accounts, 3-10
 - using rmusers, 3-10
 - using sysadm, 3-11
 - comparison of rmusers and sysadm, 3-10
 - manually, 3-10
- repartitioning
 - see partitioning, 11-20
- repartitioning for nonstandard partitions, 11-20
- respawn delimiter, 2-6
- restore command, 9-19
- restoring files, 9-1
- rmusers, 3-10
- root account, 3-3
 - removing password from, 3-17
- root partition, 11-21
- RS-232 interface assembly, 2-6
- RS-232 parameters
 - auxiliary port settings
 - server, 10-37
 - workstation, 10-8
- rtc, 9-4, 9-5, 9-22
- run levels, 2-3
 - processes invoked in run level 2, 2-3
- running command at specific time, 4-14
- running commands at regular intervals, 4-15
- rwX permissions modes, 3-20
 - changing, 3-22

S

- saturation, hue, and lightness chart, 5-21
- saving window configuration, 5-24
- scpio, 9-13
- screen
 - changing color, 5-33
 - configuring number of virtual screens, 5-30
- screen manager, 5-2
 - invoking default processes from def__proc, 5-34
 - invoking default processes from .def__proc, 5-35
 - invoking default processes from sys__proc, 5-36
- screen saver
 - setting, 10-7
- SCSI 0, LUN 0
 - creating device file for, 7-4
- SCSI 1, LUN 0
 - creating device file for, 7-5
- SCSI 2, LUN 0
 - creating device file for, 7-6

- SCSI 3, LUN 0
 - creating device file for, 7-7
- SCSI ID
 - setting on Utility Pages
 - server, 10-41
 - workstation, 10-14
- security
 - root account, 3-3
- server
 - backing up to remote tape drive using, 9-4
 - Utility Pages for, 10-30
- setting
 - default processes, 5-34 to 5-36
 - display characteristics, 5-12
 - tab stops, 5-14
 - search path, 4-7, 4-8
 - up reminder service, 4-14
 - up terminal characteristics, 5-10
 - up initial system, 1-1
- setting nodename on Utility Pages
 - server, 10-40
 - workstation, 10-12
- setting bell tone, 10-8
- setting up a line printer, 6-1, 6-3
- setup menus
 - catalog, 5-24
 - display menu, 5-12
 - general, 5-10
 - keyboard characteristics, 5-13
 - macro, 5-16
 - printer, 5-15
 - tabs, 5-14
- shells, 3-7, 3-9, 4-5
- showconfig command, 2-6
- single user
 - booting from Utility Page
 - server, 10-40
 - workstation, 10-11
- size
 - displaying file size, 7-14
- smgr
 - customizing, 5-2
- smgr_vlt, 5-33
- software
 - loading after rebuild, 11-41
 - removing software products, 7-16

- space
 - allocating to each partition, 11-20
 - monitoring disk space used and free, 7-13, 7-14
 - reallocating to another partition, 11-23
- special files, 7-3 to 7-8
- starting accounting and lp, 2-10
- startup
 - files executed, 4-1
 - files executed at system startup, 2-1
- stash partition, 11-21
 - determining size of, 11-21
- stop bits
 - setting on server, 10-38
 - setting on workstation, 10-10
- streaming cpio, 9-13
- su, 3-3, 2-8
- submitting a print request, 6-4
- superuser, 3-3
 - accessing, 3-3
- swap partition, 11-21
- symbols
 - creating symbol for collapsed window, 5-7
- sysadm
 - adding user accounts, 3-8
 - advantages, 3-6
 - removing user accounts, 3-11
- system configuration
 - devising backup plan, 9-4
- system maintenance
 - removing user accounts, 3-10
- system management
 - adding user accounts, 3-5
 - customizing login environment, 4-1
 - files for user accounts, 4-1
 - files executed at system startup, 2-1
 - forcing password change, 3-14
 - maintaining system, 7-13 to 7-17
 - monitoring disk space, 7-14 to 7-16
 - passwords, 3-13
 - removing user accounts, 3-10
 - setting up system, 1-1
 - system manager account, 3-3
- System Monitors (CRM), 8-7

- system security
 - adding passwords, 3-13
 - forcing password change, 3-14
 - removing passwords, 3-16
 - from root, 3-17
 - from Utility Pages, 3-19
 - root account, 3-3
- system setup
 - adding user accounts, 3-5
 - initial tasks, 1-1
- system startup
 - files executed, 2-1
 - in multiuser mode, 2-9
 - setting global variables in /etc/profile, 4-6
- systemv
 - removing products to reclaim disk space, 7-14 to 7-16

T

- tab setup menu, 5-14
- tape archive utility (tar), 9-11
- tape drive
 - allocating remote drive, 9-4, 9-22
 - backing up to, 9-4, 9-5, 9-22
 - backing up to using backup and restore, 9-18
 - using scp/cpio, 9-13
- tapes
 - exchanging workstation and VAX, 9-20, 9-21
 - using rtc command, 9-4, 9-5, 9-22
- tar, 9-11
- temporary files
 - removing, 7-14
- terminal
 - getty process, 2-5
 - setting default terminal type, 2-5
- terminal characteristics
 - setting up, 5-10
- terminal settings, 2-8
- termio, 2-8
- timezone
 - setting on Utility Page
 - server, 10-34
 - workstation, 10-4
- topcpu (CRM), 8-15
- topfault (CRM), 8-12
- topio (CRM), 8-15
- topmem (CRM), 8-14
- topsys (CRM), 8-17

troubleshooting

- rebuild options, 11-33 to 11-50
- rebuilding using nonstandard partitions, 11-20
- reasons to rebuild, 11-2
- when you receive bad block messages, 11-2

U**UID, 4-5****ulimit, 4-6****umask, 4-8****user accounts**

- /etc/passwd file, 4-4
- adding, 3-5
- adding passwords, 3-14
- adding with addusers, 3-6
- adding with sysadm, 3-8
- customizing .env file, 4-9
- customizing variables in .profile file, 4-8
- removing, 3-10
- removing passwords, 3-16

user types: owner, group, other, 3-20**users**

- adding user accounts, 3-5
- assigning passwords, 3-13
- customizing login environment, 4-1
- group identification code (GID), 4-5
- identification code (UID), 4-5
- removing passwords, 3-16
- removing user accounts, 3-10
- user accounts file, 4-4

usr partition, 11-21**utd, C-4****Utility Pages**

- format, 10-23, 10-43
- reference for workstations and servers, 10-1
- removing password, 10-4, 10-35
- server, 10-30
 - Device Image Copy Utility Page, 10-46
 - Disk Maintenance Utility Page, 10-41
 - Disk Partitioning Utility Page, 10-44
 - Hard Disk Flaw Data, 10-43
 - Main Utility Menu, 10-33
 - Peripheral Configuration Utility Page, 10-7
 - Startup Configuration Utility Menu, 10-32
 - using, 10-31
 - verify hard disk, 10-43

workstation

- Device Image Copy Utility Page, 10-26
- Disk Maintenance Utility Page, 10-13
- Disk Partitioning Utility Page, 10-23
- Format Utility Page, 10-21, 10-44
- Hard Disk Flaw Data, 10-16
- Main Utility Page, 10-3
- Operating System Parameters Page, 10-11, 10-40
- Peripheral Configuration Utility Page, 10-37
- Verify Utility Page, 10-19,

utilities

- System V and PC-DOS, C-1

V

VAX/VMS

- backing up to remote VAX tape drive, 9-21
- reading VMS-generated backup tapes, 9-21

Verify

- explanation, 11-10
- on server, 11-12
- on workstation, 11-11

Verify Utility Page

- server, 10-43
- workstation, 10-19

virtual screens, 5-30

- changing number of, 5-30

vlt

- changing default screen color, 5-33

VMSbackup, 9-21

vt200

- customizing, 5-2, 5-4

vt220, 5-2

- options for creating windows, 5-4
- setup menu, 5-8 to 5-16

vterm, 5-4

- command-line options, 5-4 to 5-8

W

watcher (CRM), 8-19

window icon

- creating symbol for collapsed state, 5-7

window size

- customizing, 5-7

windows

- attaching printer to, 5-15
 - cataloging, 5-24
 - creating, 5-4
 - a collapsed window, 5-7
 - a color window, 5-6
 - using ANSI, 5-22
 - using regis commands, 5-18, 5-19
 - a log file of activity, 5-7
 - a specific title, 5-7
 - a window that uses symbol when collapsed, 5-7
 - a window with small text, 5-6
 - an inactive window, 5-6
 - console, 5-36, 5-6
 - from command line, 5-4
 - that logs you in automatically, 5-7
 - through workstation menu, 5-8
 - to uncollapse when it receives output, 5-6
 - using catalog file, 5-24
 - with specific origin and size, 5-6, 5-7
 - creating log file, 5-7
 - customizing defaults in `.def__proc` file, 5-35
 - customizing defaults in `def__proc` file, 5-34
 - customizing defaults in `sys__proc` file, 5-37
 - saving window configuration, 5-24
 - setting display characteristics, 5-12
 - setting up terminal characteristics, 5-10
 - vterm options, 5-4 to 5-6
- word size**
- server, 10-38
 - workstation, 10-9
- workstation menu**
- commands associated with menu options, 5-27
 - configure, add option, 5-26
 - configure, background, 5-32
 - configure, delete option, 5-28
 - configure, modify option, 5-28
 - configure, screens option, 5-30
 - creating windows, 5-4
 - customizing
 - adding processes to, 5-26
 - deleting processes on, 5-28
 - modifying processes on, 5-28
- write access, 3-20**

X

xon/xoff

setting on server, 10-39

setting on workstation, 10-10

Z

z-buffering, 10-12