68NW9209H44A

SYSTEM V/88 Release 3.2 System Administrator's Guide

(Part 1)

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 2565 PHOENIX, ARIZONA

POSTAGE WILL BE PAID BY ADDRESSEE

MOTOROLA INC.

Computer Group, Microcomputer Division, DW164
2900 South Diablo Way
Tempe, AZ 85282-9741

MOTOROL

Motorola welcomes your

Manual Title _____

Part Number _____

Your Name _____

Your Title _____

Company _____

Address _____

_____

_____

General Information:

Do you read this manu

☐ Install the produc

☐ Reference infor

In general, how do yo

☐ Index ☐ Table

Completeness: ☐ E

What topic would you

_____

_____

Presentation: ☐ Excellent ☐ Very Good ☐ Good ☐ Fair ☐ Poor

What features of the manual are most useful (tables, figures, appendixes, index, etc.)?

Is the information easy to understand? ☐ Yes ☐ No If you checked no, please explain:

Is the information easy to find? ☐ Yes ☐ No If you checked no, please explain:

Technical Accuracy: ☐ Excellent ☐ Very Good ☐ Good ☐ Fair ☐ Poor

If you have found technical or typographical errors, please list them here.

Page Number	Description of Error

SYSTEM V/88 Release 3.2

System Administrator's Guide

Part 1

(68NW9209H44A)

The information is this document has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies. Furthermore, Motorola reserves the right to make changes to any products herein to improve reliability, function, or design. Motorola does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights or the rights of the others.

PREFACE

The *SYSTEM V/88 Systems Administrator's Guide* describes procedures used in the administration of a computer running the SYSTEM V/88 Release 3.2 operating system.

This guide assumes that you know the mechanics of using a computer terminal to enter commands, and that you have an awareness of such system fundamentals as the directory structure and the shell. We also expect that you feel comfortable using your computer; you know how to turn it on and how to use the diskette drive.

First Edition February 1990

Contents

Figures

Figures

Tables

i Introduction

This guide is designed to accomplish the following objectives:

- provides clear instructions on how to perform the administrative tasks of SYSTEM V/88
- gives you background information about when and why these tasks are desirable
- serves as a quick reference to administrative procedures

What Is a System Administrator?

A System Administrator performs two main tasks:

- decides what rules are needed to govern the use of the computer system
- implements those rules so as to provide the maximum amount of computing service for the system's users, consistent with the physical limitations of the machine

If you are the only user of your system, these tasks consist of those things you do to keep the machine running and your programs and data from disappearing permanently. If, however, your system will be used by a number of other people, the tasks become more complex, and you are required to be aware of the needs of your whole user community.

How This Guide Is Organized

There are two main parts to this guide:

Part 1. Procedures
Part 1 contains ten sets of step-by-step procedures that tell you how to keep your computer in operation. Each set of tasks is related to a general topic, e.g., User Services, Processor Operations.

Part 2. Support Information
Part 2 contains more detailed information about each of the ten sets of procedures. The chapters are numbered to parallel the Part 1 procedures.

This guide also includes four appendices and a glossary.

How to Use This Guide

1. Use the procedures in Part 1 to begin. They lead you through administrative tasks without requiring preliminary knowledge or experience in that area.
2. Use the chapters in Part 2 to learn more about what the procedures do. They explain what is going on in the procedures and provide background information about the basic elements of SYSTEM V/88.
3. Use the Glossary to look up definitions of terms that are unfamiliar.
4. As you gain experience, use the guide for reference.

About the Procedures

A table at the beginning of each procedure gives you information in capsule form. Table entries appear only when the information is relevant. The tables follow this style:

PURPOSE	Summary of procedures.
When Performed	When you should schedule the procedure.
Starting Conditions	The state the computer should be in when you begin the procedure. Any special login requirements.
sysadm menu	The part of the System Administration Menu package that contains the subcommands to perform the procedure.
Commands	The commands used to perform the procedure.
Bootable Programs	The names of programs used to boot the system.
Media	Disks or tapes used in the procedure.
Caution	Special instructions you must follow before or during the procedure to ensure the integrity of your system software and user files.
Reference	The chapter and section in Part 2 where this topic is more fully discussed. Other SYSTEM V/88 manuals containing relevant information.

System Administration Commands

The majority of the procedures are based on menus of the System Administration Menu package. This package consists of a hierarchical arrangement of interactive screens that lead you through system administration tasks. Appendix D contains the System Administration Menu screens used to perform system administration tasks using the **sysadm** commands.

To enter the System Administration Menu, type:

```
$ sysadm
Password:  (If you are root, you are not prompted for a password.)
```

The system responds with a screen showing the different submenus of the **sysadm** package. Type the name or number to display the screen that shows the subcommands for the submenu (see Appendix D). Subcommands are the equivalent of menu selections from lower level menus.

You can get to the submenu level with the following command:

```
$ sysadm subcommand
```

When you bypass the System Administration Main Menu, you can only perform the one requested subcommand before being returned to the shell. Some subcommands, when accessed through the menu hierarchy, however, allow you to remain in that particular submenu when the task is complete. If you are performing more than one task, consider using the System Administration Menu hierarchy. The procedures shown in this guide bypass the higher level System Administration Menus and take you directly to the subcommands.

Using Special Subcommands

Another way to bypass the System Administration Menu hierarchy is to use special administrative commands. Special administrative subcommands are password-protected, thereby allowing some users to perform certain administrative tasks without giving them access to the entire System Administration structure. If, for example, you want to allow several users to power off the computer but not let them perform any other administrative tasks, you would assign a password to the **powerdown** subcommand and give that password to those users. For an explanation and list of special administrative subcommands, see Chapter 1, *Special Administrative Passwords.*

System States

In some procedures, a particular system state is required. In most cases this means that the system must be in either the single- or multi-user state. The single-user state corresponds to run level 1, while the multi-user state corresponds to run levels 2 or 3. Procedures for bringing the system to different system states are found in Part 1 under Processor Operations Procedures. See the section on *Operating Levels*, in Chapter 3, *Processor Operations*, for more information on system states.

Logins

In some procedures, a particular login is required. This usually means that you must be logged in as **root** to do the procedure. The phrase "an authorized login" is also used. The standard meaning of this term is that you must log in using a special administrative or system login name to do the procedure (see Chapter 1, *System Security*, for a list of these logins).

Passwords

It is strongly recommended that you set up and use passwords for administrative and system logins (see Procedure 1.4 for information on how to do this). In the procedures, we assume that such password protection has been established. Therefore, when you enter a **sysadm** command as an ordinary user, you are prompted for a password, for example:

```
$ sysadm adduser
Password:
```

At this point, to proceed with the procedure, you are required to enter the password for **sysadm**. As is always the case in SYSTEM V/88, the password is not echoed to your screen.

In procedures that require you to be logged in as **root** (e.g., the super user), you are not prompted for the **sysadm** password. Also, the pound sign (#) prompt is used for the **root** login. For example:

```
# sysadm modtty
```

Notation Conventions

Whenever the text includes examples of output from the computer and/or commands you enter, the following standard notation scheme is common throughout this documentation:

- Text that you type from your terminal is shown in **bold** type.
- Text that is printed on your terminal by the computer is shown in `constant width` type.
- Comments and explanations within a display are shown in *italic* type and are indented to separate them from the text that represents computer output or input.

  Italics are also used to show substitutable values, e.g., *file*, when the format of a command is shown.
- There is an implied **RETURN** at the end of each command and menu response you enter.
- Where you may be expected to enter only a **RETURN** (as in the case where you are accepting a menu default), the symbol <**CR**> is used.
- The dollar sign ($) and pound sign (#) symbols are the standard default prompt signs for an ordinary user and **root**, respectively.

  $ means you are logged in as an ordinary user.
  # means you are logged in as **root**

- When the # prompt is used in an example, it means the command illustrated may be used only by **root**.
- When the full path name of a command is shown in an example (e.g., **/etc/fsck**) the command must be entered that way.

Command References

When commands are mentioned in a section of the text for the first time, a reference to the manual section where the command is described is included in parentheses: **command**(section). The numbered sections are located in the following manuals:

Sections (1), (1C)	*User's Reference Manual*
Sections (1M), (7), (8)	*System Administrator's Reference Manual*
Sections (1), (2), (3), (4), (5)	*Programmer's Reference Manual*

Information in the Examples

While every effort has been made to present displays of information just as they appear on your terminal, it is possible that your system will produce slightly different output. Some displays depend on a particular machine configuration that may differ from yours. Changes between releases of the SYSTEM V/88 software may cause small differences in what appears on your terminal.

P1 System Identification and Security Procedures

Introduction

The following procedures are covered in this section:

Procedure 1.1 **Check Console Terminal Configuration**
To assure that the system console terminal is properly configured.

Procedure 1.2 **Set Time and Date**
To set the time and date of the internal system clock.

Procedure 1.3 **Establish or Change System Node Name**
To define the formal system name, especially for the computer to be a node in a network.

Procedure 1.4 **Assign Special Administrative Passwords**
To assign special passwords to administrative and system logins.

Procedure 1.5 **Forgotten Root Password Recovery**
To recover from forgetting or the corruption of the root password.

Procedure 1.1: Check Console Terminal Configuration

Purpose	To assure that communication with the system is maintained.
References	Delta Series System Manual The Operator's Guide for your console terminal

All system administration functions are performed at the console terminal, which should be set at a 9600 baud rate. If the console port is operated at another data rate, you may lose communication with the system when you shut down to the firmware mode (see Chapter 3, *Processor Operations* for an explanation of firmware mode and other operating modes).

To ensure that your terminal is configured properly, perform the following tasks. Use the Operator's Guide for your terminal to learn how to make these equipment checks.

- Set the input/output terminal speed option to 9600.
- Set the interface to 8-bit ASCII, full duplex, with a parity of "none" or "space" depending on the terminal you have.
- If you lose communication with the system, check to see if the terminal is still plugged in.
- If you lose communication with the system when going to the firmware mode, make sure the input/output terminal speed option is set to match that expected by your computer's firmware (usually 9600 baud). See also the troubleshooting information in your computer's operator's guide.

A printer should be part of the console equipment configuration because it provides a record of exactly what was done and how the system responded. It is essentially the system log, and advantageous when you run diagnostics. If your console terminal has this capability, the best method is to hook the printer directly to the system console. The console printer should be independent of the LP Spooler system. Alternatively, the **/dev/osm** device can be used to keep a log of system messages.

Procedure 1.2: Set Time and Date

Purpose	To synchronize system time with clock time or to reset the system time after it has been corrupted.
Starting Conditions	System state —multi-user, for synchronizing system time with current clock time —single-user, for setting the date ahead Login —**root**, to reset clock with **date**(1)
sysadm menu	SYSTEM SETUP
Commands	**sysadm datetime**(1) **date**(1) —requires logging in as **root**
Caution	Go to single-user mode (Procedure 3.3) if you are setting the date ahead.

Setting the date ahead by one or more days should be done in the single-user mode. Setting the date ahead while in the multi-user mode with **cron** running should be avoided. The **cron** program will try to "catch-up" for the time interval involved. All the processes that were scheduled to run in the time interval are started by **cron**.

Method 1: To set the time and the date while in the operating system, use the System Administration Menu **datetime**. For example:

```
# sysadm datetime
Running Subcommand 'datetime' from menu 'syssetup',
SYSTEM SETUP

Current time and time zone is: 04:59 EDT
Change the time zone? [y, n, q, ?] n
Current date and time:  Tue. 08/28/85 05:00
Change the date and time: [y, n, q, ?] y
Month   default 08       (1-12): <CR>
              (Using <CR> to accept the default)
Day     default 28       (1-31): <CR>
Year    default 84       (70-99): <CR>
Hour    default 05       (0-23): <CR>
Minute  default 00       (0-59): 04
Date and time will be set to:  08/28/84 05:04. OK? [y, n, q] y
The date and time are now changed.
```

Method 2: The clock also can be set using the **date** command when you are in the operating system. You must be logged in as **root** to use **date**; you must first take the system to single-user mode. The arguments to the **date** command are in the sequence of month, day, hour, minute, and year.

```
# date 0216131686
Sun Feb 16 13:16:00 EST 1986
```

Procedure 1.3: Establish or Change System and Node Names

Purpose	To define the system and node names —for a new system —when reconfiguring the system to include the Network Services Extension product.
Starting Conditions	System state —multi-user or single-user Login —authorized login or **root**
sysadm menu	SYSTEM SETUP
Commands	**sysadm nodename**(1) **uname**(1) **–S** —need **root** login
Caution	Do not change system and node names arbitrarily. Changes must be coordinated with your networking connections.
References	Chapter 9, *Basic Networking* Chapter 10, *Remote File Sharing (RFS)*

The system and node names of the system can be set by any of the following:

- Using the **sysadm nodename** command.
- Using the **uname** command.
- Reconfiguring the operating system after changing the names of SYS and NODE tunable parameters (see Chapter 6, *Performance Management*).

Choose one of the following methods for establishing or changing the system and node name.

System Administration Menu —nodename

The following command line entries and system responses show the setting of the node name using the **sysadm nodename** command. The node name is then output using the **uname** command. The contents of the **/etc/init.d/nodename** file is the result of the execution of **sysadm nodename**. This file is linked to **/etc/rc2.d/s00nodename.**

```
# sysadm nodename
Running subcommand 'nodename' from menu 'syssetup',
SYSTEM SETUP

This machine is currently called "unix".
Do you want to change it? [y, n, ?, q] y
What name do you want to give it? [q] Abcd5678
# uname -a
Abcd5678 Abcd5678 3.2 1.0C M88100
# cat /etc/init.d/nodename
#         Node name changed 02/01/86 14:55:49.
#
```

Command—uname

The following shows how to display and change the system and node names:

```
# uname -a
unix unix 3.2 1.0C M88100
# uname -S abcdefghijk
uname: name must be <= 8 letters
# uname -S Abcd5678
# uname -a
Abcd5678 Abcd5678 3.2 1.0C M88100
#
```

NOTE

Using **uname** to change the system and node names is not as permanent as using **sysadm nodename**. Whenever the system is rebooted, the system and node names assigned to the system are those last entered through **sysadm nodename** and residing in the file **/etc/init.d/nodename**. This file is not set up or changed by **uname**. Thus, it is a good practice to use the command **sysadm nodename** to change the system and node names. The **/etc/init.d/nodename** file also overrides the values of SYS and NODE compiled into the kernel.

Procedure 1.4: Assign Special Administrative Passwords

Purpose	To permit controlled access to various administrative and special system functions.
Starting Conditions	Multi-user or single-user state, any login.
sysadm menu	SYSTEM SETUP
Commands	**sysadm admpasswd**(1) **sysadm syspasswd**(1)
Reference	*Special Administrative Passwords* in Chapter 1, *System Security*

After you have set up your system, you should assign passwords to the special administrative and system logins (see Chapter 1, *System Security,* for definitions of these logins). The administrative logins are: **br, setup, powerdown, sysadm, checkfsys, makefsys, mountfsys,** and **umountfsys.**

The system logins are: **root, sys, adm, bin, uucp, nuucp, lp,** and **daemon**.

Step 1: The following command displays all the special administrative logins available to allow you to assign or change any of the passwords:

```
# sysadm admpasswd
Running subcommand 'admpasswd' from menu 'syssetup',
SYSTEM SETUP
Do you want to give passwords to administrative logins?
[y, n, ?, q] y
        If you enter "y", you are prompted about each administrative login.
The login 'setup' already has a password.
Do you want to change the password, delete it, or skip it?
[c, d, s, q, ?] s
Password unchanged.
The login 'powerdown' already has a password.
Do you want to change the password, delete it, or skip it?
[c, d, s, q, ?] q
Password unchanged.
```

Step 2: To assign a password to the special system logins, enter **sysadm syspasswd**. If you want to change any of the following passwords, you must either be logged in as **root** or as one of these logins, then execute the **passwd** command. For example, the following displays:

```
# sysadm syspasswd
Running subcommand 'syspasswd' from menu 'syssetup',
SYSTEM SETUP
Do you want to give passwords to system logins? [y, n, ?, q] y
If you enter y, you are prompted about each system login.
Do you want to give the 'daemon' login a password?
[y, n, ?, q] n
The following system logins still do not have passwords: daemon
```

This command allows you to assign passwords only to those logins that have never received a password in the first place.

Procedure 1.5: Forgotten Root Password Recovery

CAUTION

Restoring the original operating system is a drastic way of recovering from a forgotten root password. Make every effort to remember or discover the root password before performing this procedure.

Purpose	To recover from a forgotten **root** password by booting from tape, mounting the root disk, and editing the **passwd** file.
Media	SYSTEM V/88 Release 3.2 boot tape.

The following steps are necessary to recover the ability to log in as **root** based on the assumption that you are not able to log in (as a conventional user) and restore the **/etc/passwd** file from another login.

Step 1: If the system is in the multi-user state, follow this step; otherwise, go to Step 2.

Shutdown the system:

```
# shutdown -y -g0 -i6
```

Wait for the message:

```
NOTICE: System secured for RESET.
```

Step 2: Insert the SYSTEM V/88 BOS tape into the tape drive. To boot from tape, you must first enter the system debugger. When the boot process begins, press the **BREAK** key on the system console to interrupt the automatic boot sequence. This places you in the debugger.

The debugger command **bo** is used to boot from the tape drive (Unit 4) in your tape controller in a 323, type:

```
bo 4
```

If it is a 327 (see the SRG for more information on booting from tape), type:

```
bo 2 40
```

The system boots from tape, printing a number of messages. When you see the initial menu, type **'q** to exit to a shell.

Step 3: Mount the root file system (in this example, **/dev/dsk/m323_0s0**):

 # mount /dev/dsk/m323_0s0 /mnt

The system then prints a warning message:

`mount: warning` *root* `mounted as` *mnt*

Step 4: Using the editor **vi**(1), edit the password file **/mnt/etc/passwd** to delete the password for root (user id **0**). (You must run: **TERM=vt100 export TERM** before using **vi**.) For example, the root password entry might initially be:

root:TFGezR7KF1q:0:3:0000-Admin(0000):/:

This should be changed to:

root::0:3:0000-Admin(0000):/:

NOTE

If shadow passwords have been enabled, you must instead edit the shadow password file **/mnt/etc/shadow**.

Step 5: Unmount the root filesystem:

 # umount /mnt

Step 6: Shutdown and reboot from the boot disk (using the automatic boot process):

 # uadmin 2 0

Wait for the message:

`NOTICE: System secured for powering down.`

Press the reset switch and reboot from the disk.

Step 7: When the system has been rebooted, follow Procedure 1.4 for setting the root password.

> **CAUTION**
>
> **Serious security problems may arise if a new password is not assigned at this point.**

P2 User Services Procedures

Introduction

The following procedures are covered in this section:

Procedure 2.1 **Add Users or Groups**
To add information about new users of the system, or to name groups.

Procedure 2.2 **Modify User Information**
To change information about users or groups.

Procedure 2.3 **Delete Users or Groups**
To remove users or groups from the system.

Procedure 2.4 **List Users or Groups**
To display information about users or groups.

Procedure 2.5 **Write to All Users**
To send a message to all users logged in.

Procedure 2.1: Add Users or Groups

Purpose	To identify new users or groups of users to the system.
Starting Conditions	System state —multi-user
sysadm menu	USER MANAGEMENT
Commands	**sysadm adduser**(1) **sysadm addgroup**(1)
Reference	*Login Administration* in Chapter 2, *User Services*

Step 1: Enter one of the following commands:

```
$ sysadm adduser
Password:

      or

$ sysadm addgroup
Password:
```

Step 2: If you enter the command **sysadm adduser**, you are led through the following sequence:

```
Running subcommand 'adduser' from menu 'usermgmt',
USER MANAGEMENT

Any time you want to quit, type "q".
If you are not sure how to answer any prompt, type "?" for help,
or see the Administrator's Guide.

If a default appears in the question, press <RETURN> for the default.

Enter user's full name [?, q]: John Q. Public
Enter user's login ID [?, q]: jqp
Enter user's ID number (default 102) [?, q]: <CR>
   (Accepting defaults by entering <CR>)
Enter group ID number or group name
(default 1)[?, q]: <CR>
Enter user's login (home) directory name.
(default '/usr/jqp')[?, q]:<CR>

This is the information for the new login:
   User's name:     John Q. Public
   login ID:        jqp
   user ID:         102
   group ID:        1
   home directory: /usr/jqp
Do you want to install, edit or skip this entry [i, e, s, q]? i
Login installed
Do you want to give the user a password? [y, n] y
New password:
        (Enter at least six characters, one of them a numeral.)
Re-enter new password:

Do you want to add another login?[y, n, q]  n
```

Step 3: If you enter the command **sysadm addgroup**, this sequence appears:

```
Running subcommand 'addgroup' from menu 'usermgmt',
USER MANAGEMENT

Anytime you want to quit, type "q".
If you are not sure how to answer any prompt, type "?" for help,
or see the Administrator's Guide.

If a default appears in the question, press <RETURN> for the default.

Enter group name [?, q]: seventy7
Enter group ID number (default 101) [?, q]: <CR>
   (Accepting default by entering <CR>)
This is the information for the new group:
   Group name:      seventy7
   group ID:        101
Do you want to install, edit or skip this entry [i, e, s, q]? i
Group installed

Do you want to add another group?[y, n, q]  n
```

Procedure 2.2: Modify User Information

Purpose	To change stored information about users.
Starting Conditions	System state —multi-user
sysadm menu	USER MANAGEMENT
Commands	**sysadm modadduser**(1) **sysadm modgroup**(1) **sysadm moduser**(1) **sysadm chgloginid**(1) **sysadm chgpasswd**(1) **sysadm chgshell**(1)
Reference	*Login Administration* in Chapter 2, User Services

This procedure covers three separate functions:

- Changing the default values that apply to the **adduser** sequence (**modadduser**)
- Changing the name of a group (**modgroup**)
- Changing three of the attributes of user information (**moduser**)

Function 1, Step 1: Enter the command:

```
$ sysadm modadduser
Password:
```

Function 1, Step 2: The **sysadm modadduser** command gives you the opportunity to change either or both of the default values for group ID and home (parent) directory that appear on the **adduser** form. The following screen shows an example of changing the default group number from **1** to **100**:

```
Running subcommand 'modadduser' from menu 'usermgmt'
USER MANAGEMENT


Anytime you want to quit, type "q".
If you are not sure how to answer any prompt, type "?" for help,
or see the Administrator's Guide.

Current defaults for adduser:
group ID                1       (other)
parent directory        /usr
Do you want to change the default group ID? [y, n, ?, q] y
Enter group ID number or group name [?, q] 100
Do you want to change the default parent directory? [y, n, ?, q] n

These will be the new defaults:
group ID:               100
parent directory:       /usr
Do you want to keep these values? [y, n, q] y
Defaults installed.
#
```

Function 2, Step 1: To change the value of a group ID name, enter the command:

```
$ sysadm modgroup
Password:
```

Function 3, Step 1: To change the values for an individual user's login, enter the command:

```
$ sysadm moduser
Password:
```

The following menu displays on your terminal:

```
MODIFY USER'S LOGIN

1 chgloginid      change a user's login
2 chgpassword     change a user's password
3 chgshell        change a user's login shell

Enter a number, a name, the initial part of a name, or
? or <number>? for HELP, q to QUIT:
```

Function 3, Step 2: Selecting an item from this menu starts a prompt sequence that helps you make the required change. When a user is first given a login ID, a default shell (**/bin/sh**) is assigned. The **chgshell** subcommand enables you to assign a different shell.

Function 3, Step 3: When you select Item 3 from the menu, or if you entered the command: **sysadm chgshell**, the following sequence appears on your terminal:

```
Running subcommand 'chgshell' from menu 'moduser',
MODIFY USER'S LOGIN

Enter user's login ID [?, q]: jqp
The current shell is /bin/sh
Enter new shell command[q]: /bin/rsh

Do you want to change the login shell of another login?[y, n, q] q
```

The above sequence assigns a restricted shell to user **jqp**.

Procedure 2.3: Delete Users or Groups

Purpose	To clear the system of an inactive user. To eliminate a group name that is no longer needed.
Starting Conditions	System state —multi-user
sysadm menu	USER MANAGEMENT
Commands	**sysadm deluser**(1) **sysadm delgroup**(1)
Caution	When you delete a user's ID, all the files and directories owned by that ID are deleted too.
Reference	*Login Administration* in Chapter 2, *User Services*

Step 1: Deleting a group ID is done with this command:

```
$ sysadm delgroup
Password:
```

Step 2: The prompt sequence is:

```
Which group name do you wish to delete?[q] seventy7
Do you want to delete group name 'seventy7', group ID 101?
[y, n, ?, q] y
seventy7 has been deleted
Do you want to delete any other group?[y, n, q] q
```

NOTE

The **sysadm delgroup** command deletes only the specified group and not the user login(s) assigned to that group. The logins belonging to the group must be deleted separately using **sysadm deluser**.

Step 3: Deleting a user's login ID requires more persistence. The user's home directory and all the files in and below that directory are deleted as well. The following is the sequence:

```
$ sysadm deluser
Password:

Running subcommand 'deluser' from menu 'usermgmt',
USER MANAGEMENT

This function COMPLETELY REMOVES THE USER, their mail file,
home directory and all files below their home directory from
the machine.  Once this is done, there is no way guaranteed to get
them all back.
         BE SURE THIS IS WHAT YOU WANT TO DO!

Enter login ID you wish to remove[q]: jqp
         'jqp' belongs to 'John Q. Public'
         whose home directory is /usr/jqp
Do you want to remove login ID 'jqp'?[y, n, ?, q] y

/usr/jqp and all files under it have been removed.

Enter login ID you wish to remove [q]: q
```

Procedure 2.4: List Users or Groups

Purpose	To see what users or groups are known to the system.
Starting Conditions	System state —multi-user
sysadm menu	USER MANAGEMENT
Commands	**sysadm lsuser**(1) **sysadm lsgroup**(1)

Step 1: The two **sysadm** subcommands in this procedure enable you to see what groups and what users are in the computer. The following command is to list groups:

```
$ sysadm lsgroup
Password:
```

The command produces a report with these column headings:

```
Groups currently in the computer
(press <RETURN> to start listing each time you hear the bell)

group     group     logins permitted to become
name      number     members using newgrp
-----     ------     --------------------------
adm         4        root,adm,daemon
bin         2        root,bin,daemon
daemon     12        root,daemon
mail        6        root
other       1
rje         8        rje,shqer
root        0        root
sys         3        root,bin,sys,adm
$
```

Step 2: If you enter this command, the following lines appear on your terminal:

$ sysadm lsuser

```
Users currently in the computer
(press <RETURN> to start listing each time you hear the bell)
```

When you press **<RETURN>** a list in the following form displays:

```
login name        user name
----------        ---------
adm               0000-Admin(0000)
bin               0000-Admin(0000)
checkfsys         check disk file system
daemon            0000-Admin(0000)
listen            0000-listen(0000)
lp                0000-lp(0000)
makefsys          make disk file system
mount fsys        mount disk file system
nuucp             0000-uucp(0000)
powerdown         general system administration
rje               0000-rje(0000)
root              0000-Admin(0000)
setup             general system administration
sys               0000-Admin(0000)
sysadmin          general system administration
trouble           trouble(0000)
umountfsys        unmount disk file system
uucp              0000-uucp(0000)
uucpa             Uucp login
$
```

Procedure 2.5: Write to All Users

Purpose	To send urgent messages to all users logged in.
Starting Conditions	System state —multi-user Login —**root** required to prevent users from blocking messages.
Commands	**wall**(1)
Reference	*Write to All Users* in Chapter 2, *User Services*

Step 1: For times when it is necessary to communicate with all users on the system at once, the SYSTEM V/88 **wall** command is used:

 # wall

The command reads whatever you type in at your terminal until it reads an end-of-file (indicated by typing in a **control-d**).

Step 2: The message you type in is sent immediately to the terminal of all users logged in. It is preceded by:

```
Broadcast Message from ...
```

A typical use of the **wall** command is to warn users that the system is about to be shut down:

```
Broadcast Message from root:  System coming down in
ten minutes.  Please log off.
```

P3 Processor Operations Procedures

Introduction

The following procedures are covered in this section:

Procedure 3.1 **Powerup**
To power up the system to the multi-user state.

Procedure 3.2 **Powerdown**
To halt the system and turn the power off.

Procedure 3.3 **Shutdown to Single-User**
To bring the system to the single-user state to do administrative tasks.

Procedure 3.4 **Return to Multi-User**
To return the system to the multi-user state after it was brought to another state for administrative purposes.

Procedure 3.5 **Shutdown to Firmware Mode**
To bring the system to firmware mode to run diagnostics.

Procedure 3.1: Powerup

Purpose	To turn on the system and make it available for use.
Starting Conditions	System state —(power off).
Time	Approximately 2 minutes (depending on configuration).
Reference	*Powering Up* section in your Delta Series system manual.

To power up the computer from a halted state (power is off), perform the following procedure:

Step 1: Turn on the console and wait for the cursor to appear.

Step 2: Follow the procedures in a your Delta Series system manual to apply power to the system. By default, most systems perform system self tests and boot the system automatically. By default, the operating system proceeds immediately to multi-user mode. At this point, the standard powerup sequence occurs. Several messages appear on the screen concerning release information and the amount of real and available system memory.

Step 3: Log on to the system when the prompt `Console login:` appears. You can log in with a system administrative login or a user login.

Procedure 3.2: Powerdown

Purpose	To halt the system and turn off the power.
Starting Conditions	System state —multi-user or single-user. You must **mount /usr** to run **sysadm** in single-user mode. Login —**root**, unless you know the **sysadm** password.
sysadm menu	MACHINE MANAGEMENT
Commands	**sysadm whoson**(1) **sysadm powerdown**(1) **powerdown**(1M) **shutdown**(1M)—**root** login only
Caution	Do not remove power until the powerdown procedure is completely finished.
Reference	*Turning the System Off* section in your Delta Series system manual.

There are differences in the procedure depending on whether you are in multi-user or single-user state.

From Multi-User

The best way to turn off the computer while the system is in the multi-user state is to use the **sysadm powerdown** command. This command causes the system to flush the system buffers, close any open files, stop all user processes and daemons currently running, unmount file systems, and then remove power from the computer, if possible. For machines that do not provide for software control of power removal, the system prints a message and enters an idle state when it is safe to remove power manually.

Step 1: Check who is logged in before taking any action that would affect a logged-in user. Enter the following command:

```
$ sysadm whoson
Password:
```

A typical response might be:

```
These users are currently logged in:
   ID    terminal number sign-on time
------- --------------- -------------
root      console         Jan 31  18:06
jaf       tty02           Jan 30  22:30
```

Step 2: Notify any users that the system is shutting down via the **/etc/wall**(1) command (see Procedure 2.5). For example:

```
# /etc/wall <CR>
The system will be coming down in 5 minutes. <CR>
Please log off. <CTRL-D>
Broadcast Message from root (console) on sys2 Wed Feb 26 07:30:27...
The system will be coming down in 5 minutes.
Please log off.
```

Step 3: Enter the following command:

```
$ sysadm powerdown
Password:
```

Observe the following output:

```
Running subcommand 'powerdown' from 'machinemgmt',
MACHINE MANAGEMENT

Once started, a powerdown CANNOT BE STOPPED.
Do you want to start an express powerdown? [y, n, ?, q] n
Enter the number of seconds to allow
between the warning messages (default 60): [?, q] 30

Shutdown started.    Thu May 16 17:10:57...

Broadcase Message from adm (console)  Thu May 16 17:10:59
The system will be shut down in 30 seconds.
Please log off now.

Broadcast Message from adm (console)  Thu May 16 17:11:30
THE SYSTEM IS BEING SHUT DOWN NOW ! ! !
Log off now or risk your files being damaged.

INIT: New run level: 0
The system is coming down.  Please wait.

The system is down.
NOTICE:  System Halt Requested
NOTICE:  System secured for powering down.
```

At this point, the power can be removed. (It will be removed automatically where supported by the machine.) To protect your system from unauthorized system powerdown (from a user terminal), assign a password both to the **sysadm** login as well as to the **powerdown**(1M) command (see Procedure 1.4, *Assign Special Administrative Passwords*).

From Single-User

If the system is in the single-user state, use the following command to power down the system:

3

Step 1: To remove power and guarantee file system integrity, use the **shutdown** command:

shutdown –y –i0 –g0

The arguments have the following meanings:

–y assume yes answers to all questions
–i0 go to state 0 (off)
–g0 allow grace period of 0 seconds

The following displays on the console:

```
Shutdown started.

The system will be shutdown in 0 seconds.
Please log off now.
THE SYSTEM IS BEING SHUTDOWN NOW ! ! !
Log off now or risk your files being damaged.
The system is coming down. Please wait.

The system is down.
NOTICE:  System Halt Requested
NOTICE:  System secured for powering down.
```

All services are stopped and power can be removed from the machine.

Procedure 3.3: Shutdown to Single-User

Purpose	To perform administrative tasks that should be done when no other users are on the system, for example: —software installation —file backup and restore —hard disk formatting —system reconfiguration
Starting Conditions	System state —multi-user. Login —root
Commands	**shutdown**(1M)
Reference	*Going to Single-User Mode* section in your Delta Series system manual.

Shutting the system down to the single-user state should be done as much as possible during off-hours, since only the console has access to the system in the single-user state.

Step 1: Log in as **root** at the console.

Step 2: Type:

 # shutdown

By default, **shutdown** prompts you about the various broadcast messages, provides a 60 second grace period between each message, and brings the system to the single-user state. All processes are killed and all file systems (except **root**) are unmounted. When you arrive in the single-user state, the following displays:

```
INIT: SINGLE USER MODE
```

You may now proceed with your intended tasks.

Procedure 3.4: Return to Multi-User

Purpose	To make the system available to users after administrative duties have been performed in either the single-user state or the firmware mode.
Starting Conditions	System state —single-user or firmware. Login —**root**
Commands	**init**(1M)
Bootable Programs	**/stand/unix**
Reference	*A Look at Entering the Multi-User State* section I in your Delta Series system manual.

There are two system states from which you can return the system to the multi-user state. You can bring the system back from the single-user state or the firmware mode.

From Single-User

After administrative tasks are finished, you can bring the system back to the multi-user state from the single-user state via the **init** command.

Step 1: At the console, enter:

 # init 2

This causes **init** to inspect **/etc/inittab** and execute entries that will initialize the system to the multi-user state and displays the following:

```
INIT: New run level: 2

The system is coming up.  Please wait.
```

The file systems are checked and the current system configuration may be printed out, the following displays:

Note that run level 3 should be used if RFS services are required.

```
The system is ready.

Console login:
```

Now you can log in either as **root** or as a conventional user, since the system is in the multi-user state.

From Firmware

After firmware programs have been run, you can bring the system back from the firmware mode by following the instructions in your Delta Series system manual for loading the boot program **/stand/unix** from the hard disk.

After the the sanity of the root file system is checked (via **fsstat**(1M)), a file system check is performed if necessary (via **fsck**(1M)), the system configuration will be printed out, and the system is placed in the multi-user state. When the following prompt appears you may log in with an appropriate system administrative login or user login:


```
Console login:
```

Procedure 3.5: Shutdown to Firmware Mode

Purpose	To perform the following functions: —run hardware/firmware diagnostics —dump the system image to tape
Starting Conditions	System state —multi-user or single-user. You must **mount /usr** to run **sysadm** in single-user mode. Login —**root**, unless you know the **sysadm** password.
sysadm menu	MACHINE MANAGEMENT
Commands	**sysadm firmware**(1) —may require a password **/etc/shutdown**(1M) —superuser only
Bootable Programs	**/stand/unix**
Reference	Your Delta Series system manual.

There are usually a number of programs that can be run from the firmware mode of a computer. These programs may be ROM, RAM, or hard-disk based.

Step 1: To execute any of these programs, enter the firmware mode via **sysadm firmware** (or **shutdown –i5**).

```
# sysadm firmware
Running subcommand 'firmware' from menu 'machinemgmt',
MACHINE MANAGEMENT
Once started, this procedure CANNOT BE STOPPED.
Do you want to go to firmware "express"? [y, n, ?, q]
```

Step 2: As with shutting down or powering down the system, you have the option to specify a delay between the warning messages. If the system is in the multi-user state, answer **n** (for no) to the question, so that the system will shut down with a small delay. If you answer no, you are prompted for the amount of the time delay.

```
Enter the number of seconds to allow
between the warning messages (default 60): [?, q] 30
```

Step 3: Now the shutdown process starts.

```
Shutdown started.     Thu May 16 17:21:32 EDT 1985
Broadcast Message from root  (console)  Thu May 16 17:21:34...
THE SYSTEM IS BEING SHUT DOWN NOW ! ! !
Log off now or risk your files being damaged.
INIT: New run level: 5
The system is coming down.  Please wait.
The system is down.
NOTICE:  Return to Firmware Requested
NOTICE:  System secured for RESET.
```

Step 4: If the machine can be forced to automatically enter the firmware mode (and display the "Power-Up Menu"), it will do so now. If not, you must enter the firmware mode manually. Consult your Delta Series system manual for instructions on how to reach the Power-Up Menu.

Procedure 3.6: Halt and Reboot the Operating System

Purpose	To effect an orderly shutdown and automatic reboot of **/stand/unix**. Normally used after reconfiguring the kernel.
Starting Conditions	System state —multi-user and single-user. You must **mount /usr** to run **sysadm** in single-user mode. Login —**root**, unless you know the **sysadm** password.
sysadm menu	MACHINE MANAGEMENT
Commands	**sysadm reboot**(1) **shutdown**(1M)
Reference	Chapter 6, *Performance Management*

Where supported by the underlying hardware, this procedure allows you to halt the system and reboot from the hard disk automatically. Its main purpose is to boot a new configuration of the operating system (after hardware and/or software modifications). See Procedure 6.1 and Chapter 6, *Performance Management*, for more about reconfiguration.

Step 1: To halt and reboot the system from the hard disk, enter **sysadm reboot** (or **shutdown –i6**):

```
# sysadm reboot
Password:
Running subcommand 'reboot' from menu 'machinemgmt',
MACHINE MANAGEMENT
Once started, a reboot CANNOT BE STOPPED.
Do you want to start an express reboot? [y, n, ?, q] y
Shutdown started.    Thu May 16 17:45:07 EDT 1988
Broadcast Message from root (console) Thu May 16 17:45:09...
THE SYSTEM IS BEING SHUT DOWN NOW ! ! !
Log off now or risk your files being damaged.

INIT: New run level: 6
The system is coming down.  Please wait.
System services are now being stopped.
The system is down.
NOTICE:  System Reboot Requested.
NOTICE:  System secured for RESET.
```

If the machine can be forced to reboot automatically from the hard disk, it will do so now.

```
1) Continue System Start-up
2) Select Alternate Boot Device
3) Go To System Debugger
4) Initiate Service Call
5) Display System Test Errors
6) Dump Memory to Tape
Enter menu #: 3<CR>

181-Bug> ab
Controller LUN = 00?   (Type the correct parameters for your system.)
Device LUN     = 00?
Default string   =  ?
Boot at power-up only    [Y,N]?  N
```

If auto-reboot is not supported, you must manually reset the system. Consult your Delta Series system manual for booting/resetting instructions.

Once reboot has begun, the messages are the same as those that appear when the system is powered up from a halted state. The system is placed in the multi-user state, and you receive the prompt:

```
Console login:
```

P4 Disk Management Procedures

Introduction

The following procedures are covered in this section:

Procedure 4.1 **Format Diskettes**
To prepare diskettes for use and verify they are usable.

Procedure 4.2 **Duplicate Diskettes**
To make exact copies of diskettes.

Procedure 4.3 **Check for Hard-Disk Errors**
To see if disk errors have been logged.

Procedure 4.4 **Set Up Hard Disk**
To format and initialize a hard disk and install a new bootloader.

Procedure 4.5 **Handle Bad Tracks**
To redirect bad tracks, fix file system damage, and update bad tracks.

NOTE

Some variation in these procedures may occur depending on the configuration of your system. If you have a cartridge tape unit or a second disk drive, the **sysadm** menus and prompts will reflect that.

Procedure 4.1: Format Diskettes

Purpose	To format a diskette so it can be used by the system.
Starting Conditions	System state —multi-user or single-user You must **mount /usr** to run this procedure in single-user mode. You must be at the computer to insert and remove the diskettes. Login —an authorized login
sysadm menu	DISK MANAGEMENT
Commands	**sysadm format**(1)
Media	Unformatted diskettes.
Time	Approximately 1 minute per diskette. Approximately 3½ minutes per diskette with verification.
Reference	*Formatting Disks* in Chapter 4, *Disk Management*

Step 1: Enter the following command:

```
$ sysadm format
Password:
```

Step 2: You are prompted to select a drive or to insert the medium to be formatted, and to say whether the format should be verified, for example:

```
Do you want each format verified?[(default: yes)[y, n, ?, q]:y
Insert disk. Press <RETURN> key when ready.
```

Step 3: At the conclusion of the formatting process a message like the following one is displayed on your terminal:

```
The removable disk is now formatted and can be
removed.
```

At this point you can remove the diskette.

Procedure 4.2: Duplicate Diskettes

Purpose	To make a copy of the contents of a diskette.
Starting Conditions	System state —multi-user or single-user You must **mount /usr** to run this procedure in single-user mode. You must be at the computer to insert and remove diskettes. Login —an authorized login
sysadm menu	DISK MANAGEMENT
Commands	**sysadm cpdisk**(1)
Media	A new formatted diskette for each one to be copied.
Time	Approximately 6 minutes per copy (3 minutes to read in to hard disk; 3 minutes to read out to new diskette)
Reference	*Duplicating Diskettes* in Chapter 4, *Disk Management*

With a single diskette drive, the technique for duplicating diskettes is to read the contents into a temporary file from the source diskette, then write it out to a new diskette. If your computer is equipped with two diskette drives, duplication is done drive-to-drive.

Step 1: From the console, enter the command:

```
$ sysadm cpdisk
Password:
```

Step 2: A prompt sequence like this appears on your terminal:

```
Insert the original medium to be copied into the diskette drive.
Press <RETURN> when ready[q]:<CR>
                    (Accepting default by entering <CR>)
            (The sequence below is for a system with one diskette drive.)
The original is being copied in.
Copy in complete
insert a writable medium into the diskette drive.
Press <RETURN> when ready[q]: <CR>
The original is being copied out onto the duplicate medium.
Copy out complete
You may now remove the medium from the diskette drive.
```

Step 3: At this point you may take the diskette out of the drive. You receive the following prompt:

```
To make another copy of the original
insert a writable medium into the diskette1 drive.
Press <RETURN> when ready.  Type q to quit.
```

NOTE

If you want another copy of what is already on the hard disk, you can insert a new diskette into the drive. You do not have to put the first diskette back in the drive. The copy that you put on the hard disk stays there until you quit this procedure. You can make as many copies to as many diskettes as you want, all from the one copy on the hard disk.

If you are through duplicating, you must type **q** to stop the process.

Procedure 4.3: Check for Hard-Disk Errors

Purpose	To check hard disk error report.
Starting Conditions	System state —multi-user or single-user You must **mount /usr** to run this procedure in single-user mode. Login —an authorized login
sysadm menu	SYSTEM DIAGNOSTICS
Commands	**sysadm diskreport**(1)

While new disk errors occur rarely, you should check the disk report regularly. The damage caused by uncorrected errors can be considerable.

Type the command:

$ sysadm diskreport

The following report displays (broken every 23 lines with the prompt `[<return for more>, q]:`).

```
ERROR LOGGING SYSTEM STARTED - Mon Dec 8 05:00:10 1988

        System Profile:
             SYSV/88/3.0 Operating System (node)
             Motorola MC88100 Processor

M320    Error Logged On    Wed Dec 10 08:57:50 1988

        Minor Device Number                          0x10
        Logical Device                               7 (07)
        Device Address                               000F35F4
        Retry Count                                  4
        Error Diagnosis                              Unrecovered
        Simultaneous Bus Activity                          M320

        Registers at Error time
                             0188
                             2000
                             0000
                             0000
                             0005

        Physical Buffer Start Address                0012B570
        Transfer Size in Bytes                           8192
        Type of Transfer                                 Read
        Block No. in Logical File System               130544
        I/O Type                                     Physical


        Statistics on Device to date:
                R/W Operations                       230270
                Other Operations                          0
                Unrecorded Errors                         0
```

```
M320   Error Logged On   Wed Dec 10 13:18:17 1988

       Minor Device Number                    0x10
       Logical Device                         8 (10)
       Device Address                         000F35F4
       Retry Count                            2
       Error Diagnosis                        Recovered
       Simultaneous Bus Activity                    M320

       Registers at Error time
                          0080
                          0400
                          0400
                          0000
                          0005

       Physical Buffer Start Address          000A1C00
       Transfer Size in Bytes                     1024
       Type of Transfer                           Read
       Block No. in Logical File System           1588
       I/O Type                                   Buffered


       Statistics on Device to date:
              R/W Operations                      8565
              Other Operations                       0
              Unrecorded Errors                      0
```

The **Logical Device** and **Block No. in Logical File System** values may be used as input to the **sysadm diskrepair** facility.

Procedure 4.4: Set Up Hard Disk

Purpose	To format and initialize a hard disk and/or install a new bootloader.
Starting Conditions	System state —multi-user or single-user if using a disk other than the primary one. You must boot from tape if using the primary disk. (Not all bootable tapes support **sysadm**.) Login —an authorized login
sysadm menu	SYSTEM DIAGNOSTICS
Commands	**sysadm fmthdisk**(1) **sysadm inithdisk**(1) **sysadm bootloader**(1)
Reference	Chapter 4, *Disk Management*

Formatting a Hard Disk

Formatting the disk always includes an initialization of the bad track information, so **inithdisk** need not be run. If a disk type is not supported or does not use **sysadm**, refer to **dinit**(1M).

Step 1: Type the following command:

```
$ sysadm fmthdisk
Password:

Any time you want to quit, type "q".
If you are not sure how to answer any prompt, type "?" for help,
or see the Administrator's manual.

If a default appears in the question, press <RETURN> for
the default.
```

Step 2: Answer the following questions about your disk:

```
Valid Devices Are:
1.  m323_0
2.  m323_1
3.  m323_2
4.  m323_3
5.  m327_00
6.  m327_10
7.  m327_20
8.  m327_30
9.  m327_1d00
10. m327_1d10
11. m327_1d20
12. m327_1d30
```

(*continued*)

(If you select an MV ME327 or MV ME323 device (Items 5-12 or 13-24), you may not be asked to give the disk size nor the manufacturer.)

```
enter the device (default: 1) [q]:

      Valid Disk Sizes Are:
      1. 40  Meg
      2. 70  Meg

What size is your disk (default: 2)?

      Valid Manufacturers Are:
      1. Toshiba
      2. Micropolis
      3. Fujitsu
      4. Priam

Who manufactures your winchester (default: 2)?
```

(For an MV ME323 device (items 9-12), the following appears)

```
      1. m323182  -CDC 182Mb
      2. m323390  -Wren V

What is the disk type (default: 1)?
```

(For an MV ME323 device (items 5-12), these questions appear)

```
Do you wish to use sector slipping (default:  y) ?

Use the manufacturer's defect list (default: n) ?
```

(For an MV ME327 device only)

```
        1.  m327cdcIII - Wren III
        2.  m327cdcIV  - Wren IV

What is the disk type (default: n)?
```

(For all types)

```
Create root and user file systems (default: n) ?
```

Step 3: If you are: 1) formatting an MVME323 device and have selected to read the manufacturer's defect list off the disk or 2) formatting an MVME327 device, skip to Step 7. A screen similar to the one shown in Step 7 appears.

Now update the badtrack list. If there is not a file in **/etc/badtracks** listing the badtracks for this device (identified as **/etc/badtracks/***device*), one is created; the bad track list contained in the configuration block of the disk block is put into the file. This list is only on the disk if the disk was set up by **sysadm format, sysadm inithdsk,** or **dinit**(1M). The retrieved list is stored in <*Head*> <*Cylinder*> <*BFI*> format (BFI means Bytes From Index).

If a badspot file exists in **/etc/badtracks** for the selected device, you are asked if you would like to delete it, thereby assuring that the current disk copy is used.

```
Delete existing badspot list (default: y) ?
```

You should expect a message similar to one of these if the file is created:

(*182Mb disk, list in config block, FE list is found:*)

```
The file /etc/badtracks/m320_0 has been updated with the
badtrack list, including FE cylinders.
```

(*182Mb disk, NO list in config block, FE list is found:*)

```
The file /etc/badtracks/m320_0 has been updated with
only the list of FE cylinders.
```

(*182Mb disk, list in config block, FE list is NOT found:*)

```
Cannot get list of FE cylinders. The file
/etc/badtracks/m320_0 contains only the list from the
disk.
```

(182Mb disk, NO list in config block, FE list is NOT found:)

```
Cannot get list of FE cylinders. The file
/etc/badtracks/m320_0 contains no bad tracks.
```

(NOT 182Mb disk, list in config block:)

```
The file /etc/badtracks/m360_0 has been updated with the
badtrack list.
```

(NOT 182Mb disk, NO list in config block:)

```
The file /etc/badtracks/m360_0 contains no bad tracks.
```

If you have an MVME323 device formatted by a previous release of the format program, you will be unable to read the list off the disk. You will be instructed to rerun the format menu, selecting the option to read the manufacturer's list from the disk.

Step 4: Continue answering the prompts:

```
Do you want to view the badtrack list, add to the list,
or continue [v,a,c,q]?
```

If you do not want to view or add to the list, select **c** (continue) and go to Step 7.

An MVME320-type device may have a badspot list in one of three formats. If you want to view or add to the list, but the file format is not known, you must enter the format. It will be known if the file was just created through this menu, or if you have already answered this question while in this loop. Once the format is known, you not prompted for it again.

```
        Valid FILE formats are:
        1. <Track>
        2. <Head> <Cylinder>
        3. <Log Device> <Block>
In what format is the file /etc.badtracks/m320_0 (default: 2)?
```

Step 5: To view the badtrack list you must select a format for the display. You will be asked for the format type each time you view the list. This allows you to see it in different forms immediately.

```
        Valid PRINT formats are:
        1. <Track>
        2. <Head> <Cylinder>
        3. <Log Device> <Block>
Dislay file in which format (default: 2)?
```

NOTE

For a MVME323 device, the file format is always <Head><Cylinder><BFI>. You will not be prompted to select a format.

The longer the list, the longer it takes to read the data, convert it, and print it. You will see the message `*** Working ***` while it works. The list will be in five columns, with each column headed by `HD,CYL, TRACK, DEV,BLK`, or `HD, CYL, BFI`. For example:

```
TRACK   TRACK   TRACK   TRACK   TRACK
 1234    5678    2345    3456    4567
 6789    9876    8765    7654    6543
 4321
```

Because <track> and <head><cylinder> numbers cannot be converted into <dev><block> format, you can only view in the latter format if the file is currently formatted as such.

Step 6: If you choose to add to the list for an MVME320 device, you must specify the format in which you will enter the data. The data you enter is converted to the file format before storing. All other uses require data entry in <*Head*><*Cylinder*><*BFI*> form.

If you have previously viewed the file, the default format is the one previously selected. Depending on your last answer, you are prompted as follows:

```
Enter the track numbers, ending with a <.>
{enter a number}
{enter a number}
                                    OR
Enter the track numbers as <Head> <Cylinder>, ending with a <.>
{enter number pairs}
{enter number pairs}
                                    OR
Enter the track numbers as <Head> <Cylinder> <BFI>, (Use -1 for an
unknown BFI field); End with a period <.>:
{enter number triplets}
                                    OR
Enter the track numbers as <logical dev> <block>, ending with a <.>
{enter number pairs}
{enter number pairs}
```

(Expect a confirmation and a repeat option like this:)

```
      The new bad tracks have
      been appended to /etc/badtracks/m320_0
Do you want to view the badtrack list, add to the list,
or continue [v,a,c,q]?
```

Step 7: Confirm input. If it is not correct, enter **c**. You are returned to Step 2.

```
This is the information for the operation:

      operation:              FORMAT
      entered device:         m323_0
      manufacturer:           CDC
      disk size:              182  Meg
      badtrack file:          /etc/badtracks/m323_0
      file format:            <Head><Cyl><BFI>
      device name:            /dev/rdsk/m323_0s7
      device type:            m323182
      sector slip:            NA
      use manuf list:         NA

Do you want to begin operation or change the information  [b,c,q]?
```

NOTE:
The device name and type are not values that were entered.
These values are generated from the device and manufacturer information provided.

Step 8: The procedure now invokes **dinit**. A message similar to the following displays:

```
*** Begin format of /dev/rdsk/m323_0s7
 (At completion:)
      The operation is complete.

      The current bad track list for disk m323_0
      has been preserved in /etc/badtracks/m323_0
```

Initialize a Hard Disk

Initialization entails setting up the bad track for the disk and initially redirecting those bad tracks elsewhere. If new bad tracks develop on an active disk, see Chapter 4 and Procedure 4.3.

Step 1: Type the following command:

```
$ sysadm inithdisk
Password:

Any time you want to quit, type "q".
If you are not sure how to answer any prompt, type "?" for help,
or see the Administrator's manual.

If a default appears in the question, press <RETURN> for
the default.
```

Step 2: Perform Steps 2 through 7 in the section *Formatting a Hard Disk*, above.

Step 3: Confirm input. If it is not correct, enter **c**. You are returned to Step 2.

```
This is the information for the operation:
      operation:              INITIALIZE
      entered device:         m323_0
      manufacturer:           CDC
      disk size:              182  Meg
      badtrack file:          /etc/badtracks/m323_0
      file format:            <Head><Cyl><BFI>
      device name:            /dev/rdsk/m323_0s7
      device type:            m323182
      sector slip:            NA
      use manuf list:         NA

Do you want to begin operation, or change the information [b,c,q]?
```

NOTE:
The device name and type are not values that were entered.
These values are generated from the device and manufacturer information provided.

Step 4: The procedure now invokes **dinit**. A message similar to the following displays:

```
*** Begin initialization configuration on /dev/rdsk/m323_0s7
```

(*At completion:*)

```
      The operation is complete.

      The current bad track list for disk m323_0
      has been preserved in /etc/badtracks/m323_0
```

Install a New Bootloader

Step 1: Type the following command:

```
$ sysadm bootloader
Password:

Any time you want to quit, type "q".
If you are not sure how to answer any prompt, type "?" for help,
or see the Administrator's manual.

If a default appears in the question, press <RETURN> for
the default.
```

Step 2: Answer the following questions about your disk:

```
Valid Devices Are:
1.  m323_0
2.  m323_1
3.  m323_2
4.  m323_3
5.  m327_00
6.  m327_10
7.  m327_20
8.  m327_30
9.  m327_1d00
10. m327_1d10
11. m327_1d20
12. m327_1d30
```

(continued)

(If you select an MV ME327 or MV ME323 device (Items 5-12 or 13-24), you may not be asked to give the disk size nor the manufacturer.)

```
enter the device (default: 1) [q]:

      Valid Disk Sizes Are:
      1. 40  Meg
      2. 70  Meg

What size is your disk (default: 2)?

      Valid Manufacturers Are:
      1. Toshiba
      2. Micropolis
      3. Fujitsu
      4. Priam

Who manufactures your winchester (default: 2)?
```

(For an MV ME323, the following appears:)

```
      1. m323182  -CDC 182Mb
      2. m323390  -Wren V

What is the disk type (default: 1)?
```

(For an MV ME327 device only)

```
       1.  m327cdcIII - Wren III
       2.  m327micro  - Micropolis
       3.  m327cdcIV  - Wren IV
       4.  m327sea40  - Seagate 48Mb
       5.  m327sea80  - Seagate 85Mb
       6.  m327maxtor - Maxtor

What is the disk type (default: n)?
```

Step 3: Confirm input. If it is not correct, enter **c**; you are returned to Step 2.

```
This is the information for the operation:

      operation:              INSTALL BOOTLOADER
      entered device:         m323_0
      manufacturer:           CDC
      disk size:              182  Meg
      badtrack file:          /etc/badtracks/m323_0
      file format:            <Head><Cyl><BFI>
      device name:            /dev/rdsk/m323_0s7
      device type:            m323182
      sector slip:            NA
      use manuf list:         NA

Do you want to begin operation, or change the information [b,c,q]?
```

NOTE:
The device name and type are not values that were entered.
These values are generated from the device and manufacturer information provided.

Step 4: The procedure now invokes **dinit** with **-b** option.

```
*** Begin vmeboot install as bootloader on /dev/rdsk/m323_0s7
```

(*At completion:*)

```
      The bootloader installation is complete.
```

Procedure 4.5: Handling Bad Tracks

Purpose	To redirect bad tracks, fix file system damage, and update bad tracks.
Starting Conditions	System state to redirect bad tracks —single-user. You must boot from tape if damage to root is extensive. (Not all bootable tapes support **sysadm** menus. If yours does not, refer to *Dynamic Bad Track Redirection* in Chapter 4, *Disk Management*) System state to update bad track file or clear **fsck** input files —single-user or multi-user You must mount **/usr** to run this procedure in single-user mode. Login —an authorized login
sysadm menu	SYSTEM DIAGNOSTICS
Commands	**sysadm redirect**(1) **sysadm fixfsys**(1) **sysadm updbadlist**(1) **sysadm delfsckfiles**(1)
Reference	*Dynamic Bad Track Redirection* in Chapter 4, *Disk Management*

If the disk to be repaired contains the **/usr** file system, you must mount **/usr** as **READ-ONLY**:

```
# mount -r /usr
```

The recommended procedure for dynamically redirecting a bad track is:

1. Run **redirect** in a no-write mode to generate **fsck** input files.

2. Run **fixfsys** in check mode to list the files that will be corrupted by the redirection process (some of which may already be corrupted due to the bad track). Write down the names of any files listed as CORRUPT or UNALLOCATED.
3. Back up as many of the above listed files as can be read, either by copying them to another place or using **cpio**.
4. Run **redirect** in write mode to redirect the new bad tracks.
5. Restore the data previously saved. Restore from earlier backups those files that were rendered unreadable due to the bad track.

Dynamically Redirecting Bad Tracks

Prepare file input with track information. If you choose to enter the bad track data interactively, skip to Step 2.

There are three formats that the bad track file may take. The entire file must be in a single format and the same for all types; the entries must be one per line. The following are some format examples:

track numbers:	*head,cylinder* numbers:	*logical device,block* no.:
1501	4 10293	11 1234
123456	6 9182	73 5678
13579		

NOTE

This logical device, block number format is provided because the system error report (**errpt**(1M)) provides these numbers. After a track redirection, the bad track file in **/etc/badtracks** is stored in <*Head*><*Cylinder*><*BFI*> format.

Step 1: Each device should have a corresponding file in **/etc/badtracks** that contains the list of known bad tracks on the disk. The file name is the device name. For example, **/etc/badtracks/m323_0** contains the known bad tracks for device m323_0.

You may add the new bad tracks to this file and use it as an input file, or you may create another file containing only the new bad tracks. The format program will redirect only new track numbers. There are no added restrictions on your new file name or location, but we recommend that it be located in **/etc/badtracks**, using an identifying name such as **tmp.m323_0**.

If you do not update the device badtrack file before the track redirection (for devices other than MVME327), it is updated after redirection to reflect the new bad tracks. If it does not exist, it is created.

Step 2: Type the following command:

```
$ sysadm redirect
Password:

Any time you want to quit, type "q".
If you are not sure how to answer any prompt, type "?" for help,
or see the Administrator's manual.

If a default appears in the question, press <RETURN> for
the default.

 Do you wish to WRITE (default:n)? [y,n,?,q]
       {select your answer}
```

(A no (n) response means the redirection will not take place, but the files to be used as input for **fsck** *will still be created. You will then be able to see what would happen and which files will be affected.)*

Step 3: Answer the following questions about your disk:

```
Valid Devices Are:
1.  m323_0
2.  m323_1
3.  m323_2
4.  m323_3
5.  m327_00
6.  m327_01
7.  m327_02
8.  m327_03
9.  m327_1d00
10. m327_1d10
11. m327_1d20
12. m327_1d30
```

(If you select an MV ME327 or MV ME323 device (Items 5-12 or 13-24), you may not be asked to give the disk size nor the manufacturer.)

```
enter the device (default: 1) [q]:

Valid Disk Sizes Are:
1. 40  Meg
2. 70  Meg
```

(continued)

```
What size is your disk (default: 2)?
      Valid Manufacturers Are:
      1. Toshiba
      2. Micropolis
      3. Fujitsu
      4. Priam
What who manufactures your winchester (default: 2)?
```

(For an MV ME323 device (items 9-12), the following appears:)

```
      1. m323182  -CDC 182Mb
      2. m323390  -Wren V
What is the disk type (default: 1)?
```

(For an MV ME327 device (Items 13-24) .)

```
      1.  m327cdcIII - Wren III
      2.  m327micro  - Micropolis
      3.  m327cdcIV  - Wren IV
      4.  m327sea40  - Seagate 48Mb
      5.  m327sea80  - Seagate 85Mb
      6.  m327maxtor - Maxtor
What is the disk type (default: n)?
```

(For an MV ME323 device (items 5-12), this question appears:)

```
Was device formatted to use sector slip (default: n) ?
```

Step 4: Describe the method of entering track information:

(For all disk types)

```
Will the track information be from a file (default:y)? [y,n,?,q]
```

(If you answered yes (y), you are prompted about the file location and format. The default path name is the file in /etc/badtracks with the device name. If no file exists, the word `none` *will appear in the default field).*

```
Enter full path name of bad track file
       (default:/etc/badtracks/m320_0) [q]:
       {Enter the name here}
```

(For an MVME320 device only)

```
       Valid File Formats Are:
       1. <Track>
       2. <Head> <Cylinder>
       3. <Log device> <Block>
```

(For all other devices)

```
      Valid File Formats Are:
      1. <Head> <Cylinder> <Log Sector>
      2. <Log Device> <Block>
```

```
In what format is the bad track file (default: 2)?
```

(If the input is not from a file (`n` *response above), one of the following messages appears instead.)*

(For an MVME320 device only)

```
       Valid Input Formats Are:
       1. <Track>
       2. <Head> <Cylinder>
       3. <Log device> <Block>
```

(For all other devices)

```
       Valid Input Formats Are:
       1. <Head> <Cylinder> <Log Sector>
       2. <Log Device> <Block>
```

```
 In what format will you enter the data (default: 2)?
```

Step 5: Confirm the information. If it is not correct enter **c**. You are returned to step 3 to fix it. All defaults will be set to your previous choices. The following example is based on default values:

```
This is the information for the track redirection:

        entered device:          m323_0
        manufacturer:            CDC
        disk size:               182  Meg
        badtrack file:           /etc/badtracks/m323_0
        file format:             <Head><Cyl><BFI>
        device name:             /dev/rdsk/m323_0s7
        device type:             m323182
        write:                   n
        sector slip:             NA

Do you wish to begin redirection, or change this data [b,c,q]?
```

NOTES:
1) If the badtrack information is not to be entered via a file, the word **interactive** *appears in the badtrack file field.*
2) The device name and type are not values that were entered. These values are generated from the device and manufacturer information provided.
3) If the device is not a 323, **NA** *appears in the manufacturer and disk size fields.*

Step 6: If there are old **fsck** input files left in **/etc/badtracks** from a previous invocation, they must be removed before continuing. If there are none, skip this step.

```
These fsck input files exist:
        /etc/badtracks/F.m323_0s2
        /etc/badtracks/F.m323_0s5
 delete them (default:y)? [y,n,q] <CR>
```

If you choose not to delete the files, this utility is stopped. In general, once the system is back in use for any length of time, the information within the files is obsolete.

Step 7: If you answered no (**n**) to the following question. You will now be asked to enter the bad track data.

```
Will the track information be from a file (default:y)? [y,n,q]
```

```
Enter bad track numbers; end with a period (.):
       {begin entering numbers}
```

(Depending on the format you chose, the message may appear as either:)

```
Enter bad track numbers; end with a period (.):
                                        OR
Enter bad track numbers as <head> <cylinder>; end with a period (.):
                                        OR
Enter bad track numbers as <head> <cylinder> <BFI>; (Use -1 for an
unknown BFI field); End with a period (.):
                                        OR
Enter bad track numbers as <logical dev> <block>; end with
a period (.):
```

Step 8: Because of the destructive nature of the reformat operation, the bad spots are redirected one at a time, with **fsck** run after each to check for file system damage.

```
Begin redirection of 3/ 248/:
```

Msg printed:

```
fsck will now be run on the disk to repair any
damage done by redirecting 3/ 248/
                                    OR
Track not in filesystem; no file system repair needed.
```

Step 9: After completion of the no-write option, there may be files placed in **/etc/badtracks** that contain file system logical block numbers. These are the file system blocks that are on the bad track. The names of these files are derived from the device name, a partition number, and prefixed with **F.**, for example:

/etc/badtracks/F.m323_0s5
/etc/badtracks/F.m323_0s2

If any of these files are created, you see the following message displays:

```
(For no-write option only)
These fsck input files have been created:

      /etc/badtracks/F.m323_0s2
      /etc/badtracks/F.m323_0s5

You may now select the 'fixfsys' option of this
menu to find out which files are affected by the track
redirection.

(If no files were created, this message appears:)

No file systems will be affected by the track redirection.
```

Step 10: After successful completion of the redirections, the bad track list is updated. A message similar to the following appears:

```
The file /etc/badtracks/m323_0 has been updated to reflect
the bad track list contained on the disk.
```

Fixing (or Checking) the File Systems

NOTE

This procedure checks only those file systems that may have corrupted files due to a track redirection. They are described by each file's name. For example, a file named **F.m323_0s2** causes a check of **/dev/rdsk/m323_0s2**.

Step 1: Type the following command:

```
$ sysadm fixfsys
Password:
```

(A list of the input files existing in **/etc/badtracks** *appears. This is informational only.)*

```
These fsck input files exist

        /etc/badtracks/F.m323_2s3
        /etc/badtracks/F.m323_2s5
```

(If no files exist, a message like this appears:)

```
There are no fsck input files.
No files systems will be checked.
```

Step 2: You are given the option to do the checking for each file system. This allows you to selectively check file systems. For example:

```
Check '/dev/rdsk/m323_0s3' (default: y)? [y,n,q]
```

(You are then asked to choose a type of checking from this display:)

```
Select:
      check (no repairs attempted)
      interactive repair
      automatic repair

[c,i a,q,?]
```

Step 3: Following a yes (**y**) response, file checking proceeds. If you selected **interactive repair** (**i**), and if an error is detected, you receive a message on your screen that describes the error and asks for a yes or no response.

The file system must be unmounted before **fsck** will check it. If it is currently mounted, an attempt is made to unmount it. If it is busy and cannot be unmounted, such as the **usr** file system when in multi-user mode, a message like this appears:

```
Cannot unmount /dev/rdsk/m323_0s1. Unmount the file system
manually before repeating this file system check.
```

If a file is found that may be corrupted due to lost data from the track redirection or that may already be corrupt due to the bad track itself, a message like this appears:

```
FILE/DIR  CORRUPT   I=47  OWNER=bin MODE=100755
SIZE=35497 MTIME=Jun 24 10:07 1986
FILE/DIR = [/dev/rdsk/m323_0s2]/bin/create
```

Note that the file name is relative to the file system described by the device. In the above example, **m323_0s2** is the **usr** file system, so the corrupt file is **/usr/bin/create**.

If any of the file system blocks in **/etc/badtracks/F.m323.0s5** are included in the I-list, a message like the following appears:

```
I-BLOCK CORRUPT - BLOCK=100
INODES 1569-1584 UNRECOVERABLE
```

This is be followed in Phase 2 with an `UNALLOCATED` message for each of the files affected (16 files for a 1K-block file system).

You should write down the names of the files affected (denoted in the **fsck** output as `FILE/DIR CORRUPT` or `UNALLOCATED`) because once the redirection is done, some of that information may be lost.

It may now be possible to copy most of the listed files elsewhere (or use **br** to back them up) before the redirection. The files directly affected by the bad spot on the track are likely unreadable. They must to be restored from a previous backup.

The data files are not automatically cleared. This allows you to repeat the file checking procedure to again see the list of corrupted files.

For a complete list of error messages see *Running* **fsck** in Chapter 5.

There is a slight possibility that a file of affected block numbers was created for an area that looked like a file system but in fact, was not. Such an error is discovered here. You are asked if you want to remove the file that caused this file system to be checked. Unless you have a specific use for it, you should answer yes (**y**).

```
/dev/rdsk/m323_0s6 does not contain a real file system. Remove
the file /etc/badtracks/F.m323_0s6 (default: y) ? y
```

Step 4: If the **root** file system has been affected by the track redirection, it is checked last if the machine is in single-user mode. If there is damage, the file system is repaired and the system rebooted.

If you are not in single-user mode, the following message appears:

```
Not in Single-User mode. Cannot check root /dev/rdsk/m323_0s0
```

Updating Bad Track Files

NOTE

This section does not apply to MVME327 device.

Each device should have a corresponding file in **/etc/badtracks** that contains the list of known bad tracks on the disk. The file name is the device name, for example: **m323_0**, and **m323_1**.

It may be necessary to update this copy of the bad track list for several reasons:

- The current copy was destroyed
- Track redirection was done from a bootable tape
- The list was not kept up to date.

Step 1: Type the following command:

```
$ sysadm updbadlist
Password:
```

Step 2: Answer the following questions about your disk:

```
      Valid devices Are:
         1. m323_1
         2. m323_1d0
         3. m323_1d1
```

(If you select an MV ME323 device (Items 5-12), you may not be asked to give the disk size nor manufacturer.)

```
enter the device  (default: 1) [q]:

    Valid Disk Sizes Are:
    1.  40 Meg
    2.  70 Meg

What size is your disk (default: 2)?

    Valid Manufacturers Are:
    1.  Toshiba
    2.  Micropolis
    3.  Fujitsu
    4.  Priam

Who manufactures your winchester (default: 2)?
```

Step 3: The MVME323 devices have a manufacturer's list on the disk in addition to the one placed there by **dinit**. Choose the list you want to retrieve.

```
Valid options:
1.  Use existing disk bad track file
2.  Use manufacturer's original list
Which bad track list do you want (default: 1)?
```

Step 4: Confirm the information. If it is not correct, enter **c**; you are returned to Step 2 to fix it. The following is an example based on the default values:

```
This is the information for the bad track file update:
      entered device:          m323_0
      manufacturer:            CDC
      disk size:               182  Meg
      device name:             /dev/rdsk/m323_0s7
      device type:             m323182
      use mfr list:            NA
Do you want to begin the update, or change this data [b,c,q]?
```

(The device name and type are not values that were entered. These values are generated from the device and manufacturer information provided.)

Step 5: When the update is complete, a message like one of the following appears:

```
The file '/etc/badtracks/m323_0' has been updated.
                                        OR
There are no bad tracks.
                                        OR
Can't read disk bad track list.
```

(If the disk has not previously been formatted, the read attempt will fail and the menu will be exited. There is no option to scan an unformatted disk.)

Step 6: View the badtrack list:

```
Do you want to view the badtrack list, add to the list,
or continue [v,a,c,q]?
```

If you do not want to view or add to the list, select **c** (continue) and go to Step 7.

To view the badtrack list for an MVME320 device, you must select a format for the display. You are asked for the format type each time you view the list. This allows you to see it in different forms immediately. All other devices displays the data in *<Head><Cylinder> <BFI>* form.

```
        Valid PRINT Formats are:
        1. <Track>
        2. <Head> <Cylinder>
        3. <Log Device> <Block>
Display file in which format (default: 2)?
```

The longer the list, the longer it takes to read the data, convert it, and print it. You will see the message `*** Working ***` while it works. The list is in five columns, with each column headed by `HD,CYL,TRACK,DEV,BLK,BFI` or `HD,CYL,BFI` as shown in the following example.

```
TRACK   TRACK   TRACK   TRACK   TRACK
 1234    5678    2345    3456    4567
 6789    9876    8765    7654    6543
 4321
```

Because <Track> and <Head><Cylinder> numbers are not converted into <Dev> <Block> format, you can only view in the latter format if the file is currently formatted as such.

Step 7: If you choose to add to the badtrack list for an MVME320 device, you must specify the format in which you enter the data. The data you enter is converted to the file format before storing. All other devices require data entry in *<Head><Cylinder><BFI>* form. You are prompted as follows:

```
        Valid INPUT Formats are:
        1. <Track>
        2. <Head> <Cylinder>
        3. <Log Device> <Block>
In what format will you enter the data (default: 2)?
```

Depending on your last answer, you are prompted as follows:

```
Enter the track numbers, ending with a <.>
{enter a number}
{enter a number}
                        OR
Enter the track numbers as <Head> <Cylinder>, ending with a <.>
{enter number pairs}
{enter number pairs}
                        OR
Enter bad track numbers as <Head><Cylinders><BFI>; (Use -1 for
an unknown BFI field); end with a period (.):
{enter number triplets}
{enter number triplets}
                        OR
```

Continued

```
Enter the track numbers as <logical dev> <block>, ending with a <.>
{enter number pairs}
{enter number pairs}
```

(Expect a confirmation and a repeat option like this:)

```
      The new bad track have
      been appended to /etc/badtracks/m320_0

Do you want to view the badtrack list, add to the list,
or continue [v,a,c,q}?
```

If, for some reason, the data could not be collected, translated or entered into the file, a message like this appears:

```
Unable to get new input for the bad track file.  The
file /etc/badtracks/m323_0 remains unchanged.
```

Step 8: When the update is complete, the following message appears:

```
The operation is finished.
```

Clearing fsck input files

After a track redirection there may be files placed in **/etc/badtracks** (which contain file system block numbers) meant as data files for **fsck**. In general, once the system is back in use for any length of time, the information within the files is obsolete.

Each invocation of the track redirection program clears the files. You may wish to do this yourself, however, if the files are no longer needed and you want to free space.

The files of concern here are those whose names are derived from the device name, a partition number, and prefixed with **F.**. For example:

/etc/badtracks/F.m323_0s5
/etc/badtracks/F.m323_1s5

Step 1: Type the following command:

```
$sysadm delfsckfiles
Password
(The file names are listed. You must confirm the request before they are deleted.)
These fsck input files exist:
   /etc/badtracks/F.m323_0s5
   /etc/badtracks/F.m323_1s5
delete them (default:y)? [y,n,q] <CR>
(If no files exist, a message like this appears:)
There are no fsck input files to clear.
(At completion:)
The operation is finished.
```

P5 File System Administration Procedures

Introduction

The following procedures are covered in this section:

Procedure 5.1 **Create File Systems on Diskette**

Procedure 5.2 **Create File Systems on Hard Disk**
To define additional file systems when more than one disk device is available.

Procedure 5.3 **Maintain File Systems**
To check and possibly repair file systems.
To monitor disk space usage.
To reorganize disk space.
To compress file systems using tape.

Procedure 5.4 **File System Backup and Restore**
To provide a storage copy of active files.
To archive unneeded files.
To bring files and file systems back from storage.

Procedure 5.5 **Alternate File System Backup and Restore**
To provide a storage copy of active files.
To archive unneeded files.
To bring files and file systems back from storage.

Procedure 5.6 **Archive Description Change**
To describe archive devices in preparation for backup and restore.

NOTE

Some variation in these procedures may occur depending on the configuration of your system. If you have a second diskette drive, the **sysadm** menus and prompts reflect it exists.

Procedure 5.1: Create File Systems on Diskette

Purpose	To define file systems that are removable for reasons of privacy or security. To write identifying labels on the magnetic medium so the system can know what is brought on line. To bring a file system under operating system control (**mount**), or to release it so it can be removed from the system (**unmount**).
Starting Conditions	System state —(multi-user) or (single-user) You must **mount /usr** to run this procedure in single-user mode. You must be at the computer to insert and remove diskettes. Login —an authorized login or knowledge of the **sysadm** password
sysadm menus	DISK MANAGEMENT
Commands	**sysadm diskmgmt**(1) **sysadm makefsys**(1) **makefsys**(1M) **sysadm mountfsys**(1) **sysadm umountfsys**(1) **mountfsys**(1M) **umountfsys**(1M)
Media	One formatted diskette for each file system to be created. A diskette with a file system that needs to be mounted.
Time	Approximately 3½ minutes per diskette.
Reference	Chapter 5, *File System Administration*

Before doing this procedure, make sure that the diskettes you plan to use have been formatted and are not write-protected (see Procedure 4.1).

In this procedure you are prompted to name a directory that will be the mount point for a file system. Inform users not to keep other files in that directory.

Step 1: Enter one of the two commands below.

```
$ sysadm makefsys
Password:

     or

$ makefsys
Password:
          (Note that these two passwords may be different.)
```

Step 2: You are prompted for further information:

```
Insert the medium in the diskette drive. Press <RETURN> when ready:<CR>
Enter the label to be put on the medium[?, q]? fsys01
```

(This writes a label on the magnetic medium. The name should be six characters or less. A paper label with the same file system name should be affixed to the front of the diskette protective envelope.)

```
Enter the file system name[?, q]? dirfs1
```

(This makes a directory that will be used as the mount point for the file system. The name should be six characters or less.)

```
Enter the maximum number of files and directories on this medium
(default 144)[q]: <CR>
```

(The default for 1.2 Mbyte diskettes is 304.)

```
Building 'dirfs1' file system on 'fsys01'.
Initializing 'dirfs01' file system.
Do you want to leave 'dirfs01' mounted?[y, n, ?, q]: y
/dirfs1 mounted.  DO NOT REMOVE THE MEDIUM UNTIL IT IS UNMOUNTED!
```

The names and other responses used in the above scenario are arbitrary. Use names and responses appropriate for your situation.

Step 3: If you want to mount a file system that has already been created, you may select **mountfsys** from the **sysadm diskmgmt** menu or invoke the **mountfsys**(1M) command. The procedure is:

```
$ sysadm mountfsys
Password:
```

or

```
$ mountfsys
Password:
```

(Note that these two passwords may be different.)

You are prompted for further information:

```
Running subcommand "mountfsys" from menu "diskmgmt"
DISK MANAGEMENT

Insert the medium in the diskette1 drive. Press <RETURN> when ready, [q]
Disk 'fsys01', file system '/dirfs1', mount it? [y, n, q] y
/dirfs1 mounted.  DO NOT REMOVE THE MEDIUM UNTIL IT IS UNMOUNTED!
#
```

Step 4: If you want to unmount a file system that is mounted on the system, you may select **umountfsys** from the **sysadm diskmgmt** menu or invoke the **umountfsys**(1M) command. Do not use **umount**(1M) if the file system was mounted with **mountfsys**.

Procedure 5.2: Create File Systems on Hard Disk

Purpose	To define additional file systems on a second hard disk device: —to give a user group a dedicated portion of the disk space —to balance the distribution of data on the disk To write identifying labels on the magnetic medium so the system can know what is being brought on line. To bring a file system under operating system control (**mount**), or to release it so it can be removed from the system (**unmount**).
Starting Conditions	System state —(single-user) Login —**root**
Commands	**labelit**(1M) **mkfs**(1M) **mkdir**(1) **mount**(1M) **umount**(1M) **sledit**(1M)
Media	A second hard disk or an unused slice on the primary hard disk.
Time	Approximately 3 minutes for **mkfs**.
Reference	*Using* **mkfs** in Chapter 5, *File System Administration*

Your computer was probably delivered with the file systems **root** and **/usr** already defined on the primary hard disk device. If you need to use this procedure, obtain the following information before you begin:

- The name of the *special* device file that is to contain the file system.
- The number of blocks (and optionally, i-nodes) the file system is to have.

You may package some of this information, plus details of directories and files to be copied to the new file system, in a prototype file. See **mkfs**(1M) in the *System Administrator's Reference Manual* for details.

The procedure includes the **mount** and **umount** commands, which are used to make file systems available for use or to remove them from use. Because file systems can be automatically mounted and unmounted during the startup and shutdown procedures, you seldom have to use these commands.

Step 1: Log in as **root**.

Step 2: Make a directory to use as the mount point for the new file system.

NOTE

The default blocksize for file systems created with **mkfs** is 4K. See the **mkfs**(1M) manual entry for information on how to create other types of file systems.

```
# mkdir /usr2
        (/usr2 is the name of the directory where
        the new file system will be mounted.)
```

Step 3: Use the **mkfs** command:

```
# mkfs /dev/rdsk/m323_1s0 26880:5000 1 324
```

The arguments on the command line are:

/dev/rdsk/m323_1s0
the name of the device on which the file system resides

26880:5000
the number of physical blocks and i-nodes

1 324
the interblock gap and blks/cylinder *(see Appendix A for recommended values for your disk.)*

Step 4: After you enter the command, you receive a prompt like this:

```
Mkfs: /dev/rdsk/m323_1s0?
(DEL if wrong)
```

The command waits for 10 seconds before proceeding. If anything on the command line looks incorrect, you have a chance to cancel the command by hitting the DELETE key.

Step 5: **mkfs** next reports some of the attributes of the file system:

```
bytes per logical block = 1024
total logical blocks = 13440
total inodes = 4992
gap (physical blocks) = 1
cyl size (physical blocks) = 324
mkfs: Available blocks = 26566
#   (root prompt)
```

The figures that appear on your screen may not correspond exactly to those shown.

Step 6: Label the file system (to eliminate warnings from **mount**). The label should match the first six characters of the last component of the mount point directory.

```
#   labelit /dev/rdsk/m323_1s0 usr2 1s0
Current fsname:   , Current volname:   , Blocks: 26880, Inodes: 5000
FS Units: 1Kb, Date last modified: Thu Jan 8 11:53:04 1987
NEW fsname = usr2, NEW volname = 1s0 --DEL if wrong!!
#
```

The command waits for 10 seconds before proceeding. If anything on the command line looks incorrect, you have a chance to cancel the command by pressing the DELETE key.

Step 7: Mount the new file system with the following command:

```
# mount /dev/dsk/m323_1s0 /usr2
```

(The arguments on the command line are **/dev/dsk/m323_1s0**, *the name of the device on which the file system resides, and* **/usr2**, *the mount point directory.)*

Step 8: Make a directory in the new file system called **lost+found**; create a few files in the directory and remove them.

```
# /etc/mklost+found /usr2
```

The **lost+found** directory is used by the file system checking utility, **fsck**. Adding and removing files sets up some available directory entries to which **fsck** can assign lost files. The recommended number of empty entries for the **lost+found** directory is roughly 1/50th of the i-node count for the file system. The **mklost+found** utility follows this rule of thumb.

Step 9: The file system is now available for use. If the file system is to be used regularly, it should be added to **/etc/fstab**. See Chapter 3 for details. You may also use this procedure to define file systems on diskettes.

Procedure 5.3: Maintain File Systems

Purpose	To check and possibly repair file systems on removable media so the integrity of the file system is assured. To be informed on how disk space is being used so adequate resources can be provided to users.
When Performed	Before mounting the file system. On a schedule appropriate for your circumstances.
Starting Conditions	System state —(multi-user) or (single-user) For checking, the file system must *not* be mounted. For other maintenance, the file system must be mounted. To run this procedure in single-user mode, you must **mount/usr** You must be at the computer to insert and remove the media.
sysadm menus	FILE MANAGEMENT DISK MANAGEMENT
Commands	**sysadm checkfys**(1) **checkfsys**(1) **sysadm diskuse**(1) **sysadm fileage**(1) **sysadm filesize**(1)
Media	Diskette that contains file system to be checked.
Time	Approximately 3½ minutes per file system.
Reference	*Maintaining a File System* in Chapter 5, *File System Administration*

File System Checking

Step 1: Insert the diskette into its drive and close the latch.

Step 2: Enter the command:

```
$ sysadm checkfsys
Password:

     or

$ checkfsys
Password:
        (Note that these two passwords may be different.)
```

Step 3: You are prompted to select the appropriate drive and press **RETURN** when ready.

Step 4: You are then asked to choose a type of checking from this display:

```
Disk `fs1.v1', file system `/fsys01'
Select:
   check              (Any errors detected are reported, but not fixed.)
   interactive        (You are asked to approve/disapprove fixes.)
   automatic          (Any errors detected are automatically fixed.)
[c, i, a, q, ?]:
```

Step 5: Following your response, file checking proceeds. If you selected **interactive,** and if an error is detected, you receive a message on your screen that describes the error and asks for a **yes** or **no** response.

NOTE

The error messages are in Chapter 5 under *Running* **fsck.**

Step 6: When the check is completed, a message appears similar to the following:

```
27 files 94 blocks 1106 free
You may now remove the medium from the diskette drive.
```

The operation is complete.

Monitoring Disk Usage

This second part of the file system maintenance procedure involves various ways of making sure that enough space is available on hard disk to accommodate the users needs.

Step 1: Enter the command:

```
$ sysadm diskuse
Password:
```

5

Step 2: This display appears on your screen:

```
FILE SYSTEM USAGE AS OF 05/04/86  13:23:30
File     Free     Total    Percent
System   Blocks   Blocks   Full
------   ------   ------   -------
/        6810     26880    75%
/usr     15448    84480    82%
```

Step 3: If a more detailed look at files is indicated, you can use one of the following commands:

```
$ sysadm fileage

  and

$ sysadm filesize
```

Step 4: The **fileage** command causes you to be prompted for two pieces of information:

1. The full pathname of the directory to search

   It is important to be specific in your response. If you select a high-level directory, such as **/usr**, you receive more information than you want.

2. The number of days to go back

   The default is 90 days.

Step 5: **filesize** displays information on the *n* largest files (default is 10) in a directory named by you.

The heading for the information displays:

```
                file size          date of
owner           (characters)       last access          filename
-----           ------------       -----------          --------
```

(The information in the display depends on your application.)

Procedure 5.4: File System Backup and Restore

Purpose	To store file systems or parts of file systems: —to guard against loss of data —to free up space on the disk
When Performed	On a schedule developed to fit the needs of your system.
Starting Conditions	System state —(single-user) You must **mount /usr** to run this procedure in single-user mode. You must be at the computer to insert and remove the media. Login —**root**
sysadm menu	FILE MANAGEMENT
Commands	**sysadm backup**(1) **sysadm store**(1) **sysadm restore**(1) **sysadm bupsched**(1) **mount**(1M)
Media	Tapes in enough quantity to hold the files or file systems you are backing up. Cartridge tapes hold about 300,000 512-byte blocks.
Time	Approximately 25 minutes per cartridge tape.
Reference	*File System Backup and Restore* in Chapter 5, *File System Administration*

Complete Backup

Step 1: Login as **root**.

Step 2: Take the system to the single-user mode (run-level S or 1; see Procedure 3.3, *Shutdown to Single-User*).

Step 3: The System Administration Menu package resides in **/usr** so you must **mount** that file system:

```
# mount /dev/dsk/m323_0s2 /usr
```

Step 4: Enter the command:

```
# sysadm backup
```

Because you logged in as **root**, you are not prompted for the **sysadm** password. The following screen displays:

```
Running subcommand 'backup' from menu 'filemgmt',

FILE MANAGEMENT

Available file systems:
/ /usr /usr2 ALL
Enter file system(s) you want to backup [q, ?]: /usr2
Select complete or incremental backup [c, i, q, ?]: c
Print each file name as it is copied? [q, ?] [y, n, q, ?]: y

Select which drive to use:
   1 ctape1        2 diskette1
Enter a number, a name, the initial part of a name, or
? for HELP, q to QUIT:
```

(Select the storage medium to be used. In the example, we have specified cartridge tapes. Select **ctape1** *for complete and incremental backups.)*

```
Before inserting the tape into the drive, mark it as follows:
        Complete Backup of /usr2,
        Sat. 09/08/84, 05:08:31 AM
                part 1
```

(continued)

```
Insert the medium in the ctape drive.
Press <RETURN> when ready. [q] <CR>
..... (If you asked for file names to be printed, they will appear here.)
Reached end of medium on output.
Remove medium.
Before inserting the next part into the drive, mark it:
        Complete Backup of /usr,
        Sat. 09/08/84, 05:08:31 AM
                part 2
Insert the tape.
Press <RETURN> when ready. [q] <CR>
..... (If you asked for file names to be printed, they will appear here.)
Reached end of medium on output.
  .
  .
  .
  .
  .
  .
            (Process continues until all data are copied
            to the tapes.)
  .
  .
  .
Reached end of medium on output.
Remove medium.
Before inserting the next part into the drive, mark it:
        Complete Backup of /usr,
        Sat. 09/08/84, 05:08:31 AM
                part n
Insert the tape.  Press <RETURN> when ready. [q] <CR>
..... (If you asked for file names to be printed, they will appear here.)
13089 blocks
Complete backup of  /usr2  finished.
You may now remove the medium.
#
```

Step 5: Label each tape used for the backup. Include a sequence number as part of the label (e.g., Volume 1, Volume 2).

Step 6: Return the system to the normal operating configuration (see Procedure 3.4, *Return to Multi-User*).

Incremental Backup

Incremental backups can be used after an initial full backup has been done. See the discussion on backup strategies in Chapter 5 under *File System Backup and Restore.*

Step 1: Login as **root.**

Step 2: Take the system to the single-user mode (run-level S or 1).

Step 3: The System Administration Menu package resides in **/usr** so you must **mount** that file system:

```
# mount /dev/dsk/m323_0s2 /usr
```

Step 4: Execute the following System Administration backup command (**sysadm backup**) and follow the displayed instructions:

```
# sysadm backup
Running subcommand 'backup' from menu 'filemgmt',
FILE MANAGEMENT

Available file systems:
/ /usr /usr2 ALL
Enter file system(s) you want to backup [q, ?]: /usr2<CR>
Select complete or incremental backup [c, i, q, ?]: I<CR>
Print each file name as it is copied? [q, ?] [y, n, q, ?]: y<CR>

Select which drive to use:
   1 ctape1        2 diskette1
Enter a number, a name, the initial part of a name, or
? for HELP, q to QUIT:
```

(Select the storage medium to be used. Select **ctape1** *for complete and incremental backups.)*

```
Before inserting the tape into the drive, mark it as follows:

         Incremental Backup of /usr2,
         Sat. 09/08/84, 05:08:31 AM to
         Sat. 09/08/84, 07:04:21 AM
                Part 1
Insert the medium in the drive.  Press <RETURN> when ready. [q] <CR>
/usr2/abc/file01
    .
    .
    .
83 blocks

Incremental backup of  /usr2  has finished.
You may now remove the medium.
#
```

Step 5: Label the medium.

Step 6: Return the system to the normal operating configuration (see Procedure 3.4, *Return to Multi-User*).

Selective Backup

Step 1: Login as **root**.

Step 2: Take the system to the single-user mode, run-level S or 1 (see Procedure 3.3).

Step 3: The System Administration Menu package resides in **/usr** so you must **mount** that file system:

 # mount /dev/dsk/m323_0s2 /usr

Label a formatted tape to indicate the directory or file.

Step 4: Insert the formatted tape in the cartridge tape drive.

Step 5: Enter the command:

 # sysadm store

The display on your terminal is:

```
Running subcommand 'store' from menu 'filemgmt',
FILE MANAGEMENT

            (Prompt for device)

  1. Select a single file for storing
  2. Select all files under a directory for storing
Enter a number: 1
Enter full path name of file to be stored [q]:
      /usr/abc/file1

  1. Select a single file for storing
  2. Select all files under a directory for storing
  3. List files selected so far
  4. Store selected files
Enter a number: 4

  1 files selected
```

Step 6: Each time you provide a file or directory name, you are prompted to enter more names, to review what has been entered, or to proceed with the storing process. When you have entered all the names of files to be stored, enter **4**; the following displays:

```
Files stored on:
Sat 05/04/85, 03:18:53 PM
     part 1
/usr/abc/file1
28 blocks
    .
    .
    .
Store complete.
Do you want to verify that your file(s) were stored properly
[y,n,q,?] y<CR>
PLEASE NOTE:
To verify that the store worked properly, you must
re-insert all parts that were just written to, starting with "part 1"
Insert the medium in the tape drive. Press <RETURN> when ready. <CR>
Verification complete. You may remove the medium.
Should the stored files be removed from the built-in disk[y,n,q,?]
```

This last question gives you a chance to remove the files just written to tape. You would probably elect to do that if you were in the process of freeing up storage space.

Step 7: Return the system to the normal operating configuration.

Restore

Step 1: Login as **root**.

Step 2: Take the system to the single-user mode (run-level S or 1).

Step 3: The System Administration Menu package resides in **/usr** so you must **mount** that file system:

```
# mount /dev/dsk/m323_0s2 /usr
```

Step 4: Enter the command and follow the displayed instructions:

$ sysadm restore

when restoring from a complete or incremental backup, all tapes of that series must be loaded. Even if you intend to restore only a single file, all tapes of the backup series must be loaded in sequence.

```
# sysadm restore
Running subcommand `restore' from menu `filemgmt',
FILE MANAGEMENT
Select which drive to use
   1 ctape1   2 diskette
   Enter a number, a name, the initial part of a name, or
   ? for HELP, q to QUIT: 1
   Select:
           1. restore a single file
           2. restore a directory of files
           3. restore all files
           4. list all the files
   Enter a number [q,?]: 1
   Insert the medium in the ctape drive.
   Press <RETURN> when ready. [q]<CR>
   Enter full path name of file(s) to be restored [q, ?]: /usr/abc/file1
   Do you want to rename the file as it is copied in? [y, n, q]: y
   WARNING:
     Be very careful when you rename a file.  Files incorrectly named
     by typing errors are difficult to find and repair.
     Remember that only the first 14 characters of each part of the
     file name (i.e., the characters between the "/"s) are significant.
```

(You will be asked to rename each file in turn. An empty response (<CR>) skips that file. An answer of period (.) restores the file with its original name.)

```
   Rename </usr/abc/file1>
   /usr/abc/datafile3
       .
       .
       .
   83 blocks
```

(continued)

(continued)

```
Restoration complete.
You may now remove the medium from the tape drive.
Select:
        1. restore a single file
        2. restore a directory of files
        3. restore all files
        4. list all the files
Enter a number [q,?]: q

#
```

Step 5: Return the system to the normal operating configuration (see Procedure 3.4, *Return to Multi-User*).

Step 6: Store the backup media in a safe place.

Procedure 5.5: Alternate File System Backup and Restore

Purpose	To store file systems or parts of file systems. —to guard against loss of data —to free up space on the disk
When Performed	On a schedule developed to fit the needs of your system.
Starting Conditions	System state —(single-user) You must **mount /usr** to run this procedure in single-user mode. You must be at the computer to insert and remove the media. Login —**root** Login —**br** Login —**sysadm backup_rest**
sysadm menu	FILE MANAGEMENT
Commands	**sysadm backup_rest**(1) **br**(1M) **mount**(1M)
Media	Formatted diskettes or tapes in enough quantity to hold the files or file systems you are backing up. A minimum of one piece of media per file system for a full backup. Diskettes hold about 1200 512-byte blocks; tapes about 120,000.
Caution	Creating backups in a run level other than single user can make it impossible to recover files from an archive.
Time	Approximately 6 minutes per diskette. Approximately 35 minutes per tape.
Reference	*File System Backup and Restore* in Chapter 5, *File System Administration*

Full Backup

Step 1: Log in as **root**.

Step 2: Take the system to the single-user mode (run level S; see Procedure 3.3, *Shutdown to Single-User*).

Step 3: The System Administration Menu package resides in **/usr** so you must **mount** that file system as shown below.

```
# mount /usr
```

Step 4: Enter the command:

```
# sysadm backup_rest
```

or

```
# br
```

Because you logged in as **root**, you are not prompted for the **sysadm** or **br** passwords.

The following screen displays:

```
Backup & Restore functions are:
  1. INCREMENTAL backup
  2. FULL backup
  3. SELECTED backup
  4. RESTORE files from archive
  5. ARCHIVE description change
  6. PREFERENCES
  0. QUIT this level
-- Your choice? [0-6]
```

NOTE

Carefully read through this entire procedure and *File System Backup and Restore* in Chapter 5 before attempting a backup or restore.

Step 5: Type **2** to access the FULL backup menu:

```
Last full backup completed:
          date, at time

File-Systems not yet backed up are:
        /usr
        /

-------------------------
Full backup functions are:
  1. ESTIMATE archive requirements for '/usr'
  2. BACKUP '/usr' now
  0. QUIT this level
-- Your choice? [0-2]
```

Step 6: Type **1** for an estimate of the archive space required for a given file system. A full backup entails backing up each file system in turn onto its own archive volume(s), thereby allowing each file system to be recovered individually in the future.

Step 7: Type **2** to begin the full backup. The name of the first file system appears in the menu items and automatically changes as file systems are successfully backed up. As file systems are backed up, they disappear from the "not yet done" list and appear in a "done" list. If a file system is not backed up (e.g., a bad tape read during verification), then its name is rotated to the end of the "not yet done" list. This allows you to try again to back up file systems that failed the first time. When all file systems have been successfully backed up, the starting time and date of the full backup are saved, so that future incrementals can save files modified since that date.

If you decide to quit (by typing 0) before all file systems are backed up, the list of remaining file systems is saved, and are read when you next enter full backup from the top menu. You can ignore this partial list and backup all the file systems if you want. You are asked about this when you first enter full backup, but only if the previous full backup was not completed. An incomplete full backup does not change the reference time for incremental backups. Each time you resume a previous incomplete full backup, the old starting time is the potential new reference point, since incrementals are taken for all selected file systems at once, using this reference point.

If you quit before all volumes of a multivolume archive are complete, you must begin at volume 1 again the next time you try to back up the file system.

Step 8: Label each diskette or tape used for the backup. Include a sequence number as part of the label (Volume 1, Volume 2).

Step 9: Return the system to the normal operating configuration (see Procedure 3.4, *Return to Multi-User*).

Incremental Backup

Incremental backups can be performed after an initial full backup has been done. See the discussion on backup strategies in Chapter 5 under *File System Backup and Restore.* A separate piece (or set of pieces) of media should be reserved for each incremental backup between full backups, thus, maintaining a full (daily) record of all changes since the last full backup.

Step 1: If you are only scheduling an incremental backup, or you do not wish to take the system to single-user mode (because you know that the system is quiescent), log in as (or execute) **sysadm backup_rest** or **br** (both commands require a password) and skip to Step 5. Otherwise, log in as **root** and proceed to Step 2.

Step 2: Take the system to the single-user mode (run level S).

Step 3: The System Administration Menu package resides in **/usr** so you must mount that file system:

```
# mount /usr
```

Step 4: Type:

```
# sysadm backup_rest
```

or

```
# br
```

Step 5: Type **1** in response to the **backup_rest** main menu. The following screen displays:

```
Last incremental completed:
          date, at time

Last full backup completed:
          date, at time

Incremental backup functions are:
  1. DIRECTORY other than '/'
  2. ESTIMATE archive requirements for '/'
  3. SCHEDULE an unattended backup for '/'
  4. BACKUP '/' now
  0. QUIT this level
-- Your choice? [0-4]
```

Step 6: Choose the proper menu item:

Type **1** to set up for an incremental backup on an area other than the entire system. When the directory is "/", only certain file systems will be examined for changed files. See *Describing Your System* in Chapter 5 for how to identify file systems for incremental backup. For directories other than "/", that entire directory tree is examined, including any mounted (local or remote) resources.

Type **2** to find out how many archive volumes the backup will use, how many files will be backed up, and how many kilobytes (1024 bytes) the archive will occupy. This may take many minutes.

Type **3** to schedule an incremental backup to occur automatically at a pre-set time in the next 24 hours (see *Preparing for File System Backup and Restore* in Chapter 5 to change the preset time). If an estimate has not been done this day, one will be performed automatically now before the incremental is scheduled.

Type **4** to incrementally back up the chosen directory. If the directory is "/", this usually incrementally backs up those subdirectories (file systems) that have had a full backup.

If you have a particularly large incremental backup, you may need multiple archive volumes (unless the backup is scheduled, in which case it must fit on one volume). If you estimate before you do the actual backup, you will know how many volumes to use. Be sure you have enough on hand before you start the backup, otherwise you may have to quit in the middle, which invalidates all previous volumes.

Step 7: Label the medium.

Step 8: Return the system to the normal operating configuration (see Procedure 3.4, *Return to Multi-User*).

Selected Backup

Step 1: If you know that the system is quiescent, and you do not wish to take the system to single-user mode, log in (or execute) **sysadm backup_rest** or **br** (both commands require a password) and skip to Step 5. Otherwise, log in as **root** and proceed to Step 2.

Step 2: Take the system to the single-user mode, run level S (see Procedure 3.3).

Step 3: The System Administration Menu package resides in **/usr** so you must mount that file system:

```
# mount /usr
```

Step 4: Type the command:

```
# sysadm backup_rest
```

Step 5: Type **3** in response to the **backup_rest** main menu. The following screen displays:

```
A backup now would select every file.   (current backup selection criteria)
------------------------------
Selected-backup functions are:
  1. CHANGE selection criteria
  2. USE existing RESTORE criteria
  3. DIRECTORY other than '/'
  4. ESTIMATE archive requirements for selected files
  5. BACKUP selected files now
  0. QUIT this level
-- Your choice? [0-5]
```

Step 6: Type `1` to change the current selection criteria.

When you first enter the **backup_restore** main menu, any previous selection criteria are erased. However, any criteria you enter during the session are retained throughout the session, even if you quit the selected backup menu.

You can perform several selected backups, changing the criteria and directory between backups. To leave the selected backup function, you must explicitly choose the QUIT option.

Currently, you must enter the selection criteria as ordinary **bru**(1) options.

Step 7: Follow the instructions under *Selected Backup* in Chapter 5, *File System Administration* or see **bru**(1) in the *User's Reference Manual* for information on **bru** options.

Step 8: Return the system to the normal operating configuration.

Restore

Step 1: If you know that the system is quiescent, and you do not wish to take the system to single-user mode, you may log in (or execute) **sysadm backup-restore** or **br** (both commands require a password) and skip to Step 5. Otherwise, log in as **root** and go to Step 2.

NOTE

When doing a complete restore of **root** or **usr**, the system should be in single-user mode (with the **usr** file system mounted).

Step 2: Take the system to the single-user mode (run level S).

Step 3: The System Administration Menu package resides in **/usr** so you must mount that file system:

```
# mount /usr
```

Step 4: Enter the command:

 # sysadm backup_restore

Step 5: Type **4** in response to the **backup_restore** menu. The following screen displays:

```
A restore now would select every file.   (current restore selection criteria)

----------------------
Restore functions are:
  1. WHOLE archive view
  2. SELECTED archive view
  3. VERIFY an existing archive
  4. CHANGE selection criteria
  5. USE existing SELECTED-BACKUP criteria
  6. DIRECTORY other than '/'
  7. EXAMINE archive header and EXTRACT recovery info
  8. FILE-SYSTEM REBUILD from recovery info
  9. RESTORE selected files from archive to '/'
  0. QUIT
-- Your choice? [0-9]
```

Step 6: Choose the proper menu item:

Type **1** or **2** to list files on the archive. The selected view (**2**) is useful to preview the effect of the restore selection criteria.

Type **3** to check the internal consistency and general readability of an archive. If you suspect that an archive volume may be damaged or worn, use this function to see if it is readable. Even if some blocks are damaged, the archiver will try to regain synchronization and read good blocks following the damaged ones.

In the same way that files can be selectively backed up, they can also be selectively restored. This is useful when only a single file system must be recovered from an incremental backup, or when only part of a file system must be recovered from any type of backup. To restore a particular file system from incremental, the base directory should be "/", and the selection criteria should select the name of the file system directory with a leading "./" (e.g., **./usr**). Files can be restored into any base directory (not just their original location). You can also use this method to move entire user hierarchies around, even between machines.

Type **4, 5,** or **6** to set the selection criteria and base directory (see **bru** *Options* in Chapter 5 or **bru**(1) in the *User's Reference Manual*).

Type **7** to examine the header of volume 1 of an existing archive and perhaps extract some recovery information from it. By default, archive header labels have information about the type of backup (full, incremental, or selected) and the base directory from which the backup was taken. Full backups also include file system recovery information that can be used with menu item 8 to rebuild completely a file system. The header information displays on the terminal, and the label also has the information in it extracted. If the label has recognizable information, then it is used; otherwise, the previous recovery information is kept. If the information is used, it automatically changes the base directory, which appears in the menu items offered at the next chance for input. In this way you can read the header of an archive and have the base directory automatically set for the proper recovery of the archive (using item 9). Item **7** should be selected before restoring a file system from a full backup (or a directory from a selected backup).

Type **8** if the file system has been completely destroyed, or if you want to transfer a complete file system to an unused device on your machine or on another machine. This option uses recovery information written in the archive header during a full backup to build a new file system (using **mkfs**) that is the same size as the original. This action destroys any data already in an existing file system. The selection criteria are then automatically set so that a RESTORE operation will restore the entire file system.

Type **9** to read the files from the archive and place them in the base directory and its descendants. Subdirectories are made as necessary. The RESTORE criteria are different from those used in selected backup (they are remembered as two different items), but the base directory is the same. A full restore of a file system entails (1) restoring the last full backup of the file system and (2) restoring selected file system files from the latest incremental backup of the entire system.

If you have selected only certain files to be restored, and you know in which volume of the archive they are contained (from the optional listing generated during the backup), you may insert only that volume. This will generate warnings and prompts from **bru**(1). Respond to them as shown.

```
bru: warning - found volume 2, expecting 1
bru: c -> continue  q -> quit  r-> reload [default q] >> c<CR>
bru: warning - missing archive header block; starting at volume 2
bru: c -> continue  q -> quit  r-> reload [default q] >> c<CR>
```

Step 7: Return the system to the normal operating configuration (see Procedure 3.4, *Return to Multi-User*).

Step 8: Store the backup media in a safe place.

Multiple Volumes

If more than one volume is required during backup and restore, the following prompt is displayed by **bru**(1).

bru: load volume *n* **and enter device [default:** *archive device*] >>

Enter <**CR**> when you have the next volume loaded. You may change the device at this prompt (if, for example, you have two tape drives), but the archive description information (e.g., size of medium) will be used. (The default archive device used when **backup_restore** is invoked is not changed by this prompt, just the current **bru**(1) device.)

Procedure 5.6: Archive Description Change

Purpose	To describe archive devices in preparation for backup and restore.
When Performed	Before performing a backup or restore, or when an archive device is added to or deleted from the system.
Starting Conditions	System state —(any state) You must **mount /usr** to run this procedure in single-user mode. Login —any valid login, **root**, **sysadm**, or **br**
sysadm menu	FILE MANAGEMENT
Commands	**sysadm backup_restore**(1) **br**(1M)
Reference	*Preparing for File System Backup and Restore* in Chapter 5, *File System Administration*

Step 1: Log in as yourself, **root**, **br**, or **sysadm backup_restore**.

Step 2: If you logged in as yourself or **root**, enter the command:

```
# sysadm backup_restore
Password:
  or
# br
Password:
```

If you logged in as **root**, you are not prompted for the **sysadm** password.

Step 3: Type **5** to access the Archive Description menu. An Archive Description menu similar to the following displays:

```
Current archive description of Streaming-Tape...
 archive name:  /dev/rmt/m350_0t
 media size:    59M
 buffer size:   128K
 skip by:       reading
 last changed:  Fri  07 Feb 1986, at 09:43:58 AM

--------
You may:
  1. CHANGE individual values
  2. READ all-new values from library
  3. SAVE the values as shown
  0. QUIT this level
-- Your choice?  [0-3]
```

Step 4: Choose the appropriate menu item:

Type **1** to change individual values for the current archive device. More menus are presented that prompt you to enter the new values. When all changes have been made, type **3** to save the changes or **0** to ignore the changes you just made.

Type **2** to replace the current archive description with the standard description of a device. These descriptions are kept in a library of several complete descriptions. If you READ a complete description from this library, it replaces the entire current description. Before you save it, you can individually change any of the items. Once the new description is saved, any changes to the previous description are lost. See *The Archive Description Library* in Chapter 5 for additional information.

P6 System Reconfiguration Procedures

Introduction

The following procedures are covered in this section:

Procedure 6.1 **Reconfigure the Operating System**
To rebuild the operating system after tuning resulting from modifications to hardware and/or operating system software.

Procedure 6.2 **Unbootable Operating System Recovery**
To recover if you create an unbootable operating system after attempting a system reconfiguration.

Procedure 6.1: Reconfigure the System

Purpose	To make a new **/unix**. To incorporate tunable changes that result from hardware and/or software changes to the system.
When Performed	Only when the system must be tuned.
Starting Conditions	System state —(multi-user) Login —**root**
Commands	**sysgen**(1M)
Reference	Chapter 6, *Performance Management*

System reconfiguration is necessary whenever the physical configuration of the computer or the software configuration of the operating system changes. This happens when you upgrade hardware (e.g., as add more memory or disks) or edit tunable parameters to improve performance. The interactive **sysgen** program allows you to reconfigure your system. Refer to the following table when using **sysgen**.

Sysgen Typing Conventions	
Key(s)	**Meaning**
arrow left or <^h>	backspace
c	change field; enable line
arrow down or **j**	move cursor down
arrow up or **k**	move cursor up
n	**no** to **sysgen** prompt
o	open a new screen
q	close current screen
s	select or deselect an item
y	**yes** to **sysgen** prompt

Modifying System Configuration Information

The steps to incorporate changes to the new system configuration are:

Step 1: Log in as **root**.

Step 2: Type:

 # sysgen

A screen similar to the following displays:

```
                              SYSGEN V1.5
------------------------------------------------------------
Standard Configuration                                standard
(Your sysgen screen displays may differ from the example screens.)
(The cursor ( -> ) in the leftmost column is pointing to the current configuration.)
```

CONFIGURATION SELECTION SCREEN

NOTE

It is suggested that you use the **D** command in **sysgen** to create a copy of the "standard" configuration under a new name, and make changes to the copy. This way, if there are problems, you can return to the standard configuration.

Step 3: Using the cursor-move keys, go to the line showing the configuration you want to modify and type **o** to open the "Configuration Items" screen.

```
                                SYSGEN V1.5
------------------------------------------------------------------
Standard Configuration                                    standard
------------------------------------------------------------------

     Distributed Unix File System (DU)                      du
*    File System Handlers (S5)                              filsys
*    Generic CPU Support                                    vmecpu
* -> Kernel and Paging Parameters                           kernel
       (cursor at line to be modified)
        .
        .
        .
*    MVME323 ESDI Disk Controller Board                     vme323/f1
        .
        .
        .
(An asterisk (*) in the leftmost column indicates that an item and its subitems are
part of the current configuration.)
```

CONFIGURATION ITEMS SCREEN

Step 4a: To remove an item and its subitems from the current configuration perform this step. Move the cursor to the line you want to modify and type **s**. That item and its subitems are removed from the current configuration (indicated by the removal of the * from that line).

Step 4b: To add an item and its subitems to the current configuration perform this step. Move the cursor to the line you want to modify and type **s**. The * appears in the leftmost column of that line.

Step 5: To remove a subitem within a set of items shown in the "Configuration Items" screen, perform this step. Move the cursor to the item containing the subitems to be modified and type **o** to open the "Configuration Subitems" screen. A screen similar to the following displays the subitems in alphabetical order:

```
                              SYSGEN V1.5
------------------------------------------------------------------------
   Standard Configuration                                  standard
------------------------------------------------------------------------
*  Kernel and Paging Parameters                            kernel
------------------------------------------------------------------------
*  # Of Additional Event Cell Memory Pages     MAXSEPGCNT     Parameter
*  # Of System Reserved Memory/Swap Pages      MINASMEM       Parameter
*  # of Hash Buckets for buffers               NHBUF          Parameter
*  # of bad disk track mappings supported      BADDISKS       Parameter
*  Auto FS Buffer Update Rate (Seconds)        NAUTOUP        Parameter
     .
     .
     .
* -> System Buffers Count                      NBUF           Parameter
       .
       .
       .
```

CONFIGURATION SUBITEMS SCREEN

Move the cursor to the line you want to modify and type **s**. The system removes that item from the current configuration (indicated by the removal of the * from that line).

Step 6: To enter new configuration values perform Steps 6 and 7. Move the cursor to the line showing the item whose value you want to change and type **o** to open the "Subitem Values" screen.

```
                                SYSGEN V1.5
-------------------------------------------------------------------
   Standard Configuration                                 standard
-------------------------------------------------------------------
*  Kernel and Paging Parameters                           kernel
-------------------------------------------------------------------
*  System Buffers Count                        NBUF       Parameter
-------------------------------------------------------------------
        Parameter Name:     buffers  (cannot be modified)

           Conf.c Name:     NBUF  (cannot be modified)
         Default Value:     0     (cannot be modified)
                 Value: ->  0     (type in new value)
```

SUBITEM VALUES SCREEN

Move the cursor to the item that can be modified and type in the modification

In addition to parameter values, you can also modify the "Number of Devices" field in board descriptions using Steps 6 and 7. One series of screens related to boards follows:

```
                                SYSGEN V1.5
------------------------------------------------------------------
Standard Configuration                                    standard
------------------------------------------------------------------

      Distributed SYSTEM V/88 File System (DU)      du
*     File System Handlers (S5)                      filsys
*     Generic CPU Support                            vmecpu
*     Kernel and Paging Parameters                   kernel
        .
        .
        .
* -> MVME323 ESDI Disk Controller Board             vme323
        (cursor at line to be modified; type o to open next screen)
        .
        .
        .
```

CONFIGURATION ITEMS SCREEN

```
                                SYSGEN V1.5
-------------------------------------------------------------------
  Standard Configuration                                  standard
-------------------------------------------------------------------
* MVME323 ESDI Disk Controller Board                     vme323
-

* -> ESDI Controller board #1          VME323 #1    Device
     (cursor at line to be modified; type O to open next screen)
* Enhanced Small Disk Interface Driver  vme323    Driver
```

CONFIGURATION SUBITEMS SCREEN

```
                              SYSGEN V1.5
-------------------------------------------------------------------
  Standard Configuration                                  standard
-------------------------------------------------------------------
* MVME323 ESDI Disk Controller Board                     vme323
-------------------------------------------------------------------
* ESDI Controller board #1        Mvme323 #1        Device
-------------------------------------------------------------------

                    Device Type:     vme323

             Number of Devices: -> 0  (type in new value)
       Default Number of Devices:    0

       Interrupt Vector Location:    324
                         Address:    FFFFA000
                Bus Request Level:   5
```

SUBITEM VALUES SCREEN

Rebuilding the Operating System

Step 8: After modifying the current configuration as described in Steps 1 through 7, return to the "Configuration Selection" screen by typing **q** at each lower level screen.

Step 9: Verify that the cursor is on the line showing the new configuration for the system and type **q**. The following sequence of prompts appears:

```
Save changes to keyword? [y/n] y
   (Typing n discards modifications.)
Update system configuration files? [y/n] y
   (Typing n prevents an update to these files.)
Rebuild the operating system? [y/n] y
   (Typing n prevents rebuilding of the operating system.)
Install the new operating system to be used on the next
reboot? [y/n] y
   (Typing n prevents an update to the operating system.)
```

Booting the New Operating System

Step 10: Follow Procedure 3.6 to halt and reboot on the newly created operating system. If the system does not boot properly, perform Procedure 6.2.

Procedure 6.2: Unbootable Operating System Recovery

Purpose	To recover from an unbootable **/stand/unix**. To get a viable version of the system running after an unsuccessful attempt at reconfiguring the system.
Starting Conditions	System state —variable. Login —**root**
Commands	**sysadm firmware**(1) —to get to firmware
Bootable Programs	**/unix** **/oldunix** **/unix.rel**
Reference	*Performance Management* in your Delta Series System Manual

If you create a **/stand/unix** that is unbootable or operates so poorly that recovery while operating in that version is impossible, do the following:

Step 1a: If the system boots to the point where you get the `Console login:` prompt, log in as root. Refer to Procedure 3.5 for instructions on how to shut down the system and reach the "Power-Up Menu."

Step 1b: If you cannot log in, consult your Delta Series system manual for instructions on how to reach the "Power-Up Menu."

Step 2: Refer again to your Delta Series system manual for instructions on how to explicitly boot a disk file. Use the file names in the following order specified until one boots properly:

1. **/unix**
2. **/oldunix**
3. **/unix.rel**

Step 3: If none of these files boots the system, consult your *SYSTEM V/88 Release 3.2 Software Release Guide* to determine how to reload the operating system from the release media. Go to Step 5.

Step 4: After the system has rebooted and you have logged in again, remove **/stand/unix** and copy the file that was booted to **/stand/unix**. For example, having failed to properly boot the new operating system, you successfully boot the file **/unix**. Next type:

```
# rm /stand/unix
# cp /unix /stand/unix
# sync
```

Once again, shut down the system and reboot. The system should come up automatically.

Step 5: Finally, try to determine what went wrong. Consider first that you may have placed an incorrect value in the parameter(s) with which you were working.

Step 6: Repeat Procedure 6.1 to correct errors in the previous reconfiguration. If you cannot determine what went wrong, document what happened as thoroughly as possible and contact your Field Service Representative.

P7 LP Print Service Administration Procedures

Introduction

This section provides the following procedures:

Procedure 7.1 **Stop the Print Service**
Instructions for stopping the LP print service.

Procedure 7.2 **Restart the Print Service**
Instructions for restarting the LP print service.

Procedure 7.3 **Set Up the Print Service**
Instructions for adding or deleting an LP printer from the current configuration, or for changing the configuration of a printer.

Procedure 7.4 **Set Up Forms**
Instructions for adding, changing, or deleting a form from the LP print service.

Procedure 7.5 **Set Up Filters**
Instructions for adding, changing or deleting a filter from the LP print service.

Procedure 7.1: Stop the Print Service

Purpose	To stop the print service.
Starting Conditions	Multi-user or single-user state (If you are in single-user state, you must run the command **mount /usr** before following this procedure.) Login —root
Commands Used in 7.1	**sysadm packagemgmt/lpmgmt/service/stop**
Reference	Summary of *Administrative Commands* in Chapter 7

Step 1: Enter the following command to stop the print service:

$ sysadm stop

This stops the print service completely. You are notified by a screen message when the printer has stopped; no further requests for printing will be acted on.

Procedure 7.2: Restart the Print Service

Purpose	To make the LP system available again after having been stopped.
Starting Conditions	Multi-user or single-user state (If you are in single-user state, you must run the command **mount /usr** before following this procedure.) Login —root
Commands Used in 7.2	**sysadm packagemgmt/lpmgmt/service/start**
Reference	Summary of *Administrative Commands* in Chapter 7

Step 1: After you completed those administrative tasks for which you stopped the print service, restart the print service by entering the following command:

$ sysadm start

A screen message notifies you when the print services are available again.

Procedure 7.3: Set Up the Print Service

NOTE

Procedure 7.3 provides instructions for adding a printer to your print service, changing a printer configuration, and deleting a printer from your print service. Before you can follow any of these procedures, the printer in question must be connected to an available port.

Purpose	To add or delete an LP printer from your system, or to change the configuration of an LP printer in your system.
Starting Conditions	Multi-user or single-user state (If you are in single-user state, you must run the command **mount /usr** before following this procedure.) Login —root
Commands Used in 7.3	**sysadm packagemgmt/lpmgmt/printers/add_p** **sysadm packagemgmt/lpmgmt/printers/change_p** **sysadm packagemgmt/lpmgmt/printers/accept_p** **sysadm packagemgmt/lpmgmt/printers/enable_p** **sysadm packagemgmt/lpmgmt/service/default** **sysadm packagemgmt/lpmgmt/printers/reject_p** **sysadm packagemgmt/lpmgmt/printers/disable_p** **sysadm packagemgmt/lpmgmt/printers/delete_p**
Reference	Summary of *Administrative Commands* in Chapter 7

Add an LP Printer

Step 1: To add a printer, enter the following command:

$ sysadm printers/add_p

If other printers have already been added to the system, you are asked if the printer you are now adding is similar to any of them. If it is, the configuration of the similar printer is used as a template for the configuration of the new printer. If not, standard defaults are given in a template. In either case, the system prompts you to define the configuration for the new printer, as shown in the following example.

If you defined the alert-type for the printer to be **mail** or **write** and did not specify a login, the login **sysadm** is assumed. In addition, the alert-type **quiet** can only be used to terminate an active alert and, therefore, cannot be used as the alert-type for a new printer.

NOTE

In the following example, the printer is an AT&T Model 455 daisy-wheel printer, connected to a port on the Expanded Input/Output Capability Feature card. For more information on how to identify ports, see Chapter 7, *Printer Management*, of this guide.

```
Enter the name of the new printer:  printer2
Is this printer to be similar to one already added? [y, n, ?, q] n
Printer class to add it to:  (default: none)  letterquality
```
(for the next two prompts, substitute in formation for your printer)
```
Enter one line that describes the printer for your users.
(default:  none)  AT&T Model 455 daisy-wheel printer
Type of printer: (default:  unknown)  455
Types of files printable without filtering:
(default:  simple) simple,nroff

Printer interface to use: (default:  standard)  /usr/kol/interface
Can a user skip the banner page?  (default:  no) [y, n, ?, q]

Use the default page size and printing pitch?
(default:  yes) [y, n, ?, q]

Enter a command to run to alert you when the printer faults.
(default:  mail lp)  write lp
How often should you be alerted (minutes, 0-once)?  (default:  0)  3
How should printing restart after you fix a printer fault?
(default:  continue)

Is the printer directly connected?  (default:  yes) [y, n, ?, q]
Printer port to use:  /dev/tty11
Is the printer also a login terminal?  (default:  no) [y, n, ?, q]

The default port settings will be:
9600 cs8 cstopb -parenb -parodd ixon -ixany opost -olcuc
onlcr -ocrnl -onocr -onlret -ofill n10 cr0 tab0 bs0 vt0 ff0
Enter any changes or additional settings that must be in
effect:

List the print wheels available:
(default:  none) elite, courier
```

(continued)

```
                                                    (continued)
List the users who are denied access to this printer.
(default:  none)
List the only users who are allowed access to this printer.
(default:  all)  merlyn ehr3 gig

List the only forms that can be used with this printer.
(default:  none)  payroll_check, order_form

install, edit, or skip this printer entry: (default: install)
```

If, while setting up the print service, you have trouble defining any of the fields for which you must supply values, be sure to read the help messages and look for further details in Chapter 7 of this guide. Also, see *How to Define Printer Ports* and *Printer Port Characteristics* in Chapter 7.

When you have finished this session, the printer you have defined is added to the print service for your machine. Next, you must specify certain operational parameters for this printer.

Step 2: To tell the print service to accept requests for print jobs on the printer you are adding, enter the following command:

$ sysadm printers/accept_p

You are asked to list the printers or printer classes that can start accepting print requests.

Step 3: When you are ready to start printing, be sure that the printer is ready to receive output. For several printers, this means that the top of the form has been adjusted and that the printer is on-line. To enable printing to occur on the printer, enter the following command:

$ sysadm printers/enable_p

You are asked to list the printers that can start printing requests.

Step 4: (This step is optional.) Set the destination of print requests to a default printer by entering the following command:

$ sysadm default

You are asked to name a printer or printer class to use as the default destination for print requests.

Step 5: You can verify that the new printer has been added according to your specifications, and that your final configuration is correct, by entering the following command:

$ sysadm printers/list_p

The command prompts you for the names of printers for which you want to see a complete configuration; it gives you a configuration listing of those printers. If you specify **all**, you receive a configuration listing of all available printers, including the one you have just added.

Change the Configuration of an LP Printer

If you change your system by changing a printer configuration or by deleting a printer, two things may change as a result: 1) the appearance of the text of printed files, and 2) the appropriateness of the changed or deleted printer for jobs already in the queue. To safeguard requests for printing, consider taking the following preliminary actions for a printer that you are going to drop from the system or for which you are going to change the configuration: 1) stop the printer from taking requests for printing, and/or 2) disable the printer.

Once you are satisfied that no jobs are in danger of being ruined and no new jobs will be queued for the affected printer, you are ready to add or delete a printer to your system, or to change the configuration of an existing printer. The procedures for these precautionary measures are given in Steps 1 and 2.

Step 1: (This step is optional.) To safeguard against losing print jobs requested from a printer that you are reconfiguring, you may want to stop that printer from accepting print requests. Enter the following command:

$ sysadm printers/reject_p

The **printers** menu asks you to specify targeted printers or printer classes and to provide a reason for turning off their ability to accept requests.

Step 2: (This step is optional.) Disable your printer by entering the following command:

$ sysadm printers/disable_p

The **printers** menu then gives you the option of preserving or deleting current print requests before the specified printer is disabled:

```
Enter the names of the printers that should stop printing requests.
Separate multiple names with a space or comma.
(default:  printer2)
Cancel any requests currently printing?  (default:  no) [y, n, ?, q]
Wait for any requests currently printing?
(default:  no) [y, n, ?, q] yes
Reason for disabling:
(default:  none)  to reconfigure printer2
```

The reason that the default printer listed in Step 2 is **printer2** is because it was the last printer added to the system with the **add** command from the **printers** menu. If no printers have been added in this way, the default value is "none."

Step 3: Now you are ready to change the configuration of the appropriate printer. Enter the following command:

$ sysadm printers/change_p

First, you are prompted for the name of the printer. Then you are asked to provide details about the new configuration, e.g., page dimensions.

If you defined the alert-type for the printer to be **mail** or **write** and did not specify a login, the login **sysadm** is assumed. In addition, the alert-type **quiet** can only be used to terminate an active alert.

```
Enter the name of the printer:  (default:  printer2)
Printer class to add it to:  (default:  letterquality)
Enter one line that describes the printer for your users.
(default:  the printer specified in Step 1)
Type of printer: (default:  the type specified in Step 1)
Types of files printable without filtering: (default:  simple, nroff)

Printer interface to use: (default:  /usr/kol/interface)
Can a user skip the banner page?  (default:  no) [y, n, ?, q]

Use the default page size and printing pitch?
(default:  yes) [y, n, ?, q]

Enter a command to run to alert you when the printer faults.
(default:  write lp)
How often should you be alerted (minutes, 0-once)?  (default:  3)  10
How should printing restart after you fix a printer fault?
(default:  continue)

Is the printer directly connected?  (default:  yes) [y, n, ?, q]
Printer port to use:  (default:  /dev/tty11)
Is the printer also a login terminal?  (default:  no) [y, n, ?, q]

The default port settings will be:
9600 cs8 cstopb -parenb -parodd ixon -ixany opost -olcuc
onlcr -ocrnl -onocr -onlret -ofill n10 cr0 tab0 bs0 vt0 ff0
Enter any changes or additional settings that must be in
effect: (default:  as listed above)
List the print wheels available:
(default: elite, courier)

List the users who are denied access to this printer.
(default:  none)
List the only users who are allowed access to this printer.
(default:  merlyn ehr3 glg) |woo

List the only forms that can be used with this printer.
(default:  payroll_check, order_form)

install, edit, or skip this printer entry:  (default:  install)
```

Step 4: "Authorize" the newly configured printer to accept requests for print jobs by entering the following command:

$ sysadm printers/accept_p

Step 5: Re-enable the newly configured printer to print by entering the following command:

$ sysadm printers/enable_p

The menu prompts you to provide the names of the printers that can start printing requests.

Step 6: (This step is optional.) To define a default destination for print requests, enter the following command:

$ sysadm default

You are prompted for the name of the printer or printer class to be used as the default destination for print requests.

Step 7: The reconfiguration of your printer is now complete. If you want to verify that your final configuration is correct, enter the following command:

$ sysadm printers/list_p

The command prompts you for the names of printers for which you want to see a complete configuration; it gives you a configuration listing of those printers. If you specify **all**, you receive a configuration listing of all available printers, including the one you have just changed.

Delete a Printer

Step 1: Enter the following command to delete a printer:

$ sysadm printers/delete_p

You are asked to name the printer to be deleted. The command will also give you an opportunity to reconsider your request by asking, immediately, for verification of it (the message `Are you sure?` displays).

Step 2: To verify that the appropriate printer was deleted, enter the following command:

```
$ sysadm printers/list_p
```

The command prompts you for the names of printers for which you want to see a complete configuration; it gives you a configuration listing of those printers. If you specify **all**, you receive a configuration listing of all available printers, minus the one you have just deleted.

Procedure 7.4: Set Up Forms

Purpose	To add, change or delete a form for the print service.
Starting Conditions	Multi-user or single-user state (If you are in single-user state, you must run the command **mount /usr** before following this procedure.) Login —root
Commands Used in 7.4	**sysadm packagemgmt/lpmgmt/forms/add_f** **sysadm packagemgmt/lpmgmt/forms/change_f** **sysadm packagemgmt/lpmgmt/forms/delete_f** **sysadm packagemgmt/lpmgmt/forms/mount_f** **sysadm packagemgmt/lpmgmt/forms/list_f**
Reference	Summary of *Administrative Commands* in Chapter 7

About Using Forms

Your printer can print on a variety of pre-printed forms, e.g., checks and invoices. The LP print service can help you take advantage of this capability, but you must supply additional software, as well as the forms. Once you have all the necessary components in place, the LP print service can help you use your forms and the software you have to support them in the following ways:

- The print service can schedule the work of one or more printers, allowing you to assign the task of printing particular forms to various printers at specific time periods. (If you have only one printer, the scheduling service allows you to allocate its time among multiple forms.)
- The print service can keep track of forms currently mounted and alert you to mount forms as needed.

The various steps involved in setting up a system and using it to print on pre-printed forms are shown in the following diagram. This diagram leads you through a sample case, showing the steps taken by a company that wants to automate the printing of payroll checks for its employees. (The payroll database mentioned in Step 1, which contains the information that appears on the checks, e.g., the payee's name and the amount, is assumed to exist on the company's computer.)

The company buys (or asks a programmer to write) application software (a program) that takes information from the payroll database to print a standard check.

↓

The system administrator installs the application on the system.

↓

The system administrator notifies users (in this case, a company clerk responsible for preparing payroll checks) that the application program is available to prepare the checks.

↓

The system administrator uses the new sysadm menu for LP forms to identify, to the print service, the available forms, including checks, invoices, and time cards. ← *Provided by the LP print service*

↓

The payroll clerk issues a request to the print service to print checks.

↓

The print service notifies the administrator of the payroll clerk's request and alerts the administrator to change the paper in the printer from the current stock to checks.

↓

The system administrator loads checks into the printer.

↓

The printer prints the payroll checks.

As you can see, the LP print service does not provide any application software like that described in Step 1 (a program that converts a file to a format suitable for a particular form, e.g., an invoice). As stated in Step 1, you can purchase application software separately or you can write your own applications.

Add a Form

Step 1: Enter the following command to add a form for the print service:

```
$ sysadm forms/add_f
```

You are asked to name the desired form and provide specifications for it, as follows:

```
Name of form:  payroll
Page length: (default: 66)
Page width: (default: 80)
Lines per inch: (default: 6)
Characters per inch: (default: 10)
Number of pages in the form: (default: 1)  2
Character set to use with the form:  (default:  any)
Ribbon color to use with the form:  (default:  any)  black
Enter a one-line description of this form for your users:
payroll check for non-supervisory staff
Enter the full pathname of a file containing an alignment pattern:
(default:  none)  /usr/sam/align.check)
What type of file is this? (default:  simple)

Enter a command to run to alert you when the form needs mounting.
(default:  mail lp)  write lp
How many print requests should be waiting before you are alerted?
(default:  1)  4
How often should you be alerted (minutes, 0-once)?
(default:  0)  3

List the users who are denied access to this form.
(default:  none)
List the only users who are allowed access to this form.
(default:  all)  merlyn ehr3 gig

install, edit, or skip this form:   (default: install)
```

Step 2: (This step is optional.) If you would like to mount this form, enter the following command:

$ sysadm forms/mount_f

You are prompted to answer the following questions:

```
Enter the name of the printer:  (default:  printer2)  printer3
Enter the name of the form you are mounting:  (default:  payroll)
Enter the name of the print wheel you are mounting:  (default:  none)
Print an alignment pattern?  [y, n, ?, q]  yes

Press return to print an alignment pattern [q to quit]:
```

You see the third prompt (**`Enter the name of the print wheel you are mounting:`**) only if your printer can take print wheels. If it does, and you are prompted for the names of both a form and a print wheel, be sure that you assign a value other than "none" to one of them.

The default form is **payroll** because that was the last form added to the system with the **add** command from the **forms** menu. If the form can be mounted, an alignment pattern is to be printed, the prompt **`Press return to print an alignment pattern [q to quit]:`** appears repeatedly until the user indicates that he or she has finished. Each time the **`RETURN`** key is pressed, an alignment pattern will be printed on the printer named. (Enter **q** to stop printing the alignment pattern.)

NOTE

An alternative way to mount a new form is through the Printer Management Menu. The following command line is equivalent to the mount command shown above: **sysadm mount_p**.

Step 3: To verify that the new form has been added, enter the following command:

$ sysadm forms/list_f

The command prompts you to name the forms for which you want to see a complete description; it gives you a list of those forms. If you specify **all**, you receive a list of all available forms, including the one you have just added.

Change a Form

Step 1: Enter the following command to change a form for the print service:

$ sysadm forms/change_f

You are prompted for the name of the form you want to change. (By default, the form to be changed is considered the form you added last by selecting the menu item **add form**.) Then you are asked to describe the form and list users who are or are not allowed to use it, through a series of questions, shown in the following example.

Notice that the default values for various specifications are the same as the values for the corresponding specifications of the last form added (see the previous section).

```
Name of form: (default: payroll)
Page length: (default: 66)
Page width: (default: 80)
Lines per inch: (default: 6)
Characters per inch: (default: 10)
Number of pages in the form: (default:  2)
Character set to use with the form:  (default:  any)
Ribbon color to use with the form:  (default:  black)
Enter a one-line description of this form for your users:
(default: payroll check for non-supervisory staff)

Do you want to see the current alignment pattern?
(default: no) [y, n, ?, q]

Do you want to keep the current alignment pattern?
(default: yes) [y, n, ?, q] n

Enter the full pathname of a file containing an alignment pattern:
(default:  /usr/sam/align.check) /usr/joanne/align.new
What type of file is this? (default:  simple)

Enter a command to run to alert you when the form needs mounting.
(default:  write lp)  mail lp
How many print requests should be waiting before you are alerted?
(default:  4)  10
How often should you be alerted (minutes)?
(default:  3)

List the users who are denied access to this form.
(default:  none)
List the users who are allowed access to this form
(default: merlyn, ehr3, glg)

install, edit, or skip this form:  (default:  install)
```

(These are the same questions you are asked when adding a new form.)

Step 2: Now the changes to your form are complete. If you want to verify that these changes are correct, enter the following command:

```
$ sysadm forms/list_f
```

The command prompts you for the names of forms for which you want to see a complete description; it then gives you a list of those forms. If you specify **all**, you receive a list of all available forms, including the one you have just changed.

Delete a Form

Step 1: Enter the following command to delete a form for the print service:

$ sysadm forms/delete_f

You are asked to name the forms you want to remove. Then, before the forms are deleted, you have a chance to reconsider your request (the message `Are you sure?` displays).

Step 2: The appropriate form has now been deleted. If you want to verify this, enter the following command:

$ sysadm forms/list_f

The command prompts you for the names of forms for which you want to see a complete description; it then gives you a list of those forms. If you specify `all`, you will receive a list of all available forms, minus the one you have just deleted.

Procedure 7.5: Set Up Filters

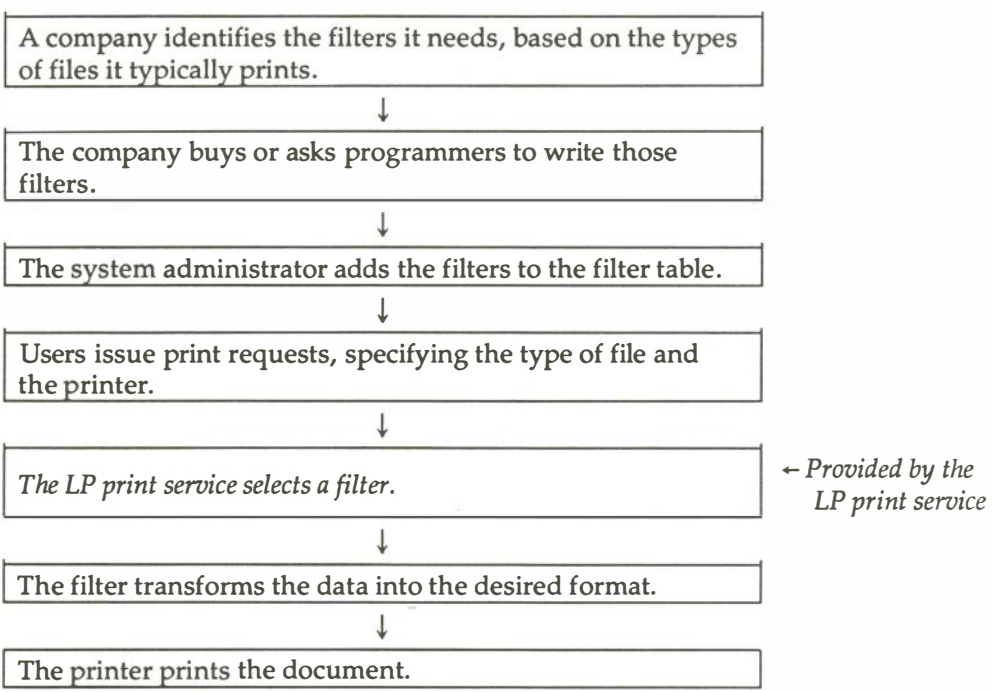
Purpose	To add, change, or delete a filter from the print service.
Starting Conditions	Multi-user or single-user state (If you are in single-user state, you must run the command **mount /usr** before following this procedure.) Login —root
Commands Used in 7.5	**sysadm packagemgmt/lpmgmt/filters/add_f** **sysadm packagemgmt/lpmgmt/filters/change_f** **sysadm packagemgmt/lpmgmt/filters/delete_f** **sysadm packagemgmt/lpmgmt/filters/list_f**
Reference	Summary of *Administrative Commands* in Chapter 7

About Using Filters

In addition to allowing you to print a variety of documents, the LP print service enables you to have data that appears in your input file in one format printed on paper in another format. The device used by the print service to transform formats is a program called a filter. You must provide any filters you want to use on your system (either by buying or writing them); the LP print service does not provide filters.

The print service helps you manage filters that you have already installed on your system. Specifically, it oversees the use of filters, checking to find out when they are needed for print jobs, and then matching the appropriate filter with a user's file and a printer.

The following diagram shows a sample scenario of how filters may be used with the help of the LP print service.



Add a Filter

Step 1: Enter the following command to add a filter for the print service:

$ sysadm filters/add_f

The system prompts you for the name of the desired filter. Then it will present a series of questions through which you can submit specifications for it. In the following example, the type of printer is an AT&T Model 455:

```
Filter name: 450
Input types it can convert:  (default: any)  nroff
Types it can produce:  (default: any )  450
     (substitute your printer for the following)
Types of printers it is restricted to:  (default: any)  455
Printers it is restricted to:  (default:  any)
Is this a slow filter?  (default:  yes) [y, n, ?, q]
Enter the filter command and any fixed options:
450
        .
        .
        .
        .
Keyword:  MODES
Pattern:  landscape
Template:  -l
Keyword:  done
install, edit, or skip this filter:  (default:  install)
```

Step 2: To verify that the new filter has been added, enter the following command:

$ sysadm filters/list_f

The system prompts you for a list of the filters for which you want to see a complete description; it then gives you a list of those filters. If you specify **all**, you receive a list of all available filters, including the one you have just added.

Change a Filter

Step 1: Enter the following command to change a filter for the print service:

$ sysadm filters/change_f

The system prompts you for the name of the filter you want to change.

If the filter was delivered with the LP print service, you are then asked whether you want to restore the filter to the "factory setting." If you answer no, the system then asks you through a series of questions to provide a new specification for the filter.

In the following example, the specification for the filter added in the previous section is changed. (Because you are not dealing with a filter that was delivered with the LP print service, you are not given the choice of restoring the filter to the "factory setting.") Two changes are made: add another type of printer to which the filter is restricted (qume) and specify that this filter is not to be a "slow" filter (as specified by default).

```
Filter name (default: 450)
Input types it can convert:  (default: nroff)
Types it can produce:  (default:  450)
Types of printers it is restricted to:  (default: 455) 455,qume
Printers it is restricted to:  (default:  any)
Is this a slow filter?  (default: yes) [y, n, ?, q]  no
Enter the filter command and any fixed options:
(default:  450)
        .
        .
        .
        .
Keyword:  (default:  MODES)
Pattern:  (default:  landscape)
Template:  (default:  -l)

Keyword: done

install, edit, or skip this filter:  (default:  install)
```

Step 2: To verify that the appropriate filter has been changed, enter the following command:

$ sysadm filters/list_f

The command prompts you for the names of filters for which you want to see a complete description; it then gives you a list of those filters. If you specify **all**, you receive a list of all available filters, including the one you have just changed.

Delete a Filter

Step 1: Enter the following command to delete a filter for the print service:

$ sysadm filters/delete_f

The system then prompts you for the name of the filter you want to remove and give you an opportunity to reconsider your request (the message `Are you sure?` displays).

Step 2: To verify that the appropriate filter has been deleted, enter the following command:

$ sysadm filters/list_f

The command prompts you for the names of filters for which you want to see a complete description; it then gives you a list of those filters. If you specify **all**, you receive a list of all available filters, minus the one you have just deleted.

P8 TTY Management Procedures

Introduction

Procedure 8.1 **Check TTY Line Settings**
To tell what line settings are defined.

Procedure 8.2 **Make TTY Line Settings**
To create new TTY line settings and hunt sequences.

Procedure 8.3 **Modify TTY Line Characteristics**
To change the characteristics of TTY lines. To turn lines on or off.

Procedure 8.1: Check TTY Line Settings

Purpose	To tell what line settings are defined.
Starting Conditions	System state —multi-user or single-user Login —an authorized login or knowledge of the **sysadm** password.
sysadm menu	TTY MANAGEMENT
Commands	**sysadm lineset**(1)
Reference	*How the TTY System Works* in Chapter 8, *TTY Management*

Step 1: Enter this command to go directly to the lineset display:

```
$ sysadm lineset
Password:
```

Step 2: This following displays on your terminal:

```
Running subcommand `lineset' from menu `ttymgmt',
TTY MANAGEMENT
Tty Line Settings and Sequences

console1    console2   console3   console4   console5   console
pc220       (does not sequence)
300         19200      9600       4800       2400       1200
300H        4800H      9600H      19200H     2400H      1200H
1200UUCP    (does not sequence)
1200UUCPH   (does not sequence)
```

Each line setting is a name used to identify a set of TTY line characteristics. During the **login** process, the line settings on one line "hunt" from left to right, moving from one to the next on receiving a BREAK signal. The rightmost setting on each line hunts to the first one again, forming a circular hunt sequence.

Note that the **pc220, 1200UUCP**, and **1200UUCPH** settings do not sequence. Sending a BREAK does not make it change in any way.

Step 3: To look at a line setting in detail:

```
Select one line setting to see it in detail [?, q]: 1200

Line Setting:              1200
       Initial Flags:      B1200 HUPCL
       Final Flags:        B1200 SANE IXANY TAB3 HUPCL
       Login Prompt:       login:
       Next Setting:       300

B1200        1200 Baud
HUPCL        Hang Up on Last Close
IXANY        Enable Any Character to Restart Output
SANE         Set All Modes To "Traditionally Reasonable" Values
TAB3         Expand Horizontal-tab To Spaces
```

Step 4: Notice you do not have to start with the leftmost entry of a row; you can specify any entry:

```
Select another line setting or
<RETURN> to see the original list [?, q]: 300

Line Setting:                300
        Initial Flags:       B300 HUPCL
        Final Flags:         B300 SANE IXANY TAB3 HUPCL
        Login Prompt:        login:
        Next Setting:        19200

B300          300 Baud
HUPCL         Hang Up on Last Close
IXANY         Enable Any Character to Restart Output
SANE          Set All Modes To "Traditionally Reasonable" Values
TAB3          Expand Horizontal-tab To Spaces

Select another line setting or
<RETURN> to see the original list [?, q]: q
```

(Depending upon whether you entered from the first **sysadm** *menu or directly from the command line, you will get either the following message or your prompt:)*

```
Press the RETURN key to see the ttymgmt menu [?, q]: q
```

Procedure 8.2: Make TTY Line Settings

Purpose	To create new line settings and hunt sequences.
Starting Conditions	System state —multi-user or single-user Login —an authorized login or knowledge of the **sysadm** password.
sysadm menu	TTY MANAGEMENT
Commands	**sysadm mklineset**(1)
Reference	*How to Create New Line Settings* and *Hunt Sequences* in Chapter 8, *TTY Management*

In this procedure, how to connect a dual-speed modem to the computer to handle 300 baud and 1200 baud. There is a 1200 setting already in the table, but if the user misses the speed, **BREAK** must be pressed many times to hunt for 1200. Because only two speeds are required, create a new 1200-300 sequence:

Step 1: Enter this command to go directly to the **mklineset** display:

```
$ sysadm mklineset
Password:
```

Step 2: The following sequence of prompts appears on your terminal:

```
Running subcommand 'mklineset' from menu 'ttymgmt',
TTY MANAGEMENT

Enter the name of the new tty line setting [?, q]: 1200300
Select a baud rate [?, q]: ?  (To ask for HELP)

Available baud rates:
    50      110     150       300     1200      2400      9600     38400
    75      134     200       600     1800      4800     19200
Select a baud rate [?, q]: 1200

Enter the login prompt you want (default - "login: ") [?, q]: <CR>

     (Accepting the default)

Do you want to add another tty line setting to the sequence? [y,n,q] y

Enter the name of the new tty line setting [?, q]: 3001200
Select a baud rate [?, q]: 300
Enter the login prompt you want (default - "login: ") [?, q]: <CR>
Do you want to add another tty line setting to the sequence? [y,n,q] n

Here is the tty line setting sequence you created:

1200300      3001200
Line Setting:         1200300
    Initial Flags:     B1200 HUPCL
    Final Flags:       B1200 SANE IXANY HUPCL TAB3
    Login Prompt:      login:
    Next Setting:      3001200
Line Setting:          3001200
    Initial Flags:     B300 HUPCL
    Final Flags:       B300 SANE IXANY HUPCL TAB3
    Login Prompt:      login:
    Next Setting:      1200300
 .
 .
 .
```

```
.
.
.
B1200     1200 Baud
B 300     300 Baud
HUPCL     Hang Up on Last Close
IXANY     Enable Any Character to Restart Output
SANE      Set All Modes To "Traditionally Reasonable" Values
TAB3      Expand Horizontal-tab To Spaces
Do you want to install this sequence? [y, n, q] y
Installed.
```

(Depending upon whether you entered from the first **sysadm** *menu or directly from the command line, you will get either the following message or your prompt:)*

```
Press the RETURN key to see the ttymgmt menu [?, q]: q
```

Procedure 8.3: Modify TTY Line Characteristics

Purpose	To modify TTY line settings or turn line on or off.
Starting Conditions	System state —multi-user or single-user Login —an authorized login or knowledge of the **sysadm** password.
sysadm menu	TTY MANAGEMENT
Commands	**sysadm modtty**(1)
Reference	*How to Modify TTY Line Characteristics* in Chapter 8, *TTY Management*

The objective here is to tell the computer which port to use with the line settings defined in Procedure 8.2.

Step 1: Enter the following command to go directly to the **modtty** display:

```
$ sysadm modtty
Password:
```

Step 2: The following sequence of prompts appears on your terminal:

```
Running subcommand 'modtty' from menu 'ttymgmt',
TTY MANAGEMENT

Changeable tty lines:
     contty      tty00       tty01       tty02     tty03
Select the tty you wish to modify,
or enter ALL to see a report of all ttys [?, q]: ALL

Changeable tty lines:
                  Hangup     Line
Tty      State    Delay      Setting     Description
---      -----    -----      -------     -----------
contty    off      off        9600       2nd port on processor card
tty00     off      off        9600       1st port on mvme335 card
tty01     off      off        9600       2nd port on mvme335 card
tty02     off      off        9600       3rd port on mvme335 card
tty03     off      off        9600       4th port on mvme335 card

Continue (default: y)? [y, n, q] <CR>

Changeable tty lines:
     contty      tty00       tty01       tty02     tty03
Select the tty you wish to modify,
or enter ALL to see a report of all ttys [?, q]: tty03

tty03:  current characteristics:
     State             off
     Hangup Delay      off
     Line Setting      9600
     Description

Available states:
     off     on
Select a state (default: off) [?, q]:  on

Enter a hangup delay, in seconds, or 'off' (default: off) [?, q]:  45
```

(*Because this is a dial-up line, we want to specify a* **timeout** *figure.*)

.
.
.

```
.
.
.
Available line settings:
     console   console4     pc220   1200H     4800     9600    19200H
     console1  console5     300     1200UUCP  4800H    9600H   1200300
     console2               300H    2400                19200   3001200
     console3               1200    2400H
Select a line setting (default: 9600) [?, q]:  1200300

Current description:

Enter a new description (default: current description) [?, q]:
      1200/300 baud dial in line

tty03: new characteristics:
     State          on
     Hangup Delay   45
     Line Setting   1200300
    Description     1200/300 baud dial in line
Do you want to install these new characteristics? [y, n, q] y
tty14 now has new characteristics.

Changeable tty lines:
    contty     tty00      tty01       tty02    tty03
Select the tty you wish to modify,
or enter ALL to see a report of all ttys [?, q]: ALL

Changeable tty lines:
                  Hangup   Line
Tty       State   Delay    Setting    Description
---       -----   -----    -------    -----------
contty     off    off      9600       2nd port on processor card
tty00      off    off      9600       1st port on mvme335 card
tty01      off    off      9600       2nd port on mvme335 card
tty02      off    off      9600       3rd port on mvme335 card
tty03      on     45       1200300    1200/300 baud dial in line

Continue (default: y)? [y, n, q] q
```

P9 Basic Network Procedures

Introduction

The following procedures are covered in this section:

Procedure 9.1 **Set Up Basic Networking Files**
To configure basic networking files.

Procedure 9.2 **Basic Networking Maintenance**
To maintain basic networking files and operations.

Procedure 9.3 **Basic Networking Debugging**
To track down problems in basic networking.

Procedure 9.4 **Set Up BNU TCP/IP NETWORK**
To show how to set up BNU files for a NETWORK to handle **uucico** services.

Procedure 9.1: Set Up Basic Networking Files

Purpose	To configure basic networking files. To ensure proper communications links.
When Performed	Initial setup and when adding new devices or remote systems.
Starting Conditions	System state —2 (multi-user) or 1 (single-user). You must **mount /usr** to run this procedure in single-user mode. Login —an authorized login
sysadm menu	PACKAGE MANAGEMENT
Commands	**sysadm uucpmgmt**(1) **sysadm devicemgmt**(1) **sysadm portmgmt**(1) **sysadm systemmgmt**(1) **sysadm pollmgmt**(1)
Reference	Chapter 9, *Basic Networking*

The procedure that follows provides instructions for setting up the Basic Networking facility and putting it into operation. This is done using **sysadm** subcommands and a text editor.

The following steps provide instructions on adding entries to three of the necessary support files: **Devices**, **Systems**, and **Permissions**. Instructions are also provided to modify existing entries in the **/etc/inittab** file for use with Basic Networking. Finally, the set up of several optional files is described.

Display the **uucpmgmt** System Administration submenu by entering:

```
$ sysadm uucpmgmt
Password:
                        BASIC NETWORKING UTILITIES MANAGEMENT

  1 devicemgmt  manage devices (list, add, delete)
  2 pollmgmt    manage poll entries (list, add, delete)
  3 portmgmt    manage I/O ports (list, modify)
  4 systemmgmt  manage remote systems entries (list, add, delete, call)

Enter a number, a name, the initial part of a name, or
? or <number>? for HELP,  q to QUIT:
```

Each Basic Networking subcommand described in the following sections can be accessed from this display or directly from the shell.

Set Up Devices File - devicemgmt

The **Devices** file (**/usr/lib/uucp/Devices**) contains information about the devices used to call other machines. For details on this file, refer to Chapter 9.

Step 1: To add entries into the **Devices** file, type **sysadm devicemgmt**, then select **2** (add).

```
$ sysadm devicemgmt
Password:

Running subcommand 'devicemgmt' from menu 'uucpmgmt',
BASIC NETWORKING UTILITIES MANAGEMENT

This procedure is used to list, add, and delete entries
in the Basic Networking Utilities '/usr/lib/uucp/Devices' file.
This file contains information about devices
available for calling out using the commands: uucp, cu, and ct.

Type 'q' at any time to quit the present operation.
If a '?' appears as a choice, type '?' for help.

If a default appears in the question type <RETURN> for the default.

Enter the operation you want to perform:

        1 list
        2 add
        3 delete

(default list) [q]: 2
```

The subcommand prompts you for information on the devices used by Basic Networking.

port name
: The name of the port to which the device will be connected.

device name
: The name of the device that is being connected to the above port. Pick the one you are using from the list that displays; the default is **penril**. If ACU is specified as the device type to be connected to the port, two entries are created, one for 300 baud and one for 1200 baud.

Step 2: After you have entered the requested information, it displays before it is entered into the **Devices** file.

The **/etc/inittab** file may not contain a correct entry for BNU use (e.g., baud rate, incoming/outgoing use) the port just assigned. You can change the port now or later using the **portmgmt** subcommand in the next procedure.

Set Up /etc/inittab - portmgmt

The **inittab** file (**/etc/inittab**) contains information on the ports to which the devices are connected. For further information on this file, see Chapter 3, *Processor Operations*.

Step 1: To add BNU entries to the **inittab** file type **sysadm portmgmt**, then select **2** (modify).

```
$ sysadm portmgmt
Password:

Running subcommand 'portmgmt' from menu 'uucpmgmt',
BASIC NETWORKING UTILITIES MANAGEMENT

This procedure is used to list and modify
the entries that control the direction of traffic
on the Basic Networking Utilities I/O ports used by uucp, cu, and ct
commands.

Type 'q' at any time to quit the present operation.
If a '?' appears as a choice, type '?' for help.

If a default appears in the question, type <RETURN> for the default.

Enter the operation you want to perform:

        1 list
        2 modify


(default list)[q]:  2
```

The subcommand lists the ports available to be used by Basic Networking, then asks you to choose which one you want to modify. It then prompts you for the following information:

port name
: Name of the port you want to add (must be a port shown in the list).

traffic direction
: The direction of the traffic on the port. You must specify whether the traffic will be **incoming** only, **outgoing** only, or **bidirectional**.

baud rate
: Enter the speed (baud rate) of the selected port.

After you have entered the requested information, it displays before it is entered into the **/etc/inittab** file.

NOTE

Since adding a device (**sysadm devicemgmt**) automatically creates a port entry in **/etc/inittab**, you may only need to use **sysadm portmgmt** for modifications.

Set Up Systems File - systemmgmt

The **Systems** file (**/usr/lib/uucp/Systems**) contains the information needed by **uucp** to call and log on to a remote machine. Each entry represents one remote machine that can be called by your Basic Networking programs.

If the **Systems** entry is to be used to contact a machine that is hardwired to your system, refer to Chapter 9 for special instructions on setting up the **Systems** file.

Step 1: To add other machines to your **Systems** file, type **sysadm systemmgmt**, then select **2**.

```
# sysadm systemmgmt
Password:

Running subcommand 'systemmgmt' from menu 'uucpmgmt',
BASIC NETWORKING UTILITIES MANAGEMENT

This procedure is used to list, add, and delete entries in the
Basic Networking Utilities '/usr/lib/uucp/Systems' file.
This file contains information about what remote systems
can be called by cu and uucp commands.
You can also try to call a remote system that appears
in the '/usr/lib/uucp/Systems' file.

Type 'q' at any time to quit the current operation.
If a '?' appears as a choice, type '?' for help.

If a default appears in the question, type <RETURN> for the default.

Enter the operation you want to perform:

        1 list
        2 add
        3 delete
        4 call

(default list)[q]:  2
```

After you select **2** (add), the subcommand prompts you for the following information:

node name
: Node name of the system you want to call.

device type
: Type of device used to establish connection (e.g, acu).

baud rate
: The speed at which the device will place the call.

phone number
The telephone number of the remote machine. Special symbols can be embedded in the phone number, including abbreviations from the **Dialcodes** file (**/usr/lib/uucp/Dialcodes**).

login ID
Used by **uucp** to login on the remote machine.

password
The password associated with the above login.

Step 2: After you have entered the requested information it displays before it is entered into the **Systems** file.

Set Up Poll File - pollmgmt

The **Poll** file (**/usr/lib/uucp/Poll**) contains a list of the machines that are to be called (polled) by your system to see if they have anything to transmit to you. It also contains the times they are to be polled.

Step 1: To add entries to the **Poll** file, type **sysadm pollmgmt,** then select **2** (add).

```
$ sysadm pollmgmt
Password:

Running subcommand 'pollmgmt' from menu 'uucpmgmt',
BASIC NETWORKING UTILITIES MANAGEMENT

This procedure is used to list, add, and delete entries in the
Basic Networking Utilities '/usr/lib/uucp/Poll' file.
This file contains information about what systems and
the times (hours) the systems should be polled.

Type 'q' at any time to quit the current operation.
If a '?' appears as a choice, type '?' for help.

If a default appears in the question, type <RETURN> for the default.

Enter the operation you want to perform:

        1 list
        2 add
        3 delete

(default list) [q]:
```

The **pollmgmt** subcommand prompts you for the following information:

system name
: Name of the system you want to poll.

polling hours
: The hours you want to poll the system; must be an integer number between 0 and 23 (e.g., 0 4 8 12 16 20 is every four hours).

Step 2: After you have entered the requested information, it displays before it is entered into the **Poll** file.

Set Up Permissions File

The default **/usr/lib/uucp/Permissions** file provides the maximum amount of security for your system. The file, as delivered, contains the following entry:

```
LOGNAME=nuucp
```

You can set additional parameters for each machine to define:

- the ways it can receive files from your machine
- the directories it can read and write in
- the commands it can use for remote execution

See Chapter 9 for information on how to set up this file. If you want to change the contents of this file, you must edit it to modify the file and make the entries you desire.

Set Up Devconfig File

The **/usr/lib/uucp/Devconfig** file is only needed if you are using BNU a Streams-based transport provider that conforms to the Transport Interface (TI). If you have installed the Network Services Extension (NSE) and are using the TCP/IP transport provider supplied, the entry shown in the **Devconfig** file is all you need in this file. (**cu** using NSE is not supported.)

```
service=cu      device=tcpip  push=tirdwr
service=uucico  device=tcpip  push=tirdwr
```

You would also have to create an entry for tcpip in your **Devices** file. Description in the **Devices** file tell how to define TI devices. See Procedure 9.4 (supplied with NSE) for a complete example of setting up BNU on a LAN.

Devconfig entries define the STREAMS modules that are used for a particular TI device. (The **push=** variable shows the modules and the order they are pushed on to a stream.) Different modules and devices can be defined for **cu** and **uucp** services. If you want to change the contents of this file, you must use one of the editors (**ed** or **vi**) to modify the file and make the entries you desire.

Set Up Sysfiles File

/usr/lib/uucp/Sysfiles lets you assign different files to be used by **uucp** and **cu** as **Systems, Devices,** and **Dialers** files. Here are some cases where this optional file may be useful:

- You may want different **Systems** files so requests for **cu** login services can be made to addresses other than **uucp** services.
- You may want different **Dialers** files to use different chat scripts for **cu** and **uucp**.
- You may want to have multiple **Systems, Dialers,** and **Devices** files. The **Systems** file in particular may become large, making it convenient to split it into several smaller files.

The format of the **Sysfiles** file is described in Chapter 9. The following is an example of the file:

```
service=uucico   systems=Systems.cico:Systems
                 dialers=Dialers.cico:Dialers
                 devices=Devices.cico:Devices
service=cu       systems=Systems.cu:Systems
                 dialers=Dialers.cu:Dialers
                 devices=Devices.cu:Devices
```

If you want to change the contents of this file, you must use one of the editors (**ed** or **vi**) to modify the file and make the entries you desire.

Add uucp logins

You must add one or more administrative logins to your system so incoming **uucp** (**uucico**) requests from remote machines can be handled properly. Each remote machine should have an entry in its **Systems** file for your machine that contains the login ID and password that you add to your **/etc/passwd** file.

An example of a common entry in the **/etc/passwd** file is shown below.

```
nuucp:????:6:1:UUCP.Admin:/usr/spool/uucppublic:
/usr/lib/uucp/uucico
```

This entry shows that a login request by **nuucp** is answered by **/usr/lib/uucp/uucico**. The home directory is **/usr/spool/uucppublic**. The *????* will be replaced by an encrypted password that would be added using **passwd nuucp**.

Other Networking Files

There are three other files that affect the use of Basic Networking facilities. In most cases, the default values are fine and no changes are needed. If you want to change them, however, use a standard editor (**ed** or **vi**).

Maxuuxqts
defines the maximum number of **uuxqt** programs that can run at once.

Maxuuscheds
defines the maximum number of **uusched** programs that can run at once.

remote.unknown
is a shell script that executes when an unknown machine starts a conversation. It logs the conversation attempt that fails to make a connection. (If you change the permissions of this file so it cannot execute, your system will accept any conversation requests.)

Procedure 9.2: Basic Networking Maintenance

Purpose	To keep files related to basic networking from consuming too much disk space.
When Performed	Automatically with **cron**(1M) or as needed.
Starting Conditions	System state —2 (multi-user) or 1 (single-user)

Basic Networking Utilities comes with four shell scripts that poll remote machines, reschedule transmissions, and clean up old log files and unsuccessful transmissions. These shell scripts should be executed regularly to keep your basic networking running smoothly. Normally, they are run automatically with **cron**(1M), though they can also be run manually. The few areas needing clean up that are not handled by these shell scripts should be maintained manually.

Automated Networking Maintenance (cron)

The Basic Networking Utilities are delivered with entries for **uudemon** shell scripts in the **/usr/spool/cron/crontabs/root** file. These entries automatically handle some BNU administrative tasks for you. Each shell script is in **/usr/lib/uucp**.

When the system is in run state 2 (multi-user), **cron** scans the **/usr/spool/cron/crontabs/root** file every minute for entries scheduled to execute at that time. As the UUCP administrator, you should become familiar with **cron** and the four **uudemon** shell scripts.

uudemon.poll

The **uudemon.poll** shell script, as delivered, does the following:

- Reads the **Poll** file (**/usr/lib/uucp/Poll**) once an hour.

- If any of the machines in the **Poll** file are scheduled to be polled, a work file (**C.sysnxxxx**) is placed in the **/usr/spool/uucp/***nodename* directory, where *nodename* is replaced by the name of the machine.

The shell script is scheduled to run twice an hour just before **uudemon.hour** so that the work files will be there when **uudemon.hour** is called. The default root crontab entry for **uudemon.poll** is:

```
1,30 * * * * "/usr/lib/uucp/uudemon.poll > /dev/null"
```

uudemon.hour

The **uudemon.hour** shell script you receive with your machine does the following:

- Calls the **uusched** program to search the spool directories for work files (C.) that have not been processed and schedules these files for transfer to a remote machine.
- Calls the **uuxqt** daemon to search the spool directories for execute files (X.) that have been transferred to your system and were not processed at the time they were transferred.

The default root crontab entry for **uudemon.hour** is as follows:

```
41,11 * * * * /usr/lib/uucp/uudemon.hour > /dev/null
```

As delivered, this is run twice an hour. You may want it to run more often if you expect high failure rates.

uudemon.admin

The **uudemon.admin** shell script, as delivered, does the following:

- Runs the **uustat** command with **–p** and **–q** options. The **–q** reports on the status of work files (C.), data files (D.), and execute files (X.) that are queued. The **–p** prints process information for networking processes listed in the lock files (**/usr/spool/locks**).
- Sends resulting status information to the **uucp** administrative login via mail.

There is no default entry **/usr/spool/cron/crontabs/root** for **uudemon.admin**. The following is recommended:

```
48 8,12,16 * * * /bin/su uucp -c
 "/usr/lib/uucp/uudemon.admin"> /dev/null
```

uudemon.cleanup

The delivered **uudemon.cleanu** shell script does the following:

- Takes log files for individual machines from the **/usr/spool/uucp/.Log** directory, merges them, and places them in the **/usr/spool/uucp/.Old** directory with other old log information. If log files get large, the ulimit may need to be increased.
- Removes work files (C.) 7 days old or older, data files (D.) 7 days old or older, and execute files (X.) 2 days old or older from the spool files.
- Returns to the sender mail that cannot be delivered.
- Mails a summary of the status information gathered during the current day to the UUCP administrative login (**uucp**).

No default root crontab entry for **uudemon.cleanup** is delivered. This is a recommended entry:

```
45 23 * * * ulimit 5000; /bin/su uucp -c
"/usr/lib/uucp/uudemon.cleanup" > /dev/null 2>&1
```

Manual Maintenance

Some files may grow indirectly from **uucp** and other basic networking activities. Here are two files you should check and delete if they have become too large:

/usr/adm/sulog
: This file keeps a history of all super user commands. Since the uudemon entries in the **/usr/cron/root** file use the **su** command, the **sulog** will grow over time. You should delete this file if it becomes too large.

/usr/lib/cron/log
: This file is a log of cron activities. While it grows with use, it is automatically truncated when the system goes to the multi-user state.

Procedure 9.3: Basic Networking Debugging

Purpose	To use available monitoring tools to solve basic networking problems.
Starting Conditions	System state —2 (multi-user) or 1 (single-user)
Commands	**uustat**(1C) **cu**(1C) **Uutry**(1M) **uuname**(1C) **uulog**(1C) **uucheck**(1M)

These procedures describe how to solve common problems that may be encountered with Basic Networking Utilities.

Check for Faulty ACU/Modem

You can check if the automatic call units or modems are not working properly in several ways:

- Run **uustat –q**. This gives counts and reasons for contact failure.
- Run **cu –d –l***line*. This lets you call over a particular line and print debugging information on the attempt. The line must be defined as Direct in the devices file. (You must add a telephone number to the end of the command line if the line is connected to an autodialer or the device must be set up as **direct**.)

Check Systems File

Check that you have up-to-date information in your systems file if you are having trouble contacting a particular machine. Some things that may be out of date for a machine are its:

- Phone number
- Login
- Password

Debug Transmissions

If you are unable to contact a particular machine, you can check out communications to that machine with **Uutry** and **uucp**.

Step 1: To try to make contact, run:

$ /usr/lib/uucp/Uutry –r *machine*

where *machine* is replaced with the node name of the machine you are having problems contacting. This command:

1. Starts the transfer daemon (**uucico**(1M)) with debugging. You get more debugging information if you are **root**.
2. Directs the debugging output to **/tmp/***machine*,
3. Prints the debugging output to your terminal (**tail –f**). Hit **BREAK** to end output.

You can copy the output from **/tmp/***machine* if you want to save it.

Step 2: If **Uutry** doesn't isolate the problem, try to queue a job by running:

$ uucp –r *file machine*!*/dir/file*

where *file* is replaced by the file you want to transfer, *machine* is replaced by the machine you want to copy to, and *dir/file* is where the file will be placed on the other machine. The **–r** option queues a job but does not start the transfer.

Now use **Uutry** again. If you still cannot solve the problem, you may need to call support personnel. Save the debugging output; it will help diagnose the problem.

Check Error Messages

There are two types of error messages for Basic Networking Utilities: ASSERT and STATUS. See Appendix C for a listing of these messages.

ASSERT Error Messages

When a process is aborted, ASSERT error messages are recorded in **/usr/spool/uucp/.Admin/errors**. These messages include the file name, **sccsid**, line number, and text. These messages usually result from system problems.

STATUS Error Messages

Status error messages are stored in the **/usr/spool/uucp/.Status** directory. The directory contains a separate file for each remote machine your system attempts to communicate with. These files contain status information on the attempted communication and whether it was successful.

Check Basic Information

There are several commands you can use to check for basic networking information.

uuname

Use this command to list those machines your machine can contact.

uulog

Use this command to display the contents of the log directories for particular hosts.

uucheck –v

Run this command to check for the presence of files and directories needed by **uucp**. This command also checks the **Permissions** file and outputs information on the permissions you have set up.

Procedure 9.4: Set Up BNU TCP/IP NETWORK

Purpose	To show how to set up BNU files for a TCP/IP NETWORK to handle **uucico** services.
Starting Conditions	System state —2 (multi-user) or 1 (single-user) Login —**root**
Commands	Any text editor (**ed**(1), **vi**(1))

To improve performance of **uucico** services, you can have **uucico** requests connect directly to remote machines on the TCP/IP NETWORK without going through login.

You should only set up **uucico** as described above if you trust all machines on your network. The reason is that there is a potential security breach since you would not be requesting a password when a machine tries to transfer files to your machine.

Step 1: The systems you want to transfer files to on your network without going through login must be listed in the **Systems** file.

Edit **/usr/lib/uucp/Systems** and add entries similar to the following entry for each system.

```
M68K Any tcpip - \000\002\004\001\131\002\002\001
```

where:

M68K
is the system name

Any
is when to call

tcpip
is a device type in the **Devices** file

-
is a field place holder

`\000\002`
is the address family, which is constant for all *tcpip* device type entries.

`\004\001`
is the **tcpip** port ID (1025 decimal) of the listener, which is constant for all *tcpip* device type entries.

`\131\002\002\001`
is the octal representation of the M68K internet address 89.2.2.1.

Step 2: Continuing with the example above, from the entry in the **Systems** file, add the following entry called **tcpip** to the **/usr/lib/uucp/Devices** file:

```
tcpip,e tcpip - - TLIS \D NLS
```

where:

first `tcpip`
is the device name

`,e`
is the protocol selection for guaranteed error-free delivery

second `tcpip`
is the network name

`- -`
is the field place holders

`TLIS`
is the network type

`\D`
is the instruction: read in the port ID and network address `\000\002\004\001\131\002\002\001`

`NLS`
is the dialer type

Step 3: The **/usr/lib/uucp/Devconfig** file entries must be created to define the **STREAMS** modules to **push** for **uucico** service:

```
service=uucico  device=tcpip  push=tirdwr
```

The line is already in the **Devconfig** file. To activate it, remove the comment character (#) in front of each line.

Step 4: From the entry in the **Devices** file, you must add an entry to the **/usr/lib/uucp/Dialers** file called **NLS**. The following is what you should enter:

```
NLS "" "" NLPS:000:001:101\N\c
```

Step 5: You must register the **uucico** service with the network listener. The following command must all be typed on one line:

 # nlsadmin –a 101 –c "/usr/lib/uucp/uucico –r 0 –i TLIS –u \ nuucp" –y "uucico server with NO login checking" tcpip

This entry says to answer requests for service code 101 with **uucico** directly as it is used for a **TLIS** (Transport Interface) network. Notice that the default **nuucp** login ID is used; you can use a different one if you choose.

P10 Remote File Sharing Procedures

Introduction

Procedure 10.1 **Set Up Remote File Sharing (setuprfs)**
To set up all basic information needed to run Remote File Sharing (RFS).

Procedure 10.2 **Start/Stop RFS (startstop)**
To start and stop RFS, check if it is currently running, and set up RFS to start automatically at system boot time.

Procedure 10.3 **Local Resource Advertising (advmgmt)**
To manage the local resources you make available to other machines.

Procedure 10.4 **Remote Resource Mounting (mountmgmt)**
To manage remote resources made available to your machine.

Procedure 10.5 **Change RFS Configuration (confgmgmt)**
To change your ID mapping, show your current RFS configuration, or update the domain member list.

These procedures are designed to help you set up and maintain RFS Utilities on your computer. Before you run any of these procedures, do the following:

- Read the *SYSTEM V/88 Software Release Guide (SRG)*. This will tell you how to install RFS and help you with some special problems you may run into.

The first time you set up RFS you should use the **sysadm** interface available with RFS. The interface not only lets you add all basic RFS configuration information, but it also acts as a tutorial by introducing and explaining key RFS concepts.

The following are the two recommended ways you can use the **sysadm** interface for RFS.

- Type **sysadm rfsmgmt**. This displays the top Remote File Sharing Menu. From there you can select the function or submenu you want, such as **setuprfs** to set up RFS for your system. Browsing through the **sysadm rfsmgmt** menus is a good way to familiarize yourself with the functions available.

- Type **sysadm** *subcommand*, where *subcommand* is replaced by the specific **sysadm** subcommand you want to use. Procedures 10.1 through 10.5 describe how to go directly to the **sysadm** subcommand you need for different RFS administrative tasks.

A diagram of the **sysadm rfsmgmt** subcommand tree is shown below. The diagram also notes the section of this document where each subcommand is described.

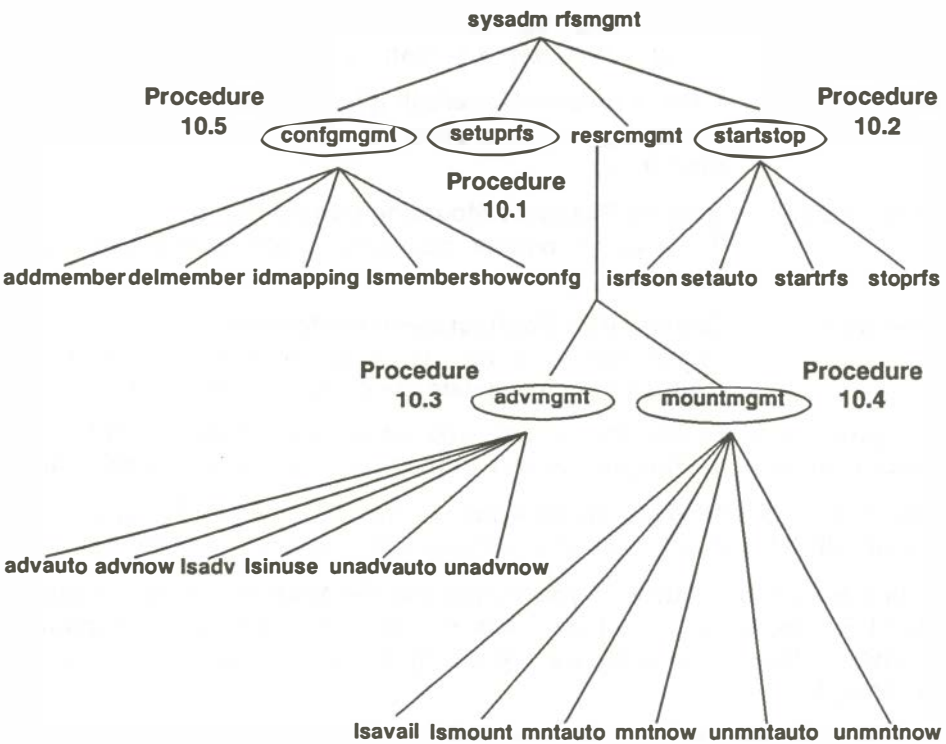


Figure 10-1. sysadm rfsmgmt Subcommands

Should you need more information as you are setting up RFS, there are several things you can do:

- Type **?**. A message is printed at your terminal that further describes what you need to know to complete the step.
- Check the glossary. A glossary of terms is provided at the end of this section to explain important RFS terms.
- Read Chapter 10. The *Remote File Sharing* chapter provides the most complete description of the RFS software. The first-time RFS administrator will be particularly interested in the *Overview* section.

The **sysadm** interface lets you do everything necessary to set up and run RFS in a basic configuration. There are several optional features that are not available through the **sysadm** interface, however.

The optional features not available using the **sysadm** interface are described at the end of the *Setting Up RFS* section of Chapter 10. The word "Optional" is placed in the heading of each optional feature. The features include the following:

Remote Computer Verification
By default, when a machine requests the use of one of your resources, your machine will process the request without verifying the remote machine's password. This procedure describes how to restrict access of all your resources to a limited group of remote machines whose names and passwords match those in lists you set up.

Complex user ID/group ID mapping
ID mapping defines the permissions remote users have to your resources. The choices of ID mapping schemes are limited when you use the **sysadm** interface to set up mapping. This procedure gives you more flexibility in setting up permissions for remote users.

Multiple Domain Resource Sharing
The **sysadm** interface assumes that you are only sharing resources within one domain. However, it is possible to have more than one domain on a network. This procedure describes how to share resources among multiple domains on the same network.

Multiple Domain Name Service
When you define the primary and secondary name servers using **sysadm**, you are defining them to serve a single domain. You can, however, define the same set of machines to be the name server for several domains using this procedure.

More experienced RFS administrators will also be interested in the *Monitoring* and *Parameter Tuning* sections of Chapter 10. Information in these sections will help you fine tune your system so RFS can make the most efficient use of your system's resources.

RFS Glossary

Advertise
To make a local resource available to other computers using RFS. The **adv**(1M) (or **sysadm advnow**) command is used by administrators to advertise a resource.

Advertise Table
An internal list of available resources. An advertise table on each computer running RFS has the name of each resource the computer has made available.

Automatic Advertise List
A list of local resources that are automatically offered to other computers when RFS is started. The list consists of full **adv** command lines placed inside the **/etc/rstab** file. The command lines are added to **/etc/rstab** using the **sysadm advauto** command or by any standard file editor.

Automatic Mount List
A list of remote resources that are mounted on the local system when RFS is started. The list is contained in the **/etc/fstab** file. (See the **fstab**(4) manual page in the *System Administrator's Reference Manual* for the format of the file.) Automatic mount information is added to **/etc/fstab** using the **sysadm mntauto** command or by any standard file editor.

Client
A RFS computer that is using a remote resource.

Client Caching

The ability of an RFS computer that is using a remote resource to store remote data blocks in its local buffer pools. This technique improves RFS performance by reducing the number of times data must be read across the network.

Client List

When an RFS administrator advertises a resource, the administrator can restrict the resource so only certain remote machines can use it. This list of machines is added to the **adv**(1M) command line when a resource is advertised.

Client Permissions

When an RFS administrator advertises a resource, the administrator can set permissions for the resource. The permissions are assigned on the **adv**(1M) command line. If the permissions are read-only, the client computers can only mount the resource with read permissions. If they are read/write, a client can mount the resource read/write or read only.

Current Name Server

When a domain is set up, a primary and zero or more secondary domain name servers are assigned. Only one of those machines is actually handling domain name server responsibilities at a time. That machine is referred to as the current name server. Normally, the primary is the current name server. However, if the secondary has taken over temporarily, it is the responsibility of the secondary's administrator to pass the responsibility back to the primary whenever the primary resumes running RFS. (See also **Domain**, **Primary Name Server**, and **Secondary Name Server**.)

Directory Pathname

RFS asks you for a full pathname to a directory in two instances. When you advertise a local resource, you need the full pathname of the directory you are advertising. When you mount a remote resource, you need the full pathname of the directory where the remote resource should be attached.

Domain
A logical grouping of computers in an RFS environment. A domain name is like a telephone area code, acting as an addressing prefix to attach to a computer name or a resource. Assigning a domain name server for a domain provides a central location where lists of resources and network addresses for the group of computers can be stored. Domains also provide a level of security.

Domain Information (rfmaster)
The primary and secondary name server assignments for a domain are stored in the **/usr/nserve/rfmaster** file. The primary keeps the definitive copy of this file and distributes it automatically to each computer in the domain when each starts RFS. It also contains the network address of each name server.

Domain Member List
The list of the computers that make up an RFS domain. This list is stored in the **/usr/nserve/auth.info/***domain***/passwd** file on the primary name server, where *domain* is replaced by the name of the domain. Members are added on the primary using the **rfadmin –a** or **sysadm addmember** commands.

Forced Unmount
To unmount one of your local resources from all remote machines that have mounted it. This has the effect of killing all processes that are currently using the resource on all client machines.

ID Mapping
To define the permissions remote users and groups have to your advertised resources. The tools available for mapping let you set permissions on a per-computer basis and on a global basis. You can then map individual users or groups by ID name or number. When you map IDs for RFS, it is easiest to do so with ID numbers since mapping by name requires that you have copies of the remote machines' **/etc/passwd** and **/etc/group** files.

Local Resource

A directory that resides on your machine that you have made available for other computers running RFS to use. You must advertise the directory (**adv**(1M) command) to offer it to other computers. If a remote machine mounts your resource, it could have access to all subdirectories, files, named pipes, and devices within your directory (depending on file permissions you set up).

Mount

The special use of **mount**(1M) in RFS is to attach a remote resource to a directory on your system so local users can access the remote resource. Once mounted, the remote resource appears to be just another part of the local SYSTEM V/88 file system tree. See the **mount –d**(1M) command and the **sysadm mountmgmt** procedures.

Network Specification

The name that identifies a networking product that is compatible with the Transport Interface (TI). This is also referred to as the transport provider. RFS requires a transport provider to communicate with other machines. The network specification is used to tell RFS the exact device to use for communications.

Network Listener

The process used by a transport provider to wait for any type of incoming requests from the network. Once a request comes in, the listener directs it to one of the processes registered with the listener. The process represents a service, such as **uucp** or RFS.

Networking Support Utilities

A software package that contains the network listener. This package must be installed to use RFS.

Network Address

The address by which a computer is known to a particular network. An RFS administrator needs to know the network address of the primary to start RFS for the first time. Address information of other machines is handled internally by RFS.

Node Name
The name you assign to your computer to use for communications needs (use **sysadm nodename** to change it). Networking software, e.g, BNU and RFS, use this name to identify your machine. A full RFS computer name is *domain.nodename*, where *domain* is the name of the computer's RFS domain.

Primary Name Server
The computer that is assigned to provide a central location for addressing and information collection for an RFS domain. Information includes a list of domain members, resources offered by domain members, and optional user ID mapping information. Secondary name servers can be assigned to continue limited name service when the primary is down. For example, a secondary cannot add or delete domain members.

RFS Automatic Startup
RFS can be set to start automatically when your machine is booted. This is done by changing the **initdefault** line in the **/etc/inittab** file from **2** to **3**. The **sysadm setauto** command does this for you automatically. (**init** state **3** is the RFS/Multi-user state.)

RFS Daemon (rfudaemon)
A daemon process that runs when RFS is running. When network connections to remote resources are broken, **rfudaemon** sends a message to **rfuadmin**, which then continuously tries to remount the resource. (See **rfudaemon**(1M) and **rfuadmin**(1M) for further information.)

RFS Password
A password assigned by the primary name server for every computer in its domain. Each computer must enter its password the first time it starts RFS. After that the password is stored locally in **/usr/nserve/loc.passwd**. By copying the domain password file from the primary (**/usr/nserve/auth.info/***domain*/**passwd**), a computer can verify that a remote machine trying to mount its resource is the machine it claims to be.

Remote File Sharing State (init 3)
The special initialization state used to start RFS. When you type **init 3** or set the **initdefault** line in **/etc/inittab** to **3**, your system will start RFS, advertise all resources in your automatic advertise list, and mount all resources in your automatic mount list.

Remote Resource
A directory that resides on a remote machine that is available for you to connect to using RFS. You must mount the resource (**mount**(1M) or **sysadm mntnow** commands) to make it available to users on your system. Once you mount the remote resource, your users could have access to all subdirectories, files, named pipes and devices related to your directory (depending on file permissions the remote machine set up).

Resource
See **Remote Resource** and **Local Resource**.

Resource Identifier
The name assigned to a resource when it is advertised. The name is limited to 14 printable ASCII characters; slash (/), period (.), and white space may not be used.

Secondary Name Server
A computer designated to take over name server responsibilities temporarily should the primary domain name server fail. The secondary cannot change any domain information. It can, and should, only pass name server responsibility back to the primary when RFS is running on the primary again.

Server
A RFS computer that offers a resource to others.

Transport Provider
The software that provides a path through which network applications can communicate. RFS can communicate over any transport provider that meets the criteria in Appendix B of the *NSE, Part 1*.

User/Group Name
The names associated with each local user and group that is allowed access to your computer. This information can be found in the first field of the **/etc/passwd** or **/etc/group** files, respectively. Remote users and groups can be assigned the same permissions as the local users and groups by using RFS ID mapping.

User/Group ID Number

Every user and group name has a corresponding number that is used by the SYSTEM V/88 operating system to handle permissions to files, directories, devices, etc. These numbers are defined in the third field of the **/etc/passwd** or **/etc/group** files, respectively. Remote users and groups can be assigned the same permissions as the local users and groups by using RFS ID mapping.

Procedure 10.1: Set Up Remote File Sharing (setuprfs)

Purpose	To set up everything needed to run RFS Utilities.
Starting Conditions	System state —**2** (multi-user) You must **mount /usr** to run this procedure in single-user mode. Login —**root**
Commands	**sysadm setuprfs**
Reference	Chapter 10, *Remote File Sharing*

This procedure is used to set up RFS on your machine. When the procedure is done, you will have completed everything needed to run RFS on your system.

NOTE

When you complete this procedure, **sysadm** creates the file **/usr/nserve/rfmaster**. If you had previously created that file, the old version is placed in **/usr/nserve/Orfmaster**.

Prerequisites

Before you begin setting up RFS, you must do the following.

- Install software and hardware as described in the *SYSTEM V/88 Software Release Guide*. (The order of software installation in the SRG is very important!)

- Choose one or more computers on the network to act as domain name servers. Exactly one primary is required. All domain administration is done from the primary. You can choose zero or more secondary domain name servers. These are defined simply to keep the name server running temporarily, should the primary fail. (You can configure RFS on your machine before the primary is configured and running RFS. However, you cannot start RFS until the primary begins running RFS.)
- Log in as **root**.

Set Up RFS

Step 1: Type **sysadm setuprfs**. The following information displays:

```
                    SET UP RFS

If you have not yet set up RFS on your machine, this subcommand will
walk you through all steps necessary to set up RFS.  These steps are:

   * entering the name of the transport provider
   * designating your machine as a primary name server or a non-primary
   * entering the domain name
   * primary only: adding members to domain (optional)
   * primary only: designating secondary name servers (optional)
   * setting up ID mappings (optional)
   * putting local resources on the Automatic Advertise List (optional)
   * putting remote resources on the Automatic Mount List (optional)
   * setting up RFS to start automatically (optional)
   * starting RFS (optional)

If these terms are unfamiliar do not be concerned.  All new terms and
concepts will be explained.  Explanations of RFS terms and concepts can
also be found in your System Administrator's Guide.
```

continued

In most cases, the "setuprfs" subcommand will call other RFS sysadm menus to do the specific tasks. Later, when you want to do these tasks as part of your ongoing RFS administration you will use these specific RFS sysadm menus.

There are a few tasks (that you won't need to do very often) that can not be done through the specific menus. To do these you will have to stop RFS and rerun this "setuprfs" subcommand. These tasks are:

- **changing the name of the transport provider**
- **changing the name of the domain**
- **recognizing a new primary name server**
- **primary only: adding secondary name servers**

If you are running this command to change something in your domain configuration, any old domain configuration information will be saved for you in /usr/nserve/Orfmaster.

This "setuprfs" subcommand assumes you have already installed a transport provider and that the network listener is properly set up.

You will be unable to set up RFS until these assumptions are met.
SET THE TRANSPORT PROVIDER ...

To share resources between machines, the machines must be connected in some way. Because RFS is media independent, it is not tied to any one type of connection medium. The connection between machines, the TRANSPORT PROVIDER, may be any network that is compatible to the Transport Interface. The System Administrator's Guide explains the Network Support Utilities and how they work together with RFS.

RFS assumes that this transport provider has been installed.

Enter the name of your transport provider [?, q]:

Step 2: Answer each question as it is asked. If you are setting up RFS for the first time, you should answer all questions. If you make a mistake or want to change something later, you can run **sysadm setuprfs** again.

The following is a list of information you are asked:

Transport provider
The name of the transport provider that RFS will use to communicate across the network. If you are told that the listener is not running, see the *Set Up Network Listener* section of Chapter 10.

Machine type
This tells your machine whether or not it is the primary name server for your domain.

Domain name
This is the name of your domain. It can be up to 14 characters in any combination of letters, digits, hyphens, and underscores. This name is the same for all computers in your domain.

Non-primary
If your machine *is not* the primary name server, you are asked to enter the following:

Primary's node name
You must enter the primary's node name, then its network address; address formats are different for different networks.

RFS password
The RFS password for your machine. This must match the password entered for your machine by the primary domain name server.

Primary
If your machine *is* the primary, you are asked for the following:

RFS password
You are asked twice to enter a password for your machine. Enter the same password both times.

Add domain members

You are asked to enter the machine names of every machine in your domain. Do not worry if you do not know them all now, you can always add and delete members later.

Define secondaries

You are asked to enter the name and network address of the secondary name server(s) for your domain; this is optional. Secondaries are only there to take over temporarily when the primary is not available. The network address you add is in the same format as that of the primary.

User and group ID mapping

Since user ID and group ID mapping can be complex, this interface presents you with several standard options. Choice **a** (all remote IDs into a guest ID) is the best choice in most cases since it provides the maximum security and the minimum complexity. Choices **b** or **c** (each remote ID number to the corresponding local ID number) are only valuable if you have identical **/etc/password** and **/etc/group** files among machines with which you share resources. The difference between **b** and **c** is that **b** protects all administrative logins and **c** only protects **root** login. (See Chapter 10 for information on more complex ID mapping schemes.)

Auto advertise list

If there are some local directories you are ready to share, you can add them to a list of those that are automatically offered when you start RFS. You must know the pathname to that directory, make up a name to identify it (resource identifier), decide if you want to restrict it to read-only access, enter a short description, and enter a list of client machines (if you only want certain machines to access the resource).

Auto mount list

If there are remote resources that will be available from other machines in your domain, you can add them to your automatic mount list. Those resources are automatically mounted on your machine when RFS is started. Usually, **/mnt** is a safe place to use as a mount point directory for the first remote resource you try with RFS. (Read the *Mount Guidelines* section in Chapter 10 for information on where you should not mount remote resources.)

Automatic RFS start

If you select the automatic start feature, your machine automatically starts RFS when it is booted. This places your machine in **init** state **3**. (See Chapter 10 for a detailed description of **init 3** processing.)

Start RFS now

If the primary name server is up and running RFS and if everything was done correctly in the previous steps, you should be able to start RFS immediately. Be patient. It may take several minutes.

CONFIGURATION IS COMPLETE!

To make sure RFS is working, you could return to the shell and do the following:

- Type **rfadmin –q** (or **sysadm isrfson**) to see if RFS is running.
- Type **adv** (or **sysadm lsadv**) to list the local resource you have advertised.
- Type **mount** (or **sysadm lsmount**) to list the remote resources you have mounted. (**mount** also lists locally mounted devices.)

If you want more information on some of the more complex security and resource sharing facilities available, see Chapter 10. The following chart describes the **sysadm setuprfs** functions and lists dependent commands and files of the procedure you just completed. This information should help you relate the **sysadm** processing to the descriptions of RFS components in Chapter 10.

Table 10-1. **sysadm setuprfs** Description

Sysadm Commands	Dependent Commands	Files	Description
setuprfs		**rfmaster** *	Set up basic RFS information.
	dname -N		Define the Transport Provider used by RFS.
	dname -D		Define the domain your machine is a member of.
	idload	**uid.rules** ** **gid.rules** **	Choose mapping scheme. (See **sysadm idmapping**.)
	adv	**/etc/rstab**	Advertise automatically. (See **sysadm advauto**.)
	adv **nsquery** **mount –d**	**/etc/fstab**	Mount automatically. (See **sysadm mntauto**.)
		/etc/inittab	Set RFS to start automatically. (See **sysadm setauto**.)
	rfstart	**/etc/rstab** **/etc/fstab**	Start RFS now. (See **sysadm startrfs**.)
***	**rfadmin –a**		Add machines to domain. (See **sysadm addmember**.)
***		**rfmaster** *	Define secondary machine(s).

* This file appears in the directory **/usr/nserve**.

** This file appears in the directory **/usr/nserve/auth.info**.

*** You will be prompted for this information only if you are on the primary name server.

Procedure 10.2: Start/Stop RFS (startstop)

Purpose	Start and stop RFS.
Starting Conditions	System state —**2** (multi-user) or **3** (if RFS is running) Login —**root**
Commands	**sysadm startstop**
Reference	Chapter 10, *Remote File Sharing*

The **sysadm startstop** command presents a menu of subcommands that can be used to start RFS, stop RFS, check to see if RFS is currently running, and set RFS to start automatically at boot time.

Prerequisites

Before you run **sysadm startstop** subcommands, do the following.

- Set up RFS. Use **sysadm setuprfs** to enter basic RFS information.

The following chart describes the **sysadm startstop** subcommands and provides information on dependent commands and files. This will help you relate **sysadm** processing with the RFS components described in Chapter 10.

Table 10-2. sysadm startstop Subcommands

Sysadm Commands	Dependent Commands	Files	Description
isrfson			Reports whether RFS is running.
setauto		**/etc/inittab**	Set up your system so that when it is booted RFS is automatically started. This command also lets you set up your system to NOT start RFS automatically when it is booted.
startrfs	**rfstart**(1M) **rmountall**(1M)	**/etc/rstab** **/etc/fstab**	Start RFS immediately. When you start RFS with this command, the command also advertises and mounts all resources in your Automatic Advertise List and Automatic Mount List, respectively.
stoprfs	**rfstop**(1M) **fumount**(1M) **unadv**(1M) **fuser**(1M)		Stop RFS immediately. Unmount any resources mounted on your machine. Unadvertise all of your resources and force them to be unmounted from other machines.

Check if RFS is Running

Type **sysadm isrfson** to see if RFS is currently running. If it is, you see the following:

```
RFS is running.
```

Set RFS to Start Automatically

Step 1: Type **sysadm setauto**. You then see the following:

```
SET UP / TURN OFF AUTOMATIC START OF RFS

You can set up your system so that any time you start your machine the
system automatically starts RFS.  When RFS is automatically started,
all the resources you have set up to be automatically advertised and
mounted are advertised and mounted just as if you had manually started
RFS by using the "startrfs" subcommand.

You also use this "setauto" subcommand to turn off automatic starting
of RFS.  When you turn off the automatic start option the machine will
be set to come up in multi-user mode when the system starts.
```

Step 2: If RFS automatic start-up was off, type **y** to turn it on. If RFS automatic start-up was on, type **y** to turn it off.

The **sysadm setauto** command sets up automatic start-up by changing the **initdefault** line in the **/etc/inittab** file to **3** (RFS state). When you turn it off, **initdefault** is set back to **2**. See Chapter 10 for other information on the RFS state (**init 3**).

Start RFS Now

Step 1: Type **sysadm startrfs** to start RFS. If RFS starts successfully, you see the following:

```
                START REMOTE FILE SHARING

Attempting to start RFS.
This will take a few moments.  Please wait...

    The system will now ATTEMPT to mount the resources you have
    set up to be automatically mounted.  There are several conditions
    that may exist that would prevent a resource from being
    successfully mounted.  These are:
      - The machine owning the resource is not running RFS,
      - The machine owning the resource does not have it advertised,
      - The resource identifier in the Automatic Mount List is incorrect,
      - The local directory you chose as a mount point does not exist,
      - The local directory you chose as a mount point is busy.
    Your system will continue to try to mount a resource if it
    cannot be mounted when RFS is first started.

                RFS has been started.
```

If RFS does not start, see the *Starting/Stopping RFS* section of Chapter 10 for a list of possible problems.

Stop RFS Now

Step 1: Type **sysadm stoprfs** to stop RFS and answer **y** to the first prompt. If RFS was running and could be stopped successfully, you see the following:

```
                    STOP REMOTE FILE SHARING

Stopping RFS will make remote resources unavailable to your local users
and will make your local resources unavailable to remote users.
Processes using these resources will be killed.  Remote users currently
in one of your directories will be logged off.  Be sure to consider the
affect to local and remote users before stopping RFS.

Do you want to continue to stop RFS? [y, n, q]
```

Procedure 10.3: Local Resource Advertising (advmgmt)

Purpose	Share your resources with other machines.
Starting Conditions	System state —**2** (multi-user) or **3** (RFS state) Login —**root**
Commands and subcommands	**sysadm advmgmt** **sysadm advauto** **sysadm advnow** **sysadm lsadv** **sysadm lsinuse** **sysadm unadvnow** **sysadm unadvauto**
Reference	Chapter 10, *Remote File Sharing*

You can selectively share parts of your computer's file system with other machines on your RFS network using the **sysadm advmgmt** subcommands.

Prerequisites

Before you run **sysadm advmgmt** subcommands, you should do the following:

- Set up RFS. Use the **sysadm setuprfs** command to enter the basic information needed to run RFS.
- Start RFS. To use **advnow**, **unadvnow**, or **lsinuse** subcommands, RFS should be running. The other subcommands do not require that RFS be running.

The following chart describes the **sysadm advmgmt** subcommands and lists dependent commands and files of the procedures in this section. This information should help you relate the **sysadm** processing to the descriptions of RFS components in Chapter 10.

NOTE

You will note variations between the screen interfaces for the subcommands listed previously.

Table 10-3. sysadm advmgmt Subcommands

Sysadm Commands	Dependent Commands	Files	Description
advauto	**adv**(1M)	**/etc/rstab**	Set up a resource to be advertised each time RFS is started. The information you enter for each resource will be added to the Automatic Advertise List.
advnow	**adv**(1M) **nsquery**(1M)		Advertise a resource immediately.
unadvauto		**/etc/rstab**	Remove a resource from the Automatic Advertise List. Once removed, the resource will not be advertised the next time RFS is started.
unadvnow	**unadv**(1M) **fumount**(1M) **fuser –k**		Immediately unadvertise a resource, then (optional) unmount it from all client machines.
lsinuse	**rmntstat**(1M)		Print a list of all the machines that have mounted your advertised resources. For each resource the list includes: the Resource Identifier, the pathname to the resource, and the name of each client machine.
lsadv	**adv**(1M)	**/etc/rstab**	Print a list of all your resources that are currently advertised and those that are on your automatic advertise list.

Advertise Automatically

Step 1: Type **sysadm advauto** to add an entry to the Automatic Advertise List. You then see the following:

```
        ADD RESOURCES TO THE AUTOMATIC ADVERTISE LIST

You share your local resources with other machines by advertising
them.  The machines that can use the resources you advertise are
called clients.

This subcommand lets you add resources to the Automatic Advertise List
to be advertised every time RFS is started.  You will be asked for all
the information needed.

NOTE:  Adding resources to the Automatic Advertise List does NOT
make the resources immediately available to clients.  If RFS is
running when you access this command you may be able to immediately
advertise the resources.  The next time RFS is started the resources
will be automatically advertised for you.

Enter the full pathname of the local directory to be added to
the automatic advertise list [?, q]:
```

Step 2: Enter each item of information as it is asked. This information includes:

Directory pathname
The local directory you want to share.

Resource identifier
The name you assign to the resource.

Remote permissions
Read-only or read/write permission.

Description
Up to 32-character description of the resource.

Client list
The machines that can use the resource.

You will also be asked if you want to advertise the resource immediately if RFS is running, the pathname exists, and the resource is not being used.

The **sysadm advauto** command sets up a resource to be automatically advertised by adding a complete **adv**(1M) command line in the **/etc/rstab** file. See Chapter 10 for information on the **adv** command.

Remove Automatic Advertises

Step 1: Type **sysadm unadvauto** to remove an entry from the Automatic Advertise List. You see the following:

```
REMOVE LOCAL RESOURCES FROM AUTOMATIC ADVERTISE LIST

When you no longer want to share a resource with client machines
you unadvertise it.  Resources that have been set up to be
advertised automatically, however, will be readvertised by the
system the next time RFS is started unless they are removed from
the Automatic Advertise List.

This subcommand lets you remove resources from the Automatic
Advertise List.
NOTE: This subcommand does NOT unadvertise immediately.  To
unadvertise immediately use "unadvnow".

Enter the resource identifier of the resource to be removed from
the Automatic Advertise List [?, q]:
```

Step 2: Type the resource identifier of the resource you want to remove.

The **sysadm unadvauto** command removes the automatic advertise by deleting an **adv**(1M) command line from the **/etc/rstab** file.

Advertise Immediately

Step 1: Type **sysadm advnow** to advertise a local resource immediately. You see the following:

```
ADVERTISE LOCAL RESOURCES IMMEDIATELY
You share your local resources with other machines by advertising them.
The machines that can use the resources are called clients.

This subcommand lets you immediately advertise resources.  As soon
as a resource is advertised it is available to client machines.
You will be asked for all the information needed.

NOTE:  This subcommand does NOT add resources to the Automatic
Advertise List.  Use the subcommand "advauto" to add resources
to this list.

Enter the full pathname of the local directory to
advertise [?, q]:
```

Step 2: Enter each item of information as it is asked. This information will include the following:

Directory pathname
The local directory you want to share.

Resource identifier
The name you assign to the resource.

Remote permissions
Read-only or read/write permission.

Description
Up to 32-character description of the resource.

Client list
The machines that can use the resource.

Unadvertise Immediately

Step 1: Type **sysadm unadvnow** to unadvertise a local resource immediately. You see the following:

```
                UNADVERTISE LOCAL RESOURCES IMMEDIATELY

When you no longer want to share a resource with client machines you
unadvertise the resource.  This subcommand lets you unadvertise
currently advertised resources.

When a resource is unadvertised, no new clients may access it.
Unadvertising does not take a resource away from client machines that
have already mounted it.  If any clients have mounted the resource you
are unadvertising, the subcommand lets you force the resource to
be unmounted.

NOTE:  This subcommand does NOT remove resources from the Automatic
Advertise List.  Use the subcommand "unadvauto" to remove resources
from this list.

Enter the resource identifier of the resource to be
unadvertised [?, q]:
```

Step 2: Type the name of the resource you want to unadvertise.

List Remotely Mounted Resources

Step 1: Type **sysadm lsinuse** to list your resources that are currently mounted and in use on remote machines. The following is an example of what displays:

```
        LOCAL RESOURCES CURRENTLY MOUNTED BY CLIENT MACHINES
The fields in this list are in the following order:
Resource identifier, Local pathname, Client name.
-------------------------------------------------------------

CROOT    /       peanuts.snoopy peanuts.linus
CDEV    /dev     peanuts.lucy peanuts.patti
```

List Locally Advertised Resources

Step 1: Type **sysadm lsadv** to list the local resources that are currently or automatically advertised. The following is an example of what displays:

```
                ADVERTISED LOCAL RESOURCES

The fields in this list are in the following order:
Resource Identifier, Local Pathname, Client Access Permissions,
Resource Status, Resource Description, Client List.

The status of a resource is "C" (Current) if RFS is running and
the resource is now available to client machines.  The status is
"A" (Automatic) if the resource is in the Automatic Advertise List.
A resource may be both "A" and "C".  Please wait for the list ...
-------------------------------------------------------------------

CROOT /  read/write A/C "Charlie root file system" peanuts.linus peanuts.snoopy
CDEV  /dev   read/write C "Charlie device directory"   unrestricted
```

Procedure 10.4: Remote Resource Mounting (mountmgmt)

Purpose	Mount remote resources on your machine.
Starting Conditions	System state —**2** (multi-user) or **3** (RFS state) Login —**root**
Commands	**sysadm mountmgmt** **sysadm mntauto** **sysadm unmntauto** **sysadm mntnow** **sysadm unmntnow** **sysadm lsavail** **sysadm lsmount**
Reference	Chapter 10, *Remote File Sharing*

You can attach another machine's advertised resource to your file system using the **sysadm mountmgmt** subcommands.

Prerequisites

Before you run **sysadm mountmgmt** subcommands, you should do the following:

- Set up RFS. Use the **sysadm setuprfs** to enter the basic information needed to run RFS.
- *Start RFS. To use* ***mntnow, unmntnow,*** *or* ***lsavail*** *subcommands, RFS should be running. The other subcommands do not require that RFS be running to provide useful information.*

The following chart describes the **sysadm mountmgmt** subcommands and lists dependent commands and files of the procedures in this section. This information should help you relate the **sysadm** processing to the descriptions of RFS components in Chapter 10.

Table 10-4. sysadm mountmgmt Subcommands

Sysadm Commands	Dependent Commands	Files	Description
mntauto	**mount –d**(1M) **nsquery**	**/etc/fstab**	Set up a remote resource to be mounted each time RFS is started. The information you enter for each resource will be added to the Automatic Mount List.
mntnow	**mount**(1M) **nsquery**(1M)		Mount a resource immediately.
unmntauto		**/etc/fstab**	Remove a remote resource from the Automatic Mount List. Once removed, the resource will no longer be mounted when RFS is started.
unmntnow	**umount**(1M) **fuser –k**(1M) **mount**(1M)		Immediately unmount a resource and kill all local processes associated with it.
lsavail	**nsquery**(1M)		Print a list of all remote resources available for you to mount.
lsmount	**mount –d**(1M)	**/etc/fstab**	Print a list of all your currently-mounted and automatically-mounted resources.

Mount Automatically

Step 1: Type **sysadm mntauto** to add an entry to the Automatic Mount List. You then see the following:

```
ADD REMOTE RESOURCES TO AUTOMATIC MOUNT LIST

To use remote resources that have been made available to you from
remote machines, you mount the resources on local directories.
Users on your machine can then access the resources as if
they were local.

This subcommand lets you add remote resources to the Automatic Mount
List. Your system will attempt to mount these resources each time
RFS is started.  If RFS is running on your machine, you may also be
able to immediately mount the resources you have added to the Automatic
Mount List.

Enter the full pathname of the local directory to be
used as the mount point [?, q]:
```

Step 2: Enter each item of information as it is asked. This information will include the following:

Local mount point
The local directory to use as a mount point.

Resource identifier
The name of the remote resource.

Local permissions
The permissions (read-only or read/write).

The **sysadm mntauto** command sets up the automatic mount by adding information to the **/etc/fstab** file that is needed with the **mount**(1M) command. See Chapter 10 for information on the **mount** command.

Once the resource is added, you are asked if you want to mount the resource immediately if the following are true: RFS is on, the mount point exists, the mount point is not busy, and the resource is not already in use.

NOTE

When using the **sysadm** interface, you can only mount resources available from your own domain. To mount resources available from other domains, see the *Remote Resource Mounting* section of Chapter 10 for information.

Remove Automatic Mounts

Step 1: Type **sysadm unmntauto** to remove an entry from the Automatic Mount List. You see the following:

```
REMOVE REMOTE RESOURCES FROM THE AUTOMATIC MOUNT LIST

When you no longer want to share remote resources you can unmount them
from your file system.  If the resources have been added to the
Automatic Mount List and you do not want the system to try to mount
them the next time RFS is started, you use this subcommand to remove
the resources from the Automatic Mount List.

NOTE: This subcommand does not immediately unmount resources.
Use the subcommand "unmntnow" to immediately unmount resources.

Enter the resource identifier of the remote resource to be removed
from the Automatic Mount List [?, q]:
```

Step 2: Type the resource identifier you want to remove.

The **sysadm unmntauto** command removes the automatic mount by deleting the mount information for the resource from the **/etc/fstab** file.

NOTE

Using the **sysadm** interface, you can only remove entries for resources within your own domain.

Mount Immediately

Step 1: Type **sysadm mntnow** to mount a remote resource immediately. You see the following:

```
MOUNT REMOTE RESOURCES IMMEDIATELY

To use remote resources that have been made available to you
from other machines you mount the resources on local directories.
Users on your machine can then access the resources as if they
were local.

This subcommand lets you immediately mount remote resources when
RFS is running on your machine.

NOTE:  This subcommand will not add resources to the Automatic
Mount List.
Use the subcommand "mntauto" to add resources to this list.

Now checking for available remote resources.  Please wait ...

Enter the full pathname of the local directory to be
used as the mount point [?, q]:
```

Step 2: Enter each item of information as it is asked. This information will include the following:

Local mount point
The local directory to use as a mount point.

Resource identifier
The name of the remote resource.

Local permissions
The permissions (read-only or read/write).

Unmount Immediately

Step 1: Type **sysadm unmntnow** to unmount a remote resource immediately. You see the following:

```
                UNMOUNT REMOTE RESOURCES IMMEDIATELY

When you no longer want to use remote resources you can unmount them
from your file system.  If RFS is running and the resources are
currently mounted you use this subcommand to immediately unmount
the resources.

NOTE: This subcommand does not remove the resources from the Automatic
Mount List. Use the subcommand "unmntauto" to remove resources from
this list.

Enter the resource identifier of the remote resource
to be unmounted [?, q]:
```

Step 2: Type the name of the resource you want to unmount. The system kills all local processes currently accessing the resource and then unmount the resource. (If your current directory is within the resource, you are logged off.)

List Available Remote Resources

Step 1: Type **sysadm lsavail** to list Remote Resources that are available for you to mount from other members of your domain. The following is an example of the output that displays:

```
                REMOTE RESOURCES CURRENTLY AVAILABLE
Remote resources are available if they have been advertised by other
machines.  Resources can only be available when RFS is running on your
machine.

The fields on this list are in the following order:
Resource identifier, Access permissions, Advertising machine,
Description.
Please wait for the list...
-----------------------------------------------------------------

LROOT      read/write  peanuts.linus      Linus root file system
```

List Locally Mounted Resources

Step 1: Type **sysadm lsmount** to list the remote resources that are currently or automatically mounted on your system. The following is an example of the output that displays:

```
                REMOTE RESOURCES MOUNTED

This list shows remote resources that are mounted or set up to be
mounted on your machine.  The order of the fields is:
Resource identifier, Local mount-point, Local access permissions,
Resource status.

The status of a resource is "C" (Current) if RFS is running and the
remote resource is mounted on a local directory.  The status is "A"
(Auto) if the remote resource is on the Automatic Mount List.
Resources may be both "C" and "A".   Please wait for the list...
-----------------------------------------------------------------

LROOT /usr/Lroot read-write A/C
```

Procedure 10.5: Change RFS Configuration (confgmgmt)

Purpose	Change ID mapping, show current RFS configuration, or update domain member list.
Starting Conditions	System state—**2** (multi-user) or **3** (RFS state) Login—**root**
Commands	**sysadm confgmgmt** **sysadm showconfg** **sysadm idmapping** **sysadm lsmember** (primary only) **sysadm addmember** (primary only) **sysadm delmember** (primary only)
Reference	Chapter 10, *Remote File Sharing*

The **sysadm confgmgmt** subcommands let you perform several separate RFS configuration tasks. The tasks include listing basic RFS configuration information and setting up basic ID mapping strategies. If your machine is a primary name server, commands are also available to maintain the domain member list.

Each of these commands is also available through the **sysadm setuprfs** command. The advantage of using them separately is that you don't have to go through the whole **setuprfs** procedure if you are only interested in one function.

Prerequisites

Before you run **sysadm confgmgmt** subcommands, you should do the following:

- Set up RFS. Use the **sysadm setuprfs** to enter the basic information needed.

The following chart describes the **sysadm confgmgmt** subcommands and lists dependent commands and files. This information should help you relate the **sysadm** processing to the descriptions of RFS components in Chapter 10.

Table 10-5. sysadm confgmgmt Subcommands

Sysadm Commands	Dependent Commands	Files	Description
showconfg	**uname**(1) **dname**(1M) **dname –n**(1M)	**rfmaster** *	List the following RFS information for your domain: Your machine's node name. The name of your machine's domain. The Transport Provider used by RFS. The primary and secondary name servers' names and network addresses.
idmapping	**idload**(1M)	**uid.rules** ** **gid.rules** **	Choose one of three basic strategies for defining the permissions remote users will have to your resources.
addmember	**rfadmin –a** (1M)	*dom*/**passwd** **	Add a machine and its password to the domain member list. This is only used on the primary name server machine. (*dom* is replaced by the domain name.)
delmember	**rfadmin –r** (1M)	*dom*/**passwd** **	Delete a machine and its password from the domain member list. This is only used on the primary name server machine. (*dom* is replaced by the domain name.)
lsmember		*dom*/**passwd** **	List all machines in the domain member list, noting those that are primaries and secondaries. This is only used on the primary name server machine. (*dom* is replaced by the domain name.)

* This file appears in the directory **/usr/nserve**.
** This file appears in the directory **/usr/nserve/auth.info**.

Show RFS Configuration

Step 1: Type **sysadm showconfg** to show RFS configuration information for your machine. The following is an example of the output that displays:

```
CURRENT RFS CONFIGURATION

Node name:                      charlie
Domain name:                    peanuts
Transport provider:             network
Primary name server:            charlie
Address of primary:             charlie.serve
Secondary name server:          linus
Address of linus:               linus.serve
```

Choose ID Mapping Scheme

NOTE

The **sysadm idmapping** command provides you with the choice of three of the most common ways of defining remote users' permissions to your resources. For more complex methods of ID mapping, such as mapping by name or mapping by individual machines, Chapter 10 describes how to manually map IDs.

CAUTION

Do not use the ID mapping available through the sysadm interface if you want to keep any ID mapping you enter manually from descriptions in Chapter 10.

Step 1: Type **sysadm idmapping** to change or list the user and group ID mapping for your machine. You see the following:

```
USER ID AND GROUP ID MAPPING

Every machine in a domain defines how remote users will be allowed
to access its local resources.  You define this access by setting
up a mapping table of how remote user IDs and remote group IDs will
be mapped to local user and group IDs.

Since setting up mapping can be a complex procedure, this subcommand
allows you to choose from three pre-defined mappings.
Selecting option a, b, or c will replace any current mapping that
you may have previously set up.  To set up mappings not defined
by this subcommand see your System Administrator's Guide.

Do you need a more detailed explanation of ID mapping? [y, n, q]
```

Step 2: Type **y** if you want more details, **n** if you do not. After the detailed explanation (if requested) you see the following:

```
Choose option desired for uid/gid mappings:
  a) all remote IDs map to guest ID with "other" permissions.
  b) remote IDs map directly to same local IDs except uids 0-99,
     gids 0-10.
  c) remote IDs map directly to same local IDs except uid 0,
     gid 0 (root).
  d) show current ID mapping.
[a, b, c, d, ?, q]:
```

Step 3: Since user ID and group ID mapping can be complex, this interface presents you with several standard options. All these options represent global mapping schemes. This means that the same mapping rules are applied to users of all machines that use your resources. For information on other mapping schemes, see the *Mapping Remote Users* section of Chapter 10.

Type a **?** if you need more information on one of the three basic mapping types. Then type one of the three mapping choices (**a**, **b**, or **c**) or type **d** to show the current ID mapping.

The following paragraphs describe the three mapping choices.

a
This is usually the best choice since all remote users on all machines that share your resources will be assigned to a special guest login ID. It provides the maximum security and the minimum complexity.

b or **c**
These options are only valuable if you have identical **/etc/password** and **/etc/group** files among machines with which you share resources. The difference between **b** and **c** is that **b** protects all administrative logins and **c** only protects the **root** login.

Add Domain Members

Each time a machine is added to an RFS network, the machine's node name and password must be entered on the primary name server machine.

Step 1: Type **sysadm addmember** to add a machine to the domain member list. You see the following:

```
ADD MEMBER MACHINES TO YOUR DOMAIN

Every machine must be a member of a domain. The primary name server
machine for a domain maintains the list of machines that are members
of its domain.

This subcommand allows the primary name server to add machines to the
list of domain members.  The addition takes effect immediately.  When
a machine is added as a member of a domain it may share resources
available in that domain.

Enter the node name of a machine to be added as a member of your
domain [?, q]:
```

Step 2: Type the node name of the machine you want to add to the domain.

Step 3: Type a password for the machine you want to add to the domain. This password should match the password entered by the machine when it first starts RFS. This is the password the machine will use when it wants to enter your domain. (The password may simply be a carriage return.)

Delete Domain Members

Step 1: From the primary name server machine, type **sysadm delmember** to delete a machine from the domain member list. You cannot delete primary or secondary name servers. You see the following:

```
DELETE MEMBER MACHINES FROM YOUR DOMAIN

The primary name server machine for a domain maintains the list of
machines that are members of its domain.

This subcommand allows the primary name server to delete
machines from the list of domain members.  The deletion takes
effect immediately.  When a machine is deleted from a domain it may
no longer access resources available in that domain.

Enter the node name of a machine to be added as a member of your
domain [?, q]:
```

Step 2: Type the name of the machine you want to delete from the domain member list.

List Domain Members

Step 1: From the primary name server machine, type **sysadm lsmember** to list machines in the domain member list. The following is an example of the output that displays:

```
MEMBERS OF DOMAIN: peanuts

Node name: charlie          Primary name server
Node name: linus            Secondary name server
Node name: lucy
Node name: patti
Node name: snoopy
```

MOTOROLA INC.

Microcomputer Division
2900 South Diablo Way
Tempe, Arizona 85282
P.O. Box 2953
Phoenix, Arizona 85062