ULTRIX-32
Guide to
System Backup and Restore

Order No. AA-ME92A-TE

ULTRIX-32 Operating System, Version 3.0

Digital Equipment Corporation

This manual was written and produced by the ULTRIX Documentation Group in Nashua, New Hampshire.

Contents

About This Manual

1 Backup and Restore Methods and Strategies

2 Backup Commands and Procedures

3 Local File Restoration

4 Remote Backup

5 Remote File Restoration

A Device Mnemonics

Examples

Figures

Tables

About This Manual

The objective of this guide is to provide you with information on backing up and restoring files and file systems using ULTRIX system commands and utilities. The guide will assist you in developing system backup strategies locally and in a network environment. It also presents guidelines from which you can develop specific procedures for your site.

Audience

The *ULTRIX–32 Guide to System Backup and Restore* is written for the person responsible for managing and maintaining an ULTRIX system. It assumes that this individual is familiar with ULTRIX commands, the system configuration, the system's controller/drive unit number assignments and naming conventions, and an editor such as vi or ed. You do not need to be a programmer to use this guide.

Organization

This manual consists of 5 chapters, one appendix, and an index. The chapters and the appendix are:

Chapter 1: Backup and Restore Methods and Strategies
Identifies backup and restore methods and suggests certain strategies for backing up and restoring files and file systems.

Chapter 2: Backup Commands and Procedures
Identifies and explains how to use the various ULTRIX backup commands and utilities.

Chapter 3: Local File Restoration
Describes how to use the restore command and provides procedures for restoring file systems.

Chapter 4: Remote Backup
Explains the network environment and how to use the opser utility to perform remote backups.

Chapter 5: Remote File Restoration
Explains how to restore files and file systems using the rrestore command. This chapter also contains procedures for restoring the root and /usr file systems after a head crash has occurred.

Appendix A: Device Mnemonics
Lists the supported device mnemonics and explains how to obtain detailed reference page information on devices.

Related Documents

You should have the hardware documentation for your system and peripherals.

Conventions

The following conventions are used in this manual:

special	In text, each mention of a specific command, option, partition, pathname, directory, or file is presented in this type.
command(x)	In text, cross-references to the command documentation include the section number in the reference manual where the commands are documented. For example: See the cat(1) command. This indicates that you can find information about the cat command in Section 1 of the reference pages.
literal	In syntax descriptions, this type indicates terms that are constant and must be typed just as they are presented.
italics	In syntax descriptions, this type indicates terms that are variable.
[ ]	In syntax descriptions, square brackets indicate terms that are optional.
. . .	In syntax descriptions, a horizontal ellipsis indicates that the preceding item can be repeated one or more times.

function	In function definitions, the function itself is shown in this type. The function arguments are shown in italics.
UPPERCASE	The ULTRIX system differentiates between lowercase and uppercase characters. Enter uppercase characters only where specifically indicated by an example or a syntax line.
example	In examples, computer output text is printed in this type.
example	In examples, user input is printed in this bold type.
%	This is the default user prompt in multiuser mode.
#	This is the default superuser prompt.
>>>	This is the console subsystem prompt.
.	In examples, a vertical ellipsis indicates that not all of the
.	lines of the example are shown.
.	
<KEYNAME>	In examples, a word or abbreviation in angle brackets indicates that you must press the named key on the terminal keyboard.
<CTRL/x>	In examples, symbols like this indicate that you must hold down the CTRL key while you type the key that follows the slash. Use of this combination of keys may appear on your terminal screen as the letter preceded by the circumflex character. In some instances, it may not appear at all.

Backup and Restore Methods and Strategies 1

This chapter provides information on the backup and restore process and offers practical suggestions for establishing local and remote backup strategies. The chapter:

- Explains default specifications used by the backup and restore commands
- Describes backup and restore methodologies
- Offers backup and restore strategy suggestions

1.1 Default Specifications

In this guide, many of the examples do not specify a device. The tar, dump, and restore commands, when used without specifying a device, default to the /dev/rmt0h device. If you use the dump command and your backup media is a diskette, you must use the -f option and specify the backup device. The default for the mdtar command is /dev/rra1a. On a VAX processor with an installed tape drive, the default device is /dev/rmt0h, which is the high density, rewind tape drive.

For more information on these commands and their default devices, see tar(1), mdtar(1), dump(8), and restore(8) in the ULTRIX Reference Pages.

1.2 Overview of the Backup and Restore Methodologies

The method you use to backup and restore entire file systems or individual files and directories depends on several factors. For example, you must know if the target is local or remote, what hardware is attached to the system (disks, tapes, or diskettes), and whether the work must be done in single-user or multiuser mode. The following sections briefly describe three backup and restore methods; subsequent chapters provide more details.

1.2.1 Local Backup and Restore -- Single-user Mode

You use the dump and restore or opser commands to backup and restore file systems residing on a local system. To use these commands, you must be in single-user mode.

- The dump command performs the file system backup. See dump(8) in the ULTRIX Reference Pages for more information on this command.
- The restore command performs a partial or complete restore of the file system(s) that you backed up with the dump command. See restore(8) in the ULTRIX Reference Pages for more information on this command.
- The opser command performs an interactive file system backup. See opser(8) in the ULTRIX Reference Pages for more information on this command.

When performing a local backup and restore using these commands, you place the system in single-user mode, unmount the file systems to be backed up, perform checks of those file systems, and back up and restore the file systems. See Chapter 2 and Chapter 3 for a detailed description and examples of these commands and procedures.

1.2.2 Local Backup and Restore -- Multiuser Mode

You use the tar or mdtar command to backup and restore individual files and directories that reside on a local system. You can use these commands while the system is in multiuser mode.

- The tar command performs the backup and restore to tape. See tar(1) in the ULTRIX Reference Pages for more information on this command.
- The mdtar command performs the backup and restore to diskette. See mdtar(1) in the ULTRIX Reference Pages for more information on this command.

These commands enable users to back up their own files or directories while the system is in multiuser mode. The user does not need superuser privileges, but must own the target file(s) or directory. See Chapter 2 and Chapter 3 for a detailed description and examples of these commands and procedures.

1.2.3 Remote Backup and Restore -- Single-user Mode

You use the rdump, the rrestore, or the opser commands to backup and restore a remote system across the network.

- The rdump command performs a remote file system backup. See rdump(8) in the ULTRIX Reference Pages for more information on this command.

- The rrestore command performs a remote file, directory, or file system restore. See rrestore(8) in the ULTRIX Reference Pages for more information on this command.
- The opser command performs an interactive remote backup and restore. See opser(8) in the ULTRIX Reference Pages for more information on this command.

When performing a remote backup and restore using these commands, you must be in single-user mode and have superuser privileges. See Chapter 2, Chapter 4, and Chapter 5 for a detailed description and examples of these commands and procedures.

1.3 Developing a Backup and Restore Strategy

Regardless of whether you need to perform local or remote backups, you should develop a backup strategy that minimizes both the time and number of tapes or diskettes required. Your backup strategy depends on the needs and resources of your site. Because the dump command performs incremental backups that can, if unplanned, waste both time and media, you should develop a practical way of backing up your file systems.

1.3.1 Sample Backup Sequence

A simple backup sequence could start with a monthly level-0 backup, followed by weekly level-5 backups, and daily level-9 backups. This sequence ensures that all files that have changed since the date of a lower level are backed up.

The following example shows a sample backup sequence that backs up the files in the root file system to the default tape drive. It also shows how to use the u option. This option instructs the dump command to update the /etc/dumpdates file after it successfully completes the backup.

```
# /etc/dump 0u /   (1st of month, dumps the whole file system)
# /etc/dump 9u /   (daily)
          .
          .
          .
# /etc/dump 5u /   (1st full week)
# /etc/dump 9u /   (daily)
          .
          .
          .
# /etc/dump 5u /   (last full week)
# /etc/dump 9u /   (daily)
          .
          .
          .
```

The dump command uses the date stored in the /etc/dumpdates file as a time stamp for determining which files are to be backed up. A Level-0 backup is defined as the highest and a level-9 backup is defined as the lowest in order of precedence. The dump command compares the last modified date of a given file with that of the specified backup level's last backup date. Then it backs up only those files that have been changed since the appropriate date and time stamp of a lower level number.

Backup Commands and Procedures 2

This chapter explains how to back up files and file systems using the ULTRIX backup and file copy/transfer commands and utilities. The backup and file transfer commands described in this chapter enable you to:

- Back up file systems
- Transfer files and file systems to tapes or diskettes
- Perform backups from the operator account

The following sections describe how to use the ULTRIX commands to perform these tasks.

2.1 Backing up File Systems

Use the dump command to back up file systems. This command enables you to back up a file system to a tape or disk device. To do a file system backup, you must shut down the system to single-user mode. The following procedure shows you how to perform a file system backup of the root (/) and the /usr file systems:

1. Shut down the system. To shut down the system in five minutes and give users periodic warning messages, type:

```
# /etc/shutdown +5 'System going down to perform backups'
```

2. Determine the names of the devices containing the file systems to be backed up. You can determine the device names by displaying the contents of the /etc/fstab file and noting the device names associated with the file systems that you want to back up. For example:

```
# cat /etc/fstab
/dev/ra0a:/:rw:1:1:ufs::
/dev/ra0g:/usr:rw:1:3:ufs::
```

This example shows that the / and /usr file systems are on the ra0 device, partitions a and g respectively.

You can also use the dump command with the w option to determine which file systems are to be backed up. The w option displays the file systems in the /etc/dumpdates file that have to be backed up. The w option will work provided that you have been updating the

/etc/dumpdates file by using the u option when performing backups.

3. Unmount the file systems to be backed up by using the umount command with the -a option. The root (/) file system remains mounted.

```
# umount -a
```

4. Perform a file system check on the file system that you want to back up by using the fsck command. The fsck command will check the file system and provide you with messages and prompts. This will ensure the integrity of the file system. For example:

```
# fsck /dev/ra0a
```

5. Perform a level-9 backup of the root (/) file system to the default tape device (/dev/rmt0h, tape drive 0). For example, type:

```
# /etc/dump 9u /
```

6. Perform a level-9 backup of the /usr file system to a disk device with a unit number of 2. For example, type:

```
# /etc/dump 9uf /dev/rra2a /usr
```

   Notice that the f option immediately precedes the /dev/rra2a device specification.

In these level-9 backups, the dump command provides you with several messages and prompts you to insert either a tape or a disk, depending on the device type that you specify. It also prompts you with a number of yes and no questions. Answer all of the questions.

Change the tape or disk when the program prompts you to do so. Be sure to label all of the disks or tapes with the date of the backup, the file system name, the starting inode number for that volume, and the backup level. This will make it easier to locate files and file systems that you want to restore.

When the last disk or tape is filled, remove it from the drive and store it in a safe, secure place.

If you want to tailor the backup process so that certain options are in effect, see dump(8) in the ULTRIX Reference Pages.

2.2 Transferring Files to Tape or Diskette

The tar and mdtar commands allow you to transfer an individual user's directories or files to tapes or diskettes. You do not need superuser privileges, nor do you have to be in single-user mode to use these commands. However, you do need to have read permission on the files

you are backing up.

Use the tar command when magnetic tape is the primary backup medium. It is a tape archive facility that supports single-volume and multivolume archives.

System users should invoke the tar command to back up their files provided that the system has a tape drive. The following example shows how to back up a directory named /usr/staff/jmf to tape. This example assumes that tape drive 0, the default device, is being used:

```
# cd /usr/staff
# tar c ./jmf
```

To use the tar command along with the device name, type:

```
# tar cf /dev/rmt1h ./jmf
```

Use the mdtar command when diskettes are the primary backup medium. Diskettes have less storage capacity than magnetic tape reels, so more than one diskette may be needed. Given a directory pathname, the mdtar command makes a list of the files and decides how to fit them on multiple diskettes.

The following example shows how to back up the directory /usr/staff/jcj to diskette(s):

```
# cd /usr/staff
# mdtar c ./jcj
```

To use the mdtar command along with the device name, type:

```
# mdtar cf /dev/rmt1h ./jcj
```

The mdtar command prompts you to insert and remove additional diskettes as needed to back up the files.

2.3 Backing up From the Operator Account

When you log in to the operator account, the system automatically invokes the opser utility. The opser utility, /opr/opser, provides a simple and concise interface for ULTRIX system maintenance, primarily file system backups. The opser utility runs as a shell environment in place of the system shell when you log in as the operator. The operator password must be set up by the superuser.

The opser utility enables you to:

- Determine who is currently logged in to the system
- Shut down multiuser mode, enter single-user mode, and restart multiuser mode

- Check file system consistency
- Perform incremental or full file system backup to tape
- Perform incremental or full file system backup over the network
- Halt the processor
- Execute opser options on a remote system.

The opser utility also allows you to escape to the shell, execute ULTRIX system commands, and return to the opser utility. It also provides on-line help in the form of an options menu. For more information, see opser(8) in the ULTRIX Reference Pages.

2.3.1 Hardware Requirements

To perform system maintenance, you should run the opser utility from the operator's console. Running the opser services from a terminal limits you to the help, user, and shell functions.

The opser utility supports these tape devices for backup operations: TE16, TS11, TSV05, TU77, TU78, TU79, TU80, TU81, TU81E, TK50, and TK70. It also supports the network as a backup device.

2.3.2 Running the opser Utility

To run the opser utility, log in to the operator account. Once running, the opser utility prints a sign-on message, two informational messages, and a command prompt:

```
ULTRIX-32 Operator Services

Line editing: delete - erase one character, ^U - kill entire line

For help, type h followed by a return

opr>
```

The opr> prompt informs you that the opser utility is running in place of the shell and is ready to accept opser options.

To use an opser option, type the appropriate letter or name. Then, press the RETURN key. For example, to display on-line help about all opser options, type:

```
opr> h
```

The opser utility will display the following:

```
() - may use first letter in place of full name
Valid commands for Local Opser are:

!sh                        - shell escape (execute ULTRIX-32 commands)
                             (Type control d to return from shell)
(u)sers                    - show logged in users
(s)hutdown                 - stop time-sharing
(d)ismount                 - unmount file systems
(f)sck                     - file system checks
(r)estart                  - restart time-sharing
(h)elp                     - print this help message
(b)ackup                   - file system backup
halt                       - halt processor
(n)etwork [Slave]          - initiate Remote Opser
(q)uit                     - exit from opser
```

To end an opser session and return to a login prompt, type q.

The next sections provide detailed discussions of the opser options.

2.3.3 Determining Who Is Logged In

To determine how many users currently are logged in to the system, enter the u (users) option at the opr> prompt. For example:

```
opr> u
```

When you enter the u option, the opser utility displays a list of users who are currently logged in.

2.3.4 Shutting Down Multiuser Mode

Use the s (shutdown) option to place the ULTRIX system in single-user mode. You must be in the single-user mode before you can run a file system check and before you can back up the file systems. To shut down multiuser mode and leave the system in single-user mode, enter the s (shutdown) option at the opr> prompt. For example:

```
opr> s
```

When you enter the s option, the opser utility first displays the names of those users currently logged into the system. Then, it prompts for the number of minutes to delay before shutting down from multiuser to single-user mode. The number of minutes that you enter must be between 1 and 99. This delay gives users enough time to finish their work and to log out. During this delay period, the opser utility broadcasts warnings of the impending shutdown. Five minutes prior to the designated time, it disables logins. Finally, at the designated time, it broadcasts a final

message and then shuts down multiuser mode.

Prior to entering single-user mode from multiuser mode, the opser utility kills all running processes and synchronizes the file systems (using the sync utility). When the opser utility redisplays the opr> prompt, you are automatically in a single-user environment. You can now enter other opser options and continue with system maintenance.

2.3.5 Unmounting File Systems

To unmount the file systems when running opser in single-user mode, enter the d (dismount) option at the opr> prompt. For example:

```
opr> d
```

When you enter the d option, the opser utility invokes the umount command to unmount file systems listed in the /etc/fstab file. For more information on the umount command, see mount(8) in the ULTRIX Reference Pages.

To ensure that all of the file systems have been unmounted, escape to the shell, using the !sh option as described in Section 2.3.8, and enter the mount command. Without options, the mount command displays all of the currently mounted file systems. At this point, you will have to unmount the file systems by issuing the appropriate umount command.

The following example shows the sequence of commands that you would enter assuming that the /usr/staff3 file system is still mounted after you entered the d option.

```
opr> !sh
Password:
type ^D to return to opser

# /etc/mount
/dev/ra0a on / type ufs
/dev/ra1h on /usr/staff3 type ufs
# /etc/umount  /dev/ra1h
# <CTRL/D>
opr>
```

The sequence of commands shows the shell escape option, !sh, which causes the system to respond with a # shell prompt after the root password is entered. Next, the sequence shows the system response to the mount command, which shows that the root (/) and /usr/staff3 file systems are still mounted (the root file system should be mounted). Notice that the file system /usr/staff3 is unmounted by unmounting /dev/ra1h. This is the device (ra1) and partition (h) on which the /usr/staff3 file system is mounted. Lastly, the sequence shows the use of a

CTRL/D to return to the opr> prompt.

Note

When you unmount the /usr file system, some ULTRIX commands may be unavailable.

2.3.6 Checking File System Consistency

The file systems should be checked prior to performing a backup. You should not back up a corrupted file system. To check the consistency of your file systems when running opser in single-user mode, enter the f (file system check) option at the opr> prompt. For example:

```
opr> f
```

When you enter the f option, the opser utility invokes the fsck command to check the file systems indicated in the /etc/fstab file.

2.3.7 Backing Up File Systems

You should unmount and then check all file systems before backing them up. Use the d option to unmount the file systems. Use the f option to check the file systems for consistency. To shut down multiuser mode prior to unmounting and checking file systems, specify the s (shutdown) option.

To back up file systems when running the opser utility in single-user mode, enter the b (backup) option at the opr> prompt. Enter the option using the format: b *command file name*

The *command file name* argument specifies the command file that contains the ULTRIX system commands required for that backup. If you do not use the *command file name* argument, the opser utility calls the default backup command file, /opr/backup. This script enables you to perform level 0 or level 9 backups to supported tape drives.

The /opr/backup command file prompts you to specify the tape drive number, the logical unit number of the tape drive, and the tape density. For example:

```
opr> b backup

ULTRIX-32 File System Backup Procedure

start errlog daemon - elcsd
elcsd:

Please specify the tape drive type. (eg. te16):
Please specify the logical unit number for the tape drive. (eg. 0):
Please specify tape density. (eg. 1600):
```

If you specify either a TK50 or a TK70 for the tape drive type, this backup script will not prompt you for a tape density. If any of your entries are invalid, the command file responds with the message:

```
Error in specifying tape drive - repeating questions.
```

Once you have correctly specified the device, the backup script prompts:

```
Backup to be full or incremental? < f or i >:
```

If you enter an f, a level 0 backup is performed. If you enter an i, a level 9 backup is performed.

After you have correctly responded to the prompts, the file system dump process begins.

The default backup command file also invokes the error logger daemon in single-user mode, so that error logging is enabled during backups. The error logger daemon is described in the Guide to the Error Logger System.

As the superuser, you should set up backup command files in the way that you want your backups performed.

2.3.8 Escaping to the Shell

To escape to the shell at any time when running the opser utility, enter !sh at the opr> prompt. The opser utility prompts you to enter the root password. This is the only password you can enter. For example:

```
opr> !sh
Password:
type ^D to return to opser

#
```

After receiving a shell prompt, you can execute any ULTRIX command. The !sh command is useful for using the talk command to tell users that you intend to shut down the system. After you finish entering ULTRIX commands, press CTRL/D to return to the opr> prompt.

2.3.9 Restarting Multiuser Mode

To restart multiuser mode when running the opser utility in single-user mode, enter the r (restart) option at the opr> prompt. For example:

```
opr> r
```

This option causes the system to restart multiuser mode. The system displays the same startup messages and starts the same daemons as though a normal startup had occurred. Once the system is back in multiuser mode, it again displays the standard login prompt.

2.3.10 Performing a Remote Backup

The opser utility enables you to backup a file system over the network. Chapter 4 explains how to use the opser utility to perform a remote backup.

2.3.11 Halting the Processor

To halt the processor when running the opser utility in single-user mode, enter the h (halt) option at the opr> prompt. For example:

```
opr> halt
```

This command synchronizes the disks (writes out in-memory file system information to the disks) and then halts the processor. You will then have to reboot your system manually as described in the Guide to System Shutdown and Startup.

Note

To shut down multiuser mode prior to halting the processor, specify the s (shutdown) option.

2.3.12 Exiting opser

You can exit the opser utility and return to the login prompt by using the q option, provided that you have not shut the system down to single-user mode. Once you shut the system down, you must use the r (restart) option to exit. Section 2.3.9 describes how to use the r option. If you enter q at the opr> prompt after shutting down the system, opser displays the message:

```
Time-sharing stopped
opr>
```

If you have not shut the system down and you want to exit opser, enter the q (quit) option at the opr> prompt. For example:

```
opr >q
```

The opser utility will log you out of the operator account and the system will redisplay the login prompt.

2.4 The Labeled Tape Facility

In addition to the backup methods already described, the system contains a labeled tape facility command, ltf. The ltf command reads and writes single-volume, versions three and four, ANSI-compatible tape volumes. This provides you with a way of accurately transferring information between ULTRIX and other systems. For information on this command, refer to ltf(1) and ltf(5) in the ULTRIX Reference Pages.

Local File Restoration 3

Local file restoration can be accomplished in a number of ways. This chapter identifies the more commonly used methods. The chapter:

- Explains how to restore files, directories, or file systems on different media using the restore command
- Explains how to use the tar and mdtar commands to recover files
- Provides step by step procedures for restoring a complete file system
- Contains procedures for restoring the root (/) and /usr file systems after a catastrophic event has occurred.

3.1 Using the restore Program

You use the restore program to restore files, file systems, and directories that you backed up with the dump program.

Note

If you used the tar or mdtar command to backup your files, you must use the tar or mdtar command again to restore the backup media. See section 3.7 for details.

With the restore program, you can do restores either interactively or noninteractively. The following sections explain both methods with examples of how to perform each.

3.2 Preparing For An Interactive Restore

Before starting the actual restoration, you should prepare in three ways:

- Log in as yourself or superuser to root
- Mount the dump media
- Change your working directory to the dump media mount point.

The following sections describe each of these preparatory tasks.

3.2.1 Logging In

Some restore functions require superuser privileges. For example, if you intend to change file and directory attributes when restoring, you must be superuser. See restore(8) in the ULTRIX Reference Pages for details on the restore command options and required permissions.

3.2.2 Mounting the Dump Media

Before invoking the restore program, ensure that the backup media containing the dump image is mounted on the appropriate device. This is usually the default tape device /dev/rmt0h, but it could be any device or a file system in the form of an on-line backup file. In a multi-volume dump set, you must mount volume 1 first since it contains information on the directory structure and inodes for all files on the dump image. Once you have established the restore environment, you can mount other volumes in the set.

3.2.3 Changing Your Working Directory

Change your current working directory to the mount point of the file system that was backed up. For example, if the /usr file system was backed up from root, first position yourself at root, then enter the command to invoke restore from this location.

Note

If you are unsure of the mount point, enter the /etc/restore command with the -t option for a complete listing of the dump media contents.

Be aware that once you invoke the restore program you enter the restore environment. While in this environment, you get directory information about the dump media only, not your working environment.

Remember, the restore program restores files to the current working directory (the directory from which you invoked the program) using the relative pathname of the dump media file(s) that are being restored.

3.3 Restoring Files Interactively

To restore individual files or directories from the dump image on the backup media, use the /etc/restore command with the -i option. The -i option specifies that you want to do the restore in interactive mode.

3.3.1 Invoking the Interactive restore Program

The location of the dump image determines what command format you use to invoke the restore program.

- If the dump image is located on a device other than the default (/dev/rmt0h), use the format:

  ```
  # /etc/restore -if device
  ```

  The -i option specifies interactive mode. The -f option and *device* argument let you specify the name of the archive. Include the -f option and *device* argument when the archive is a device other than /dev/rmt0h. For example, to restore files interactively from a diskette with a *device* name of rra2a, type:

  ```
  # /etc/restore -if /dev/rra2a
  ```

  However, if the device is the default tape drive /dev/rmt0h, do not include the -f option and *device* argument on the command line. Use this format:

  ```
  # /etc/restore -i
  ```

- If the dump image is an on-line backup file on a mounted disk, use the format:

  ```
  # /etc/restore -if file
  ```

  Here, *file* specifies the dump image backup file. Specify either the full or relative pathname depending on your current working directory. For example, to restore the /usr file system from a dump image named /usr/backups, position yourself at root and type:

  ```
  # /etc/restore -if /usr/backups
  ```

3.3.2 Using Interactive Mode Commands

When you enter the /etc/restore command with the -i option, the restore> prompt appears on your screen. The prompt signals that the restore program is waiting for a command.

3.3.2.1 Listing the Commands – To get a listing of the available commands, enter an h or a ? at the restore > prompt. The program responds with a listing. For example:

```
restore >h
Available commands are:

ls [arg] - list directory
cd arg - change directory
pwd - print current directory
add [arg] - add `arg' to list of files to be extracted
delete [arg] - delete `arg' from list of files to be extracted
extract - extract requested files
quit - immediately exit program
verbose - toggle verbose flag (useful with "ls")
help or `?' - print this list
If no `arg' is supplied, the current directory is used
restore >
```

Note

Remember, after invoking the restore program in interactive mode, you are in the restore environment. The program responds to your subsequent commands with information about the dump media, not about your standard environment.

3.3.2.2 Listing the Pathname of the restore Working Directory – To list the pathname of the working directory within the restore environment, type pwd at the restore > prompt. For example:

```
restore >pwd
/lib
restore >
```

3.3.2.3 Getting a File and Directory Listing – To get a listing of the files and directories that can be restored at the current directory level, type ls at the restore > prompt. For example:

```
restore >ls
 adm/           hosts/         man            skel/
 bin/           include/       mdec/          spool/
 dict           kits           msgs/          src/
restore >
```

To get a listing of the files and directories that can be restored at a different directory level, enter the pathname of that directory as an argument to the ls command.

3.3.2.4 Comparing Inode Numbers – By comparing the displayed inode numbers with the starting inode numbers recorded on a multivolume dump set, you can find the tape or disk that contains the file you want to restore. To see the inode numbers of the files contained within your listing, first use the verbose command to enter verbose mode, then use the ls command to get your listing. The restore program displays the files with the inode numbers to the left of the dumped file names. For example:

```
restore >verbose
verbose mode on
restore >ls
    2  *./       11971  hosts       14253  spool/
    2  *../       2177  include/     4145  src/
11969  adm/      11427  mdec/        4253  staff
 1633  bin/       2722  msgs/        3266  sys/
restore >
```

As shown above, the hosts file has an inode number of 11971 and the spool directory has an inode number of 14253.

Note

Sometimes the inodes for a given file span multiple tapes or disks. When this happens, you will need more than one tape or disk to restore the file.

After checking the inode numbers, you can turn off verbose mode. To do this, type verbose one more time. For example:

```
restore >verbose
verbose mode off
restore >
```

3.3.2.5 Changing Directories – Change directories in the restore environment with the cd command. To specify a particular directory, enter the cd command with the directory pathname given as an argument to the command. For example:

```
restore >cd /lib
restore >
```

3.3.3 Specifying What You Want Restored From the Dump Image

When working interactively with the restore program, you must first decide which files or file systems you want restored from the dump media. Use the ls command (as explained earlier) to get a file or directory listing of the dump media. Once you know what is on the media, you can specify what you want restored.

You specify what you want restored by creating (and modifing) an extraction list. Creating an extraction list involves working from the dump image and using the restore program's add command. Modifying an extraction list involves working from the dump image and using the restore program's add and delete commands.

3.3.3.1 Creating and Modifying the Extraction List – To create and modify an extraction list, you must be within the restore environment. Once you have invoked the restore program (using the guidelines given previously), follow these steps:

1. Use the pwd command to determine your position within the dump media directory structure.

2. Use the ls command to identify which files and directories are on the dump image at the current level.

3. Decide which files or directories you want to extract from the dump image.

4. Use the cd command to move around the dump image and confirm the location of files and directories.

5. Position yourself in the dump media directory structure at a point that is one level above the desired item.

   For example, suppose you want to restore /usr/lib. To do this, first position yourself at /usr. If you attempt to identify /usr/lib while positioned within /usr/lib, the program complains and cannot find it.

6. Create the extraction list by typing the add command and the name of each desired file or directory as an argument to the command.

   For instance, suppose you are in the /usr directory and you want to extract the files lpf and lpd and the subdirectory refer. To do this, enter the add command followed by the file and directory names. For example:

```
restore >add lpd lpf refer
restore >
```

Note

When you add a file to the extraction list, the program marks that file as one to restore. When you add a directory to your extraction list, the program marks that directory, its files, and all subordinate directories and files as ones to restore.

7. Use the ls command to see how the program has marked these files. For example:

```
restore >ls
diffh              libc_p.a           *lpd
dnet/              libcg.a            *lpf
how_pix            libpc_p.a          *refer/
restore >
```

Notice the asterisks (*) in front of the items that you added to your extraction list. When you view the directory listing, the asterisks point out what you put on the list. Similarly, when the restore program reads an asterisk during the extraction process, it transfers a copy of that file or directory from the dump image to your current working directory.

8. Use the delete command (with the filename given as an argument) if you change your mind and want to remove a particular file from the list. For example:

```
restore >delete lpf
restore >
```

9. Use the ls command to see what the program did in response to the delete command. For example:

```
restore >ls
diffh              libc_p.a           *lpd
dnet/              libcg.a             lpf
how_pix            libpc_p.a          *refer/
restore >
```

Notice that the lpf file has no asterisk now. This change indicates that the program took lpf from the list and will not extract it from the dump image.

10. Once you have created or modified the extraction list, you request a transfer of the files to your system. To do this, use the extract command.

3.3.4 Restoring the Specified Files and Directories

The restore program's extract command initiates the actual restore. When you enter the extract command, the program transfers a copy of each marked item to the current working directory using the relative pathname found on the dump media.

Note

Remember, the current working directory is the one from which you invoked the restore program initially. Your position within the dump media directory structure when you enter the extract command has no influence on the final location of the restored files.

For example, assume that a file (/usr/lib/lpf, for example) was dumped to tape with /usr as the mount point of the backup. In this case, ./lib/lpf is the relative pathname on the dump media. If you invoked the restore program from /usr (in other words, with /usr as your current working directory), the program restores /lib/lpf as subordinate to /usr (/usr/lib/lpf). On the other hand, if you invoked the restore program from /lib (in other words, with /lib as your current working directory), the program restores /lib/lpf as subordinate to /lib (/lib/lib/lpf).

3.3.5 Entering the extract Command

To extract the files that you placed on your extraction list, enter the extract command at the restore > prompt. For example:

```
restore >extract
```

In response, the restore program displays a few statements and prompts you for some answers. For example:

```
You have not read any tapes yet.
Unless you know which volume your file(s) are on you
should start with the last volume and work forward
towards the first.
Specify next volume #:
set owner/mode for '.'? [yn]
```

3.3.5.1 Responding to the 'Specify next volume #' Prompt – The first prompt, Specify next volume #:, assumes that you are working from a multivolume backup set.

- If you are working from a multivolume set, and you know that the files you want extracted start on a certain volume number, then enter that volume number. If you don't know which volume contains

the files that you want extracted, enter the number of the last volume of the set and work your way towards the first. For example, for a three volume backup set, enter 3 at the initial prompt. The program returns a second prompt to which you respond by entering 2, and so on.

- If you are working from a single volume backup tape, enter 1 in response to the initial prompt.

In response to your entry, the restore program searches the dump image for the items listed on the extraction list. When it finds the individual files and directories, it restores them.

3.3.5.2 Responding to the 'Set owner/mode for '.'' Prompt – The second prompt, set owner/mode for '.', lets the superuser change the attributes of the current directory (shown as '.' within the prompt) to the attributes found for that entry on the dump image. The current directory's attributes include access permissions, ownership, and the accessed and update times.

- If you want to change the attributes, enter a y.

  The restore program responds by changing the current directory's attributes to those found on the dump image for the corresponding directory. It then redisplays the restore > prompt.

  If you are not the superuser and you enter a y in response to the prompt, the program responds with an error message, and does not change the attributes.

- If you do not want to change the attributes or if you do not have superuser privileges, enter an n.

  The restore program leaves the attributes of the current directory unchanged. It then redisplays the restore > prompt.

3.3.6 Exiting the Interactive Restore

When you finish restoring files, type quit to get back to the system prompt:

```
restore >quit
#
```

3.4 Restoring Files Noninteractively

Using a noninteractive restore method is useful when you want to restore an entire file system or the entire dump image.

The following sections describe how to prepare for a restoration, and how to invoke the noninteractive restore program.

3.4.1 Preparing For a Noninteractive Restore

Before starting the actual restoration, you should prepare in three ways:

- Log in as yourself or superuser to root
- Mount the dump media
- Change your working directory to the dump media mount point.

The following sections describe each of these preparatory tasks.

3.4.1.1 Logging In – Some restore functions require superuser privileges. For example, if you intend to change directory attributes when restoring, you must be superuser. See restore(8) in the ULTRIX Reference Pages for details on the restore command options and required permissions.

3.4.1.2 Mounting the Dump Media – Before invoking the restore program, ensure that the backup media containing the dump image is mounted on the appropriate device. This is usually the default tape device /dev/rmt0h, but it could be any device or file system in the form of an on-line backup file. In a multi-volume dump set, you must mount volume 1 first since it contains information on the directory structure and inodes for all files on the dump image. Once you have established the restore environment, you can mount other volumes in the set.

3.4.1.3 Changing Your Working Directory – Before invoking restore, change your current working directory to the mount point of the file system that was backed up. For example, if the /usr file system was backed up from root, first position yourself at root, then enter the command to invoke restore from this location.

Note

If you are unsure of the mount point, enter the /etc/restore command with the -t option for a complete listing of the dump media contents.

Be aware that once you invoke the restore program you enter the restore environment. While in this environment, you get

directory information about the dump media only, not your working environment.

Remember, the program restores files to the current working directory (the directory from which you invoked the restore program) using the relative pathname contained on the dump media. If you are in the wrong directory, the program could overwrite the files in that area if they happen to have the same filename.

3.4.2 Invoking the Noninteractive restore Program

To restore files noninteractively, enter the /etc/restore command with the options and arguments that specify what you want restored. See restore(8) in the ULTRIX Reference Pages for a description of the command, its options, restrictions, and diagnostics.

The following examples demonstrate various command line entries:

- If the dump image is located on the default device, and you want to restore the entire image, use the format:

  ```
  #/etc/restore -x
  ```

  The -x option specifies that the root directory and all subordinate items are to be extracted.

- If the dump image is on a device other than the default, and you want to restore the entire image, use the format:

  ```
  #/etc/restore -xf device
  ```

  The -x option specifies that the entire image is to be restored. The -f option and *device* argument let you specify the name of the archive. For example, if the dump media resides on a device named /dev/rmt1h, and you want all items restored, enter:

  ```
  #/etc/restore -xf /dev/rmt1h
  ```

- If the dump image is located on the default device, and you want to restore specific files from the image, use the format:

  ```
  #/etc/restore -v pathname1 pathname2
  ```

  Here, the -v option specifies that you want the program to display the name and type of each file it extracts. The pathname arguments specify that the named files are to be extracted from the dump media. If these files match a directory on the dump media, the directory and subordinate files and directories are also extracted. For example, to restore (in verbose mode) the files /usr/lib/a and /usr/bin/b which reside on the default device, enter:

```
#/etc/restore -v /usr/lib/a /usr/bin/b
```

3.4.3 Exiting the Noninteractive Restore

At the end of a noninteractive restore, the restore program exits automatically and the system redisplays its prompt.

3.5 Restoring a Complete File System

To restore a complete file system, you must first unmount it. For information on unmounting a file system, see mount(8) in the ULTRIX Reference Pages. When the file system is unmounted, perform these steps:

1. Create the target file system. To create the file system, use the newfs command on an unmounted (raw) device.

   Note

   When recreating a file system, the newfs command destroys all existing data on the partition. Therefore, you should back up important data before issuing newfs. To check the partitions, use the chpt command with the -q option.

   For example, to use the newfs command to create a new file system on partition g of an RD53 disk on drive 0, enter:

   ```
   # /etc/newfs /dev/rra0g rd53
   ```

   For more information on the newfs command, see newfs(8) in the ULTRIX Reference Pages.

2. Check and mount the target file system. After creating a target file system, you must check it using the fsck command and then mount it using the mount command. The following example shows how to check the file system and mount the device /dev/ra0g onto the target directory /usr:

   ```
   # fsck /dev/rra0g
   # /etc/mount /dev/ra0g /usr
   ```

   When you invoke the fsck command, it will display certain information on the file system. Refer to fsck(8) in the ULTRIX Reference Pages for further details.

3. Change to the directory on the target file system. Once the target file system is available for use, use the cd command to position yourself at the top of the directory, in this case, /usr. For example:

   ```
   # cd /usr
   ```

4. Restore the dumped file system from the backup media. Having sufficiently prepared the target file system, restore the file system from its most recent and most complete dump tapes or disks. Use the /etc/restore command with the r option. For example:

```
# /etc/restore r
```

This command restores all of the files and directories in the file system from a dump tape mounted on the default tape device, /dev/rmt0h, to the current directory, /usr.

5. Remove the restore table by removing the file named restoresymtable that restore created in the current directory. Then reposition yourself at the root (/) directory. For example:

```
# rm restoresymtable
# cd /
```

6. Unmount the restored file system by using the umount command. For example:

```
# /etc/umount /dev/ra0g
```

This command unmounts the file system /usr which was mounted in step 2, and prepares the file system for the next step. Refer to mount(8) in the ULTRIX Reference Pages for more information on the umount command.

7. Check the file system. After unmounting the restored file system, use the fsck command to check for inconsistencies. For example:

```
# /etc/fsck  /dev/rra0g
```

The fsck command checks the named file system, notifies you of all inconsistencies, occasionally prompts you for a response to its suggested course of action, and proceeds accordingly. When unsure of the consequences of your response, you should answer no. By answering no, you leave the condition uncorrected, but create a summary from which you can decide on a plan of action. You can also use the fsck command with either the -p or -P options. Refer to fsck(8) in the ULTRIX Reference Pages for information on these

as well as other fsck command options.

Note

If you are using fsck on the root partition (/dev/rra0a) and fsck finds and corrects errors, you should halt the system without syncing the disks. To do this, press the HALT button on the MicroVAX or type CTRL/P on the VAX system console when the system returns the prompt. This action returns you to the console subsystem and allows you to reboot the system. It also ensures that the system buffer cache does not overwrite the fsck corrections. You must then reboot your system to get the corrected root file system into memory.

For further details, see the information on maintaining file system consistency in the Guide to System Crash Recovery.

3.6 Local Restoration of the Root and /usr File Systems

This section describes a procedure for restoring the root (/) and /usr file systems of your system disk.

The procedure in this section requires you to have access to the most recent dump image of your root (/) and /usr file systems. The dump information should be available either in the form of a dump tape, or as a sequence of dump diskettes.

You should use this procedure only when a catastrophic error occurs on the system disk, such as a disk crash, or when the inadvertent deletion of either the root (/) or /usr file systems renders the system inoperative.

The following example assumes that you are restoring your root (/) and /usr file systems using a MicroVAX II and an RD53 disk, drive 0, as the system disk. The names of the disk device files in this example are /dev/ra0a for the root file system and /dev/ra0g for the /usr file system.

Disk device file names are processor specific. In particular, the disk device file name for a VAXstation 2000 processor's system disk is /dev/rd0a, instead of /dev/ra0a, which is the device name of the MicroVAX II. Appendix A lists the disk device special file name mnemonics and provides information on how to find their associated special file names.

The following steps explain how to restore the root (/) and /usr file systems and how to reboot the system. These steps are based on the following assumptions:

- The system is a MicroVAX II with an RD53 disk as drive 0

- The files are being restored from a level 0 TK50 dump tape.

1. Boot the TK50 device where the stand-alone system volume (labeled ULTRIX-32_v3.0_STANDALONE) is mounted. The boot command is described in the Basic Installation Guide for your specific processor. Refer to the Basic Installation Guide for more information on how to boot the processor from the distribution media.

   In the following example, using a TK50 tape cartridge for the software distribution kit, the boot command is:

   ```
   >>> b mua0
   ```

   The system next prompts you to enter 1 for the Basic Installation, 2 for the Advanced Installation, or 3 for System Management.

2. Enter 3 for System Management.

3. Run MAKEDEV in the /dev directory to make the system disk special file and the tape device special file. The command line format for making the special files is:

   cd /dev MAKEDEV *mnemonic*

   The *mnemonic* argument refers to a device mnemonic. See Appendix A for a listing of the supported device mnemonics. The following example shows how to make the device special files needed for the MicroVAX II:

   ```
   # cd /dev
   # MAKEDEV ra0 tms0
   ```

   These commands create the necessary special files in /dev for the system disk, RD53 drive 0, and the TK50 tape drive.

4. Change to the root (/) directory.

   ```
   # cd /
   ```

5. Make a new root file system using the mkfs command. This command requires you to use the following format:

   mkfs *dev name* 15884 *sectors tracks*

   The *dev name* variable specifies the raw device special file. The number 15884 specifies the size of the root file system in 512-byte blocks. This is a constant used by all disks. The *sectors* and *tracks* variables specify the disk-specific number of sectors and number of tracks for the target disk. The following table lists the supported disk drives, the *sectors*, and the *tracks* entries.

Disk	Sectors	Tracks
ra60	42	4
ra70	42	11
ra80	31	14
ra81	51	14
ra82	57	15
ra90	70	13
rd53	18	8
rd54	17	15
rm05	32	19
rp07	50	32

As described in mkfs(8) of the ULTRIX Reference Pages, additional entries can be made for the block size and the fragment size. However, the accepted approach is to let mkfs use its defaults of 8192 and 1024 respectively.

The following example shows you the correct mkfs entry for making the root file system on partition a of an RD53 disk. Note that in this example, the raw device is /dev/rra0a. Had this disk drive been connected to a Vaxstation 2000, the device entry would have been /dev/rrd0a.

```
# /etc/mkfs /dev/rra0a 15884 18 8
```

Refer to the table for the disk-dependent number of sectors and number of tracks for your target disk.

6. Reload the boot block. This step is necessary only if the disk is physically damaged or replaced for some other reason. The boot block should normally be operational.

   To reload the boot block:

   ```
   # dd if=/vaxboot of=/dev/rra0a conv=sync
   ```

7. Check the new file system for consistency, otherwise the system will prevent you from mounting it later.

   ```
   # /etc/fsck /dev/rra0a
   ```

8. Mount the root file system:

```
# /etc/mount /dev/ra0a /mnt
```

9. Restore the root file system. To restore the root file system, you must first change to the /mnt directory and enter a command line in the format:

   ```
   restore -rf device
   ```

   The *device* argument is the tape device file name where the dump tape is mounted.

   Mount the dump tape of your root file system onto the tape drive, and then restore the root dump image to your system. For example:

   ```
   # cd /mnt
   # restore -rf /dev/rmt0h
   ```

   The device name /dev/rmt0h is the device file name of the tape drive. The /dev/rmt0h device is the default device and is used only to illustrate use of the restore command. If you are using the default tape device, you do not have to specify the f option or the device name.

10. Unmount the file system and check it for consistency:

    ```
    # cd /
    # /etc/umount /dev/ra0a
    # /etc/fsck /dev/rra0a
    ```

11. Reboot the system by using the following procedure:

    Halt the system:

    ```
    # sync
    # sync
    # halt
    ```

    Boot the newly restored disk (drive 0):

    ```
    >>> b dua0
    ```

12. To restore the /usr file system, follow the steps listed previously in the section called Restoring a Complete File System.

3.7 Using the tar or mdtar Commands To Restore Files

If you backed up your files and directories with the tar and mdtar commands, you must use these commands to do your restore. You do not need superuser privileges, nor do you have to be in single-user mode to use these commands.

The following procedure assumes that a relative pathname was used when the original tar image was created. To restore individual files:

1. Change the directory. Use the cd command to situate yourself in the home directory of the file or files that you want to extract. For instance, if you want to extract files in the /usr directory, type:

```
# cd /usr
```

When you are in the /usr directory, you can list the files stored on the archive media by typing:

```
# tar t
```

If you want to display the file pathnames on a specific device, type:

```
# tar tf /dev/rmt1h
```

The tar command displays the pathnames of the files in the tar image.

2. Extract the files. To extract a single file or a series of named files from tape, type:

```
# tar xp file1 file2 file3...
```

The tar command extracts the listed files using the path name that was placed on the tar archive media. The p option restores the named files to their original protection codes. The p option only takes effect if you are logged in as the superuser.

If you specify the tar command with the xp options and do not specify file names, you will extract all of the files on the archive media by default. For example:

```
# tar xp
```

There are many other options that you can use with the tar and mdtar commands. For more information, refer to tar(1) and to mdtar(1) in the ULTRIX Reference Pages.

3.8 The Labeled Tape Facility

In addition to the restore methods already described, the system contains a labeled tape facility, ltf. The ltf command reads and writes single-volume, versions three and four, ANSI-compatible tape volumes. This provides you with a way of accurately transferring information between ULTRIX and non-ULTRIX systems. For information on this command, refer to ltf(1) and ltf(5) in the ULTRIX Reference Pages.

Remote Backup 4

This chapter introduces conceptual and setup information needed to perform a network backup and explains how to back up file systems over the network. The chapter also explains how to use the remote opser utility.

The chapter contains the following topics:

- The network environment
- Performing remote backups

4.1 The Network Environment

This section introduces concepts that you need in order to perform a remote backup in a master/slave environment and identifies hardware and system setup considerations. This section also explains the concept of performing a network backup either directly to a tape device or through the use of a staging area.

4.1.1 The Master/Slave Relationship

A master system is the local system that you use to initiate a remote backup. You initiate a remote backup by running the remote opser utility as described in 4.2. You must be logged into the master system to perform a remote backup. The master system should have a tape drive and a disk drive that you can use to transfer data from the remote system.

The slave system is the system from which the master system takes the backup data. In some cases, a slave system may only have fixed disks and no tape drive device. Therefore, the only way to back up the slave system's files is from another system. It can also be more efficient and economical to centralize the backup process regardless of the slave systems' hardware.

Figure 4–1 shows a simple network environment that has four slave systems connected to a master system. The four slave systems are all VAXstation 2000 processors. The master system is a VAX 8600 processor that has an RA81 disk drive with a file system named /staging designated as a data transfer storage area. The RA81 disk drive is used for staging

area backups of the slave systems. The master system also has a TU81E streaming tape drive with the default device name /dev/rmt0h. The TU81E tape drive is used to perform network backups directly to tape and to transfer the staging area files to tape. Staging area and direct backups are discussed in Section 4.1.3.

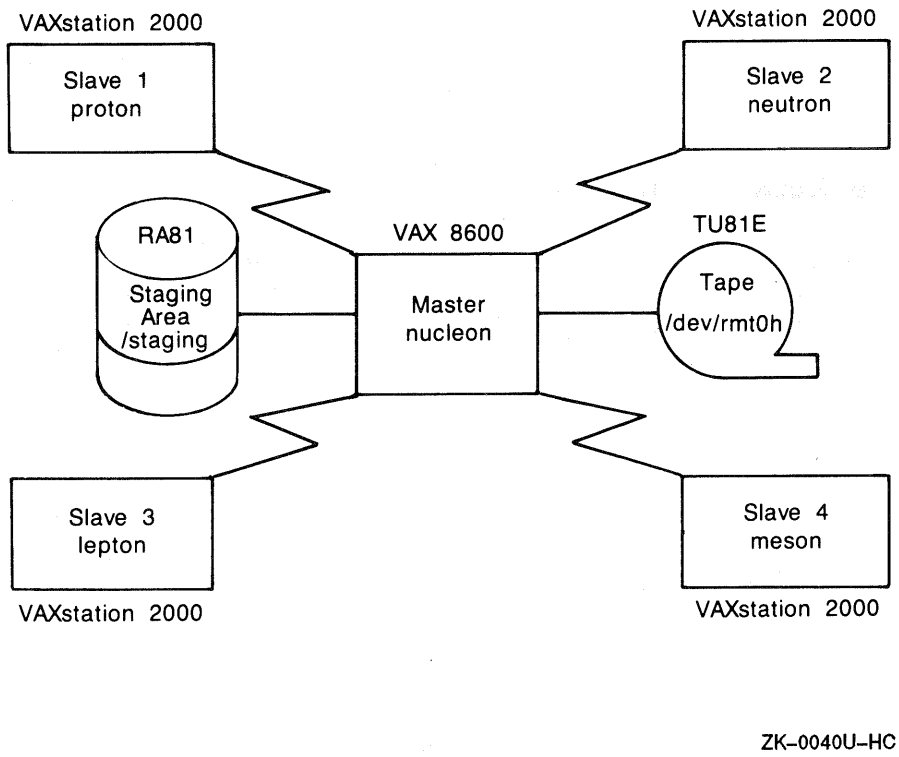

Figure 4-1: Simple Network Environment

4.1.2 Hardware and System Setup Considerations

Before performing a network backup, there are several hardware and system setup considerations to take into account. The hardware considerations described in this section focus on the tape and disk drives that you use to perform a network backup. The system setup considerations described in this section focus on the system files that have to be in place to perform a network backup.

4.1.2.1 Hardware Considerations – The primary hardware consideration deals with the type of tape drive connected to the master system. Tape drives can have either a streaming mode capability or a start/stop mode. Although having a tape drive with a streaming mode capability is not a requirement, this type of tape drive will effect a faster backup process when moving a staging file to tape. All Tape Mass Storage Control Protocol (TMSCP) tape drives are classified as streaming tape drives. Refer to tms(4), tu(4), and ts(4) in the ULTRIX Reference Pages for information on specifying these tape devices.

The tape drives that have a streaming mode capability are:

- TU80
- TU81
- TU81E
- TK50
- TK70
- TS05 (When extended characteristics are set)

The other tape drives that you can use are classified as start/stop mode tape drives. The start/stop mode tape drives are:

- TE16
- TS11
- TU77
- TU78
- TU79
- TS05 (When extended characteristics are not set)

If you have a start/stop mode tape drive, use the direct backup method described in section 4.2.5.2.

The other hardware consideration concerns the disk drives in your system configuration. To perform a remote backup using a staging area as discussed in Section 4.1.3, you must ensure that you have a partition on the master system disk that is large enough to receive all of the backup

data. You must know the partition sizes of the disks connected to your system, and how much free space they contain. The free disk space on the master system disk must be 12 percent greater than the amount of slave system backup data. This is because the software that performs the network backup adds an additional 12 percent of overhead to the data being transferred.

4.1.2.2 System Setup Considerations – There are two system files that you must set up on both the master and slave systems before you can perform a network backup. These system files are:

- The /etc/hosts file
- The /.rhosts file in the root directory

The /etc/hosts file on the master system must specify all of the slave systems on which you intend to perform remote backups. As described in the Guide to Networking, each entry in the master system /etc/hosts file must contain the slave system's Internet address, the official host name, and any aliases for that system. For example, in the case of the simple network environment shown in Figure 4–1, the master system nucleon must have a /etc/hosts file similar to the following:

```
128.45.40.1 proton john
128.45.40.2 neutron peter
128.45.40.3 lepton paul
128.45.40.4 meson ruth
128.45.40.5 nucleon mathew
```

The /etc/hosts file on the slave systems must contain an entry for the master system. For example, the /etc/hosts file for the slave system proton must contain an entry for the master system nucleon as follows:

```
128.45.40.5 nucleon mathew
```

Refer to the Guide to Networking and to hosts(5) in the ULTRIX Reference Pages for more information on the format of this file.

The /.rhosts file in the root directory on the master system must list the slave systems on which you intend to perform remote backups. The /.rhosts files on the slave systems must specify the official host name of the master system. For example, in the case of the simple network environment shown in Figure 4–1, the master system, nucleon, must have a /.rhosts file similar to the following:

```
proton root
neutron root
lepton root
meson root
```

At a minimum, the slave system proton must have a /.rhosts file that

specifies the official host name of the master system, in this case, nucleon. For example:

```
nucleon root
```

Note that the /.rhosts file on the master system specifies access to the root directory of all the slave systems, while the /.rhosts file on the slave system specifies access to the root directory of the master system. This enables the master system nucleon to perform superuser activities on the slave systems during the remote backup process. If you try to perform a network backup without this information in the /.rhosts files, the system responds with the message:

```
Permission denied
```

If either of these two files are set up incorrectly, you will be unable to perform a slave system backup using the remote opser capability described in Section 4.2.

4.1.3 Staging Area or Direct Backup Determination

The remote opser utility described in Section 4.2 provides you with two methods of performing a network backup:

- Staging area backup
- Direct backup

A staging area is designated space in a file system on the master system that has enough free disk space to receive the slave system's backup data. The staging area backup method requires you to back up the slave system's data to the master system's disk and then to transfer the data to a tape device. Transferring the backup data to tape from the master system's disk enables you to take advantage of the streaming mode capability of the tape drive.

In the streaming mode, a tape drive continuously writes data to tape and does not stop tape movement until all of the data has been written. To support the streaming mode, data must be transferred at a rate at least equal to the tape's write transfer rate. Yet, because data cannot be transmitted over a network at this rate, it must first be transmitted to a staging area on the master system's disk drive and then copied to the tape drive. Transferring the staging area file to a tape drive in this way takes advantage of this type of drive's streaming mode capability. The disadvantage is that the file restoration process has the added step of having to transfer the dump image from the tape back to the staging area on the disk.

The direct backup method allows you to back up directly to a tape or disk drive. Use the direct backup method if you have a start/stop mode tape

drive. The direct backup method has a simpler file restoration process because you can restore directly from the tape or disk without having to transfer the dump image to a staging area.

4.2 Performing Remote Backups

Remote backups are performed by using the opser utility and selecting the n (network) option from the opser menu.

To run the opser utility, log in to the operator account (operator login name). When you log in to the operator account, the system automatically invokes the opser utility in place of the shell. Once running, the opser utility prints a utility sign-on message, two informational messages, and a command options prompt:

```
ULTRIX-32 Operator Services

Line editing: delete - erase one character, ^U - kill entire line

For help, type h followed by a return

opr>
```

The opr> prompt informs you that the opser utility is running in place of the shell and is ready to accept opser command options.

To use an opser option, type the appropriate option letter or name and press the RETURN key. For example, to display on-line help about all opser options, press the RETURN key without a command option letter, or type h:

```
opr> h

() - may use first letter in place of full name
Valid commands for Local Opser are:

!sh                 - shell escape (execute ULTRIX-32 commands)
                      (Type control d to return from shell)
(u)sers             - show logged in users
(s)hutdown          - stop time-sharing
(d)ismount          - unmount file systems
(f)sck              - file system checks
(r)estart           - restart time-sharing
(h)elp              - print this help message
(b)ackup            - file system backup
halt                - halt processor
(n)etwork [Slave]   - initiate Remote Opser
(q)uit              - exit from opser

opr>
```

To end an opser session and return to a login prompt, type q.

Except for the n (network) option, all of the local opser menu selections are discussed in Chapter 2. Refer to that chapter for information on other menu selections.

The remainder of this section discusses the functions and capabilities of the remote opser network option.

To enable remote opser, enter n at the opr> prompt. You must also know the name of the slave system on which you want the remote opser running. After you enter n, the opser utility prompts you to enter the name of the slave system; however, it only allows you to enter slave system names that are stored in the master system's /etc/hosts file.

The following example shows how opser responds when you enter an invalid slave system name.

```
opr> n
Enter Slave System name: porton
Slave system name not found in /etc/hosts file
```

The following example shows how opser responds when you enter a valid slave system name.

```
opr> n
Enter Slave System name: proton
Network:  slave = proton.
alvin     ttyp0   Dec 15 09:24  (nucleon)
simon     tty01   Dec 15 08:30
teddy     tty02   Dec 15 08:33
Shutdown at 09:27  (in 3 minutes)
proton_opr>
```

This example shows that after you enter a valid slave system name, the opser utility displays the name of the slave system followed by a list of users currently logged in to that system. The opr> prompt becomes proton_opr>, which tells you that you are now running opser on the slave system named proton. You can also enter the slave system name and the n on the opser command line and achieve the same results. For example:

```
opr> n proton
Network:  slave = proton.
alvin     ttyp0   Dec 15 09:24  (nucleon)
simon     tty01   Dec 15 08:30
teddy     tty02   Dec 15 08:33
Shutdown at 09:27  (in 3 minutes)
proton_opr>
```

Regardless of how you invoke the remote opser, it performs a shutdown of the slave system in three minutes; all users on the slave system must be logged out. When the shutdown completes, the slave system will be in single-user mode and connected to the network. The slave system will then be able to receive opser commands across the network from the

master system.

If you press the RETURN key, or enter an h for help at the proton_opr> prompt, the opser utility displays the remote opser menu. For example:

```
proton_opr> h

() - may use first letter in place of full name
Valid commands for Remote Opser are:

!sh          - Slave System shell escape (execute ULTRIX-32 commands)
!sh          - Master System shell escape (execute ULTRIX-32 commands)
               (Type control d to return from shell)
(d)ismount   - unmount file systems
(f)sck       - file system checks
(r)estart    - restart time-sharing
(h)elp       - print this help message
(b)ackup     - file system backup
halt         - halt processor
(q)uit       - exit Remote Opser (do not restart Slave System)

proton_opr>
```

The following sections explain how each of these opser options work. Read these sections before trying to use these options.

4.2.1 Escaping to the Slave Shell

To escape to the slave shell, enter !sh at the slave system opr> prompt. The opser utility prompts you to enter a password. You must enter the root password for the slave system. For example:

```
proton_opr>!sh
Password:
type ^D to return to opser
#
```

After you enter the slave's root password, the system places you in the slave's shell and displays the shell prompt. You can now execute any ULTRIX command, excluding editors like vi. This option is useful for doing manual file system checks with the fsck command, displaying mounted file systems with the df command, or unmounting file systems with the umount command.

To return to the remote opser, type a CTRL/D at the shell prompt.

4.2.2 Escaping to the Master Shell

To escape to the master shell, enter the letters lsh at the slave system opr> prompt. The opser utility prompts you to enter a password. You must enter the root password for the master system. For example:

```
proton_opr>!sh
Password:
type ^D to return to opser
#
```

After you enter the master's root password, the system places you in the master's shell and displays the shell prompt. You can now execute any ULTRIX command. This option is useful for using the talk command to inform users that you intend to shut down the system. It is also useful for ensuring that the staging area that you are going to use is large enough to perform a staging area backup with the df command.

This option is also used for invoking script files before and after performing a staging area backup. As described in Section 4.2.5.1, script files can be used to clear the staging area of unnecessary files or to transfer staging area files to tape.

To return to the remote opser, type a CTRL/D at the shell prompt.

4.2.3 Unmounting the Slave's File Systems

To unmount the file systems on the slave system, enter the d (dismount) option at the slave system opr> prompt. For example:

```
proton_opr> d
```

When you use the d option, the opser utility invokes the umount command to unmount file systems listed in the slave system's /etc/fstab file. The umount command is described in mount(8) of the ULTRIX Reference Pages.

To ensure that you have unmounted all of the file systems, use the !sh option as described in Section 4.2.1 and enter the mount command. Without options, the mount command displays any file systems that are currently mounted. If any file systems are still mounted, unmount them by issuing the appropriate umount command.

The following example shows the sequence of options and commands that you would enter, assuming that the /usr/staff3 file system is still mounted after you enter the d option:

```
proton_opr> d
proton_opr>
 !sh
Password:
type ^D to return to opser
# /etc/mount
/dev/ra0a on / type ufs
/dev/ra1h on /usr/staff3 type ufs
# /etc/umount  /dev/ra1h
# <CTRL/D>
proton_opr>
```

Note

When you unmount the /usr file system, some ULTRIX commands may be unavailable.

The sequence of commands shows the shell escape, !sh, which causes the system to prompt you for the slave system's root password, after which it displays the # shell prompt. Next, the sequence shows the system response to the mount command. Notice that the file system /usr/staff3 is unmounted by unmounting /dev/ra1h. This is the device (ra1) and partition (h) on which the /usr/staff3 file system is mounted. Lastly, the sequence shows the use of a CTRL/D to return to the proton_opr> prompt.

4.2.4 File System Consistency Check on the Slave

Before you back up any file systems, you should check them for consistency. To check file systems for consistency, enter f at the slave system opr> prompt. The opser utility invokes the fsck command at the slave system to check the file systems listed in the slave's /etc/fstab file. If the fsck command finds any file system inconsistencies, it displays a message informing you that you must run fsck manually.

To run fsck manually, escape to the slave's shell and issue the appropriate fsck command.

If fsck fails on the slave's root file system, you must:

1. Quit network opser by typing q at the slave system opr> prompt. As described in Section 4.2.8, the slave system will be shutdown and in single-user mode.
2. Run fsck on the slave system. For example:

```
# /etc/fsck /dev/rra0a
```

3. On the slave system, press the HALT button on the MicroVAX or type CTRL/P on the VAX system console at the system prompt. This action returns you to the console subsystem and allows you to reboot the system. It also ensures that the system buffer cache does not overwrite the fsck corrections.
4. You must then reboot the slave system to get the corrected root file system into memory.
5. Reestablish network opser by typing n at the local opser menu at the master system.

Note

Whenever fsck corrects root file system inconsistencies, either through the f menu option or manual fsck, you must halt and reboot the slave system. Use the HALT button on the MicroVAX processors or type CTRL/P at the system console on the other VAX processors.

For further details, see the information on maintaining file system consistency in the Guide to System Crash Recovery.

Example 4–1 shows you a sequence of commands with a file system inconsistency encountered on the /dev/ra0h file system.

Example 4-1: Performing a File System Consistency Check

```
proton_opr> d

proton_opr> f

/dev/ra0a: 489 files, 6050 used, 631 free (93 frags, 160 blocks)
/dev/ra0d: 306 files, 43542 used, 51325 free (122 frags, 327 blocks)

/dev/ra0h: UNEXPECTED INCONSISTENCY; RUN fsck MANUALLY.
/dev/ra0f: 155 files, 41487 used, 147657 free (98 frags, 126 blocks)

proton_opr> !sh

Password:
type ^D to return to opser

# fsck /dev/rra0h
** Last mounted on /usr/users
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
/dev/ra0h: 3427 files, 95983 used, 14774 free (184 frags, 3362 blocks)
# /etc/mount -a
# df
Filesystem    total    kbytes   kbytes  percent
   node       kbytes    used     free    used     Mounted on
/dev/ra0a       7423     6050      631     91%     /
/dev/ra0d     105407    43542    51325     46%     /usr
/dev/ra0h     123063    95983    14774     87%     /usr/users
/dev/ra0f     210159    41487   147657     22%     /usr/server
# /etc/umount -a
# <CTRL/D>
proton_opr>
```

Example 4–1 shows that after running the fsck command, the file system inconsistencies were resolved. Note that the fsck command was executed on the raw device, /dev/rra0h. The example also shows that after the file system inconsistency was resolved, the mount command was invoked with the –a option to mount all the file systems in the /etc/fstab file so that the df command could be invoked. The df command was issued to display and confirm the contents of the file systems. The umount command with a –a option was issued to dismount the file systems so that a network backup of the file systems could be performed without having to use the remote opser dismount option.

Refer to fsck(8), mount(8), and df(8) in the ULTRIX Reference Pages for more information.

4.2.5 Backing Up the Slave's File Systems

There are two methods that you can use to perform a remote backup of file systems. One is the staging area backup method and the other is the direct backup method. You will usually use the staging area backup method if you have a streaming mode tape drive. Use the direct backup method if you have a start/stop mode tape drive.

Note

You must unmount and check all file systems prior to backing them up. Use the opser d option to unmount the file systems; use the opser f option to check them for consistency.

The following two sections explain how to use the two backup methods. The sections contain complete examples of how the opser utility responds when using either method.

4.2.5.1 Staging Area Backup Method – The staging area backup method requires you to back up the slave system data to a designated file system on the master system's disk, and then to perform a tape transfer of the backed up data.

Before performing a staging area backup, you must know:

- The backup level, which is a number from 0 to 9. Use the number 9 for a daily backup, the number 5 for a weekly backup, and the number 0 for a monthly or full backup.
- The file system that you will be using for the staging area on the master system, for example, /staging.
- The name of the file system that you want to dump to the staging area, for example, /.
- The amount of slave system data that you want to dump to the master system's staging area and the amount of free disk space in the staging area. The amount of free disk space must be at least 12 percent greater than the amount of slave system data that you want to dump.

When the backup is complete, you must use the tar command to transfer the staging area file to tape, or the mdtar command to transfer the staging area file to another disk. In either case, you should remove the staging area file from the staging area once you have transferred it to a removable media.

To use either tar or mdtar without exiting opser, you must escape to the master's shell using the lsh option as described in Section 4.2.2. If an operator is performing the staging area backup and does not have access to the superuser password, you must create some script files to perform the tape or disk transfer and to remove the staging area file from the staging area. The script files must be located in the /opr directory and have root ownership.

For example, to perform a tape transfer of the staging area file, a script file named saveit would have a one-line entry of:

```
tar -cv /staging/*
```

Similarly, to remove the staging area file from the /staging directory, a script file named removeit would have a one-line entry of:

```
rm -rf /staging/*
```

Having script files available to operators who do not have access to the superuser password keeps them from having to exit the opser utility to perform the tape or disk transfers or to remove files from the staging area.

Example 4–2 shows a backup sequence of the root (/) file system from the slave system proton to the master system named nucleon. The sequence shows that before performing the backup, the removit script file was invoked to clear the /staging directory of any files. The example also shows that the root file system was transferred to the master's /staging directory. Once the name of the file system is entered, the opser utility determines if there is enough disk space in the specified staging area to back up the slave's file system. If there is enough room, the backup process begins. If there is not enough room, the system sends a beep signal to your terminal and displays the message:

```
Couldn't allocate ##### bytes.
proton_opr>
```

The ##### signs represent the number of bytes required to perform the backup.

When the opser utility has determined that there is enough disk space in the staging area, it starts the dump sequence. The opser utility assigns a file name to the data being transferred to the staging area. The naming convention that it uses for the file created on the master system is:

slave system name.filesystem name.time stamp

The opser utility also changes any slashes (/) in the file system name to tildes (~). In Example 4–2, the file name is:

proton.~WED-Sep-24-10:32:26-1986

The dump image file is located in the master system nucleon's /staging directory. In the example, the staging area file name breaks down as the root file system of the slave system proton, which was backed up on Wednesday, September 24, 1986 at 10:32 am. The tilde sign (~) equals the root (/) file system.

Example 4-2: Staging Area Backup Method Display

```
proton_opr> lsh removeit
proton_opr> b
Enter backup level (0 - 9): 9
Use staging file or go directly to device(s = staging/d = direct)? s
Enter directory to place staging file: /staging
Enter name of filesystem to dump: /

Sizing file system to determine number of bytes to pre-allocate.

DUMP: Estimated 68608 bytes output to file named
          nucleon:/staging/proton.~.Wed-Sep-24-10:32:26-1986

Allocating 68608 bytes.

DUMP: Date of this level 9 dump: Wed Sep 24 10:32:23 1986
DUMP: Date of last level 0 dump: Fri Sep 12 09:19:58 1986
DUMP: Dumping /dev/rra0a (/) to
      /staging/proton.~.Wed-Sep-24-10:32:26-1986 on host nucleon
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]

DUMP: Estimated 68608 bytes output to file named
                /staging/proton.~.Wed-Sep-24-10:32:26-1986

DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]

DUMP: 58368 bytes were dumped to file
                /staging/proton.~.Wed-Sep-24-10:32:26-1986

DUMP: level 9 dump on Wed Sep 24 10:32:23 1986
DUMP: Dump is done

proton_opr>lsh saveit
```

Example 4–2 shows that once the dump is complete, the file named proton.~Wed-Sep-24-10:32:26-1986 is transferred to tape by using the script file saveit.

4.2.5.2 Direct Backup Method – The direct backup method enables you to back up a slave's file systems directly to a master system's tape drive or disk drive if your system configuration has the capacity to designate a disk drive for the purpose of backing up the slave systems.

Whether you are backing up to a tape or a disk drive, you must know the following information before you can use the direct backup method:

- The backup level number, which is a number from 0 to 9. Use the number 9 for a daily backup, the number 5 for a weekly backup, and the number 0 for a monthly or full backup.
- The device name where you want the slave system's backup data to go, for example, /dev/rmt0h.
- The name of the slave's file system that you want to dump.

You must also ensure that you have enough of the chosen media (tapes, disks, or diskettes) on hand when the backup process begins. Depending on the size of the backup, you may have multiple backup volumes.

Example 4–3 shows a backup sequence of the root (/) file system from the slave system named proton to the master named nucleon. This example shows that the device /dev/ra2a (an RX50 device, unit number 2), is being used as the direct backup media. When the name of the file system is entered, the opser utility sizes the slave's file system and informs the operator that, for this backup, 17 diskettes are required to transfer all of the data. Example 4–3 shows that when the data has been dumped, the opser utility redisplays the proton_opr> prompt. This prompt indicates that the backup has been successfully completed.

Example 4-3: Direct Backup Method Display

```
proton_opr> b
Enter backup level (0 - 9): 9
Use staging file or go directly to device(s = staging/d = direct)? d
Enter device name: /dev/ra2a
Enter name of filesystem to dump: /

DUMP: Place Remote  RX50 device unit #2 ONLINE
DUMP: NEEDS ATTENTION: Cannot open disk. Do you want to retry the open?
                                          ("yes" or "no") yes
DUMP: Date of this level 0 dump: Wed Sep 24 10:07:32 1986
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rra0a (/) to /dev/ra2a on host nucleon
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]

DUMP: Estimated 654 (10240 byte) blocks on 17.00 disk(s).

DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: Change Disks: Mount disk #2
DUMP: NEEDS ATTENTION: Next disk ready?: ("yes" or "no") yes
DUMP: Disk 2 begins with blocks from ino 81
DUMP: Change Disks: Mount disk #3
DUMP: NEEDS ATTENTION: Next disk ready?: ("yes" or "no") yes
DUMP: Disk 3 begins with blocks from ino 240
.
.
.
DUMP: NEEDS ATTENTION: Next disk ready?: ("yes" or "no") yes
DUMP: Disk 17 begins with blocks from ino 18745
DUMP: Dump is done
proton_opr>
```

When the backup is complete, ensure that the dump media is properly labeled and stored in a safe, secure place. You should record the disk or tape volume number, the backed up file system name, and the starting inode numbers on each tape or disk. You can now back up other file systems or return to the local opser menu.

4.2.6 Returning to the Local opser

To return to the local opser main menu, enter r at the slave system opr> prompt. For example:

```
proton_opr> r
```

This causes the remote opser running on the slave system to restart multiuser mode, disable the network connection to the slave system, and return you to the local opser that you initiated at the master system.

Note

Do not use the q command to end a remote opser session. The q command stops remote execution of the opser utility and should only be used in the event of an emergency by someone with superuser privileges. Issuing the r command is the only way to end a remote opser session properly.

4.2.7 Halting the Slave Processor

To halt the slave system processor, enter halt at the slave system opr> prompt. This command causes the remote opser to synchronize the slave system disks (which means that in-memory file system data is written to the disks), then halt the slave system processor. When you enter halt, the remote opser checks to make sure that you want to halt the slave system processor:

```
proton_opr>halt

This will leave the Slave System halted. You must then manually reboot
the Slave System.
Do you really want to do this ?(Enter y/n)y

Leaving network mode.

opr>
```

When you enter a y, the remote opser utility exits remote opser execution and leaves the slave system halted. Consequently, someone at the slave system will have to reboot it manually. The master system then returns to local execution of the opser utility as indicated by the opr> prompt, and the network connection that was established automatically disconnects. If you enter an n, the remote opser utility returns the slave system opr> prompt.

4.2.8 Stopping Remote Execution of opser

In an emergency, it may be necessary to stop running the opser utility at the slave system. To stop remote execution of opser, enter q at the slave system opr> prompt. For example:

```
proton_opr> q
This will leave the Slave System shutdown in single user mode.
Do you really want to do this?(Enter y/n)y
Leaving network mode.
opr>
```

The precedinig example shows that when you enter q, the remote opser checks to make sure that you want to quit the remote opser session. When you enter a y, the remote opser utility exits remote opser execution and leaves the slave system in single-user mode. Consequently, someone at the slave system will have to restart multiuser mode. The master system then returns to local execution of the opser utility as indicated by the opr> prompt, and the network connection that was established automatically disconnects. If you enter an n, the remote opser utility returns the slave system opr> prompt.

4.2.9 Exceptional Condition Handling

You can abort either a remote opser process (such as rdump) on the slave, or a local opser process (such as dump) on the master. To abort any processes running under the local opser on the master system, type a ^CTRL/C. Typing a ^CTRL/C causes the opser utility to return an opr> prompt. Type a CTRL/C to abort any process running under the remote opser on the slave system.

Remote File Restoration 5

You can restore files, directories, and file systems over the network with the rrestore program. This chapter describes how to:

- Restore files, directories, or file systems on different media using the rrestore program
- Restore the root and /usr file systems after a catastrophic event has occurred

5.1 Using the rrestore Program

You use the rrestore program to perform a remote file restoration either interactively or noninteractively. Subsequent sections of this chapter explain both methods with examples of how to perform each.

5.1.1 Network Considerations

Although the rrestore program is similar in function to the restore program, certain rules apply when working across a network:

1. You must make a distinction between the master and slave systems as described in Chapter 4. Typically, the dump media is located on the host or master system. You invoke the rrestore program from the slave system, and specify the host name where the dump media resides.

2. You do not have to be the superuser to restore files remotely. However, to restore a complete file system properly, you must be logged in as the superuser.

3. It is important to remember where the file system that you want to restore was originally backed up.

- If the file system was backed up to a staging area file and is still intact, then you must specify the staging area file name to the rrestore program.

- If the staging area file was moved to tape using the tar command, then you must first use tar to move the staging area file from the tape to the staging area and then remote restore the file system from the staging area file. See Chapter 4 for a description of the staging area.

4. Whatever method that you used to back up a file system must be repeated in reverse order.

5. You can only specify a device to the rrestore command when the files that you want to restore were originally backed up using the direct backup method. See Chapter 4 for a description of backup methods.

5.2 Preparing for an Interactive Restore

Before starting the remote restore, you may need to:

- Log in as yourself or as superuser to root
- Mount the first dump tape on the appropriate device
- Transfer the backed up file from tape to a staging area using the tar command.

The following sections describe each of these tasks.

5.2.1 Logging In

Some rrestore functions require superuser privileges. For example, if you intend to perform an interactive file restoration of files that you do not own, you must be superuser. Similarly, if you intend to change file and directory attributes when restoring, you must be superuser. See rrestore(8) in the ULTRIX Reference Pages for details on the command options and required permissions.

5.2.2 Mounting the Dump Media

Before invoking the rrestore program, ensure that the backup media containing the first dump tape is mounted on the appropriate device. This is normally the default tape device (/dev/rmt0h), but it could be any device or file system in the form of an on-line backup file. The first tape is critical since it contains directory and inode information about the contents of the dump image.

5.2.3 Transferring Files to the Staging Area

If the remote backup was to a staging area file and then was transferred to tape using the tar command, you must first ensure that the file is transferred back to the staging area from the tape by using the tar command with the -x option. For example, if the file name is proton.~Wed-Sep-24-10:32:26-1986, you would enter:

```
# cd /staging
# tar -x proton.~.Wed-Sep-24-10:32:26-1986
```

The proton.~Wed-Sep-24-10:32:26-1986 file name is the staging area file name assigned by the rdump command as described in Chapter 4. Once the staging area file has been transferred from tape to the staging area on the master system, you can invoke the rrestore command.

5.3 Restoring Files Interactively

Once you are logged in and the dump media is mounted on the appropriate device, use the following command format:

```
# /etc/rrestore -if host:device
```

The -i option specifies interactive mode. The -f option followed by the *host:device* string specifies the host and host device from which to obtain the data. For example, to restore files that were backed up to the master system nucleon using the direct backup method, you would enter:

```
# /etc/rrestore -if nucleon:/dev/rmt0h
```

Now, if you want to restore a file that you transferred back to the staging area from tape with the tar command, you would restore the file by entering a command line similar to:

```
# /etc/rrestore -if nucleon:/staging/proton.~.Wed-Sep-24-10:32:26-1986
```

5.3.1 Using Interactive Mode Commands

When you enter the /etc/rrestore command with the -i option, the rrestore> prompt appears on your screen. The prompt signals that the program is waiting for a command.

5.3.1.1 Listing the Commands – To get a listing of the available commands, enter an h or a ? at the rrestore> prompt. For example:

```
rrestore >h
Available commands are:

ls [arg] - list directory
cd arg - change directory
pwd - print current directory
add [arg] - add `arg' to list of files to be extracted
delete [arg] - delete `arg' from list of files to be extracted
extract - extract requested files
quit - immediately exit program
verbose - toggle verbose flag (useful with "ls")
help or `?' - print this list
If no `arg' is supplied, the current directory is used
rrestore >
```

Note

Remember, after invoking the restore program in interactive mode, you are in the restore environment. The program responds to your subsequent commands with information about the dump media, not about your standard environment.

5.3.1.2 Listing the Pathname of the restore Working Directory – To list the pathname of the working directory within the restore environment, type pwd at the rrestore > prompt. For example:

```
rrestore >pwd
/lib
rrestore >
```

5.3.1.3 Getting a File and Directory Listing – To get a listing of the files and directories that can be restored at the current directory level, type ls at the rrestore > prompt. For example:

```
rrestore >ls
 adm/          hosts/        man           skel/
 bin/          include/      mdec/         spool/
 dict          kits          msgs/         src/
rrestore >
```

To get a listing of the files and directories that can be restored at a different directory level, enter the pathname of that directory as an argument to the ls command.

5.3.1.4 Comparing Inode Numbers – By comparing the displayed inode numbers with the starting inode numbers recorded on a multivolume dump set, you can find the tape or disk that contains the file you want to restore. To see the inode numbers of the files contained within your listing, first use the verbose command to enter verbose mode, then use the ls command to get your listing. The rrestore program displays the files with the inode numbers to the left of the dumped file names. For example:

```
rrestore >verbose
verbose mode on
rrestore >ls
    2  *./        11971  hosts       14253  spool/
    2  *../        2177  include/     4145  src/
11969  adm/       11427  mdec/        4253  staff
 1633  bin/        2722  msgs/        3266  sys/
rrestore >
```

As shown above, the hosts file has an inode number of 11971 and the spool directory has an inode number of 14253.

Note

Sometimes the inodes for a given file span multiple tapes or disks. When this happens, you will need more than one tape or disk to restore the file.

After checking the inode numbers, you can turn off verbose mode. To do this, type verbose one more time. For example:

```
rrestore >verbose
verbose mode off
rrestore >
```

5.3.1.5 Changing Directories – Change directories in the restore environment with the cd command. To specify a particular directory, enter the cd command with the directory pathname given as an argument to the command. For example:

```
rrestore >cd /lib
rrestore >
```

5.3.2 Specifying What You Want Restored From the Dump Image

When working interactively with the rrestore program, you must first decide which files or file systems you want restored from the dump media. Use the ls command (as explained earlier) to get a file or directory listing of the dump media. Once you know what is on the media, you can specify

what you want restored.

You specify what you want restored by creating (and modifing) an extraction list. Creating an extraction list involves working from the dump image and using the rrestore program's add command. Modifying an extraction list involves working from the dump image and using the rrestore program's add and delete commands.

5.3.2.1 Creating and Modifying the Extraction List – To create and modify an extraction list, you must be within the rrestore environment. Once you have invoked the rrestore program (using the guidelines given previously), follow these steps:

1. Use the pwd command to determine your position within the dump media directory structure.

2. Use the ls command to identify which files and directories are on the dump image at the current level.

3. Decide which files or directories you want to extract from the dump image.

4. Use the cd command to move around the dump image and confirm the location of files and directories.

5. Position yourself in the dump media directory structure at a point that is one level above the desired item.

   For example, suppose you want to restore /usr/lib. To do this, first position yourself at /usr. If you attempt to identify /usr/lib while positioned within /usr/lib, the program complains and cannot find it.

6. Create the extraction list by typing the add command and the name of each desired file or directory as an argument to the command.

   For instance, suppose you are in the /usr directory and you want to extract the files lpf and lpd and the subdirectory refer. To do this, enter the add command followed by the file and directory names. For example:

```
rrestore >add lpd lpf refer
rrestore >
```

Note

When you add a file to the extraction list, the program marks that file as one to restore. When you add a directory to your extraction list, the program marks that directory, its files, and all subordinate directories and files as ones to restore.

7. Use the ls command to see how the program has marked these files. For example:

```
rrestore >ls
diffh              libc_p.a           *lpd
dnet/              libcg.a            *lpf
how_pix            libpc_p.a          *refer/
rrestore >
```

Notice the asterisks (*) in front of the items that you added to your extraction list. When you view the directory listing, the asterisks point out what you put on the list. Similarly, when the rrestore program reads an asterisk during the extraction process, it transfers a copy of that file or directory from the dump image to your current working directory.

8. Use the delete command (with the filename given as an argument) if you change your mind and want to remove a particular file from the list. For example:

```
rrestore >delete lpf
rrestore >
```

9. Use the ls command to see what the program did in response to the delete command. For example:

```
rrestore >ls
diffh              libc_p.a           *lpd
dnet/              libcg.a             lpf
how_pix            libpc_p.a          *refer/
rrestore >
```

Notice that the lpf file has no asterisk now. This change indicates that the program took lpf from the list and will not extract it from the dump image.

10. Once you have created or modified the extraction list, you request a transfer of the files to your system. To do this, use the extract command.

5.3.3 Restoring the Specified Files and Directories

The rrestore program's extract command initiates the actual restore. When you enter the extract command, the program transfers a copy of each marked item to the current working directory using the relative pathname found on the dump media.

Note

> Remember, the current working directory is the one from which you invoked the rrestore program initially. Your position within the dump media directory structure when you enter the extract command has no influence on the final location of the restored files.

For example, assume that a file (/usr/lib/lpf, for example) was dumped to tape with /usr as the mount point of the backup. In this case, ./lib/lpf is the relative pathname on the dump media. If you invoked the rrestore program from /usr (in other words, with /usr as your current working directory), the program restores /lib/lpf as subordinate to /usr (/usr/lib/lpf). On the other hand, if you invoked the rrestore program from /lib (in other words, with /lib as your current working directory), the program restores /lib/lpf as subordinate to /lib (/lib/lib/lpf).

5.3.4 Entering the extract Command

To extract the files that you placed on your extraction list, enter the extract command at the rrestore > prompt. For example:

```
rrestore >extract
```

In response, the rrestore program displays a few statements and prompts you for some answers. For example:

```
You have not read any tapes yet.
Unless you know which volume your file(s) are on you
should start with the last volume and work forward
towards the first.
Specify next volume #:
set owner/mode for '.'? [yn]
```

5.3.4.1 Responding to the 'Specify next volume #' Prompt – The first prompt, Specify next volume #:, assumes that you are working from a multivolume backup set.

- If you are working from a multivolume set, and you know that the files you want extracted start on a certain volume number, then enter that volume number. If you don't know which volume contains

the files that you want extracted, enter the number of the last volume of the set and work your way towards the first. For example, for a three volume backup set, enter 3 at the initial prompt. The program returns a second prompt to which you respond by entering 2, and so on.

- If you are working from a single volume backup tape, enter 1 in response to the initial prompt.

In response to your entry, the rrestore program searches the dump image for the items listed on the extraction list. When it finds the individual files and directories, it restores them.

5.3.4.2 Responding to the 'Set owner/mode for '.'' Prompt – The second prompt, set owner/mode for '.', lets the superuser change the attributes of the current directory (shown as '.' within the prompt) to the attributes found for that entry on the dump image. The current directory's attributes include access permissions, ownership, and the accessed and update times.

- If you want to change the attributes, enter a y.

  The rrestore program responds by changing the current directory's attributes to those found on the dump image for the corresponding directory. It then redisplays the rrestore > prompt.

  If you are not the superuser and you enter a y in response to the prompt, the program responds with an error message, and does not change the attributes.

- If you do not want to change the attributes or if you do not have superuser privileges, enter an n.

  The rrestore program leaves the attributes of the current directory unchanged. It then redisplays the rrestore > prompt.

5.3.5 Exiting the Interactive Restore

When you finish restoring files, type quit to get back to the system prompt:

```
rrestore >quit
#
```

5.4 Restoring Files Noninteractively

Noninteractive restoration of data is accomplished with a command line entry. The files or directories to be restored are entered as part of the command line. Before issuing an rrestore command in this way, ensure that you are in the correct directory and that you have the correct media mounted.

The following example shows you how to restore three files from the master system named nucleon. The example shows that the files named in the command line are restored from the dump tape mounted on device /dev/rmt0h on the master system nucleon.

```
#/etc/rrestore -xf nucleon:/dev/rmt0h file1 file2 file3
#
```

When the restoration process completes, the system redisplays the system prompt and the specified files are now in the current working directory. Like the interactive restoration method, you can specify directories as well as files. The rrestore command will restore all of the files and subdirectories beneath the specified directory.

In addition to file names, you can specify file systems. For example:

```
#/etc/rrestore -rf nucleon:/dev/rmt0h
#
```

In this example, the rrestore command restores all files on the dump tape to the current working directory. This shows a file system restore, which is normally performed after a new file system is created, mounted, and a change directory is executed.

5.5 Remote Restoration of the System Disk

This section describes procedures for restoring the root (/) and /usr file systems of your system disk.

These remote restoration procedures can only be used if you are restoring your disk to a MicroVAX II processor, or to either the VAXstation 2000 or MicroVAX 2000 processors. If you are on any other type of processor, these methods will not work, and you must restore your disk using local dump media. Refer to Chapter 3 for information on local restoration procedures of the root and /usr file systems.

The procedures in this section require you to have access to the most recent dump image of your root (/) and /usr file systems. The dump information should be available either in the form of a dump file image on a remote machine, a dump tape mounted on the remote system's tape drive, or a dump disk mounted on the remote system's tape or disk drive.

You should only use these procedures when a catastrophic error, such as a disk crash, occurs on the system disk or when the inadvertent deletion of either the root (/) or /usr file systems renders the system inoperative.

The examples in the steps assume that you are restoring your root (/) and /usr file systems using a MicroVax II, and the disk device files in this example are /dev/ra0a for the root file system, and /dev/ra0g for the /usr file system.

Disk device file names are processor specific, so you should refer to Appendix A for the proper disk names for your disk drive. In particular, the disk device file name for a VAXstation 2000 processor is /dev/rd0a, instead of /dev/ra0a, the name used with the MicroVAX II processor.

The examples in the steps assume:

- The system is a MicroVAX II with an RD53 disk designated as drive 0. Therefore, the device mnemonic name is ra0. If your system is a VAXstation 2000 processor type, then rd0 is the device mnemonic name that you should use.
- Files are being restored from the remote system named proton to the corrupted system named neutron.

The examples in the steps show how to restore the root (/) and /usr file systems remotely, and how to reboot the system.

5.5.1 Restoring the Root File System

To restore the root file system of a MicroVax or MicroVAX II processor remotely, follow these steps:

1. Boot the network device as described in the Basic Installation Guide. The client machine (the one to which you are restoring) must be registered for network install on a server, so it can successfully boot the stand-alone system from the network.

   If your processor is a MicroVAX II, or other MicroVAX processor, use this command to boot the network device:

   ```
   >>> b xqa0
   ```

   If your processor is a VAXstation 2000 or MicroVAX 2000, then use this command to boot the network device:

   ```
   >>> b esa0
   ```

   In either case, the command boots the system and prompts you to enter 1 for Basic Installation, 2 for Advanced Installation, or 3 for System Management.

2. Enter 3 for System Management.

3. Run the gethost command (located in the root file system).

```
# gethost
```

This creates an /etc/hosts file containing both your system's hostname and the hostname of the server where you registered for the network installation.

4. Add the hostname and internet address of the remote machine where the dump tape is mounted, or where the dump disk image resides, to the /etc/hosts file.

If the dump image is on the network server machine from which you previously booted, you can skip this step, since the server hostname is already in the /etc/hosts file.

If you do not know the internet address of the remote dump machine, enter:

```
# rsh lepton -l ris grep proton /etc/hosts
 128.45.40.115 proton pr
#
```

This command line instructs the remote shell to search for proton's /etc/host file on the remote system lepton and to display proton's internet address, 128.45.40.115. The name lepton is the name of the network server machine.

The name proton is the name of the machine that contains the dump information, either on tape or in a file system's file, and whose name you wish to add to the /etc/hosts file.

The following example shows the steps required to add the remote host proton with an internet address of 128.45.40.115 to the /etc/hosts file:

```
# ed /etc/hosts
1,$p
127.0.0.1 localhost
128.45.40.101 neutron
128.45.40.102 quasar
a
128.45.40.115 proton pr
.
w
q #
```

5. Ensure that the remote system's /.rhosts file contains your system's host name. Your system must be trusted as root by the remote system where the dump image resides for the remote restore process

to work.

An easy way to ensure that a remote system contains your hostname is with the command:

```
# rsh proton who
```

The name proton is the name of the system where your dump tape or dump image resides.

If this command is successful, you should continue on to the next step. If you receive a Permission denied message, either you or someone with superuser privileges at the remote system should add your hostname to the /.rhosts file on that system.

6. Run MAKEDEV in the /dev directory to make the ra0 device:

```
# cd /dev
# MAKEDEV ra0
```

This creates the necessary special files in /dev for the system disk, RD53 drive 0. See Appendix A for a list of device mnemonics used by MAKEDEV.

7. Make a new root file system using the mkfs command. This command requires you to use the following format:

mkfs *dev_name* 15884 *sectors tracks*

The *dev_name* variable specifies the raw device special file. The number 15884 specifies the size of the root file system in 512-byte blocks. This is a constant used by all disks. The *sectors* and *tracks* variables specify the disk-specific number of sectors and number of tracks for the target disk. The following table lists the supported disk drives, with the corresponding *sectors*, and *tracks* entries.

Disk	Sectors	Tracks
ra60	42	4
ra70	42	11
ra80	31	14
ra81	51	14
ra82	57	15
ra90	70	13
rd53	18	8
rd54	17	15
rm05	32	19
rp07	50	32

As described in mkfs(8) of the ULTRIX Reference Pages, additional

entries can be made for the block size and the fragment size. However, the accepted approach is to let mkfs use its defaults of 8192 and 1024 respectively.

The following example shows you the correct mkfs entry for making the root file system on partition a of an RD53 disk:

```
# /etc/mkfs /dev/rra0a 15884 18 8
```

Refer to the table for the disk-dependent number of sectors and number of tracks for your target disk.

8. Reload the boot block. This step is necessary only if the disk is physically damaged or must be replaced for some other reason.

   To reload the boot block:

   ```
   # dd if=/vaxboot of=/dev/rra0a conv=sync
   ```

9. Check the new file system for consistency, otherwise the system will prevent you from mounting it later. For example, type:

   ```
   # /etc/fsck /dev/rra0a
   ```

10. Mount the root file system. For example, type:

    ```
    # /etc/mount /dev/ra0a /mnt
    ```

11. Restore the root file system. The way you restore the root file system depends on whether you are restoring from a dump tape (use step 12) or from a file on a file system dump image (use step 13).

    In either case, you must first change to the /mnt directory and enter a command line in the format:

    rsh *rem_sys* dd if=*dev* bs=*n*k | restore rf -

    or

    rsh *rem_sys* dd if=*file* bs=*n*k | restore rf -

    In both cases, *rem_sys* is the name of the remote system where the dump image or dump tape resides. For tape dumps, *dev* is the special file name where the dump tape is mounted. For a dump image file on a file system, *file* is the path file name where the dump image resides. In both cases, *n*k is the variable number that specifies the block size used for the write.

12. Restore the root file system from a dump tape. If you are restoring from a dump tape, load the dump tape of your root file system onto the remote system's tape drive. Then restore the root dump image to your local system:

```
# cd /mnt
# rsh proton dd if=/dev/rmt0h bs=10k | restore rf -
```

The name proton is the name of the remote system where the tape is physically mounted. The device name /dev/rmt0h, is the special file name of the tape drive.

13. Restore the root file system from an on-line dump image. If you are restoring from an on-line dump image contained in a file on a remote system, follow these steps to restore the root file system:

```
# cd /mnt
# rsh proton dd if=/usr/backups/backup.root bs=10k | restore rf -
```

The name proton is the name of the remote system where the dump file image exists. The name /usr/backups/backup.root is the name of the actual dump file. The block size used for the write is specified as 10k.

14. Dismount the file system and check it for consistency:

```
# cd /
# /etc/umount /dev/ra0a
# /etc/fsck /dev/rra0a
```

15. Reboot the system. To reboot the system, use the following procedure:

Halt the system:

```
# sync
# sync
# halt
```

Boot the newly restored disk (drive 0):

```
>>> b dua0
```

5.5.2 Restoring the /usr File System

If you experienced a head crash that destroyed the /usr file system, or for some reason the /usr file system was deleted, you will have to restore it. To restore the /usr file system remotely, perform these steps:

1. Create the target file system. To create the file system, use the newfs command on an unmounted (raw) device.

You can use the newfs command to create a new file system on partition g of an RD53 disk on drive 0, as shown in the following example:

```
# /etc/umount /dev/ra0g
# /etc/newfs /dev/rra0g rd53
```

2. Check and mount the target file system. After creating a target file system, you must check it using the fsck command and then mount it using the mount command. The following example shows how to check the file system and mount the device /dev/rra0g on to the target directory /usr:

```
# fsck /dev/rra0g
# /etc/mount /dev/ra0g /usr
```

When you invoke the fsck command it will display certain information on the file system. Refer to fsck(8) in the ULTRIX Reference Pages for further detail.

3. Change to the directory on the target file system. Once the target file system is available for use, use the cd command to position yourself at the top of the directory, in this case, /usr. For example:

```
# cd /usr
```

4. Restore the file system from the dump media using the rrestore command with the format described in Chapter 4. For example, to restore the /usr file system from the tape device on the remote system proton, you would enter:

```
# /etc/rrestore -rf proton:/dev/rmt0h
```

This command restores all of the files and directories in the file system from a dump tape mounted on the remote system's tape device (/dev/rmt0h) to the current directory (/usr).

5. Remove the file named restoresymtable that rrestore created in the current directory. Then reposition yourself at the root (/) directory. For example:

```
# rm restoresymtable
# cd /
```

6. Use the umount command to unmount the restored file system. For example:

```
# /etc/umount /dev/ra0g
```

This command unmounts the file system /usr and prepares the file system for the next step. Refer to mount(8) in the ULTRIX Reference Pages for more information on umount.

7. After unmounting the restored file system, use the fsck command to check for inconsistencies. For example:

```
# /etc/fsck  /dev/rra0g
```

The fsck command checks the named file system, notifies you of all inconsistencies, occasionally prompts you for a response to its suggested course of action, and proceeds accordingly. When unsure of the consequences of your response, you should answer no. By answering no, you leave the condition uncorrected, but create a summary from which you can decide on a plan of action. You can also use the fsck command with either the -p or -P options. Refer to fsck(8) in the ULTRIX Reference Pages for information on these as well as other fsck options.

8. Perform a shutdown and reboot of the system with the new /usr file system:

```
# /etc/shutdown -r now
                .
                .
                .

login:
```

When the system has completed the shutdown and reboot process, it redisplays the `login:` prompt.

Device Mnemonics A

This appendix identifies and defines the mnemonics that are used to attach any hardware or software device to your system. The mnemonics are used by the /dev/MAKEDEV shell script to create the character or block special files that represent each of the devices. The mnemonics also appear in the system configuration file as described in the Guide to System Configuration File Maintenance.

Table A–1 lists the mnemonics in seven categories: generic, consoles, disks, tapes, terminals, modems, and printers. The generic category lists the mnemonics of a general nature and includes memory, null, trace, and tty devices. The consoles category lists the system console devices that the ULTRIX operating system uses. The disks, tapes, terminals, modems, and printers categories identify the appropriate mnemonics for those devices.

The description heading in Table A–1 identifies the corresponding device name. It does not define the mnemonic's use. For detailed information on the use of each mnemonic in relation to both the MAKEDEV script and the system configuration file, refer to the reference pages in Section 4 of the ULTRIX Reference Pages. If on-line reference pages are available, you can also use the man command. For instance, if you enter at the system prompt:

```
# man ra
```

the system displays the reference page for the Mass Storage Control Protocol (MSCP) disk controller driver. Where appropriate, the SYNTAX section of the reference page defines the device's syntax as it appears, or should appear, in the config file. Refer to /dev/MAKEDEV for additional software device mnemonics that MAKEDEV uses. Refer to MAKEDEV(8) in the ULTRIX Reference Pages for a description of the MAKEDEV utility.

You should note that Table A–1 uses the convention of an asterisk (*) beside a mnemonic and a question mark (?) beside a device name to mean a variable number. The range of the variable number is dependent on the particular device.

Table A-1: Devices Supported by MAKEDEV

Category	Mnemonic	Description
Generic	boot*	Boot and std devices by cpu number; e.g., boot750
	mvax*	All MicroVAX setups; e.g., mvax2000
	vaxstation*	A VAXstation 2000 setup; e.g., vaxstation2000
	std	Standard devices below with all console subsystems:
	drum	Kernel drum device
	errlog	Error log device
	kUmem	Kernel Unibus/Q-bus virtual memory
	kmem	Virtual main memory
	mem	Physical memory
	null	A null device
	trace	A trace device
	tty	A tty device
	local	Customer specific devices
Consoles	console	System console interface
	crl	Console RL02 disk interface for VAX 86?0
	cs*	Console RX50 floppy interface for VAX 8??0
	ctu*	Console TU58 cassette interface for VAX 11/750
	cty*	Console extra serial line units for VAX 8??0
	cfl	Console RX01 floppy interface for 11/78?
	ttycp	Console line used as auxiliary terminal port
Disks	hp*	MASSBUS disk interface for RM?? drives
	ra*	UNIBUS/Q-bus/BI/HSC MSCP disk controller interface
	ese*	UNIBUS/Q-bus/BI/HSC MSCP electronic ESE20 disk
	rb*	UNIBUS IDC RL02 disk controller interface for RB?? drives
	rd*	VAXstation 2000 and MicroVAX 2000 RD type drives
	rz	SCSI disks (RZ22/RZ23/RZ55/RRD40)
	rk*	UNIBUS RK?? disk controller interface
	rl*	UNIBUS/Q-bus RL?? disk controller interface
	rx*	VAXstation 2000 and MicroVAX 2000 RX type drives
Tapes	mu*	TU78 MASSBUS magtape interface
	tms*	UNIBUS/Q-bus/BI/HSC TMSCP tape controller interface
	rv*	UNIBUS/Q-bus/BI/HSC TMSCP optical disk
	ts*	UNIBUS/Q-bus TS11/TS05/TU80 magtape interface
	tu*	TE16/TU45/TU77 MASSBUS magtape interface
	st*	VAXstation 2000 and MicroVAX 2000 TZK50 cartridge tape

Category	Mnemonic	Description
	tz*	SCSI tapes (TZ30/TZK50)
Terminals	cxa*	Q-bus cxa16
	cxb*	Q-bus cxb16
	cxy*	Q-bus cxt08
	dfa*	Q-bus DFA01 comm multiplexer
	dhq*	Q-bus DHQ11 comm multiplexer
	dhu*	UNIBUS DHU11 comm multiplexer
	dhv*	Q-bus DHV11 comm multiplexer
	dmb*	BI DMB32 comm multiplexer including dmbsp serial printer/plotter
	dhb*	BI DHB32 comm multiplexer
	dmf*	UNIBUS DMF32 comm multiplexer including dmfsp serial printer/plotter
	dmz*	UNIBUS DMZ32 comm multiplexer
	dz	UNIBUS DZ11 and DZ32 comm multiplexer
	sh*	MicroVAX 2000, 8 serial line expansion option
	ss*	VAXstation 2000 and MicroVAX 2000 basic 4 serial line unit
	dzq*	Q-bus DZQ11 comm multiplexer
	dzv*	Q-bus DZV11 comm multiplexer
	lta*	Sets of 16 network local area terminals (LAT)
	pty*	Sets of 16 network pseudoterminals
	qd*	Q-bus VCB02 (QDSS) graphics controller/console
	qv*	Q-bus VCB01 (QVSS) graphics controller/console
	sm*	VAXstation 2000 monochrome bitmap graphics/console
	sg*	VAXstation 2000 color bitmap graphics console
Modems	dfa*	DFA01 integral modem communications device.
Printers	dmbsp*	BI DMB32 serial printer/plotter
	dmfsp*	UNIBUS DMF32 serial printer/plotter
	lp*	UNIBUS LP11 parallel line printer
	lpv*	Q-bus LP11 parallel line printer

Index

N

O

R

S

T

U

HOW TO ORDER ADDITIONAL DOCUMENTATION

DIRECT TELEPHONE ORDERS

In Continental USA
and New Hampshire,
Alaska or Hawaii
call **800-DIGITAL**

In Canada
call **800-267-6215**

DIRECT MAIL ORDERS (U.S. and Puerto Rico*)

DIGITAL EQUIPMENT CORPORATION
P.O. Box CS2008
Nashua, New Hampshire 03061

DIRECT MAIL ORDERS (Canada)

DIGITAL EQUIPMENT OF CANADA LTD.
100 Herzberg Road
Kanata, Ontario K2K 2A6
Attn: Direct Order Desk

INTERNATIONAL

DIGITAL EQUIPMENT CORPORATION
PSG Business Manager
c/o Digital's local subsidiary
or approved distributor

Internal orders should be placed through the Software Distribution Center (SDC), Digital Equipment Corporation, Westminster, Massachusetts 01473

*Any prepaid order from Puerto Rico must be placed
with the Local Digital Subsidiary:
809–754–7575

Reader's Comments

Note: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well-organized? Please make suggestions for improvement. _____

_____

_____

_____

Did you find errors in this manual? If so, specify the error and the page number.

_____

_____

_____

Please indicate the type of user/reader that you most nearly represent.

- ☐ Assembly language programmer
- ☐ Higher-level language programmer
- ☐ Occasional programmer (experienced)
- ☐ User with little programming experience
- ☐ Student programmer
- ☐ Other (please specify) _____

Name _____ Date _____

Organization _____

Street _____

City _____ State _____ Zip Code or Country _____

Do Not Tear - Fold Here and Tape

digital

No Postage Necessary if Mailed in the United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO.33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

Digital Equipment Corporation
Documentation Manager
ULTRIX Documentation Group
ZKO3-3/X18
Spit Brook Road
Nashua, N.H.
03063

Do Not Tear - Fold Here and Tape

Cut Along Dotted Line

Reader's Comments

Note: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well-organized? Please make suggestions for improvement. _____

_____

_____

_____

Did you find errors in this manual? If so, specify the error and the page number.

_____

_____

_____

Please indicate the type of user/reader that you most nearly represent.

- ☐ Assembly language programmer
- ☐ Higher-level language programmer
- ☐ Occasional programmer (experienced)
- ☐ User with little programming experience
- ☐ Student programmer
- ☐ Other (please specify) _____

Name _____ Date _____

Organization _____

Street _____

City _____ State _____ Zip Code or Country _____

Do Not Tear - Fold Here and Tape

digital

No Postage Necessary if Mailed in the United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO.33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

Digital Equipment Corporation
Documentation Manager
ULTRIX Documentation Group
ZKO3-3/X18
Spit Brook Road
Nashua, N.H.
03063

Do Not Tear - Fold Here and Tape

Cut Along Dotted Line