

IRIX™ Admin: Peripheral Devices

Document Number 007-2861-001

CONTRIBUTORS

Written by Susan Thomas and Jeffrey B. Zurschmeide.

Illustrated by Dany Galgani

Production by Gloria Ackley

Engineering contributions by Ray Niblett, Ewan McKissock, Kam Kashani, Andrew Cherenson, Chris Wagner, Dave Higgen, Jeff Doughty, Paul Mielke, Robert Stephens, Joe Yetter, Jack Weldon, Gretchen Helms, and Vernon Schryver.

© Copyright 1996, Silicon Graphics, Inc.— All Rights Reserved

The contents of this document may not be copied or duplicated in any form, in whole or in part, without the prior written permission of Silicon Graphics, Inc.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure of the technical data contained in this document by the Government is subject to restrictions as set forth in subdivision (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 52.227-7013 and/or in similar or successor clauses in the FAR, or in the DOD or NASA FAR Supplement. Unpublished rights reserved under the Copyright Laws of the United States. Contractor/manufacturer is Silicon Graphics, Inc., 2011 N. Shoreline Blvd., Mountain View, CA 94039-7311.

Silicon Graphics, IRIS, and CHALLENGE are registered trademarks and IRIX, Workspace, Impresario, and Onyx are trademarks of Silicon Graphics, Inc. Apollo is a registered trademark of Apollo Computer, Inc. FrameMaker is a registered trademark of Frame technology, Inc. Hewlett-Packard is a registered trademark of Hewlett-Packard Company. IBM is a registered trademark and PC/AT is a trademark of International Business Machines Corporation. Macintosh and ImageWriter are registered trademarks of Apple Computer, Inc. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company, Ltd. NFS is a registered trademark of Sun Microsystems, Inc. MS-DOS is a registered trademark of Microsoft Corporation. Centronics is a registered trademark of Centronics Data Computer Corporation.

IRIX™ Admin: Peripheral Devices
Document Number 007-2861-001

Contents

	List of Figures	vii
	List of Tables	ix
	IRIX Admin Manual Set	xi
	About This Guide	xiii
	What This Guide Contains	xiv
	Conventions Used in This Guide	xv
	Additional Resources	xvi
1.	Terminals and Modems	1
	About the TTY Subsystem	2
	Terms	3
	Installing an ASCII Terminal	4
	Connecting the Terminal Hardware	4
	Configuring the Terminal Software	4
	Setting Terminal Options	8
	Installing a Modem	9
	Before You Begin	10
	Connecting the Modem Hardware	11
	Setting Up the Modem Software	12
	Modem Software Setup, All Modes	12
	Dial-out Mode	14
	Dial-in Mode	15
	Dialing Out to Another Modem	17
	Managing the TTY Subsystem	18
	Checking Line Settings Using IRIX Shell Commands	18
	Creating and Testing Line Settings	19
	Modifying Line Characteristics	19

- Serial Port Pin Signal Tables 20
 - Defining the Serial Interface 21
 - Cabling the Serial Ports 22
 - DB-9 Serial Connector Cabling 22
 - Mini-DIN8 Serial Connector Cabling 25
- 2. **Printers** 29
 - Understanding the lp Spooler 30
 - Terms 31
 - Command Interface to lp 32
 - User Command Summary 32
 - lp: Send a Print Job to a Printer 33
 - cancel: Remove a Print Request 34
 - disable: Stop Printer From Processing Requests 35
 - enable: Allow Printer to Process Requests 36
 - lpstat: Report lp Status 36
 - Administrator Command Summary 37
 - lpsched: Start the lp Scheduler 37
 - lpshut: Stop the lp Scheduler 38
 - reject: Prevent Print Requests 38
 - accept: Allow Print Requests 38
 - lpmove: Move a Request to Another Printer 39
 - lpadmin: Configure Printers 40
 - Managing the lp System 41
 - Adding a Printer 41
 - Adding Parallel and Serial Printers 42
 - Adding a SCSI Printer 43
 - Adding a Network Print Server 44
 - Removing a Printer 46
 - Changing the Default Printer Destination 46
 - Managing Printing on Print Server Systems 47
 - Checking Print Server Status 47
 - Canceling Print Server Requests 47

Configuring the BSD lpr Spooler System	48
Verifying Installation of the BSD lpr Subsystem	49
Configuring the Printcap File	50
Printcap Examples	51
Using the lpr Command to Print	52
Printer Cable Pin Signal Tables	53
Parallel Port Pin Signal Table	53
Serial Port Pin Signal Tables	55
DB-9 Connector Cabling	55
Mini-DIN8 Connector Cabling	58
3. CD-ROM, Floptical, and Floppy Disk Drives	61
Floppy Disk and CD-ROM Filesystems	61
CD-ROM Filesystems	62
Floppy Disk Filesystems	62
Configuring Floppy Disk Drive Device Files	63
Using Floppy Disk Drives	65
Using DOS and Macintosh Floppy Disks	65
Using a Floppy Disk Drive for IRIX File Transfer	65
Floppy Disk File Transfer With tar	66
Floppy Disk File Transfer With cpio	66
Floppy Disk File Transfer With dd	66
4. Tape Drives	69
Adding a Tape Drive	70
MAKEDEV Commands for Tape Drives	71
Tape Capacities	71
Making Tape Drive Links	73
1/2-Inch Tape Drives	74
Switch Settings for the Kennedy 1/2-Inch SCSI Tape Drive	74
8mm and 4mm Tape Drives	75
Exabyte 8mm Cartridge Tape Media Specifications	75
QIC Tape Drives	76

- Using dump With DAT Tapes 78
- Tape Drive Cleaning Process 78
 - 1/2-Inch Reel-to-Reel Cipher Tape Drive Cleaning Process 79
 - 1/2-Inch Reel-to-Reel Kennedy Tape Drive Cleaning Process 81
 - 8mm and 4mm Tape Drive Cleaning Process 82
- 5. Troubleshooting 83**
 - Troubleshooting Your Modem Setup 83
 - Troubleshooting Your Printing System 84
 - Hardware Troubleshooting Checklist 84
 - Software Troubleshooting Checklist 85
 - Troubleshooting Network Printers 86
 - Emergency Measures 86
 - Troubleshooting the BSD lpr Spooling System 87
 - Troubleshooting Inaccessible Tape Drives 91
 - Checking the Hardware 91
 - Checking the Software 92
 - Troubleshooting Tape Read Errors 94
- A. Error Messages 95**
 - lp Error Messages 95
 - Tape Drive Error Indications 107
- B. Peripheral Device Files 109**
- Index 111**

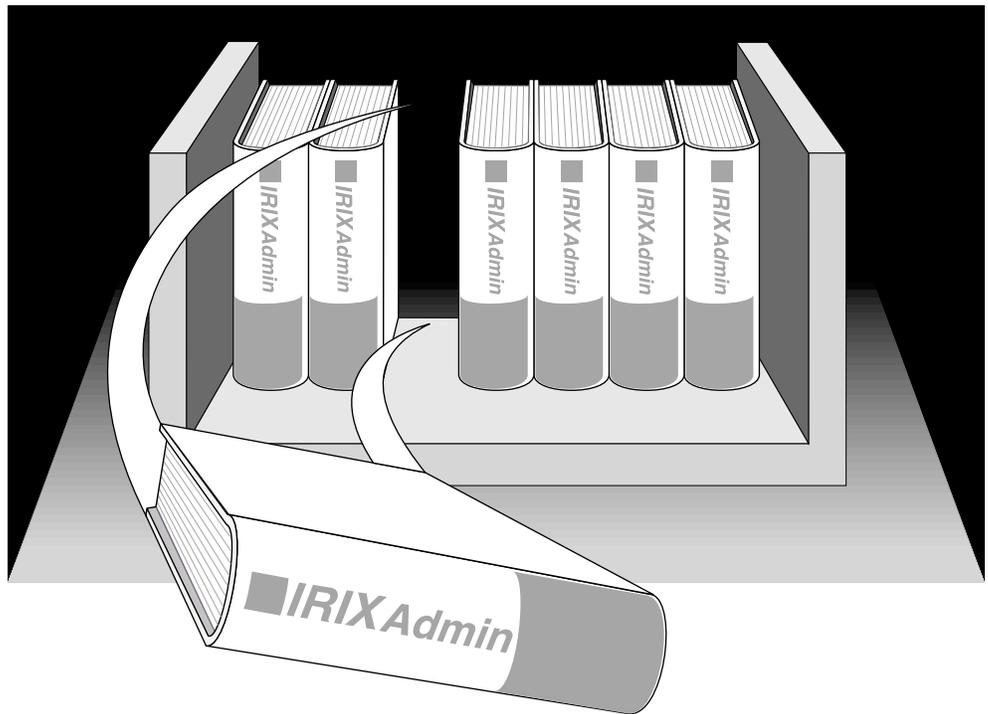
List of Figures

Figure 1-1	DB-9 Connector	22
Figure 1-2	DB-9 to DB-25 RTS/CTS Modem Control Cable	24
Figure 1-3	Mini-DIN8 to DB-25 Serial Terminal Cable	25
Figure 2-1	Print Spooling	30
Figure 2-2	Printer Classes	31
Figure 2-3	Print Servers, Print Clients, and Print Server Systems	44
Figure 2-4	Parallel Port Pin Assignment	54
Figure 2-5	DB-9 Pin Assignment	56
Figure 2-6	DB-9 to DB-25 RTS/CTS Flow Control Cable	57
Figure 2-7	Mini-DIN8 to DB-25 Serial Cable	58
Figure 4-1	Kennedy Dipswitch Bank 1	74
Figure 4-2	Kennedy Dipswitch Bank 2	74

List of Tables

Table 1-1	Modem Configuration Scripts	10
Table 1-2	Overview of Modem Software Setup	12
Table 1-3	Determining the Usage Mode for Modem Configuration	13
Table 1-4	inittab Dial-in Options by Modem Type	16
Table 1-5	Pin Definitions for a Null Modem Cable	22
Table 1-6	Pin Definitions for a Three-wire Null Modem Terminal Cable	23
Table 1-7	Pin Definitions for a DB-9 to DB-25 RTS/CTS Modem Control Cable	24
Table 1-8	Pin Definitions for a Mini-DIN8 to DB-25 Serial Terminal Cable	26
Table 1-9	Pin Definitions for a Mini-DIN8 RTS/CTS Modem Cable	27
Table 2-1	Pin Definitions for a Parallel Port Cable	54
Table 2-2	Pin Definitions for a DB-9 Serial Cable	56
Table 2-3	Pin Definitions for a DB-9 to DB-25 RTS/CTS Flow Control Cable	57
Table 2-4	Pin Definitions for a Mini-DIN8 to DB-25 Serial Cable	59
Table 4-1	Cartridge Tape and DAT Capacities	71
Table 4-2	9-Track Tape Capacities	72
Table 4-3	Exabyte 8mm Cartridge Tape Specifications	75
Table 4-4	Low-Density QIC Tape Drive Compatibility	77
Table 4-5	High-Density QIC Tape Drive Compatibility	77

IRIX Admin Manual Set



This guide is part of the *IRIX Admin* manual set, which is intended for administrators: those who are responsible for servers, multiple systems, and file structures outside the user's home directory and immediate working directories. If you find yourself in the position of maintaining systems for others or if you require more information about IRIX than is in the end-user manuals, these guides are for you. The *IRIX Admin* guides are available through the IRIS InSight online viewing system.

The set comprises these volumes:

- IRIX Admin: Software Installation and Licensing—Explains how to install and license software that runs under IRIX, the Silicon Graphics implementation of the UNIX operating system. Contains instructions for performing miniroot and live installations using Inst, the command line interface to the IRIX installation utility. Identifies the licensing products that control access to restricted applications running under IRIX and refers readers to licensing product documentation.
- IRIX Admin: System Configuration and Operation—Lists good general system administration practices and describes system administration tasks, including configuring the operating system; managing user accounts, user processes, and disk resources; interacting with the system while in the PROM monitor; and tuning system performance.
- IRIX Admin: Disks and Filesystems—Explains disk, filesystem, and logical volume concepts. Provides system administration procedures for SCSI disks, XFS and EFS filesystems, XLV and lv logical volumes, and guaranteed-rate I/O.
- IRIX Admin: Networking and Mail—Describes how to plan, set up, use, and maintain the networking and mail systems, including discussions of sendmail, UUCP, SLIP, and PPP.
- IRIX Admin: Backup, Security, and Accounting—Describes how to back up and restore files, how to protect your system's and network's security, and how to track system usage on a per-user basis.
- IRIX Admin: Peripheral Devices—Describes how to set up and maintain the software for peripheral devices such as terminals, modems, printers, and CD-ROM and tape drives. Also includes specifications for the associated cables for these devices.
- IRIX Admin: Selected Reference Pages (not available in InSight)—Provides concise reference page (manual page) information on the use of commands that may be needed while the system is down. Generally, each reference page covers one command, although some reference pages cover several closely related commands. Reference pages are available online through the man(1) command.

About This Guide

This guide explains how to use the system-level IRIX[®] utilities available with IRIS[®] workstations and servers to set up and maintain peripheral devices, such as terminals, modems, printers, and CD-ROM and tape drives.

If you have a graphics workstation, you may find it convenient to use the System Manager, which is described in the *Personal System Administration Guide*. That guide should be your first resource for administering graphics workstations. Regardless of whether you use the System Manager or the IRIX command-line interface, the results are the same. The System Manager does not create any new files on your system, unlike applications such as WorkSpace[™].

If you have a server, this book and the other books in the IRIX Admin suite are your primary guides to system administration, since without graphics, you cannot use the System Manager. This guide does not describe the System Manager in great detail. Instead, it covers the traditional shell command approach to administering peripherals with the IRIX operating system.

What This Guide Contains

IRIX Admin: Peripheral Devices contains the following chapters:

- Chapter 1 “Terminals and Modems” describes how to set up and maintain serial terminals and modems.
- Chapter 2 “Printers” provides instruction on the installation and maintenance of local and networked printers.
- Chapter 3 “CD-ROM, Floptical, and Floppy Disk Drives” describes the setup and maintenance of CD-ROM and floppy drives.
- Chapter 4 “Tape Drives” lists the steps to add, maintain, and use tape drives.
- Chapter 5 “Troubleshooting” provides general problem-solving information and check-out procedures. Also, it describes specific problems that can occur with peripheral devices and suggests what you can do to correct them.
- Appendix A “Error Messages” lists error messages, their meanings, and what you should do about them.
- Appendix B “Peripheral Device Files” contains a list of peripheral device files and directories that reside in the */dev* directory.

Conventions Used in This Guide

These type conventions and symbols are used in this guide:

Helvetica Bold Hardware labels

Italics Executable names, filenames, IRIX commands, manual/book titles, new terms, program variables, tools, utilities, variable command-line arguments, and variables to be supplied by the user in examples, code, and syntax statements

Fixed-width type

Error messages, prompts, and onscreen text

Bold fixed-width type

User input, including keyboard keys (printing and nonprinting); literals supplied by the user in examples, code, and syntax statements (*see also* <>)

"" (Double quotation marks) References in text to document section titles

() (Parentheses) Following IRIX commands—surround reference page (man page) section number

[] (Brackets) Surrounding optional syntax statement arguments

<> (Angle brackets) Surrounding nonprinting keyboard keys, for example, <Esc>, <Ctrl-D>

IRIX shell prompt for the superuser (*root*)

% IRIX shell prompt for users other than superuser

Additional Resources

Following are additional resources that may be beneficial in providing information about the UNIX[®] operating system and its administration.

Bach, M., *The Design of the UNIX Operating System* (Englewood Cliffs, NJ: Prentice Hall, 1986).

Fiedler, D., and B. Hunter, *UNIX System V Release 4 Administration* (Carmel, IN: Hayden Books, 1991).

Frisch, A., *Essential System Administration*. (Sebastopol, CA: O'Reilly & Associates, Inc., 1991).

Gilly, D., *UNIX in a Nutshell*. (Sebastopol, CA: O'Reilly & Associates, Inc., 1992).

Hunt, C., *TCP/IP Network Administration*. (Sebastopol, CA: O'Reilly & Associates, Inc., 1992).

Leffler, S., *The Design and Implementation of the 4.3 BSD UNIX Operating System*. (Menlo Park, CA: Addison Wesley, 1989).

Nemeth, E., G. Snyder, and S. Sebass, *UNIX System Administration Handbook* (Englewood Cliffs, NJ: Prentice Hall, 1989).

Thomas, R., *UNIX System Administration Guide for System V*. (Englewood Cliffs, NJ: Prentice Hall, 1989).

Todino, G., and T. O'Reilly, *Managing UUCP and Usenet* (Sebastopol, CA: O'Reilly & Associates, Inc., 1992)

Terminals and Modems

This chapter describes some of the software and hardware considerations for setting up terminals and modems. This information also applies to most generic serial devices. It does not cover specialized devices (such as the Dial & Button Box and the Spaceball). For more information on these devices, please contact Silicon Graphics. "Dumb" serial printers are also connected using the TTY subsystem, but they are covered in Chapter 2, "Printers."

Serial communication between users and serial devices, such as modems and terminals, is supported by the TTY subsystem. This chapter tells you how to administer the TTY subsystem with respect to terminals and modems. It includes information about:

- How the TTY subsystem works. See "About the TTY Subsystem" on page 2.
- Connecting an ASCII terminal. See "Installing an ASCII Terminal" on page 4.
- Connecting a modem. See "Installing a Modem" on page 9.
- Information on managing the TTY subsystem. See "Managing the TTY Subsystem" on page 18.
- Serial port cabling and pin signals for the standard serial ports. See "Serial Port Pin Signal Tables" on page 20.

To connect peripherals that are not covered in this chapter, refer to the documentation that accompanies the peripheral.

About the TTY Subsystem

A series of four processes (*init*(1M), *getty*(1M) or *uugetty*(1M), *login*(1), and one of *sh*(1), *csh*(1), or *ksh*(1)) connect a user to the IRIX system. *init* is invoked as the last step in the boot procedure and is responsible for spawning general processes. It spawns a process for each serial line that a user may log in on, as defined by instructions in the file */etc/inittab*. The *getty* or *uugetty* commands in */etc/inittab* require a *<line>* argument, which is the name of a special file in the */dev* directory. For a description of other arguments that may be used with *getty* and *uugetty*, see the *getty*(1M) and *uugetty*(1M) reference pages.

A user attempting to make a connection generates a signal that is routed by the hardware to the *getty* process (derived from “get a tty”) for one of the TTY line files in */dev*. *getty* responds by sending an entry from the file */etc/gettydefs* down the line. The *gettydefs* entry used depends on the *<speed>* argument used with the *getty* command. (In the SYNOPSIS of the *getty*(1M) reference page, the argument name is *<speed>*, but it is really a pointer to the *<label>* field of a *gettydefs* entry.) If no *<speed>* argument is provided, *getty* uses the first entry in *gettydefs*. Among the fields in the *gettydefs* entry is the login prompt (described in “Checking Line Settings Using IRIX Shell Commands” on page 18).

On receiving the login prompt, the user enters a login name. *getty* starts *login*, using the login name as an argument. *login* issues the prompt for a password, evaluates the user’s response, and if the password is acceptable, calls in the user’s shell as listed in the */etc/passwd* entry for the login name. If no shell is named, */bin/sh* is furnished by default.

/bin/sh executes */etc/profile* and then executes the user’s *.profile*, if it exists. The *.profile* or *.login* file often contains *stty* commands that reset terminal options that differ from the defaults. */bin/csh* executes */etc/cshrc*, *.cshrc*, and *.login*. The connection between the user and the IRIX system has now been made.

Terms

The following terms are used when describing serial devices and the TTY system:

TTY	The near-classic abbreviation for teletypewriter. The term covers the whole area of access between the IRIX system and peripheral devices, including the system console. It shows up in commands such as <i>getty</i> and <i>stty</i> , in the names of device special files such as <i>/dev/ttyd1</i> , and in the names of files such as <i>/etc/gettydefs</i> , which is used by <i>getty</i> .
TTY line	The cable connecting the computer to the serial device.
port	The connector on the computer where the TTY line attaches.
line settings	A set of characteristics for data transmission over the line.
baud rate	The speed at which data is transmitted over the line. A part of the line settings.
mode	The characteristics of the terminal interface, and a part of line settings. The TTY line and the terminal must be working in the same mode before communication can take place. Described in <i>termio(7)</i> .
hunt sequence	A series of line settings such as different baud rates. During the login sequence, a user looking for a compatible connection to the computer can go from one setting to the next by sending a BREAK signal. Also referred to as a rotary.
terminal options	Selectable settings that define the way a given terminal operates. Described in <i>termio(7)</i> .

Installing an ASCII Terminal

This section describes the procedures for connecting and configuring an ASCII terminal. The *diagnostics terminal*, or *console*, is the ASCII terminal connected to a serial port on the I/O panel of a server. It can be connected to any of these ports, depending on your system:

- To the alternate console port on a graphics system.
- To the console port (labeled port 1) or port 2 on a non-graphics system.
- To any of the four serial ports on larger systems such as the CHALLENGE[®] M or Onyx[™].

The messages produced by the power-on diagnostics appear on the screen of this terminal.

Connecting the Terminal Hardware

Connect an appropriate cable from the ASCII terminal to the serial port on the computer I/O panel. See “Cabling the Serial Ports” on page 22 for information about port cabling and pin signals.

Configuring the Terminal Software

This section tells you how to configure IRIX software to use an ASCII terminal with your computer system.

During configuration, you need to:

- Edit the file */etc/ttytype* to associate the TTY port you have chosen with a file that describes your terminal settings. These terminal descriptor files reside in the directory */usr/lib/terminfo*.
- Update the file */etc/inittab* to allow a login to that port.

Note: The utilities described in this section are distributed as part of the *ee2.sw.terminfo* package. You must have installed this package on your system in order to use these utilities. See IRIX Admin: Software Installation and Licensing for information and specific instructions on installing this package.

The `/usr/lib/terminfo` directory contains files that describe different terminal models, their capabilities, and how they operate. For most ASCII terminal models, you do not need to edit this database.

Note: If your terminal is not in the database, or if it does not work properly after you have configured the software, you may need to write a terminal description. Refer to the optional *Topics in IRIX Programming* and the `tset(1)`, `stty(1)`, and `terminfo(4)` reference pages. Additionally, your terminal may support a compatible terminal mode (such as VT100). Check your terminal documentation for compatibility modes.

The directory `/usr/lib/terminfo` is divided into numerical and alphabetical subdirectories. Each subdirectory contains entries for terminals whose names begin with that character. For example, `/usr/lib/terminfo/v` contains the entry for the Visual 50. The entry name listed in the subdirectory is `v50am`. Use the `fgrep` command to search for the terminal model name that corresponds to the entry name.

To find the entry name for your terminal and to configure software for an ASCII terminal, follow these steps:

1. Log in as root or become the superuser by entering the `su` command.

2. Change directories to `/usr/lib/terminfo`, type:

```
cd /usr/lib/terminfo
```

3. Find the entry name for your terminal. Issue the `fgrep` command with a string that you suspect could make up part of your terminal name.

```
ls -R | fgrep -i string
```

If this fails, examine the subdirectories of `/usr/lib/terminfo`, which contain all the terminal entries.

4. Once you find the terminal name as it appears in `/usr/lib/terminfo`, issue the `infocmp` command to determine the model name of your terminal. For example, for a Visual 50, issue this command:

```
infocmp -I v50am
```

You see a display that begins with this line:

```
v50am|visual50 (v50 emulation) with automatic margins,
```

The data in the first field (`v50am`) is the model name of your terminal.

Tip: If it looks like there are many terminal names that could apply, use this example shell script to help speed up the search. This example uses the string `wy60*` to help locate all the Wyse60 models. Substitute your own string, as appropriate.

```
sh
for i in wy60*
do
    infocmp -I $i
done
```

Note: If you choose an incorrect model name, you can change it later; the line still works, but screen-based commands do not display correctly.

5. Edit `/etc/ttytype` and enter the model name.

This file associates the model name with the port where the terminal is connected. In the line that contains the port you are using, replace `du` with the model name of your terminal.

An `/etc/ttytype` might look like this example:

```
iris-ansi systty
?v50am    ttyd1
?v50am    ttyd2
?v50am    ttyd3
?v50am    ttyd4
?v50am    ttyd5
?v50am    ttyd6
?v50am    ttyd7
?v50am    ttyd8
?v50am    ttyd9
?v50am    ttyd10
?v50am    ttyd11
?v50am    ttyd12
```

Hint: The optional question mark (?) at the beginning of a line in `/etc/ttytype` causes `tset` to display the model name as supplied (for example, `v50am` here) and prompts for the model name you are currently using. If you are not using the model name as provided, type in the name that you are using; otherwise, press **<Enter>** to accept the default. This provides an easy way to switch terminal settings if you use more than one type of terminal.

`tset` is normally called in your login startup script (`.login` or `.profile`). `tset` commands use information from `/etc/ttytype` and `/usr/lib/terminfo` to initialize the terminal. These files also provide information on setting environment variables so that editors and other programs know how to communicate with the terminal. See `tset(1)` for detailed information.

6. Edit */etc/inittab* so that you can log in. */etc/inittab* provides information about how the computer ports behave. This is a sample from an */etc/inittab* file:

```
t1:23:respawn:/etc/getty -s console ttyd1 co_9600 # port 1
t2:23:off:/etc/getty -N ttyd2 co_9600 # port 2
t3:23:off:/etc/getty -N ttyd3 co_9600 # port 3
t4:23:off:/etc/getty -N ttyd4 co_9600 # port 4
```

Here are two example entries, with an explanation of each field in the entries:

```
t1:23:respawn:/etc/getty -s console ttyd1 co_9600
t2:23:off:/etc/getty -N ttyd2 co_9600
```

t1 t2 uniquely identifies the entry.

23 defines the *init* run level in which this entry is to be processed. A *23* means this entry is to be processed in run levels two and three. Refer to *init(1M)* for more information about run levels.

off means do not perform the action on the process field of *init*.

respawn means start the program in the process field of *init* and restart it every time the process terminates. See *inittab(4)* for a description of all possible actions.

/etc/getty -s console ttyd1 co_9600

runs the *getty* process on the port labeled 1 at the baud rate and with the options specified in the *co_9600* entry in the */etc/gettydefs* file. The **-s console** option instructs *getty* that the login shell generated is a system console, and therefore receives system error messages.

/etc/getty -N ttyd2 co_9600

runs the *getty* process on the port labeled 2 at the baud rate and with the options specified in the *co_9600* entry in the */etc/gettydefs* file. The **-N** option instructs *getty* to honor the presence of the */etc/nologin* file, which does not allow remote logins over the network.

To enable you to log in to the terminal connected to the port labeled 2, find this line:

```
t2:23:off:/etc/getty -N ttyd2 co_9600
```

Change it to:

```
t2:23:respawn:/etc/getty -N ttyd2 co_9600
```

7. If you don't plan to run the terminal at 9600 baud, replace `co_9600` in `/etc/inittab` with the correct entry name from `/etc/gettydefs`.

`/etc/inittab` refers to `/etc/gettydefs` for information about the terminal line settings. In the example from `/etc/inittab` above, `co_9600` refers to the name of an entry in `/etc/gettydefs`; it defines a 9600 baud console setting.

To see what entries are defined in `/etc/gettydefs`, examine the file, or see "Checking Line Settings Using IRIX Shell Commands" on page 18. To make a new entry, see "Creating and Testing Line Settings" on page 19.

8. Inform `init` of the change to `/etc/inittab`, and start a `getty` process for the port:

```
telinit q
```

9. Power on the terminal. The workstation sends a login prompt to the terminal screen. Press **<Enter>** if the login prompt doesn't appear.

Note: If the default line speed set in `/etc/inittab` is incorrect, the prompt may be garbled or may not appear. You should choose another line speed.

Setting Terminal Options

The TTY system described thus far establishes a basic style of communication between the user's terminal and the IRIX operating system. Once a user has successfully logged in, he or she may prefer terminal options other than the default set.

The `stty` command controls terminal options. Many users add an `stty` command to their `.profile` or `.login` file so the options they want are automatically set as part of the login process. Here is an example of a simple `stty` command:

```
stty cr0 nl0 echoe -tabs erase '^H'
```

The options in the example mean:

`cr0 nl0` No delay for carriage return or new line. Delays are not used on a video display terminal, but are necessary on some printing terminals to allow time for the mechanical parts of the equipment to move.

`echoe` Erases characters as you backspace.

<i>-tabs</i>	Expand tabs to spaces when printing.
<i>erase '^H'</i>	Change the character-delete character to a Ctrl-H. The default character-delete character is the pound sign (#). Most terminals transmit a Ctrl-H when the <Backspace> key is pressed. Specifying this option makes <Backspace> delete the character that was just typed.

Installing a Modem

This section describes how to install a modem onto a serial port. It describes how to set up the hardware and software to support a UUCP modem connection.

Topics in this section are:

- “Before You Begin” on page 10 provides a list of prerequisites that you should be aware of.
- “Connecting the Modem Hardware” on page 11 describes the hardware connections that you should make before configuring the software.
- “Setting Up the Modem Software” on page 12 describes the software configuration procedure.

Silicon Graphics is compatible with most industry-standard modems. This section uses the US Robotics Sportster and Telebit modems as examples for the installation procedure. If you are using another type of modem, substitute its information where applicable.

You can use the standard system software to set up a modem for three different modes:

- As a dial-in modem—other users can call your modem to log in to your system.
- As a dial-out modem—you use the modem to call a remote modem to log in to the system to where the remote modem is connected.
- As a dial-in/dial-out modem—other users can call your modem and you can call other modems.

If you want to set up your modem for a SLIP or PPP connection, complete the modem configuration steps in the chapter and then refer to the SLIP or PPP configuration procedures in IRIX Admin: Networking and Mail.

Before You Begin

There are a few prerequisites you should be aware of before installing the modem. Review the following points before installing the modem.

- Choose a modem configuration script. Silicon Graphics provides a set of predefined configuration scripts for the many popular modems to help with your modem installation. These scripts describe the correct jumper settings for your modem and contain the expected AT commands and values for your modem to operate correctly. If your modem is not listed, you can use the *fix-hayes* script as a template and customize it to your modem’s specifications.

Table 1-1 shows the modem types and their configuration scripts.

Table 1-1 Modem Configuration Scripts

Modem	Configuration Script
Telebit T2500, T1600, QBlazer, T3000, and WorldBlazer	<i>fix-telebit</i>
ZyXEL U-1496	<i>fix-zyxel</i>
Intel 14.4ex	<i>fix-intel</i>
DSI 9624 models	<i>fix-dsi</i>
US Robotics (also called USR)	<i>fix-usr</i>
Hayes ACCURA (also called Hayes14)	<i>fix-hayes</i>

- Check that the UUCP software is installed. The *eo2.sw.uucp* subsystem must be installed on your system. It is shipped with your IRIX system software distribution, but it is not installed by default. You can determine whether *eo2.sw.uucp* is installed by using the command:

```
versions | grep uucp
```

Check the *versions* output for the following line:

```
I eo2.sw.uucp uucp utilities
```

If the line above is not present, *eo2.sw.uucp* is not installed. See IRIX Admin: Software Installation and Licensing for instructions on installing a subsystem and install *eo2.sw.uucp*.

- Check the baud rate of the modem. The installation procedure examples assume you are using a 28,800 or 14,400 baud modem. Be sure to substitute your modem baud rate where applicable.
- Check your serial port interface speed. The interface speed is the maximum speed of your computer's serial port, generally 38,400 bps. Check your system *Owner's Guide* to be sure. Use the maximum speed common to both modem and computer.
- If your modem supports hardware flow control (RTS/CTS) be sure to use it and disable support for software flow control (XON/XOFF).
- Determine how you want to use your modem: dial-in only, dial-out only, or dial-in/dial-out.

Connecting the Modem Hardware

To set up a modem, you must first make the following physical connections. With power to the modem off, connect the modem to:

- A serial port on your system I/O panel. Be sure to use the Silicon Graphics modem cable (model number X5-25TO8, P/N 018-8109-00X) or one created to the Silicon Graphics serial port specifications. See your system *Owner's Guide* or "Cabling the Serial Ports" on page 22 for cabling information.

Caution: Do not connect your system to the modem with a cable for an IBM[®]-PC/AT or an Apple Macintosh[®]. It will not function correctly. You must use a Silicon Graphics cable or one made to the specifications for the serial port pinouts as described in serial(7) and "Cabling the Serial Ports" on page 22.

- A power source. Connect the power cord that came with the modem to the modem and an AC outlet.
- A telephone jack. Attach the phone cable supplied with the modem into the telephone jack and the correct phone jack on the back of the modem.

Setting Up the Modem Software

The major tasks for setting up the modem software are fairly similar, despite how you choose to use your modem.

Table 1-2 provides an overview of the major tasks for each of the three usage modes. An “x” in the table cell denotes the task is performed.

Table 1-2 Overview of Modem Software Setup

Task	Dial-in Only	Dial-out Only	Dial-in/Dial-out
Turn off the port in <i>/etc/inittab</i>	x	x	x
Execute <i>telinit q</i> to reinitialize the port	x	x	x
Run <i>fix-modem</i> configuration script	x	x	x
Edit <i>Devices</i> file		x	x
<i>Chown</i> device to <i>uucp</i>		x	x
Update action field in <i>/etc/inittab</i>	x		x

Modem Software Setup, All Modes

The installation procedure in this section is used to start the software setup for the three usage modes. Follow the procedure until you reach the point where you are directed to the section that is specific for the usage mode you have chosen. Then continue with that section. For example, if you are configuring your modem for dial-in only, complete the procedures in this section and the “Dial-in Mode” section.

1. Log in as root.

2. Edit `/etc/inittab` to turn off the port you will be using for your modem so you can configure it correctly.

Find the line for the port. The line begins with `tportnumber`, where *portnumber* is the number of a serial port. For example, if you connected your modem to port 2, look for a line similar to the following:

```
t2:23:respawn:/etc/getty -N ttyd2 co_9600 #port 2
```

Change `respawn` to `off`. This tells the system not to restart the `/etc/getty` program when it exits. If it is already off, continue to the next step. The line should look like this when you are finished:

```
t2:23:off:/etc/getty -N ttyd2 co_9600 #port 2
```

3. Write and exit the `/etc/inittab` file.
4. Inform `init` about the change to `inittab` with the following command:
`telinit q`
5. Run the modem configuration script for your modem type (see Table 1-1). Select the appropriate argument for its usage mode from Table 1-3 for the port you are configuring.

Table 1-3 Determining the Usage Mode for Modem Configuration

Dial-in	Dial-out	Dial-in/Dial-out
-i	-o	-io

Note: Be sure to view the entire configuration script for information and command arguments specific to your modem type before running the script.

For example, if you want to configure a US Robotics Sportster modem for dial-in/dial-out, connected to port 2, use the `fix-usr` command and specify a dial-in/dial-out argument (`-io`) and the port number to which the modem is connected. For example, enter the command:

```
/etc/uucp/fix-usr -io 2
```

You can also specify the optional modem model type and the modem speed, using the command:

```
/etc/uucp/fix-usr -io -m SPORT -s 38400 2
```

As another example, to configure a Telebit modem for dial-in only mode, use the *fix-telebit* command and specify a dial-in modem (-i), the model of the modem (tb+, t1000, t1600, t2500) and its speed, and the port number to which the modem is connected.

In this example, if a T2500 modem is connected to port 2 at 19,200 baud, dial-in only, enter the following command:

```
/etc/uucp/fix-telebit -i -m t2500 -s 19200 2
```

- If you are configuring a dial-out modem, continue with the section, “Dial-out Mode” on page 14.
- If you are configuring a dial-in modem, continue with the section, “Dial-in Mode” on page 15.
- If you are configuring a dial-in/dial-out modem, continue with the section, “Dial-out Mode” on page 14, followed by “Dial-in Mode” on page 15.

Dial-out Mode

This section describes additional steps for configuring dial-out only and dial-in/dial-out modems. Be sure you have completed the steps in “Modem Software Setup, All Modes” on page 12 before starting the setup in this section.

You turn on dial-out modem software by editing the */etc/uucp/Devices* file. You also have to change ownership of the port to uucp.

1. Edit the */etc/uucp/Devices* file to specify the port to which the modem is connected and to set the correct baud rate for the modem. Also, add a “Direct” line for use in debugging problems. The lines have the following forms:

```
ACU ttynportnumber null baud 212 x dialer  
Direct ttydportnumber - baud direct
```

- If your modem runs at 2400 baud or lower, the *n* field should be **m**.
- If it runs faster than 2400 baud and supports hardware flow control, the *n* field should be **f**.
- All other cases can use **d**.

The value for *portnumber* is the number of the serial port to which the modem is connected. The *baud* variable is the baud rate at which you will run the modem. The *dialer* variable is the model tag, as listed in the first field of the */etc/uucp/Dialers* file.

Note: If your modem type is not listed in the *Dialers* file, you may want to obtain the most up-to-date version of the *Dialers* file. It is available online from Silicon Graphics with the *ftp* utility.

For example, for a US Robotics modem connected to serial port 2 that can accept and send data at 38400 baud (using RTS/CTS hardware flow control), add the following lines to the *Devices* file:

```
ACU ttyf2 null 38400 212 x usr
Direct ttyd2 - 38400 direct
Direct ttyf2 - 38400 direct
```

For a Telebit modem connected to serial port 2 and running at 19200 baud, add the following lines:

```
ACU ttyf2 null 19200 212 x telebit
Direct ttyd2 - 19200 direct
Direct ttyf2 - 19200 direct
```

Save and exit the *Devices* file.

2. Change the ownership of the device files. For example, if you are using serial port 2, give the following command (substituting the appropriate *tty* device as determined above):

```
chown uucp /dev/tty[dfm]2
```

- If you are configuring your modem for dial-in/dial-out mode, continue with the section “Dial-in Mode” on page 15.

When you have finished the above process, try to dial out through the modem to verify the dial-out setup, as described in “Dialing Out to Another Modem” on page 17. For more information on dialer options, see the file */etc/uucp/Dialers*.

Dial-in Mode

Be sure you have completed the steps in “Modem Software Setup, All Modes” on page 12. For dial-in/dial-out mode, be sure you have also completed the steps in “Dial-out Mode” on page 14.

Complete the configuration by editing the file `/etc/inittab`. Perform the following steps:

1. Edit the file `/etc/inittab` again so the system recognizes that the port is used for the modem (as opposed to a terminal). This editing also sets the interface speed on the port and enables user logins through the port.

Find the line for the port to which the modem is connected. The line begins with `tportnumber`, where `portnumber` is the number of a serial port. For example, if you connected your modem to port 2, look for a line similar to the following:

```
t2:23:off:/etc/getty -N ttyd2 co_9600 #port 2
```

Make the following changes to this line:

- Change `off` to **respawn**.
- Change `/etc/getty` to `/usr/lib/uucp/uugetty`.
- Add the dial-in **-i** option with the modem model argument. Pick the modem model variable from Table 1-4 (or check the `/etc/uucp/Dialers` file for additional variables):

Table 1-4 inittab Dial-in Options by Modem Type

Modem Type	-i Argument
Telebit T2500, T1600, QBlazer, T3000, and WorldBlazer	telebitin,conn
ZyXEL U-1496	zyin,conn
Intel 14.4ex	intelin,conn
DSI 9624 models	dsiin,conn
US Robotics (also called USR)	usrin,conn
Hayes ACCURA (also called Hayes14)	hayesin,conn

- Change `ttyd2` to **tt~~y~~f2**.
- Change `co_9600` to **d~~x~~_38400** (this is the serial port interface speed. It lets the modem answer and connect at 38400 and slower baud rates.) Do not set this value to a speed slower than your modem.
- Change the comment at the end of the line to note that this port is now used for a modem; include its phone number for reference.

For example, if you connected a US Robotics modem to serial port 2, and found a line similar to this line in */etc/inittab*:

```
t2:23:off:/etc/getty -N ttyd2 co_9600 #port 2
```

You would change it to read:

```
t2:23:respawn:/usr/lib/uucp/uugetty -Nt60 -iusrin,conn ttyf2
dx_38400 #Modem 555-2345
```

If you are setting up a Telebit modem to serial port 2, and found a line similar to this one:

```
t2:23:off:/etc/getty -N ttyd2 co_9600 #port 2
```

You would change it to read:

```
t2:23:respawn:/usr/lib/uucp/uugetty -Nt60 -itelebitin,conn ttyf2
dx_19200 #Modem 555-3456
```

2. Write and exit the */etc/inittab* file.
3. Inform *init* about the change to */etc/inittab* with the following command:

```
telinit q
```

To test the dial-in setup, have someone try to dial in to the system through this modem.

To dial out through the modem to verify the dial-out setup, continue with the section "Dialing Out to Another Modem."

Dialing Out to Another Modem

The *cu* utility dials your modem. Before you attempt to dial out, make sure your local modem is connected to both your system and a working telephone line. Also, make sure the modem is turned on. The *cu* syntax to dial the number 1-800-555-1234 is:

```
cu 18005551234
```

Refer to the *cu(1)* reference page for complete information on *cu*. If everything is working, you should hear the modem dialing, and after a short time you should see the following message on your console:

```
Connected
```

- If no login prompt is displayed, press **<Enter>** once.
- If a prompt is displayed, continue and log in to the remote system.
- To disconnect, press the **<Enter>** key, the tilde key (~), a period (.), and the **<Enter>** key again.
- If you cannot connect, refer to the troubleshooting tips in “Troubleshooting Your Modem Setup” in Chapter 5.

Managing the TTY Subsystem

This section covers the following topics:

- “Checking Line Settings Using IRIX Shell Commands”
- “Creating and Testing Line Settings”
- “Modifying Line Characteristics”

Checking Line Settings Using IRIX Shell Commands

The `/etc/gettydefs` file contains information used by the `getty` command to establish the speed and terminal settings for a line. The general format of the `gettydefs` file is:

label# initial-flags # final-flags #login-prompt #next-label

The following example shows a few lines from a `gettydefs` file:

```
dx_9600# B9600 # B9600 SANE TAB3 HUPCL #\r\n\n$HOSTNAME login: #dx_9600
dx_4800# B4800 # B4800 SANE TAB3 HUPCL #\r\n\n$HOSTNAME login: #dx_4800
dx_2400# B2400 # B2400 SANE TAB3 HUPCL #\r\n\n$HOSTNAME login: #dx_2400
dx_1200# B1200 # B1200 SANE TAB3 HUPCL #\r\n\n$HOSTNAME login: #dx_1200
du_1200# B1200 # B1200 SANE TAB3 HUPCL #\r\n\n$HOSTNAME login: #du_300
du_300# B300 # B300 SANE TAB3 HUPCL #\r\n\n$HOSTNAME login: #du_2400
du_2400# B2400 # B2400 SANE TAB3 HUPCL #\r\n\n$HOSTNAME login: #du_1200
```

The entries beginning with `dx` are typically used for terminals or modems; those beginning with `du` are typically for low-speed modems. See `gettydefs(4)` for more information on the fields of each entry.

These entries form a single, rotary hunt sequence; the last field on each line is the label of the next line. The next-label field for the last line shown points back to the first line in the sequence. The object of the hunt sequence is to link a range of line speeds. If you see garbage characters instead of a clear login prompt, press the <Esc> key to force *getty* to step to the next entry in the sequence. The hunt continues until the baud rate of the line matches the speed of the user's terminal.

The flag fields shown have the following meanings:

<i>B300-B19200</i>	The baud rate of the line.
<i>HUPCL</i>	Hang up on close.
<i>SANE</i>	A composite flag that stands for a set of normal line characteristics.
<i>IXANY</i>	Allow any character to restart output. If this flag is not specified, only DC1 <ctr1-q> restarts output.
<i>TAB3</i>	Send tabs to the terminal as spaces.

For a description of all *getty* flags, see *termio(7)*.

Creating and Testing Line Settings

Create new lines for the *gettydefs* file by following the example shown in "Checking Line Settings Using IRIX Shell Commands." Each entry in the file is followed by a blank line. After editing the file, run the command:

```
/etc/getty -c /etc/gettydefs
```

This causes *getty* to scan the file and print the results on your terminal. Any unrecognized modes or improperly constructed entries are reported.

Modifying Line Characteristics

You can modify TTY line characteristics using an IRIX editor, such as *vi*, to edit */etc/inittab*.

The */etc/inittab* file contains configuration instructions for the */etc/init* command. The general format of a line entry in the */etc/inittab* file is as follows:

```
identification:level:action:process
```

The four colon-separated fields are as follows:

<i>identification</i>	A unique one or two character identifier for the line entry.
<i>level</i>	The run-level(s) in which the entry is to be performed.
<i>action</i>	How <i>/etc/init</i> treats the process field (refer to the <i>inittab(4)</i> reference page for complete information).
<i>process</i>	The shell command to be executed.

/etc/inittab contains several entries that spawn *getty* processes. The following example is a selection of such entries from a sample */etc/inittab*:

```
t1:23:respawn:/etc/getty -s console ttyd1 co_9600
t2:23:respawn:/etc/getty ttyd2 co_9600
```

There are at least three things you might want to do to an *inittab* entry for a TTY line:

- Change the action. Two actions that apply to TTY lines are "respawn" and "off" (see the *inittab(4)* reference page for complete information on this field).
- Add or change arguments to */etc/getty* in the process field. A frequently used argument is *-tmm*. This tells *getty* to hang up if nothing is received within *mm* seconds. It's good practice to use the *-t* argument on dialup lines.
- Add or change comments. You can start your comments after a pound sign (#), and insert comments inline after a semi-colon (;).

Serial Port Pin Signal Tables

This section outlines the serial support on IRIS workstations and servers and provides instructions for connecting peripheral devices to the serial ports. For information on the optional serial ports, see the document that accompanies that hardware option.

Defining the Serial Interface

The workstation or server provides an RS-232 (or RS-423) or a DB-9 (9-pin) compatible serial interface. Additionally, some systems are equipped with the mini-DIN8 serial interface. These ports do not lay out the pins in a conventional pattern such as DB-9 or RS-232; instead, the plug is cylindrical. Adapter cables are available commercially or through Silicon Graphics to connect mini-DIN8 ports with DB-9 and RS-232 ports. Consult your system *Owner's Guide* for complete information on your system's serial ports. All serial data cables that you connect to the computer should be shielded. The computer can easily drive and receive signals on a 50-foot cable, and it typically drives and receives signals on a cable up to 200 feet long.

There are two types of serial interface equipment available: *Data Terminal Equipment (DTE)* and *Data Communications Equipment (DCE)*. The primary difference between DTE and DCE is the use of several pins on the connector. For example, DTE devices output on pin 2 and input on pin 3. DCE devices output on pin 3 and input on pin 2. You can connect a DTE interface directly to a DCE interface.

To connect either a DCE to a DCE, or a DTE to a DTE, use a *null modem* cable. A null modem cable allows equipment interfaces of the same type to be connected by swapping the appropriate input and output pins.

The serial ports for workstations and servers are all configured as DTE devices. Most terminals are also configured as DTE devices. Therefore, to connect a terminal to a workstation that has RS-232 or DB-9 connectors, use a null modem cable (if this connection does not work correctly, consult the documentation for your terminal to determine its requirements). To connect a modem to the workstation, use a cable that connects each pin of the serial port to the corresponding pin of the modem. No signals need to be swapped. Ensure that your cable is a straight connection before using it with your system. Connect other peripheral devices according to the configuration data provided with the device.

Silicon Graphics provides three special files that determine the signals that are used on each port. The special files beginning with *ttyd* are used for devices such as terminals; the files *ttymX* are used for modems that do not use hardware flow control; and *tyfX* files are used for devices that can use hardware flow control on the RTS and CTS pins.

Cabling the Serial Ports

This section describes the cables typically used to connect the computer to terminals and modems. The serial ports are designed to connect directly to Data Communications Equipment (DCE) devices such as modems through a serial cable. Each wire on this cable connects the same pin of the computer socket to the device, that is, it has a pin-to-pin correspondence. Connecting to Data Terminal Equipment (DTE) devices such as terminals requires a different cable arrangement: a *null modem* cable. A null modem cable has input and output signals swapped between the two connectors.

DB-9 Serial Connector Cabling

Table 1-5 lists the pin definitions for an example of a 9-pin (DB-9) null modem cable, as shown in Figure 1-1. Connect pins that are shown separated by commas in the Terminal column together with a wire or jumper at that end of the cable. Then connect these joined pins to the pin shown in the 9-pin column at the 9-pin end of the cable.

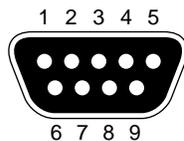


Figure 1-1 DB-9 Connector

Table 1-5 Pin Definitions for a Null Modem Cable

9-Pin	Signal	Terminal	Signal
2	Transmit Data	3	Receive Data
3	Receive Data	2	Transmit Data
4	Request To Send	5	Clear To Send
5	Clear To Send	4	Request To Send
7	Signal Ground	7	Signal Ground
8	Data Carrier Detect	6,8	Data Set Ready and Data Carrier Detect
9	Data Terminal Ready	20	Data Terminal Ready

Most terminals do not require the various handshaking lines such as *Clear To Send* or *Data Set Ready* and work with a three-wire null modem cable. To make one of these, you simply need to swap the signals for pins 2 and 3, and you need to connect pin 7 of the computer to pin 7 of the terminal. Table 1-6 lists the pin definitions for a three-wire null modem cable.

Table 1-6 Pin Definitions for a Three-wire Null Modem Terminal Cable

Computer	Signal	Terminal	Signal
2	Transmit Data	3	Receive Data
3	Receive Data	2	Transmit Data
7	Signal Ground	7	Signal Ground

Note that the pinout of these DB-9 connectors is different from that of the DIN connectors that may be next to them. These DIN connectors also have different pinouts from the mini-DIN8 connectors used on some systems and documented in “Mini-DIN8 Serial Connector Cabling.” The DB-9 and DIN connectors are connected to the same internal port hardware.

Caution: Do not use a cable designed for an IBM PC/AT™ compatible 9-pin connector. It does not work correctly with your Silicon Graphics workstation.

For modem devices that use RTS/CTS hardware flow control, the following pinout allows “full flow control.” This cable is required to implement */dev/ttyf** devices. This cable also supports */dev/tty** devices. (When using *ttym* devices, ensure that the connected device does not require flow control signals.) Figure 1-2 shows the DB-9 to DB-25 RTS/CTS cable and Table 1-7 lists the correct pin-out for hardware flow control devices.

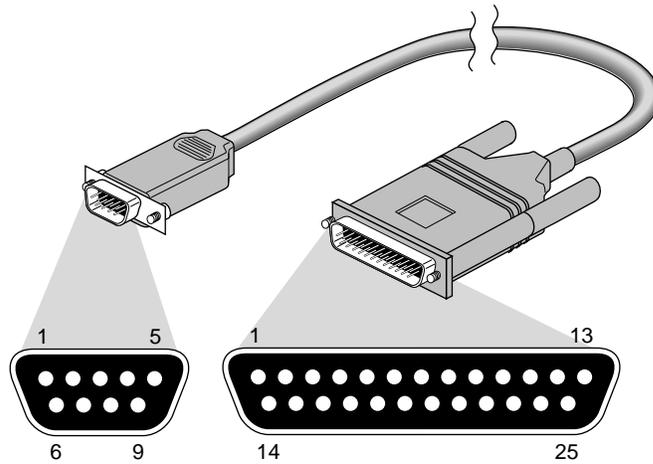


Figure 1-2 DB-9 to DB-25 RTS/CTS Modem Control Cable

Table 1-7 Pin Definitions for a DB-9 to DB-25 RTS/CTS Modem Control Cable

DB-9-Male	Signal	DB-25-Male	Signal
1	not used	1	not used
2	Transmit Data	2	Transmit Data
3	Receive Data	3	Receive Data
4	Request to Send*	4	Request to Send*
5	Clear to Send*	5	Clear to Send
6	not used	6	not used
7	Signal Ground	7	Signal Ground
8	Data Carrier Detect	8	Data Carrier Detect
9	Data Terminal Ready	20	Data Terminal Ready

* RTS and CTS are ignored (optional) if using `/dev/ttym*` but required if using `/dev/ttyf*`.

Note: This cable can be used with a null modem adapter for terminals and printers (see Table 1-5); however, it is recommended that you use this cable exclusively for modem connections. The IBM PC/AT to modem cable (“off the shelf cable”) does not work properly with your workstation. For additional information see the serial(7) reference page.

Mini-DIN8 Serial Connector Cabling

There are three basic cable configurations for the mini-DIN8 serial ports. See your *Owner's Guide* to determine if you have a mini-DIN8 port on your workstation or server. Depending on the cables used, some functionality may be sacrificed. Note that the pinout of these mini-DIN8 connectors is different from that of the DIN connectors on larger systems. These larger systems also have DB-9 connectors that are connected to the same internal port hardware.

For most dumb terminals you should use a commercially available cable, Macintosh SE[®] to ImageWriter1. This cable uses the normal three-wire connection and is used as a `/dev/ttyd*` device.

Figure 1-3 shows the mini-DIN8 to DB-25 Serial Terminal Cable and Table 1-8 lists the pin configuration.

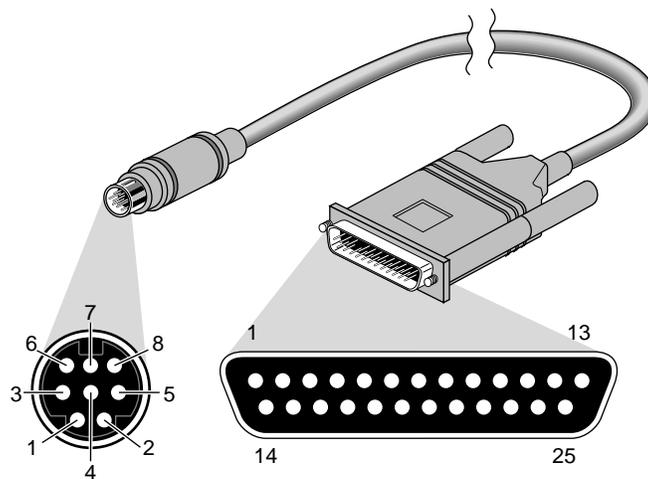


Figure 1-3 Mini-DIN8 to DB-25 Serial Terminal Cable

Table 1-8 Pin Definitions for a Mini-DIN8 to DB-25 Serial Terminal Cable

Mini-DIN8-Male	Signal	DB-25-Male	Signal
1	not used	1	not used
2	not used	2	Transmit Data
3	Transmit Data	3	Receive Data
4	Signal Ground	4	Request to Send
5	Receive Data	5	Clear to Send
6	not used	8	Data Carrier Detect*
7	Data Carrier Detect*	20	Data Terminal Ready
8	Signal Ground		

* */dev/tty** devices should be used with this cable only if the system must notice when the terminal or printer is powered off.

Note: A Macintosh SE cable has other pins connected but they can be ignored.

For modem devices using RTS/CTS hardware flow control, the following pinout allows “full flow control.” This cable is required to implement */dev/ttyf** devices. It also supports */dev/ttym** devices. Table 1-9 shows the pinout:

Table 1-9 Pin Definitions for a Mini-DIN8 RTS/CTS Modem Cable

Mini-DIN8-Male	Signal	DB-25-Male	Signal
1	Data Terminal Ready	20	Data Terminal Ready
2	Clear to Send*	5	Clear to Send*
3	Transmit Data	2	Transmit Data
4	Signal Ground	7	Signal Ground
5	Receive Data	3	Receive Data
6	Request to Send*	4	Request to Send*
7	Data Carrier Detect	8	Data Carrier Detect
8	Signal Ground	7	Signal Ground

* RTS and CTS are ignored (optional) if using */dev/ttym** but required if using */dev/ttyf**

Note: This cable is available from Silicon Graphics. Contact your sales representative. This cable can be used with a null modem adapter for terminals and printers (see Table 1-5); however, you should use this cable exclusively for modem connections. The commercially available cable that connects a Macintosh SE to a modem does not work properly with Silicon Graphics software.

Printers

Under IRIX, printer services are implemented by the line printer utility, *lp*. This utility supports a spooling service that manages print requests and a user interface to control the print service.

The easiest way for you to install and maintain a printer using IRIX is to use the graphical System Manager, which is described in the *Personal System Administration Guide*. If you have a non-graphical workstation or server, use this section for instructions on how to administer your printing system.

Note: In this chapter, the terms *workstation* and *server* are used interchangeably, because the interface described here is identical between both kinds of systems.

This chapter covers the following topics:

- Introduction to the *lp* spooling system. See “Understanding the *lp* Spooler” on page 30.
- The *lp* commands, categorized by general users and administrators. See “Command Interface to *lp*” on page 32.
- Maintaining the *lp* system. Describes tasks such as adding and removing local and network printers, changing default printers, and clearing log files. See “Managing the *lp* System” on page 41.
- Printer cabling. See “Printer Cable Pin Signal Tables” on page 53.
- The BSD *lpr* software. “Configuring the BSD *lpr* Spooler System” on page 48.

For troubleshooting and error messages, refer to:

- Troubleshooting *lp* system problems. See “Troubleshooting Your Printing System” on page 84.
- Troubleshooting BSD *lpr* system problems. See “Troubleshooting the BSD *lpr* Spooling System” on page 87.
- *lp* error messages. See “*lp* Error Messages” on page 95.

Understanding the lp Spooler

The line printer (*lp*) spooling utility is software that manages the print spooling system. Spooling is a technique that temporarily stores data until it is ready to be processed (in this case, by your printer). For *lp* spooling, a file (or group of files) to be printed is stored in a queue until a printer becomes available. When the printer is available, the next file in the queue is printed.

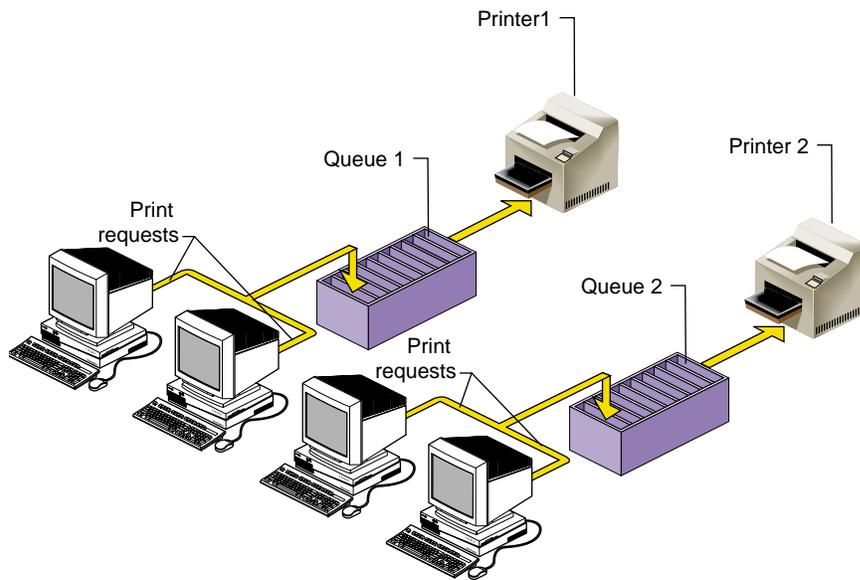


Figure 2-1 Print Spooling

lp spooling allows background printing; users can immediately use their workstations while their print jobs await the printer. With *lp* spooling, printers can be shared among many users. The flow of printing is regulated by the *lp* spooling utility.

The user interface to *lp* is a series of user and administrator commands that permit control of individual print jobs and management of printer resources.

Terms

Throughout this chapter, these terms represent important concepts in the *lp* spooling system:

- printer** A logical name that points to an interface file, which represents a physical device, that is, the actual printer.
- class** The name given to an ordered list of one or more printers. A printer may be assigned to more than one class, but need not be a member of any class. Figure 2-2 shows printers divided into two classes: one for black and white printers and one for color printers.
- destination** The place an *lp* request is sent to await printing. The destination may be a specific printer or a class of printers. An output request sent to a specific printer is printed only by that printer; a request sent to a class of printers is printed by the first available printer in its class. Default destinations are initially set up but can later be modified, as needed.

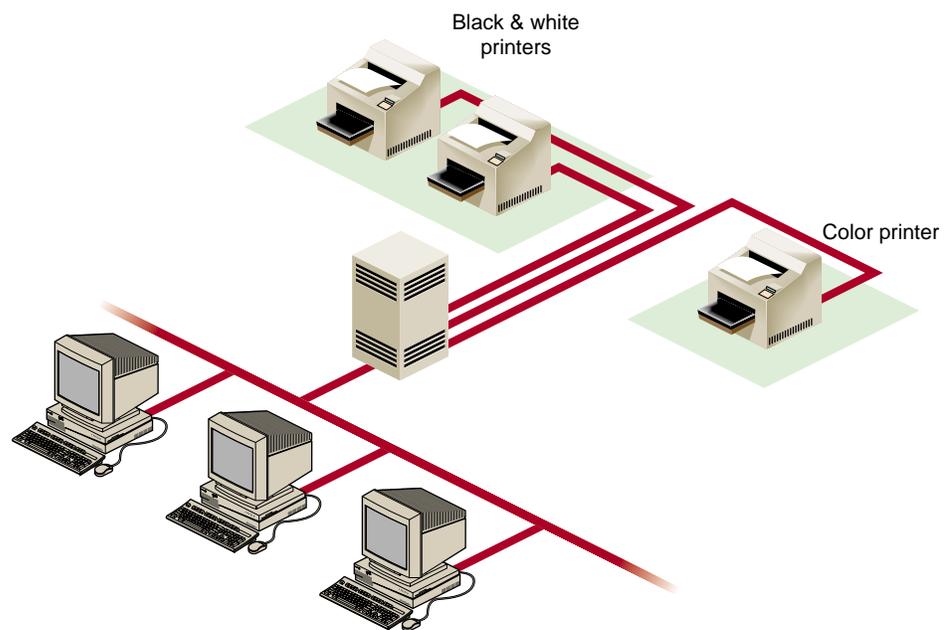


Figure 2-2 Printer Classes

Command Interface to lp

The command interface to *lp* consists of a set of commands for users sending their jobs to the printer and a set of administrator commands for managing printer resources.

The *lp* spooling utility allows:

- Grouping printers together into logical classes to maximize throughput. (These printers need not be the same type.)
- Configuring your system to spool to a class of printers.
- Queuing print requests so they (jobs) can be processed by the next available printer.
- Canceling a print request, so that an unnecessary job is not printed.
- Starting and stopping *lp* from processing print requests.
- Changing printer configurations.
- Reporting the status of the *lp* scheduler.
- Restarting any printing that was not completed if the system was powered down.
- Moving print requests and queues from one printer or class of printers to another.

User Command Summary

The commands described in this section allow users on workstations and across the network to access the printing facilities. Users should use the *lp* and *cancel* commands most frequently, the *lpstat* command occasionally, and the *enable* and *disable* commands infrequently, if ever. No special privileges are necessary to use these commands.

This section describes the five basic *lp* commands.

<i>lp</i>	Routes jobs to a destination and places them in a queue. The destination may be either a single printer or a class of printers.
<i>cancel</i>	Cancels spooled print requests.
<i>disable</i>	Prevents a printer from printing jobs that are in the queue.
<i>enable</i>	Allows a printer to print jobs in the queue.
<i>lpstat</i>	Reports the status of the <i>lp</i> spooling system.

lp: Send a Print Job to a Printer

The *lp* command routes a print job request to a destination where it is placed in a queue to await printing. The destination may be a single printer or a class of printers. If you do not specify a destination, the request is routed to the default destination. For information on how to set the default printer destination, see “Changing the Default Printer Destination” on page 46.

The form of the *lp* command is:

```
lp [options] filename...
```

Every time an *lp* request is made, a request ID is assigned to the job, and a record of the request is sent to you. The request ID has this form:

destination-seqnum

destination is the printer or class of printers to which the job has been routed. *seqnum* is an arbitrary sequence number assigned to the job by the *lp* system.

lp has three options that are particularly useful: **-n**, **-d**, and **-c**.

- Use **-n** to print more than one copy of a document:

```
lp -nnumber
```

number is the number of copies to print. Note that there is no space between **-n** and *number*.

- Use **-d** to specify a printer or class of printers other than the default printer (assuming your system has more than one printer defined):

```
lp -ddestination filename...
```

- Use **-c** (for *copy*) to send a copy of the file to the printer. This is important to ensure that edits you make to your files after you have issued a print request do not show up in the printed output:

```
lp -c filename...
```

You can combine these command options in any order. For a complete list of *lp* options, see the *lp(1)* reference page. Example 2-1 and Example 2-2 show some uses of the *lp* command.

Example 2-1 Simple *lp* Print Requests

```
lp myfile
lp < myfile
cat myfile | lp
```

To request a printout, you can use the *lp* command several different ways. The entries in Example 2-1 perform identical functions, sending a simple print request to the default printer.

Example 2-2 Complex *lp* Print Request

```
lp -n3 -dfoo -c myfile
```

The entry in Example 2-2 prints three copies on printer *foo* and creates a copy of the file for the printer to process, ensuring that if changes are made to the file after the print request, the original file is printed.

cancel: Remove a Print Request

The *cancel* command removes a job from the queue. You can cancel a job either before or after it starts printing, but you can cancel only one job at a time.

Any user can cancel any other user's job. If you cancel another user's print request, mail is sent to that user. Once you cancel a job, you can request that it be printed again only with the *lp* command:

```
cancel printer-name
cancel request-ID
```

Using the *printer-name* cancels the job currently being printed. Using the *request-ID* cancels the specified job whether or not it is currently being printed, as shown in Example 2-3.

Example 2-3 Using the *cancel* Command

```
cancel myprinter  
request "myprinter-16" cancelled  
cancel myprinter-17  
request "myprinter-17" cancelled
```

Issuing a *cancel* command does not work when the job is being printed on a remote printer. To cancel a print job on a remote printer, log in to the remote system and issue the *cancel* command.

disable: Stop Printer From Processing Requests

The *disable* command prevents the printer from printing jobs in the queue. Possible reasons for disabling the printer include malfunctioning hardware, paper jams, running out of paper, or end-of-day shutdowns. If a printer is busy at the time it is disabled, the request it was printing is reprinted in its entirety when you reenable the printer.

You can send job requests to a printer that has been disabled. The jobs are put in the queue but are not printed until the printer is enabled.

To *disable* a printer, type:

```
disable [-c] [-r"reason"] printer(s)
```

- Use the **-c** option to cancel the request currently being printed and disable the printer. This is useful if the current request causes the printer to behave abnormally.
- Use the **-r** option to tell other users why you disabled a printer. *reason* is a character string and must be enclosed in double quotation marks (" "). This string is displayed to anyone trying to use the disabled printer or to anyone issuing the *lpstat* command.

enable: Allow Printer to Process Requests

The *enable* command permits a printer that has been disabled to begin printing jobs from the queue. Example 2-4 shows how to use the *enable* command. To *enable* a printer, type:

```
enable printer...
```

Example 2-4 Using the *enable* Command

```
disable -r"paper jam" myprinter
printer "myprinter" now disabled
enable myprinter
printer "myprinter" now enabled
```

lpstat: Report lp Status

The *lpstat* command reports the status of various aspects of the *lp* system. To check *lp* status, type:

```
lpstat [options]
```

Use the **-t** option to display a complete report on the status of the *lp* system.

For a complete list of options, see the *lpstat(1)* reference page.

Example 2-5 shows an example *lpstat -t* command and its result:

Example 2-5 Using the *lpstat* Command

```
lpstat -t
scheduler is running
system default destination: myprinter
members of class foo:
myprinter
device for myprinter: /dev/plp
myprinter accepting requests since Jul 31 21:40
foo accepting requests since Jul 30 12:23
printer myprinter now printing foo-18
enabled since Aug 5 15:34
foo-18 mylogin 3156 Aug 7 17:11 on myprinter
```

Administrator Command Summary

This section summarizes the commands that are used to administer the *lp* system. To execute the administrative commands, you must be logged in as either root (that is, the superuser) or as lp. Inexperienced users should not use the *lp* administrative commands.

<i>lpsched</i>	Starts the <i>lp</i> scheduler
<i>lpshut</i>	Stops the <i>lp</i> scheduler
<i>reject</i>	Prevents jobs from queueing at a particular destination
<i>accept</i>	Permits job requests to queue at a particular destination
<i>lpmove</i>	Moves job requests from one destination to another
<i>lpadmin</i>	Configures the <i>lp</i> system

lpsched: Start the lp Scheduler

The *lpsched* command starts the *lp* scheduler, the background daemon responsible for scheduling *lp* requests. *lp* prints jobs only when the scheduler is running on that system. *lpsched* is executed automatically each time the computer is booted.

Every time *lpsched* is executed, it creates a file called *SCHEDLOCK* in */var/spool/lp*. When the scheduler is stopped under normal conditions, *SCHEDLOCK* is automatically removed. As long as this file exists, the system does not allow another *lp* scheduler process to run. If the scheduler stops abnormally (for example, if the system shuts down abnormally) you must remove *SCHEDLOCK* before you use the *lpsched* command.

To start the *lp* scheduler, type:

```
/usr/lib/lpsched
```

There is no response from the system to acknowledge the *lpsched* command; to verify that the scheduler is running, use *lpstat*.

lpshut: Stop the lp Scheduler

The *lpshut* command stops the *lp* scheduler and ends all printing activity. All requests that are being printed when you issue the *lpshut* command are reprinted in their entirety when the scheduler is restarted.

To stop the *lp* scheduler, type:

```
/usr/lib/lpshut
```

reject: Prevent Print Requests

The *reject* command stops *lp* from routing requests to a destination queue. For example, if a printer has been removed for repairs, or has received too many requests, you may wish to prevent new jobs from being queued at that destination.

If the printer is enabled, all requests that are in the queue when you issue the *reject* command are printed.

The *reject* command takes the form:

```
/usr/lib/reject [-r"reason" ] destination
```

Use the **-r** option to tell other users why print requests are being rejected. *reason* is a character string and is enclosed in double quotation marks (" "). This string is reported to anyone trying to use *lp* to send requests to the specified destination.

accept: Allow Print Requests

The *accept* command allows job requests to be placed in a queue at the named printer(s) or class(es) of printer(s). As shown in Example 2-6, *accept* allows a printer to receive job requests, and *reject* disables printing:

Example 2-6 Using the *accept* and *reject* Commands

```
/usr/lib/accept myprinter  
destination "myprinter" now accepting requests  
/usr/lib/reject -r"printer broken" myprinter  
destination "myprinter" is no longer accepting requests
```

lpmove: Move a Request to Another Printer

The *lpmove* command moves print requests from one destination to another. For example, if you have a printer removed for repairs, you may want to move all jobs pending on the queue to a destination with a working printer. You may also use *lpmove* to move specific requests from one destination to another, but only after you have halted the scheduler with the *lpshut* command. *lpmove* automatically rejects job requests rerouted to a destination without a printer. The *lpmove* command takes two forms:

```
/usr/lib/lpmove dest1 dest2
```

```
/usr/lib/lpmove request(s) destination
```

dest1, *dest2*, and *destination* are printers or classes of printers. *request* is a specific request ID.

In the first form of the command, all requests are moved from *dest1* to *dest2*. After the move, the printer or printers at *dest1* do not accept requests until you issue an *accept* command. All rerouted requests are renamed *dest2-nnn*, where *nnn* is a new sequence number in the queue for destination *dest2*. In the second form, which you can issue only after you stop the scheduler, the rerouted requests are renamed *destination-nnn*. When you restart the scheduler, the original destinations still accept new requests. The three commands in Example 2-7 demonstrate the use of the *lpmove* command.

Example 2-7 Using the *lpmove* and *lpshut* Commands

```
/usr/lib/lpmove myprinter yourprinter
```

```
lpshut
```

```
/usr/lib/lpmove foo-19 foo-20 yourprinter
```

```
total of 2 requests moved to yourprinter
```

lpadmin: Configure Printers

The *lpadmin* command has two primary uses:

- adding new printers to the system
- changing printer classes and destinations

Unlike most IRIX commands, *lpadmin* requires an option. The *lpadmin* command takes three forms:

lpadmin -d*destination*

lpadmin -x*destination*

lpadmin -p*printer* [*options*]

- Use the **-d** option to set the system default destination. The *destination* must already be installed when you issue the command.
- Use the **-x** option to remove the specified *destination* from the *lp* system. This form of the *lpadmin* command does not work while the scheduler is running.

You cannot remove a destination (printer or class) if it has pending requests; you must first either remove all requests with the *cancel* command or move them to other destinations with *lpmove*.

Removing the last remaining member of a class deletes that class from *lp*. Removal of a class, however, does not imply the removal of printers assigned to that class.

Note: The *rmprinter* command can also be used to remove a printer. See “Removing a Printer” on page 46.

- Use the **-p** option to reassign printers to different classes. This option has two options of its own. With these options, the *lpadmin* command takes the form:

lpadmin -p*printer* [**-c***class*] [**-r***class*]

- Use the **-c** option to assign a *printer* to the specified *class*
- Use the **-r** option to remove a *printer* from the specified *class*

The **-p** option does not work while the scheduler is running.

In Example 2-8, the first entry shuts down the *lp* scheduler. The second entry removes the printer *myprinter* from the *lp* spooling system. The printer *myprinter* is then removed from the *foo* class and assigned to the *boo* class with the third entry.

Example 2-8 Using the *lpadmin* Command

```
/usr/lib/lpshut
/usr/lib/lpadmin -xmyprinter
/usr/lib/lpadmin -pmyprinter -rfoo -cboo
```

For a complete list of options, see the *lpadmin(1M)* reference page.

Managing the lp System

This section contains procedures for adding a printer, removing a printer, changing your default printer, clearing printer *log* files, and printing over a network.

Adding a Printer

To send print requests to your printer, you must first add your printer by registering it with the *lp* spooler. The procedures for adding a printer with *lp* vary. The procedures are divided into the following groups, depending on the type of printer and how it is used:

- Parallel and serial printers on the local system; see “Adding Parallel and Serial Printers” on page 42.
- SCSI printers; see “Adding a SCSI Printer” on page 43.
- Printers connected to remote print server systems; see “Adding a Network Print Server” on page 44.

Note: The utilities described in this section, *mkcentpr*, *mkserialpr*, *mkscsipr*, and *mknetpr*, should be used whenever possible. For a list of supported printers, see the directory */var/spool/lp/model*. Additionally, these utilities list the printers they support when they are executed.

Adding Parallel and Serial Printers

Silicon Graphics systems are equipped with at least one parallel and one serial port. Both ports are on the back of the workstation or server and are clearly labeled.

The special file */dev/plp* is the interface to the parallel printer port. Some larger servers and workstations have multiple parallel ports. There is one parallel port on each IO4 board on CHALLENGE and Onyx systems.

If you have more than one parallel port on your system, the device files that refer to the ports are named according to the board slot in which the CPU board is installed. For example, if you have an IO4 board in slot 2 and another in slot 4, the device files for the parallel ports are */dev/plp2* and */dev/plp4*. One of these devices (the parallel port attached to the board designated as the primary board) is linked to the default */dev/plp*.

Normally, */dev/plp* is directly accessed only by a print spooling mechanism such as the *lp* subsystem. The special file */dev/plp* may only be opened for writing by one process at a time. However, several processes may open the device in read-only mode to find out the printer status. A printer reset is issued whenever the device file is opened for writing. For a complete description of the parallel printer interface, see the *plp(7)* reference page.

Parallel and serial printers are installed with a similar procedure, but to different ports. Your printer hardware documentation should detail the cabling requirements for the printer's interface. For a complete description of both types of interfaces, see "Printer Cable Pin Signal Tables" on page 53. Connect the printer interface cable to your printer and to the appropriate port on the back of your workstation.

To register printers connected directly to your computer via a parallel or serial port, follow these steps:

1. Become the superuser with the *su* command.
2. Stop the print spooler. Type:

```
/usr/lib/lpshut
```

3. Choose one of the following commands, depending on the type of port your printer is attached to:

- Assuming you have a printer attached to the parallel port, use the *mkcentpr* utility to install the printer in the *lp* system. Type:

```
mkcentpr
```

The *mkcentpr* utility is an interactive script that prompts you for all necessary information about your printer and then automatically registers it. You should be prepared to specify the device file for the parallel port (*/dev/lp* unless you have multiple CPU boards installed) and other specific information. Consult the *mkcentpr(1M)* reference page for complete information about *mkcentpr* syntax.

- Assuming you have a printer attached to the serial port, use the *mkserialpr* utility to install the printer in the *lp* system. Type:

```
mkserialpr
```

The *mkserialpr* command is similar to *mkcentpr* except that it is used for the serial port. Consult the *mkserialpr(1M)* reference page for complete information about *mkserialpr* syntax.

4. To set up the printer as the default printer, type the following command and substitute the printer name by which the printer is known to its users for the *printer-name*:

```
/usr/lib/lpadmin -dprinter-name
```

5. To restart the print spooler, type:

```
/usr/lib/lpsched
```

Your printer is now registered with the *lp* system and is ready for printing.

Adding a SCSI Printer

SCSI printers are supported with the optional product, Impressario™. If you have Impressario installed on a non-graphics system, use the *mkscsipr* command supplied with Impressario to register the printer. Consult the *mkscsipr(1M)* reference page for complete information about *mkscsipr* syntax.

Adding a Network Print Server

Network print servers allow several users to use the same printer, thus avoiding the expense of having a printer attached to each workstation. In a network, printers attached to remote systems are called print servers, remote systems with print servers attached are called print server systems, and the workstations accessing the print servers are print clients, as shown in Figure 2-3.

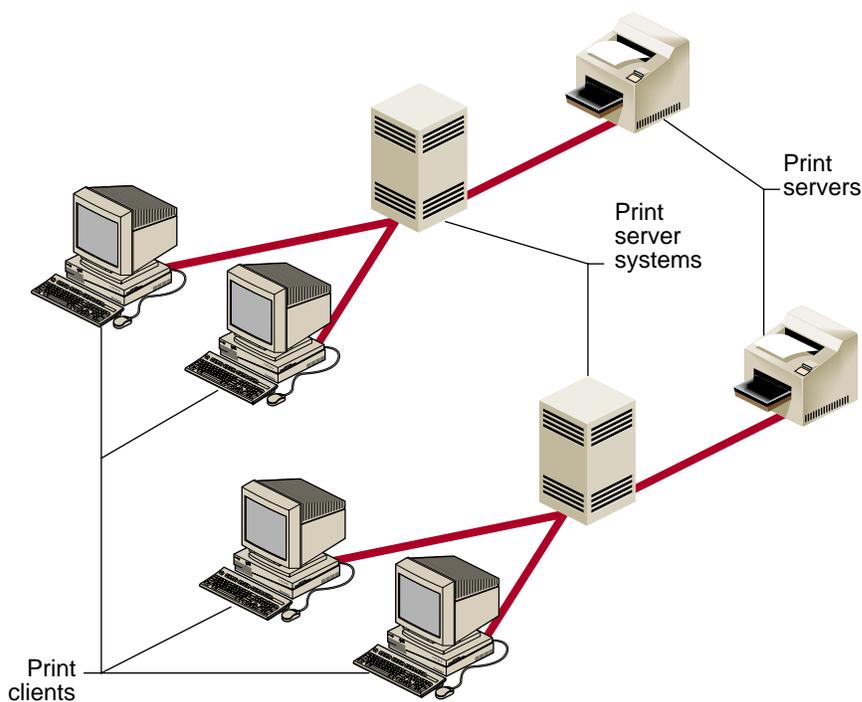


Figure 2-3 Print Servers, Print Clients, and Print Server Systems

Print servers must be configured on the print server system before remote clients can configure them successfully across the network.

To configure a print server for use across a network, follow these steps:

1. Log in as the superuser to the print server system (the system where the print server is physically attached).
2. Replace *client_name* in the command below with the hostname of the client that needs access to the print server. Type:

```
addclient client_name
```

addclient grants permission for the specified *client_name* to access print servers across the network.

Note: The *addclient* command allows anyone who has access to the *lp* account on the print client to have the privileges of the *lp* account on the print server system. This means that your *lp* system can be modified by the client system users.

If you want all remote workstations to be able to use print servers on the print server system, type:

```
addclient -a
```

3. Both print client and print server systems must be able to communicate across the network. For additional information on network communications, see IRIX Admin: Networking and Mail.
4. On the print client, become superuser.
5. To stop the spooler, type the command:

```
/usr/lib/lpshut
```

6. On the print client, add the print server to the *lp* spooler with the script *mknetpr*:

```
mknetpr queue hostname printer-name
```

queue is the name of the printer queue and should be no more than 14 characters long. *hostname* is the name of the system acting as the print server system, and *printer-name* is the name by which the print server is known to users accessing it.

7. To set up this print server as the default printer, type this command on the print client, using the *printer-name* variable from the previous step:

```
/usr/lib/lpadmin -dprinter-name
```

8. Restart the spooler with this command:

```
/usr/lib/lpsched
```

Removing a Printer

Under some circumstances, you may want to remove one or more printers from the *lp* system. The *rmprinter* utility allows you to remove a specified printer. The *preset* utility allows you to reset your entire *lp* system to the way it was when you received your workstation from Silicon Graphics. To remove a specified printer, follow these steps:

1. Become the superuser.
2. Remove the printer by entering the command below. Replace *printer-name* with the name by which the printer is known to users accessing it:

```
rmprinter printer-name
```

Your printer is now removed from the *lp* system.

To remove all printers on your system, use the *preset* command.

Caution: Use *preset* with extreme care: it removes all printer configuration information.

1. Become the superuser.
2. Type:

```
preset
```

Your *lp* system is now completely reset and all printers are removed.

Changing the Default Printer Destination

The default printer destination for a system can be an individual printer or a printer class. You can set it by using the *lpadmin* command with the **-d** option. The system default must be set by the user. A destination must already exist on the *lp* system before you can designate it as the default destination. See “Adding a Printer” on page 41 for setting up a printer.

The *lp* command determines a request’s destination by checking for a **-d** option on the command line, which if present, takes precedence over any other established destination. If **-d** is not present, *lp* checks to see if the user’s environment variable LPDEST is set. If LPDEST is set, it is used; if LPDEST is not set, then the request is routed to the default destination. Setting the environment variable LPDEST allows a user to have a default destination other than the system default.

Managing Printing on Print Server Systems

Remote printing on the print server system allows users to send print jobs over the network with the same commands that send jobs to a local printer. After the print client's *lp* spooler queues the print request, it is sent across the network to the remote print server system, where it is processed by that system's *lp* spooler. As a result of this, you cannot accurately determine the status of a remote print request by using the *lpstat* command on the local system.

This section covers two aspects of remote printing:

- "Checking Print Server Status"
- "Canceling Print Server Requests"

Checking Print Server Status

When you send a print request across the network to a print server, the local *lp* system always reports that the request is being printed, regardless of its actual status in the print server system's *lp* spooler. To check the true status, you must remotely access (using *rsh* or *rlogin*) the print server system that is processing the job. The remote *lp* scheduler changes the request ID of any job sent to it over the net to reflect the actual name of the print server and gives it a new sequence number corresponding to its place in the print server queue. To determine a specific job's status, use the *lpstat* command. Example 2-9 uses *rsh* to access the print server system:

Example 2-9 Checking the Print Server's lp Status

```
rsh hostname lpstat -t
```

hostname is the name of the print server system.

Canceling Print Server Requests

Once you know the remote print server status, you can use the *cancel* command on the print server system to cancel jobs on the print server's queue. You must cancel a remote print job from the print server once it has been sent over the network by the local *lp* system.

Example 2-10 uses *rsh* to access the print server system:

Example 2-10 Canceling Print Server Requests

```
rsh hostname cancel print-server-name
```

```
rsh hostname cancel request-ID
```

hostname is the name of the print server system. Using the *print-server-name* cancels the job currently being printed. Using the *request-ID* cancels the specified job whether or not it is currently being printed

Configuring the BSD lpr Spooler System

Silicon Graphics does not support configuring the BSD lpr print spooler locally (you cannot have the printer physically connected to a Silicon Graphics system). For information about configuring a BSD lpr print spooler on another system, refer to any documentation describing the standard BSD operating system (refer to “Additional Resources” on page xvi). Print requests can be submitted to a BSD print server once it is configured.

The BSD lpr print spooler allows you to access print servers that are attached to other systems on the network. Please be sure to check the other systems, or contact the System Administrator, to verify the type of spooling system those systems are using. Generally speaking, if a system has an */etc/printcap* file configured, it is using the BSD lpr print spooling system.

Note: Do not use the *mknetpr* command for configuring a BSD network printer. This utility supports only the System V lp Spooling System on Silicon Graphics systems.

Verify that the System Administrator of the BSD print server system includes your hostname in the print server system’s */etc/hosts.equiv* file, and that your IP address and hostname appear in the print server system’s */etc/hosts* file. You must add the print server system’s IP address and hostname to your system’s */etc/hosts* file.

If your files (documents) do not print once you have configured the BSD print spooler, see “Troubleshooting the BSD lpr Spooling System” on page 87. The troubleshooting section takes you to the point where you can see a copy of your document on the print server system. If the document disappears from the print server system’s queue without printing, contact the System Administrator of that system for further assistance.

Verifying Installation of the BSD lpr Subsystem

Enter the following command to verify that the BSD lpr print spooling system was properly installed:

```
versions -av | grep print.sw.bsdlpr
```

Example output of this command looks like this:

```
print.sw.bsdlpr etc/init.d/lpd print.sw.bsdlpr etc/printcap
print.sw.bsdlpr etc/rc0.d/K26lpd print.sw.bsdlpr etc/rc2.d/S61lpd
print.sw.bsdlpr usr/bsd/lpq print.sw.bsdlpr usr/bsd/lpr
print.sw.bsdlpr usr/bsd/lprm print.sw.bsdlpr usr/bsd/lptest
print.sw.bsdlpr usr/etc/lpc print.sw.bsdlpr usr/etc/lpd
print.sw.bsdlpr usr/etc/pac print.sw.bsdlpr usr/lib/lpf
print.sw.bsdlpr usr/spool/lpd
```

The BSD spooler is not loaded by default. Check to see if the subsystem is installed. If it is not, refer to IRIX Admin: Software Installation and Licensing and/or your release notes. Most users must use the *inst* command to install this subsystem.

Use the *versions* command (*versions -av | grep print.sw.bsdlpr*) to verify that you have the entire subsystem loaded. Creating an */etc/printcap* file is not sufficient.

After you verify that the *print.sw.bsdlpr* subsystem is installed, edit the */etc/printcap* file to configure the lpr spooling system. There are no tools to perform this function, so you need to edit the file manually. The remainder of this subsection takes you through this process. Ensure that you format the entries correctly. The */etc/printcap* file expects information in a format similar to that in the */etc/termcap* file.

Configuring the Printcap File

Before you begin editing the */etc/printcap* file, log in to your system as root.

Note: The *printcap* file is very sensitive to syntax errors. The name field must begin at the first character on a line. The print server names must be separated by pipe symbols (the vertical bar “|”). The name line must be terminated with a colon followed by a backslash (“:\”). Make sure that there are no spaces, tabs, or any other character after the backslash.

The definition lines must begin with a tab character followed by a colon (:), followed by the field you are defining, followed by an equal sign. The definition line must end with a colon. See “Printcap Examples.”

There are one name and three definition fields that must be defined. They are:

- name* This field must contain all the names the print server can be accessed as. It should include *lp* because, by default, *lpr* looks for *lp* in the name field of the */etc/printcap* file. The names are separated with pipe symbols.
- :rm* Remote print server system name. This is the name of the system that has the print server physically connected to it.
- :rp* Remote print server name. This is the name of the remote print server on the print server system that you are trying to access.
- :sd* Spool directory. This is the name of your local spool directory. If you don't use the default directory, */usr/spool/lpd*, you must create the directory using the *mkdir* command.

Printcap Examples

Following are two examples that can help you edit the */etc/printcap* file. Example 2-11 shows how to configure the printer configuration file in two lines. Example 2-12 shows how to configure each option of the *printcap* file on a separate line. The examples are followed by explanations.

Example 2-11 Printcap Example 1

```
lp|sleepy|sleepyprinter:\
    :lp=:\:rm=snowwhite.story.land:rp=doc:sd=/usr/spool/lpd:
```

Example 2-12 Printcap Example 2

```
lp|sleepy|sleepyprinter:\
    :lp=:\
    :rm=snowwhite.story.land:\
    :rp=doc:\
    :sd=/usr/spool/lpd:
```

The examples show the print server can be accessed by the names *lp*, *sleepy*, and *sleepyprinter*.

The print server system (where the print server is physically attached) is called *snowwhite.story.land*.

The name of the print server on the print server system is *doc*.

The local spool directory is called */usr/spool/lpd*.

Using the `lpr` Command to Print

Now that you have the `/etc/printcap` file configured, make sure that the daemon is running. Enter the following command:

```
ps -ef | grep lpd
```

Your system should return something similar to:

```
root 195 1 0 11:06:04 ? 0:00 /usr/etc/lpd
root 1293 753 2 13:20:39 ttyq6 0:00 grep lpd
```

The `/usr/etc/lpd` path at the end of the line indicates that the daemon is running. If only one line was returned (`grep lpd`), then start the `lpd` daemon by entering the following command:

```
/usr/etc/lpd
```

Now, if you type `ps -ef | grep lpd`, you should see two lines.

As shown in “Printcap Examples” on page 51, the print server can be accessed by three names; `lp`, `sleepy` and `sleepyprinter`. By default, the `lpr` command looks for the field `lp` in the `/etc/printcap` file. If this is not the first name in the name field, then you need to do one of two things:

- Whenever you enter the `lpr` command you must use the `-P` option to specify a printer name other than `lp`.
`lpr -Psleepyprinter filename`
- Alternately, in your shell you can set the environment variable `PRINTER` to the name of the print server you wish to use.

In the C shell:

```
setenv PRINTER sleepy
```

In the Bourne shell:

```
PRINTER=sleepy; export PRINTER
```

Now you can enter the command:

```
lpr filename
```

After submitting your request to the print server, you can see if your job has made it to the print spooling queue by entering the following command:

```
lpq
```

Your system should return something similar to:

```
lp is ready and printing
Rank Owner Job Files Total Size
1st nina 113 filename 851 bytes
```

Printer Cable Pin Signal Tables

Your workstation or server has one or more serial ports and one or more parallel ports. The serial ports on your system are either DB-9 format or mini-DIN8. The parallel port is the industry-standard 25-pin parallel port. For your use in determining correct cabling, the following sections provide pin signal tables for the serial and parallel ports on your system.

Parallel Port Pin Signal Table

The parallel port on your system uses industry standard parallel printer cables to connect to common printers. It is recommended that you use a parallel cable no longer than 10 feet. Using a parallel hookup usually results in faster printing from your system. Also, because serial ports are generally more in demand for modems, terminals, and other peripheral devices, using the parallel port, when possible, saves a serial port for other uses.

Table 2-1 lists the parallel port pin assignments, as shown in Figure 2-4.

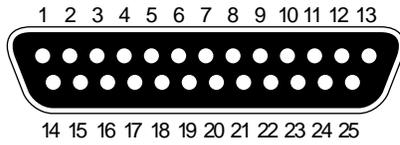


Figure 2-4 Parallel Port Pin Assignment

Table 2-1 Pin Definitions for a Parallel Port Cable

Pin	Signal
1	STB
2	DATA1
3	DATA2
4	DATA3
5	DATA4
6	DATA5
7	DATA6
8	DATA7
9	DATA8
10	ACK
11	BUSY
12	PE
13	ONLINE
14	PR/SC
15	FAULT
16	RESET
17	NO INK

Table 2-1 Pin Definitions for a Parallel Port Cable

Pin	Signal
18	NC*
19-25	Signal Ground

*NC stands for “no connect,” meaning the wire is not used.

Serial Port Pin Signal Tables

The serial port on your workstation or server is either a DB-9 (9-pin) or a mini-DIN8 port. In either case, the port is clearly labeled as a serial port. Consult your system *Owner's Guide* to determine which kind of port your system has. There are two basic printer cable configurations; one for the DB-9 serial ports and one for the mini-DIN8 serial port. Depending on the cables used some serial functionality may be sacrificed due to unconnected wires in the cable. Note that the pinout of the DB-9 connectors is different from that of the full size DIN connectors that may be next to them. These DIN connectors also have different pinouts from the mini-DIN8 connectors used on some systems and documented in this section. The DB-9 and full-sized DIN connectors are connected to the same internal port hardware.

DB-9 Connector Cabling

For most serial printers, use the following cable. This cable uses the normal 3-wire connection and be used as a */dev/ttyd** device. Table 2-2 lists typical DB-9 serial pin assignment, as shown in Figure 2-5.

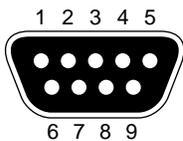


Figure 2-5 DB-9 Pin Assignment

Table 2-2 Pin Definitions for a DB-9 Serial Cable

DB-9-Male	Function
1	Transmit Data
2	Receive Data
3	
4	
5	
6	
7	Signal Ground
8	Data Carrier Detect*
9	

* Data Carrier Detect is used only with `/dev/tty*` devices if the system must notice when the printer powers off. Normally it is not used.

For printers using RTS/CTS hardware flow control, the following pinout allows “full flow control.” This cable is required to implement `/dev/tty*` devices. This cable also supports `/dev/tty*` devices. Table 2-3 lists the correct pinout for these devices, as shown in Figure 2-6.

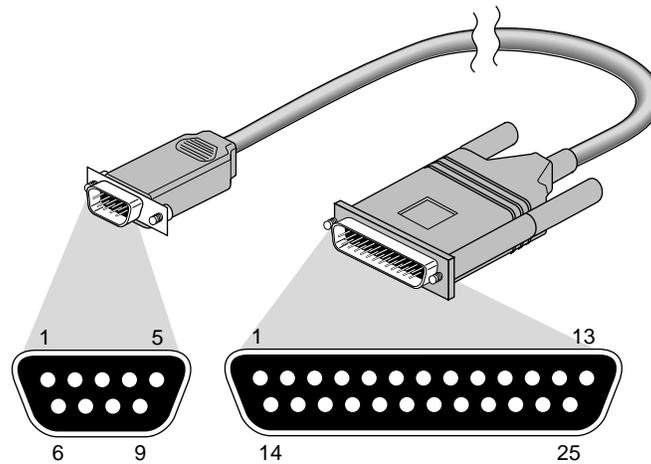


Figure 2-6 DB-9 to DB-25 RTS/CTS Flow Control Cable

Table 2-3 Pin Definitions for a DB-9 to DB-25 RTS/CTS Flow Control Cable

DB-9-Male	Signal	DB-25-Male	Signal
1	not used	1	not used
2	Transmit Data	2	Receive Data
3	Receive Data	3	Transmit Data
4	Request to Send*	4	Request to Send*
5	Clear to Send*	5	Clear to Send
6	not used	6	not used
7	Signal Ground	7	Signal Ground
8	Data Carrier Detect	8	Data Carrier Detect
9	Data Terminal Ready	20	Data Terminal Ready

* RTS and CTS are ignored (optional) if using `/dev/ttym*` but required if using `/dev/ttyf*`.

Note: This cable can be used with a null modem adapter for printers; however, it is recommended that you use this cable exclusively for modem connections. The IBM pC/AT to modem cable (“off the shelf cable”) does not work properly with your workstation. For additional information, see the serial(7) reference page.

Mini-DIN8 Connector Cabling

Many workstations and servers use the mini-DIN8 serial port. Check your system *Owner's Guide* to see if your system supports this type of connection. Note that the pinout of these mini-DIN8 connectors is different from that of the DIN connectors on larger systems. These larger systems also have DB-9 connectors that are connected to the same internal port hardware.

For most serial printers you should use a commercially available cable, Macintosh SE to ImageWriter1. This cable uses the normal three-wire connection and is used as a `/dev/ttyd*` device. Table 2-4 lists the pin configuration, as shown in Figure 2-7.

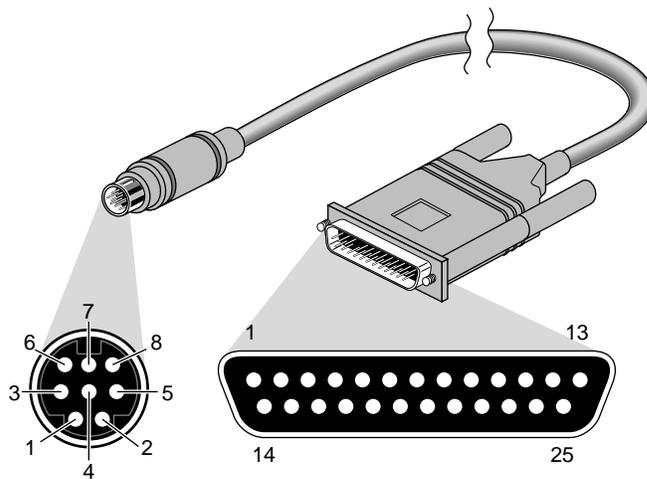


Figure 2-7 Mini-DIN8 to DB-25 Serial Cable

Table 2-4 Pin Definitions for a Mini-DIN8 to DB-25 Serial Cable

Mini-DIN8-Male	Signal	DB-25-Male	Signal
1	not used	1	not used
2	not used	2	Transmit Data
3	Transmit Data	3	Receive Data
4	Signal Ground	4	Request to Send
5	Receive Data	5	Clear to Send
6	not used	8	Data Carrier Detect*
7	Data Carrier Detect*	20	Data Terminal Ready
8	Signal Ground		

* */dev/tty** devices should be used with this cable only if the system must notice when the terminal or printer is powered off.

Note: A Macintosh SE cable has other pins connected, but they can be ignored.

CD-ROM, Floptical, and Floppy Disk Drives

This chapter discusses the software that is used to access and use CD-ROM, floptical, and floppy disk drives. It contains the following sections:

- “Floppy Disk and CD-ROM Filesystems”
- “Configuring Floppy Disk Drive Device Files”
- “Using Floppy Disk Drives”

Floppy Disk and CD-ROM Filesystems

IRIX allows you to mount and use filesystems on floppy disks, floptical disks, and CD-ROMs. You can use these filesystems on your own system, or you can export them via NFS for use on other systems (if you have NFS[®] installed). See the *ONC3/NFS Administration Guide* for more information on exporting filesystems.

The operating instructions for these kinds of filesystems are similar and are covered in detail in the *mediad(1M)* reference page.

Note: Only one instance of *mediad* is allowed per system. That is, two invocations of *mediad* for the same device generate an error.

IRIX supports the following CD-ROM, floptical disk, and floppy disk filesystem formats:

- FAT (MS-DOS[®])
- HFS (Macintosh)
- EFS and XFS (IRIX filesystem)
- ISO 9660
- Photo CD
- High Sierra
- Music CD format

CD-ROM Filesystems

mediad monitors CD-ROM drives, waiting for a CD-ROM to be inserted. When a CD-ROM is inserted, the filesystem it contains is mounted if the filesystem is in EFS, HFS, ISO 9660, or High Sierra format. If your system is running the objectserver, CD-ROM drives are monitored automatically. When a CD-ROM containing a valid filesystem is inserted, it is automatically mounted on */CDROM* (for the first CD-ROM drive), and */CDROM2*, */CDROM3*, and so on for additional drives.

If you are not running the objectserver when you invoke *mediad*, you must give instructions for the SCSI device to be monitored and a location to mount the CD-ROM filesystem. You must also have created the mount point and ensured that the directory permissions are set appropriately (755 is usually adequate for read-only filesystems). For example, if you are not running the objectserver and you wish to start *mediad* for a CD-ROM drive with SCSI identifier 4 and the mount point */CDROM* with the mount option *ro* (for read-only), issue this command:

```
mediad -o ro -ip /dev/scsi/sc0d410 /CDROM
```

Note that CD-ROM filesystems are always read-only. When you are finished using the filesystem, issue the *eject* command, and *mediad* will attempt to unmount the filesystem. If the unmount is successful, it ejects the CD-ROM. When *mediad* is running, however, any user can unmount and eject a CD-ROM with the *eject* command.

Floppy Disk Filesystems

Note: In this chapter, the term floppy disk drive also applies to a floptical drive since both are configured and used in the same manner.

Filesystems on floppy disk drives are controlled by the *mediad* daemon. *mediad* monitors a given floppy disk drive, waiting for a disk to be inserted. If your system is running the objectserver, floppy disk drives are mounted on */floppy* if the disk is in FAT (MS-DOS) or HFS (Macintosh) format. If you have more than one floppy disk drive, floppy disks in additional drives are automatically mounted on */floppy2*, */floppy3*, and so on. If your system is not running objectserver, you must provide a location for the mount point. See the *mediad(1M)* reference page for complete information. When you are through using the floppy disk, issue the *eject* command, and *mediad* attempts to unmount the filesystem. If the unmount is successful, it ejects the floppy disk immediately.

If you are not running the objectserver and you wish to start *mediad* for a high-density floppy disk drive with SCSI identifier 7 and the mount point */floppy*, use this command:

```
mediad -ip /dev/rdisk/fds0d7.3.5hi /floppy
```

This command gives instructions for the floppy disk drive to be monitored and provides a location to mount its filesystem. You must also have created the mount point and ensured that the directory permissions are set appropriately (777 is usually required for read-write filesystems).

Configuring Floppy Disk Drive Device Files

There are a number of SCSI floppy disk drives available for use with your system. To install a floppy disk drive on an IRIX system, follow the hardware documentation that is furnished with your drive to connect it to the computer.

If you are adding a floppy disk drive to a system that does not have one, the software configuration is taken care of automatically when the system boots. When the system boots, if *hinvo* indicates that a floppy disk drive is installed, but there is no link to it through the */dev* special device files, the *MAKEDEV* program is automatically invoked to add the proper device files. For more information on the *MAKEDEV* program, refer to IRIX Admin: Disks and Filesystems.

If you are installing a floppy disk drive after your initial system installation, perform the following steps:

1. Install the hardware.
2. Log in as root (the superuser) and enter these commands:

```
cd /dev  
./MAKEDEV floppy
```

The *MAKEDEV* program creates the appropriate device nodes.

If you have removed a floppy disk drive and are installing one of a different type, follow these steps:

1. Install the hardware.
2. Log in as the superuser and enter these commands:

```
cd /dev/rdisk  
rm fds*  
./MAKEDEV floppy
```

The *MAKEDEV* program creates the appropriate device nodes according the SCSI controller, and floppy disk drive number, and type of floppy disk drive. For example, a 3.5-inch floppy disk drive configured as drive 2 on SCSI controller 0 has the device node:

```
/dev/rdsk/fds0d2.3.5
```

There are various options for the different kinds of floppy disk drives that are supported. For example, your device node could use any of the following options, depending on which option suits the hardware you are installing:

3.5	(720Kb 3.5" floppy)
3.5hi	(1.44Mb 3.5" floppy)
3.5.20m	(20.1Mb floptical)
48	(360Kb 5.25" floppy)
96	(720Kb 5.25" floppy)
96hi	(1.2Mb 5.25" floppy)

3. Use the following command to link your floppy disk drive device node with a convenient filename for access, typically */dev/floppy*. Substitute the device node information for your type of floppy disk drive for the node name used here:

```
ln -s /dev/rdsk/fds0d2.3.5 /dev/floppy
```

Using Floppy Disk Drives

This section describes how to copy files to and retrieve files from floppy and floptical disks, regardless of whether the disk is in Macintosh, DOS, or IRIX filesystem format.

Note: The term floppy disk is used interchangeably with floptical disk in this section.

Using DOS and Macintosh Floppy Disks

The *mediad* daemon automatically determines the format of a floppy disk inserted in your drive and, if it is a DOS or Macintosh floppy disk, automatically mounts the filesystem on your default mount directory. Once the filesystem is mounted, you can use typical IRIX commands such as *cd*, *ls*, and *pwd* with it. See the *mediad(1M)* reference page for complete information.

Using a Floppy Disk Drive for IRIX File Transfer

You can use a floppy disk drive like a tape drive for IRIX file transfer. You can use the standard tape archive commands to write files to the floppy disk drive if it is in DOS format. Use the *mkfp* command to create the DOS filesystem on the floppy disk. See the *mkfp(1M)* reference page for complete information. You can also use the command version of *fx(1M)* to format your floppy disk for file transfer.

When you place files on a floppy disk, it is a good idea to make a note on the disk label of the format or the exact command used to place the files on the floppy disk. This makes it much easier for you (and others) to retrieve the files from the floppy disk. Also, whenever possible, change directories to the directory that contains the file and place the file on the floppy disk using a relative pathname, rather than specifying the absolute pathname.

Also, be aware that using a floppy disk to transfer files to systems made by other manufacturers may mean that the same tools are not available on the receiving system. The *tar*, *cpio*, and *dd* tools are usually available on all UNIX systems.

In the following examples, the floppy disk drive device name is given as */dev/rdisk/fds0d3.3.5*. Your actual device name may be different.

Floppy Disk File Transfer With tar

To place a copy of the file *transfer.file* on a floppy disk with the *tar* command, use the syntax:

```
tar cvf /dev/rdisk/fds0d3.3.5 transfer.file
```

To retrieve the file, use the command:

```
tar xvf /dev/rdisk/fds0d3.3.5 transfer.file
```

To retrieve all files from a *tar* floppy disk, use the command:

```
tar xvf /dev/rdisk/fds0d3.3.5
```

or for high-density floppy disks:

```
tar xvf /dev/rdisk/fds0d3.3.5hi
```

For complete information on *tar* and its options, see the *tar(1)* reference page.

Floppy Disk File Transfer With cpio

To copy files to floppy disk with *cpio*, use the command:

```
ls transfer.file | cpio -oc > /dev/rdisk/fds0d3.3.5
```

To retrieve the file again, use the command:

```
cat /dev/rdisk/fds0d3.3.5 | cpio -i
```

For complete information on *cpio* and its options, see the *cpio(1)* reference page.

Floppy Disk File Transfer With dd

This *dd* command copies a file to the floppy disk:

```
dd if=transfer.file of=/dev/rdisk/fds0d3.3.5 conv=sync
```

The following command extracts the same file:

```
dd if=/dev/rdisk/fds0d3.3.5 of=transfer.file conv=sync
```

Note: *dd* works only with single files. You can use *tar* or *cpio* to create an archive file, though, and then use *dd* to transfer that archive. If you attempt to extract the file on another brand of workstation and you experience an error, try adding the *conv=swab* statement to your extraction command line. For complete information on *dd*, see the *dd(1)* reference page.

Tape Drives

This chapter covers what you need to know about the tape drives on your workstation or server. The cartridge tape device is used primarily for filesystem backups and data transfer.

The following sections are contained in this chapter:

- “Adding a Tape Drive”
- “1/2-Inch Tape Drives”
- “8mm and 4mm Tape Drives”
- “QIC Tape Drives”
- “Using dump With DAT Tapes”

For information on backing up data onto tapes, see *IRIX Admin: Backup, Security, and Accounting*. If you are installing a tape drive, see the installation instructions furnished with the hardware.

Almost all workstations are configured with some sort of tape device for making backup copies of your files. Whether you maintain one system or a network of hundreds of workstations, you will eventually have to use and maintain some form of tape drive.

Adding a Tape Drive

To install a tape drive on an IRIX system, follow the hardware installation instructions that is furnished with your tape drive. Make sure you carefully follow any instructions regarding drive terminators.

If you are adding a tape drive to a system that does not have one, the software configuration is taken care of automatically when the system boots. When the system boots, if *hinvt* indicates that a tape drive is installed, but there is no link to it through the */dev/tape* file, the *MAKEDEV* program is automatically invoked to add the proper device nodes.

If you are installing a tape drive after your initial system installation, perform the following steps:

1. Install the hardware.
2. Log in as the superuser and enter these commands:

```
cd /dev
./MAKEDEV tape
```

The *MAKEDEV* program creates the appropriate device nodes.

If you have removed a tape drive and are installing one of a different type, follow these steps:

1. Install the hardware.
2. Log in as the superuser and enter these commands:

```
cd /dev
rm *tape
./MAKEDEV tape tapelinks
```

The *MAKEDEV* program creates the appropriate device nodes and links the correct node for the drive to */dev/tape*.

MAKEDEV Commands for Tape Drives

The *MAKEDEV* program supports these options for tape drives:

<i>tape</i>	Creates all the <i>tps</i> and <i>xmt</i> tape devices, then makes links to <i>tape</i> , <i>nrtape</i> , <i>tapens</i> , and <i>nrtapens</i> for the first tape drive found, if one exists. It first checks for <i>xmt</i> , then for SCSI in reverse target ID order.
<i>qictape</i>	Creates special files for 1/4-inch cartridge tape drives connected to an ISI QIC-O2 tape controller.
<i>magtape</i>	Creates special files for 1/2-inch tape drives connected to a Xylogics Model 772 tape controller. See <i>xmt(7M)</i> for details.
<i>links</i>	Creates both disk and tape special files.
<i>tps</i>	Creates special files for SCSI tape drives. See <i>tps(7M)</i> for details.
<i>tapelinks</i>	Makes only links to <i>tape</i> , <i>nrtape</i> , <i>tapens</i> , and <i>nrtapens</i> . Examine the target <i>tapelinks</i> in the script <i>/dev/MAKEDEV</i> for more information.

Tape Capacities

Table 4-1 and Table 4-2 list the maximum tape capacities in megabytes (MB) for the tape formats IRIX supports. Note that these are maximum, not average, capacities.

Table 4-1 Cartridge Tape and DAT Capacities

Format	Capacity (max.)
QIC24	60 MB (only reads/writes QIC24)
QIC150	150 MB with 600XTD and 6150 tapes (reads QIC24, writes QIC120, and QIC150), 120 MB with 600A tapes (writes in QIC120 format), and 250 MB with 6250 tapes
DAT	1300 MB with 60-meter (1 hr) cartridge, 2000 MB with 90 meter (1.5 hr) cartridge; uses the DDS (not DataDAT) format
8mm	2093 MB with 112 meter (120 min) P6 (US) cartridge; 2279 MB with 122 meter (90 min) P5 (European) cartridge

Note: Almost all DAT drives use DDS format. 8mm tapes are also available in P6 lengths of 15, 30, 60, and 90 minutes for the U.S., and lengths of 15, 30, and 60 minutes for Europe; the P6 cartridge is for NTSC, and the P5 is for PAL. The drive must be jumpered to match the cartridge type.

Table 4-2 shows maximum capacities for 9-track tapes. Note that 9-track tape capacities vary more than other types because of block size and tape length issues.

Table 4-2 9-Track Tape Capacities

BPI	BLKSZ	200 ft Length 6" Reel Size	600 ft Length 7" Reel Size	2400 ft Length 10.5" Reel Size	3600 ft Length 10.5" Reel Size
800	512	1	3	10	15
		8192	1.8	5.5	21
		64K	2	6	23
1600	512	1.3	4	15	22
		8192	3.5	11	41
		64K	4	12	45
6250	512	3.2	10	37	56
		8192	12	37	145
		64K	15	44	175

Note: 3600-foot tapes use thin tape (1.3 mm). BLKSZ indicates block size in bytes.

Making Tape Drive Links

For more information on making tape drive links, see the `ln(1)` and `mknod(1M)` reference pages.

If you suspect that the tape device has not been properly created or that the links between the low-level device name (for example, `/dev/mt/tps0d3`) and the symbolic name (for example, `/dev/tape`) are not correct, then you may want to log in as root and run the `MAKEDEV` script with the following command sequence:

```
cd /dev
rm *tape*
./MAKEDEV [links or device-type]
```

Device types can be:

- `tape links` (to re-create all default tape links)
- `tape` (for all tape devices)
- `qictape` (for the older QIC-02 tape)
- `tps` (for Kennedy SCSI 1/2" tape)
- `magtape` (for Xylogics 1/2" tape)

Normally, the `./MAKEDEV tapelinks` command is all you need to create links for the following default device names: `nrtape`, `nrtapens`, `tape`, `tapens`.

8mm and 4mm Tape Drives

This section provides useful information for administrators of 8 and 4 millimeter tape drives. The drives described in this section are the Exabyte 8mm and Sony Metal MP 120 (PG-120MP). They are available from Exabyte (303) 442-4333 and from Silicon Graphics.

Exabyte 8mm Cartridge Tape Media Specifications

Table 4-3 lists the various cartridge sizes and tape lengths available for the Exabyte 8mm tape drive.

Table 4-3 Exabyte 8mm Cartridge Tape Specifications

Cartridge Size	Tape Length	Formatted Capacity
256	15 m	291 MB
512	28 m	583 MB
1024	54 m	1166 MB
1536	80 m	1750 MB
2048	106 m	2332 MB

QIC Tape Drives

This section provides information useful to administrators of systems with 1/4-inch cartridge (QIC) tape drives.

The following terms are defined for QIC tapes:

QIC	Quarter-inch cartridge.
QIC-02	Host interface standard.
QIC-11	Recording format. Used on Sun Workstations with 60 MB drives. Cannot be read on Silicon Graphics systems.
QIC-24	Recording format: 9 tracks with a typical track width of .0135 inch. Density is 8000 bpi. Typical capacity is 60 MB with 6.6 MB per track.
QIC-120	Recording format: 15 tracks with a typical track width of .0065 inch. Density is 10,000 bpi (NRZI Recording Mode). Typical capacity is 120 MB, with approximately 8+ MB per track.
QIC-150	Recording format used on current Silicon Graphics drives. Uses 18 tracks.

Note: It is important to use actual QIC designations here, since many low-density drives can write (and read) in both QIC 24 and QIC11. Typically, none of the QIC150 drives can read QIC11.

QIC150 drives can write in both QIC150 (using DC6150 or DC600XTD; the name changed to the first), and in QIC120, if the tape is a 600A-style tape. Typically, QIC150 drives cannot write to QIC24.

Also note that the word *format* is misleading; there is no formatting on QIC tapes (some variants do require formatting, but Silicon Graphics does not support them). Format actually refers to the pattern of data blocks. Tapes have a cartridge type, and they are written in the format correct for that type. The type is determined by the hole pattern in the tape (preceding Beginning Of Tape, or BOT). Further, tapes written on QIC150 drives have a reference burst (magnetic pattern) written at the beginning of the drive.

The noise you often hear when you first try to read (on a QIC150 drive) a tape written on a QIC24 drive is the drive trying to figure out how the tape was written, by switching modes and retrying all the possibilities, if it doesn't see both a QIC150 cartridge and the reference burst. Physically, the noise you hear is the serve motor stepping the read/write head over each track.

The difference between a QIC150 and QIC120 (600A) cartridge is in the mechanical tolerances. The QIC150 has tighter tolerances. About the only visible difference is in the pinch roller (next to the rubber drive roller). The QIC150 has a guide slot milled into it, and the 600A does not.

Table 4-4 Low-Density QIC Tape Drive Compatibility

Tapes	Read	Write
LD tapes formatted in LDF	Yes	Yes
LD tapes formatted in HDF	Process not recommended.	Process not recommended.
HD tapes formatted in LDF	Yes	Yes
HD tapes formatted in LDF assuming LDF is QIC 24	Yes	Yes
HD tapes formatted in HDF	No	Yes (rewrite to LDF)

Table 4-5 High-Density QIC Tape Drive Compatibility

Tapes	Read	Write
HD tapes formatted in LDF	Yes	No
LD tapes formatted in HDF	Process not recommended.	Process not recommended.
HD Tapes formatted in LDF	Yes	Yes
HD tapes formatted in LDF assuming LDF is QIC 24	Yes	No
HD tapes formatted in HDF	Yes	Yes

Regarding read/write activity for a low-density tape formatted in high density, it is not only not recommended, it isn't even possible if the tape is a QIC24 (DC300XL or DC450XL) tape. If the tape is a QIC120 (DC600A) read/write does work correctly, and there is no reason to recommend against it.

Using dump With DAT Tapes

The *dump* command is used to back up all files in a filesystem, or files changed after a certain date, to magnetic tape or files.

If you are using the *dump* command to perform an incremental filesystem dump with a DAT tape drive, use the following recommendation to specify the capacity in kilobytes or megabytes of the DAT tape.

Reduce the 4mm tape-length parameter by 40% for the 60-meter tape, and leave as is for the 90-meter tape (2.0 gigabytes). You may want to trim an additional 5% or 10%, if you want to be conservative.

Tape Drive Cleaning Process

The following sections describe the recommended process for cleaning your tape drive.

- “1/2-Inch Reel-to-Reel Cipher Tape Drive Cleaning Process” on page 79
- “1/2-Inch Reel-to-Reel Kennedy Tape Drive Cleaning Process” on page 81
- “8mm and 4mm Tape Drive Cleaning Process” on page 82

To ensure data integrity, it is important to clean your tape drive on a regular basis. The process of removing the accumulation of oxide and/or dirt from the erase/write/read head surface and transport system is important for trouble-free operation of your equipment.

The tools you need are lint-free, nonabrasive cloths or cotton swabs, 90% or higher isopropyl alcohol or Freon-TF, and mild soapy water.

Caution: Never clean any plastic or rubber component (for example, the tape guide) in the tape path with 90% or higher isopropyl alcohol. Doing so degrades the composition of the component.

If Freon-TF has been banned from use in your company, you can use the 90% isopropyl alcohol. However, you must wipe the alcohol residue off with a swab and water.

1/2-Inch Reel-to-Reel Cipher Tape Drive Cleaning Process

A cleaning kit for the Cipher 880/890 can be purchased from Silicon Graphics. Contact your local sales representative for more information.

The components of the Cipher Tape Drive can be cleaned by placing the drive in the service access position. This is done by extending the unit fully on the mounting slides and opening the top plate casting. To assure safety in the open position, insert the cover stay pin (on the side of the top cover) in the hole in the chassis.

Caution: If the tape drive is located in a rack, be sure to extend the anti-tip legs at the base of the rack.

The drive should be cleaned after every four hours of tape movement.

Follow the cleaning method for each part listed:

Tachometer roller

Use a swab moistened with Freon-TF. Gently wipe the entire roller surface. The roller can be rotated by manually turning the take-up hub slowly.

Take-up hub

Use a swab moistened with Freon-TF. Rotate the hub manually while gently wiping the tape-wrapping surface.

Roller guides

Use a swab moistened with Freon-TF. Rotate each roller and gently wipe the tape contact surface and flanges or washers.

Reel hub pads

Use a swab moistened with Freon-TF. Wipe the contact surface of each pad and remove any debris around the pad.

Head

Use a swab moistened with Freon-TF. Wipe the entire face of the head, paying particular attention to the recessed areas.

Caution: Rough or abrasive materials can scratch sensitive surfaces of the head, resulting in permanent damage. Other cleaners, such as alcohol-based types, can cause read/write errors.

- Tape cleaner Use a swab moistened with Freon-TF. Wipe each blade along its length. Remove accumulated oxides from the recessed area between the blades.
- Front panel/door
 Use a cloth moistened with mild, soapy water.
- Top plate casting
 Use a cloth moistened with mild soapy water. Wipe away the oxide dust in the tape path area. Be careful not to get dirt on the head, rollers, and other nearby parts.
- Filter Locate and remove the filter from inside the air duct opening at the lower left of the front panel. Clean the filter with low-pressure compressed air (or vacuum) in the opposite direction of airflow, and reinstall.

1/2-Inch Reel-to-Reel Kennedy Tape Drive Cleaning Process

The drive should be cleaned after every four hours of operation.

Components of the Kennedy Tape Drive can be cleaned by placing the drive in the service position. This is done by extending the unit fully on the mounting slides and opening the dust cover. To open the dust cover, turn the two holding screws one quarter turn counterclockwise. To secure the dust cover in the open position, position the hole in the autolocking support bar onto the pin located on the side of the chassis. To expose the tape path components, lift the tape path dust cover next to the vacuum hub assembly.

Note: If the tape drive is located in a rack, be sure to extend the anti-tip legs at the base of the rack.

Follow the cleaning method for each part listed:

Erase/write/read head

Use a clean lint-free cloth or cotton swab dampened with 90% isopropyl alcohol or Freon-TF. Wipe the head with firm but gentle vertical strokes. Pay particular attention to the recessed areas of the head.

Note: Using non-recommended cleaning fluids or excessive amounts of the recommended cleaning fluid can damage the tape drive. Cleaning fluids allowed to run into the bearings break down the lubricant.

Tape path cleaning

Clean tape guides, rollers (except capstan roller), and the tape cleaners with a cotton swab dampened with 90% isopropyl alcohol. Clean the capstan roller with a cotton swab dampened with water or, if it is excessively dirty, dampen it with mild, soapy water. Dry thoroughly with a clean lint-free cloth.

Caution: Never clean any plastic or rubber component (for example, the tape guide) in the tape path with 90% or higher isopropyl alcohol. Doing so degrades the composition of the component.

Reel locking fingers

With the tape reel on the supply hub removed, clean the rubber pads on the fingers with a clean cotton swab or cloth dampened with water or, if it is excessively dirty, with mild, soapy water.

Caution: Do not lubricate the bearings.

8mm and 4mm Tape Drive Cleaning Process

This section provides you with a recommended process for cleaning your tape drive, including when the drive should be cleaned.

Tape Drives:

- 8mm Exabyte Tape Drive
- 4mm Archive DAT Tape Drive

The unit should be cleaned when enough data has been written to fill four tapes.

A cleaning kit for these tape drives can be purchased from Silicon Graphics. Contact your local sales representative for more information.

Perform the following steps:

1. Prepare the tape drive to be cleaned by powering on the unit. When the power-on cycle is complete, open the door and remove any data cartridges in the unit. Leave the door open.
2. Place the cleaning cartridge (from the cleaning kit) into the drive and close the door.

The remainder of the cleaning process is automatically performed by the tape drive. When the cleaning process is complete, the cleaning cartridge is automatically ejected from the drive. The average cleaning cycle is 15 seconds.

3. On the cleaning cartridge label, record the date the cleaning was performed, then store the cleaning cartridge for future use.

Do not store the cleaning cartridge if the cleaning dates have filled the cartridge label. Discard the cleaning cartridge and store a new cleaning cartridge for future use.

If the cleaning cartridge is ejected from the tape drive without performing a cleaning cycle (before 15 seconds), the cleaning cartridge has reached the end of its useful life and should be discarded.

Do not rewind and reuse the cleaning cartridge.

Troubleshooting

This chapter discusses some troubleshooting tips for problems arising with peripheral devices. It contains the following sections:

- “Troubleshooting Your Modem Setup”
- “Troubleshooting Your Printing System”
- “Troubleshooting Inaccessible Tape Drives”

Troubleshooting Your Modem Setup

If there are any problems with the *cu* dial-out process, you may want to use the **-d** option to *cu* to instruct the system to print diagnostic messages to your system console and the **-l** option to connect directly to the modem (if you added the “Direct” statement in the *Devices* file).

To test the modem connection on port 2, type:

```
cu -d -l ttyd2
```

- The `Connected` message should display on the console. Type **AT** and the `OK` message should display. If not, the modem is not correctly configured or there is a problem with the cable.
- If `Connected` does not display, check the debugging messages to determine where the connection failed.

Additionally, double-check that all your hardware connections are secure and that you are using a Silicon Graphics modem cable, or one made to the specifications described in “Serial Port Pin Signal Tables” on page 20.

Troubleshooting Your Printing System

If you send a print request to a printer with *lp* and do not receive any output, use the checklists below to make sure your system is ready for printing. Use these lists as a supplement to the troubleshooting information in the manufacturer's hardware manual.

Hardware Troubleshooting Checklist

Use the following list of questions to determine if your printer hardware is working as designed:

- Is the printer turned on?
Printers do not always indicate clearly if they are turned on. Make sure the printer is plugged into the power socket and the power switch is on.
- Does the printer have paper?
Frequently, printers run out of paper in a high-volume situation.
- Is there a paper jam?
Make sure the entire paper pathway is clear of sheets or fragments of paper. Refer to your printer hardware documentation before attempting to put any unusual paper or other media through your printer.
- Is the printer set to the correct baud?
Be sure the baud rate of the printer matches that of the serial port.
- Is the serial cable attached correctly?
Often, reseating the serial cable where it connects to the printer restores correct operation.
- Is the correct cable being used?
The use of the pins in serial cables varies somewhat in different applications. Cables designed for specific hardware may or may not function correctly with different hardware. Check your system *Owner's Guide* and the documentation supplied with your printer and cable to determine if the cable is correct for your hardware.

Software Troubleshooting Checklist

The *lp* scheduler is the program in charge of spooling your files to the printer, and it is invoked whenever you use the *lp* print command. The scheduler can be in a number of states, and each printer registered with *lp* can be in a number of states as well.

To check on the complete status of the *lp* system, type:

```
lpstat -t
```

This gives you a complete description of the status of *lp*. You may also want to examine the contents of the file */var/spool/lp/log*. Use the information you find to answer the following questions:

- Is your printer registered with *lp*?

If you do not see the name of your printer in the list of information produced by *lpstat*, then you must register your printer with *lp*.

- Is the printer enabled?

If your printer is not enabled, the *lpstat* listing contains this line:

```
printer yourprinter disabled since...
```

In order to enable the printer, type:

```
enable yourprinter
```

lp sometimes disables a printer automatically if it is unable to send a file to a print server, so a disabled printer often indicates a hardware problem, such as a host that is not communicating with the network.

- Is the printer accepting requests?

If the printer is not accepting requests, the *lpstat* listing contains this line:

```
yourprinter not accepting requests since...
```

You must use the *accept* command for that printer destination. Become the superuser (with *su*) and type:

```
/usr/lib/accept yourprinter
```

- Is the *lp* scheduler running?
If the scheduler is not running, the *lpstat* listing contains the message:
`scheduler is not running`
To restart the *lp* scheduler, become superuser (with *su*) and type:
`/usr/lib/lpsched`
- Did you specify the right printer?
If your system has more than one printer, and you wish to send a job to a printer other than the default, remember to use the **-d** option:
`lp -dotherprinter`

Troubleshooting Network Printers

If you are having trouble with a printer you are accessing over a network, check the status of the *lp* scheduler on your workstation the print server's host system.

Emergency Measures

If none of the above procedures work, there are several "last resort" procedures:

1. Stop the *lp* scheduler and then restart it. As root, type the following sequence of commands:
`/usr/lib/lpshut`
Then kill any jobs running as *lp*. You can identify these processes with the command:
`ps -fu lp`
Then type the command:
`/usr/lib/lpsched`
2. Remove the offending printer destination from the *lp* scheduler, and then register it again. Before you can do this you must either cancel any print requests going to the printer or move them to another print destination (if you have more than one).
3. As an absolute last resort, remove all printers from the *lp* system, reboot the computer, and register them all once again.

Troubleshooting the BSD lpr Spooling System

If your print request does not make it to the queue, then:

- Check for error messages.
- Double-check the command that you entered.
- Try submitting the */etc/group* file to the queue.

The file you submitted may not be in the proper format for the print server to print your request.

If your print request makes it to the queue and never gets to the print server, then:

- Do you have the print server system's IP address and hostname in the */etc/hosts* file?
- Does the print server system name match the name in the */etc/hosts* file? Do they match the hostname of the print server system?
- Did you get this error message? *waiting for remote queue to be enabled*.

This message usually means that your hostname is not in the print server system's */etc/hosts.equiv* file. If your print request disappears from the queue, and doesn't print or prints incorrect information, then:

1. Become root and enter the commands:

```
/usr/etc/lpc stop lp (or your printername)
lpr /etc/group
cd /usr/spool/lpd (or your spool directory)
ls -l
```

Your system should return something similar to:

```
-rw-rw---- 1 root lp 69 Aug 23 14:02 cfA117t1s
-rw-rw---- 1 root lp 227 Aug 23 14:02 dfA117t1s
-rwxr----- 1 root lp 0 Aug 23 14:01 lock
-rw-rw-r-- 1 root lp 25 Aug 23 14:46 status
```

2. Check the contents of the control file with the following command:

```
cat cfA117t1s
```

Your system should return something similar to:

```
Htls H the hostname that sent the print request
Proot P the person who sent the request
Jgroup J the jobname
Ctls C class/hostname
Lroot L the person who sent the request
fdfA117t1s f name of the file to print
UdfA117t1s U name of the file to remove after printing
N/etc/group N the original file name
```

3. Check the copy of the print file.

It is recommend that you use the *more* command just in case your test file is not as short as the */etc/group* file. The *df* file should look exactly like the file you attempted to print. In this case, the file *dfA117t1s* should be exactly the same as the */etc/group* file.

```
more dfA117t1s
```

The system should return something similar to:

```
sys::0:root,bin,sys,adm
root::0:root
daemon::1:root,daemon
bin::2:root,bin,daemon
adm::3:root,adm,daemon
mail::4:root
uucp::5:uucp
rje::8:rje,shqer
lp::9:
nuucp::10:nuucp
user::20:
other::995:
demos::*:997:
guest::*:998:
```

Now that you have verified that the request is properly spooling on the local system, check the print server system. First, you may need to contact the System Administrator of the print server system. You need the root password, and once you enter the *stop* command on that system, no print requests are printed. Instead, they remain in the queue. Make sure that there are no requests in the queue that are currently printing.

4. On the print server system, log in as root and enter the command:

```
/usr/etc/lpc stop lp
```

5. On the local system, enter the command:

```
/usr/etc/lpc start lp
```

6. On the print server system, *cd* to the spool directory.

If you do not know where the spool directory is, use the *cat* or *more* command with the */etc/printcap* file to look at what is set in the *sd:* variable.

7. On the print server system (after step 6), enter the following command:

```
ls -l
```

The print server system should return something similar to:

```
-rw-r----x 1 root 4 Aug 15 10:27 .seq
-rw-rw---- 1 root 69 Aug 23 14:02 cfA117tls.csd.sgi.com
-rw-rw---- 1 root 227 Aug 23 14:02 dfA117tls
-rwxr----- 1 root 0 Aug 23 14:01 lock
-rw-rw-r-- 1 root 25 Aug 23 14:46 status
```

8. Check the contents of the control file.

```
cat cfA117tls.csd.sgi.com
```

The print server system should return something similar to:

```
Htls H the hostname that sent the print request
Proot P the person who sent the request
Jgroup J the jobname
Ctls C class/hostname
Lroot L the person who sent the request
fdA117tls f name of the file to print
UdfA117tls U name of the file to remove after printing
N/etc/group N the original file name
```

9. Examine the *df** file by entering the following command:

```
more dfA117t1s
```

The system should return something similar to:

```
sys::0:root,bin,sys,adm
root::0:root
daemon::1:root,daemon
bin::2:root,bin,daemon
adm::3:root,adm,daemon
mail::4:root
uucp::5:uucp
rje::8:rje,shqer
lp::9:
nuucp::10:nuucp
user::20:
other::995:
demos::*:997:
guest::*:998:
```

The *df* file should look exactly like the file you attempted to print. In this case, the print server system's *dfA117t1s* file should be exactly the same as the *dfA117t1s* file that was on your system.

10. On the print server system, enter the following command:

```
/usr/etc/lpc start lp
```

Your file should now print on the printer. It should look exactly like the output of the *more* command. If it doesn't then contact the System Administrator of the print server system.

Troubleshooting Inaccessible Tape Drives

Note: This section does not allow for customized installations and does not address complex multiple tape drive issues. You should take care not to violate your maintenance agreements.

Checking the Hardware

Use the *hinvt* command to see if the operating system recognized the tape drive at boot time. This is one of the most basic and critical tests to check hardware. (An output similar to the following is returned with the *hinvt* command):

```
Iris Audio Processor: version A2 revision 4.1.0
1 100 MHZ IP22 Processor
FPU: MIPS R4010 Floating Point Chip Revision: 0.0
CPU: MIPS R4000 Processor Chip Revision: 3.0
On-board serial ports: 2
On-board bi-directional parallel port
Data cache size: 8 Kbytes
Instruction cache size: 8 Kbytes
Secondary unified instruction/data cache size: 1 Mbyte
Main memory size: 64 Mbytes
Integral Ethernet: ec0, version 1
Integral SCSI controller 0: Version WD33C93B, revision D
CDROM: unit 4 on SCSI controller 0
Disk drive: unit 1 on SCSI controller 0
Graphics board: Indy 24-bit
Vino video: unit 0, revision 0, Indycam connected
```

If *hinvt* does not report an attached tape drive, then your operating system can not use the drive. You need to check the installation of the hardware. What you can do at this time depends on what you can do with your computer given your maintenance support agreements.

Simple hardware checks are:

- If the tape drive is an external unit, does it have power? Simply powering it on does not cause it to be seen by the computer. The system must be shut down, power cycled, then rebooted.
- During the boot phase, do you see the access light on the tape drive light up at all? If it doesn't flash at all, chances are the operating system is still not seeing the drive.
- Is the SCSI cabling and termination correct? If visual inspection shows nothing obvious, try resetting the connectors. Any movement of hardware or cabling must be done with the system powered off.

If nothing that you do here causes *hinvt* to report the tape drive, then the most likely problem is faulty hardware. Contact your support provider.

Checking the Software

If you are reasonably sure the tape drive is correctly installed on the computer, but your software does not seem to be able to use it, a possible cause for the problem is that the tape device's SCSI address changed when other SCSI devices were added to your system.

The system assumes that if */dev/nrtape* exists and appears to be a tape drive of some kind, then it doesn't need to remake the default tape drive links of */dev/tape*, */dev/nrtape*, and so on. It also assumes that the first tape drive that it finds is the main tape drive. It searches for devices starting at the highest SCSI ID numbers, so the tape device on SCSI ID 7 gets the default links before a tape device on SCSI ID 3.

The default tape drive for most commands is */dev/tape*. If the tape drive installation proceeded correctly, you should have at least */dev/tape* and */dev/nrtape* special device files. You may have several others, depending on the type of tape drive.

The *mt* command can be used to confirm that */dev/tape* exists and that the tape drive is responding. Output similar to the following from the *mt status* command confirms that:

```
Controller: SCSI
Device: ARCHIVE: Python 25601-XXX2.63
Status: 0x20262
Drive type: DAT
Media : READY, writable, at BOT
```

The following output means that you have another process accessing the drive right now:

```
/dev/nrtape: Device or resource busy
```

The following output appears when a special device file does not exist:

```
/dev/nrtape: No such file or directory
```

The output when a device file exists, but no hardware is responding at that address, is:

```
/dev/nrtape: No such device
```

If the hardware appears to be present, but */dev/tape* does not appear to be valid, confirm the file links. Take the device unit number from *hinv* output:

```
Tape drive: unit 3 on SCSI controller 0: DAT
```

In this example the device unit number is 3 (this is likely to be different on your system). Use the following *ls* command to confirm that */dev/tape* is linked to the correct device (change the numeral 3 to the correct numeral for your drive):

```
ls -l /dev/tape /dev/mt/tps0d3*
crw-rw-rw- 2 root sys 23, 96 Sep 21 11:11 /dev/mt/tps0d3
crw-rw-rw- 2 root sys 23, 97 Jun 20 05:55 /dev/mt/tps0d3nr
crw-rw-rw- 2 root sys 23, 99 Jul 8 09:57 /dev/mt/tps0d3nrns
crw-rw-rw- 2 root sys 23,103 Jun 20 05:55 /dev/mt/tps0d3nrnsv
crw-rw-rw- 2 root sys 23,101 Jun 20 05:55 /dev/mt/tps0d3nrv
crw-rw-rw- 2 root sys 23, 98 Jun 20 05:55 /dev/mt/tps0d3ns
crw-rw-rw- 2 root sys 23,102 Jun 20 05:55 /dev/mt/tps0d3nsv
crw-rw-rw- 2 root sys 23,100 Jun 20 05:55 /dev/mt/tps0d3v
crw-rw-rw- 1 root sys 23,102 Jun 23 09:19 /dev/tape
```

The major and minor device numbers are the key here. They are the two numbers separated by a comma (23 and 102):

```
crw-rw-rw- 1 root sys 23,102 Jun 23 09:19 /dev/tape
```

Match these numbers with one of the lines from */dev/mt*. In this example, it should match to:

```
crw-rw-rw- 2 root sys 23,102 Jun 20 05:55 /dev/mt/tps0d3nsv
```

Compare the major and minor device numbers that are reported with */dev/tape* and the ones reported for */dev/mt/tps0dX**. Is there a match? If not, remove */dev/tape* and */dev/nrtape* and run *MAKEDEV* as root from the */dev* directory. Give the command:

```
./MAKEDEV tapelinks
```

The *MAKEDEV* command can be verbose in describing what it is doing. Your output may differ in the number of devices made and the unit number. Once the *MAKEDEV* program has completed, go through these same checks again to be sure of success.

The *MAKEDEV* command does not let you choose which tape device to link to. You must make the links by hand if the *MAKEDEV* program does not default to the drive that you wish to use.

This covers the basic problems that administrators experience regarding missing tape drives. See the following reference pages for more information on the commands used in this section: *mt(1)*, *ls(1)*, *hinvt(1M)*. For more technical information about tapes, see *mtio(7)*, *tps(7M)*, or *xmt(7)*.

Troubleshooting Tape Read Errors

Often there is a quick and simple fix for an error message that is caused by a tape drive malfunction or the tape itself. Both recoverable and unrecoverable errors can be caused by something as basic as a dirty read/write head, a poorly tensioned tape, or a dropout, which is a physically bad spot on the tape. An EOT message can also mean that there is no data on the tape.

The following information covers some of the basic tape maintenance/performance functions that should be considered as factors that could either prevent future error conditions from occurring or as aids in recovering from an existing error message:

- Be sure your read/write head is clean.
- Use the *hinvt* command to determine which tape drive type is connected to your system.
- Use the *mt stat* command to verify the status of the tape drive and the media.
- Use the *mt ret* command before read or write operations.

Error Messages

This appendix provides a listing of error messages, categorized by device, that may display during setup or while operating a device.

lp Error Messages

This section provides a description of the error messages that are associated with *lp* commands. The following variables are used in the error messages:

<i>file(s)</i>	Indicates the file or files that are to be printed.
<i>dest</i>	Indicates the name of the destination printer.
<i>printer-id</i>	Indicates the request identification number of the printout. For example, <i>myprinter-46</i> is the printer name followed by the request identification number.
<i>printer-name</i>	Indicates the name of the printer.
<i>program-name</i>	Indicates the program name that was executed.
<i>user</i>	Indicates the user who requested the printout.

These messages can be found in the printer log files if you miss them on the system console. Following each message is an explanation of the probable cause of the error and the corrective action to take. If you cannot correct all the error conditions you encounter, call your service representative for assistance.

- *dest* is an illegal destination name
The *dest* you used is not a valid destination name. Use the *lpstat -p* command to list valid destination names.
- *file* is a directory
The filename you typed is a directory and cannot be printed.
- *xx* is not a request id or a printer
The argument you used with the *cancel* command is not a valid request identification number or a printer name. Use the *lpstat -t* command to view a list of all the printers and requests waiting to get printed.
- *xx* is not a request
The request identification number you used with the *lpmove* command is not a valid request identification number. To find out what requests are valid, use the *lpstat -u* command.
- *xx* not a request id or a destination
You used an invalid request identification number or destination with the *lpstat* command. To find out what is valid, use the *lpstat -t* command.
- *dest* not accepting requests since *date*
Requests to the printer that you are trying to use have been stopped by the *reject* command.
- Can't access FIFO
The named pipe file */var/spool/lp/FIFO* is incorrect. The mode should be 600 with the owner *lp* and the group *lp*.
- *lp* Administrator not in password file
You must have an entry in the */etc/passwd* file for *lp*, and you must belong to the group *lp*.
- destination *printer-name* unknown
Use the *accept* command to enable the printer so that it can accept requests.

- can't access file xx
The mode could be wrong on your directory or the file that you are trying to access.
- can't create class xx—existing printer name
The class name you are trying to use has already been given to a printer. You need to use another name or remove the printer to use the class name.
- can't create new acceptance status file
The mode may be wrong on the */var/spool/lp* directory. It should be 755 with the owner lp and the group lp.
- can't create new class file
The mode may be wrong on the */var/spool/lp* directory. It should be 755 with the owner lp and the group lp.
- can't create new interface program
The mode may be wrong on the */var/spool/lp/interface* directory. It should be 755 with the owner lp and the group lp.
- can't create new member file
The mode may be wrong on the */var/spool/lp/member* directory. It should be 755 with the owner lp and the group lp.
- can't create new printer status file
The mode may be wrong on the */var/spool/lp/pstatus* file. It should be 644 with the owner lp and the group lp.
- can't create new request directory
The mode may be wrong on the */var/spool/lp/request* directory. It should be 755 with the owner lp and the group lp.
- can't create *printer-name*—existing class name
The printer name you are trying to use has already been used as a class name. You need to assign another name for the printer.
- can't create new output queue
The mode on the file */var/spool/lp/seqfile* is incorrect. It should be 644, and the mode on the directory should be 755. The owner and the group should be lp.

- can't create new sequence number file
The mode on the file */var/spool/lp/seqfile* is incorrect. It should be 644, and the mode of the directory should be 755. The owner and the group should be lp.
- can't create request file xx
The mode on the file */var/spool/lp/request/<printer-name>/<id>* is incorrect. *Printer-name* is the name of the printer such as dqp10, and *id* is the request identification number. The mode of the file should be 444, and the mode of the directory should be 755. The owner and the group should be lp.
- can't fork
You either have several processes running and are not allowed to run any more, or the system has all the processes running that it can handle. You must rerun this command later.
- can't lock acceptance status
The file */var/spool/lp/QSTATLOCK* prevents more than one *lp* request from being taken at any one time. You must rerun this command later.
- can't lock output queue
The file */var/spool/lp/QSTATLOCK* prevents more than one *lp* request from being printed on a printer at one time. You must rerun this command later.
- can't lock printer status
The temporary file */var/spool/lp/PSTATLOCK* prevents more than one *lp* request from being printed on a printer at one time. You must rerun this command later.
- can't lock sequence number file
The file */var/spool/lp/SEQLOCK* prevents more than one *lp* request from getting the next *printer-id* (request identification) number at one time. You must rerun this command later.
- can't move request *printer-id*
Printer-id is the request identification number that cannot be moved. You will probably have to change the modes on the files and directories in */var/spool/lp/request*. Also, after you shut down the *lp* scheduler, you must manually move the request from the disabled printer directory to the new destination.

- can't open class file

The *lp* program is trying to access the list of classes for printers. One reason it may not be able to open the class file is that the system might have the maximum number of files open that are allowed at any time. You can correct this by typing the command at a later time.

- can't open member file

The *lp* program is trying to access the list of members in the directory */var/spool/lp/member*. The system could have the maximum number of files open that are allowed at any time. You can correct this by typing the command at a later time.

- can't open *xx* file in MEMBER directory

There are a number of reasons why file *xx* in the */var/spool/lp/member* directory cannot be opened. The mode on the file could be incorrect; it should be 644. The system could have the maximum number of files open that are allowed at any time; you can correct this by typing the command at a later time.

- can't open *xx* file in class directory

If file *xx* cannot be opened, it's possible that the mode on the file or directory is incorrect. The file mode should be 644, and the directory mode should be 755. Another possibility is that the system has the maximum number of files open that are allowed at any time. The latter problem can be corrected by typing the command at a later time.

- can't open *xx*

You cannot print on printer *xx* because the mode is incorrect on the */dev/tty* file. The mode should be 622.

- can't open FIFO

The mode on the named pipe file */var/spool/lp/FIFO* may be incorrect. It should be 600. Or, the system could have the maximum number of files open that are allowed at any time. You can correct the latter problem by typing the command at a later time.

- can't open MEMBER directory

The mode on the directory */var/spool/lp/member* could be incorrect. It should be 755. Another possibility is that the system could have the maximum number of files open that are allowed at any time. If this is the case, try typing the command at a later time.

- can't open acceptance status file
The mode on the file `/var/spool/lp/qstatus` may not be correct; it should be 644. Another possibility is that the system could have the maximum number of files open that are allowed at any time. You can correct the latter problem by typing the command at a later time.
- can't open default destination file
Check the mode on the file `/var/spool/lp/default`; it should be 644. If the mode is correct, it could be that the system has the maximum number of files open that are allowed at any one time. You can correct this by trying the command at a later time.
- can't open file *file*
You incorrectly typed the filename or you don't have the correct modes set. If you are the owner, the mode should be at least 400.
- can't open output queue
Check the mode on the file `/var/spool/lp/outputq`; it should be 644. This error message could also be generated if the system has the maximum number of files open that are allowed at any one time. Try entering the command at a later time.
- can't open printer status file
The mode on the file `/var/spool/lp/pstatus` is incorrect; it should be 644. This message is also generated if the system has the maximum number of files open that are allowed at any one time. You can correct this by trying the command at a later time.
- can't open request directory
The mode on the directory `/var/spool/lp/request` is incorrect; it should be 655. The system may also have the maximum number of files open that are allowed at any one time. You can correct this by trying the command at a later time.
- can't open request file *xx*
The mode on the file `/var/spool/lp/member/request/<xx>` is incorrect. The mode should be 644. The system may also have the maximum number of files open that are allowed at any one time. You can correct this by trying the `lpmove` command at a later time.
- can't open system default destination file
The mode on the file `/var/spool/lp/default` is incorrect. The mode should be 644. The system may also have the maximum number of files open that are allowed at any one time. You can correct this by trying the command again at a later time.

- can't open temporary output queue
The mode on the file `/var/spool/lp/outputq` is incorrect. The mode should be 644. The system may also have the maximum number of files open that are allowed at any one time. You can correct this by trying the command at a later time.
- can't proceed—scheduler running
Many of the `lpadmin` command options cannot be executed while the scheduler is running. Stop the scheduler using the `lpshut` command and then try invoking the command again.
- can't read current directory
The `lp` and `lpadmin` commands cannot read the directory containing the file to be printed. The directory name may be incorrect or you do not have read permission on that directory.
- can't remove class file
The mode may be wrong on the file `/var/spool/lp/class`. It should be 755. The owner and the group should be `lp`. The file in that directory may also have the wrong mode; it should be 644.
- can't remove printer
The mode may be wrong on the `/var/spool/lp/member` directory. It should be 755, and the files in that directory should be 644. Both the directory and the files should be owned by `lp` and the group should be `lp`.
- can't remove request directory
The mode may be wrong on the `/var/spool/lp/request` directory. It should be 755 and should be owned by `lp`, and the group should be `lp`. The directory may still have pending requests to be printed, which must be removed before the directory can be removed.
- can't set user id to lp Administrator's user id
The `lpsched` and `lpadmin` commands can be used only when you are logged in as `lp` or `root`.
- can't unlink old output queue
The `lpsched` program cannot remove the old output queue. You must remove it manually by using the command:

```
rm /var/spool/lp/outputq
```

- can't write to *xx*
The *lpadmin* command cannot write to device *xx*. The mode is probably wrong on the */dev/ttyxx* or */dev/plp* file. It should be 622 and owned by lp.
- cannot create temp file filename
The system may be out of free space on the */var* filesystem. Use the command:
df /var
to determine the number of free blocks. Several hundred blocks are required to ensure that the system performs correctly.
- class *xx* has disappeared!
Class *xx* was probably removed after the scheduler was started. The system may be out of free space on the */var* filesystem. To find out, use the following command:
df /var
Use the *lpshut* command to stop the scheduler and restore the class from a backup.
- class *xx* non-existent
The class *xx* may have been removed because the system is out of free space on the */var* filesystem. To find out how much free space is available, use the following command:
df /var
The class will probably have to be restored from a backup.
- class directory has disappeared!
The */var/spool/lp/class* directory has been removed. The system may be out of free space on */var*; use the **df /var** command to find out. The class directory contains all the data for each printer class. To restore this directory, get these files and directory from a backup.
- corrupted member file
The */var/spool/lp/member* directory has a corrupted file in it. You should restore the directory from backup.
- default destination *dest* non-existent
Either the default destination is not assigned or the printer *dest* has been removed. Use the *lpadmin* command to set up a default destination or set your LPDEST environment variable to the value of the destination.

- destination *dest* has disappeared!
A destination printer, *dest*, has been removed after *lpsched* was started. Use the *lpadmin* command to remove the printer.
- destination printer no longer accepting requests
The printer has been disabled using the *reject* command. Use the *accept* command to re-enable the printer.
- destination *dest* non-existent
The destination printer you specified as an argument to the *accept* or *lpadmin* command is not a valid destination name, or it has been removed after the scheduler was started.
- destination printer was already accepting requests
The destination printer was previously enabled. Once a printer is accepting requests, any further *accept* commands are ignored.
- destination printer already not accepting requests
A *reject* command was already sent to the printer. Use the *accept* command to allow the printer to start accepting requests again.
- destination *printer-name* is not accepting requests – move in progress ...
The printer has been disabled by the *reject* command, and requests are being moved from the disabled printer to another printer. The printer can be enabled again by the *accept* command.
- destinations are identical
When using the *lpmove* command, you need to specify a printer to move the print requests from and a different printer to move the requests to.
- disabled by scheduler: login terminal
The login terminal has been disabled by the *lp* scheduler. Use the *enable* command to re-enable the printer.
- error in printer request *printer-id*
Printer-id is the actual request identification number. An error has likely occurred in the printer. Check the printer, and reset it if needed.
- illegal keyletter *xx*
An invalid option, *xx*, was used. See the reference page for the correct options.

- keyletters *-xx* and *-yy* are contradictory
This combination of options to the *lpadmin* program cannot be used together.
- keyletter *xx* requires a value
The option *xx* requires an argument. For example, in the command line:

```
lpadmin -m model
```


the argument to the *-m* option is the name of a model interface program.
- keyletters *-e*, *-i*, and *-m* are mutually exclusive
These options to the *lpadmin* command cannot be used together. Refer to the reference page for more information on usage.
- lp: *xx*
In this message the variable *xx* could be one of several arguments. Typically, it is a message telling you the default destination is not assigned.
- member directory has disappeared!
The */var/spool/lp/member* directory has been removed. The system is probably out of free disk space in the */var* filesystem. You need to clean up the */var* filesystem, and then install the *lp* commands or retrieve them from a backup.
- model *xx* non-existent
The name that you are using for a model interface program is not valid. A list of valid models is in the */var/spool/lp/model* directory.
- new printers require *-v* and either *-e*, *-i*, or *-m*
A printer must have an interface program, and this is specified by *-e*, *-i*, or *-m* options. The *-v* option specifies the device file for the printer. For more information on these options, refer to the *lpadmin* reference page.
- no destinations specified
There are no destination printers specified. Use the *lpadmin* command to set one up.
- no printers specified
There are no printers specified. Use the *lpadmin* command to set one up.
- non-existent printer *xx* in PSTATUS
A printer with the name *xx* is in the */var/spool/lp/pstatus* file but no longer exists. Use the *lpadmin* command to remove the printer.

- non-existent printer *printer-name* in class *xx*
The printer that you are trying to address in class *xx* has been removed from that class.
- out of memory
The message implies that there is not enough memory to contain the text to be printed.
- printer *printer-name* already in class *xx*
The printer you are trying to move to class *xx* is already in that class. You cannot move a printer to a class that it is already in.
- printer *printer-name* has disappeared!
The printer has been removed, and the *enable* command cannot find it. The printer was most likely removed after the workstation was rebooted or after the scheduler was started.
- printer *printer-name* non-existent
Printer-name is the name of a printer that has been removed after the scheduler has been started. You must use the command:
lpadmin -xprinter-name
- printer status entry for printer has disappeared
The */var/spool/lp/pstatus* file must have been corrupted. You need to resubmit the printer request.
- printer *printer-name* was not busy
The printer is not printing a request at this time. Either the request you wanted to cancel is finished printing or you have specified the wrong printer.
- request *printer-id* non-existent
You are attempting to cancel a request that does not exist. You may have given the wrong printer name or wrong request ID number or the request may have finished printing.
- request not accepted
The request was not accepted by *lp*. The scheduler may not be running. Use the *lpstat -t* command to find out more information.

- requests still queued for *printer-name*—use `lpmove`
Printer-name is the printer that still has requests waiting to be printed. Use the `lpmove` command to move those requests to another printer.
- scheduler is still running—can't proceed
You cannot perform this command while the scheduler is running. You need to use the `lpshut` command first.
- spool directory non-existent
The directory `/var/spool` has been removed. You need to use the `mkdir` command to restore the directory. This has probably removed some of the necessary `lp` files. You may have to reinstall the `lp` commands.
- standard input is empty
You specified an invalid filename either by incorrectly typing a name or by specifying a nonexistent file. Nothing will be printed on the printers from this request.
- this command for use only by lp Administrators
This command is restricted to someone logged in as root or lp.
- too many options for interface program
The `lp` command called the appropriate interface program with too many arguments. For more information on the options and arguments that can be used with the `lp` command, refer to the `lp(1)` reference page.

Tape Drive Error Indications

Following are some examples of commands and error messages. This is not an exhaustive list.

- **tar tvf /dev/nrtape**
tar: /dev/nrtape: No such device
- **cpio -itvI /dev/nrtape**
cpio: ERROR: Cannot open </dev/nrtape> for input. No such device
- **tar t**
tar: archive file /dev/tape does not exist or is a regular file
- **/usr/etc/restore t**
/dev/tape: No such file or directory

Peripheral Device Files

This appendix contains a list of peripheral device files and directories that reside in the */dev* directory. For a complete listing of IRIX device files and directories, refer to IRIX Admin: System Configuration and Operation.

<i>mt/</i>	Directory containing block device files for tapes; see <i>tps(7)</i> for SCSI 1/4-inch tape drive device names; see <i>xmt(7)</i> for Xylogics 1/2-inch tape drive names.
<i>rmt/</i>	Directory containing raw device files for tapes; see <i>tps(7)</i> for SCSI 1/4-inch tape drive device names; see <i>xmt(7)</i> for Xylogics 1/2-inch tape drive names.
<i>tape</i>	Generic tape device; see <i>mtio(7)</i> .
<i>nrtape</i>	Generic no-rewind tape device; see <i>mtio(7)</i> .
<i>tapens</i>	Generic tape device; bytes are not swapped; see <i>mtio(7)</i> .
<i>nrtapens</i>	Generic no-rewind tape device; bytes are not swapped; see <i>mtio(7)</i> .
<i>cent</i>	Parallel printer device.
<i>tek</i>	Tektronics color graphics printer device.
<i>vers</i>	Versatek color graphics printer device.
<i>plp</i>	Parallel line printer interface; see <i>plp(7)</i> .
<i>ptc</i>	Clonable pseudo-tty controller; see <i>clone(7)</i> , <i>ptc(7)</i> .
<i>grconc</i>	Master pseudo-teletype for the graphics console; see <i>pty(7)</i> .
<i>grcons</i>	Slave pseudo-teletype for the graphics console; see <i>pty(7)</i> .
<i>gm</i>	Logical console device for the Graphics Manager on the IRIS GT and GTX model systems. Messages from the software running on the 68020 on the GM board appear as output on this device.

<i>grin/</i>	Directory containing the individual logical graphics input devices.
<i>console</i>	System console device.
<i>syscon</i>	Hard link to <i>/dev/console</i> .
<i>systty</i>	Hard link to <i>/dev/console</i> .
<i>queue</i>	Graphics queue device. Graphics programs call "select" on this device in order to be notified when there is input in their graphics queue. This device can't be actually read or written.
<i>dials</i>	Device for serial port connected to dial and button box.
<i>keybd</i>	Device for serial port connected to keyboard.
<i>mouse</i>	Device for serial port connected to mouse.
<i>tablet</i>	Device for serial port connected to digitizing tablet.
<i>ttyd[1-12]</i>	Serial ports 1–12.
<i>ttyf[1-12]</i>	Serial ports 1–12 for devices that understand hardware flow control.
<i>ttym[1-12]</i>	Serial ports 1–12 for modems.
<i>ttyq*</i>	Pseudo tty devices; see <i>pty(7)</i> .

Index

A

administration, system
documentation, xi-xii

C

cables, null modem, 21, 22
cabling
serial interface, 21
serial ports, 22
CD-ROM filesystems, 61
compact disk filesystems, 61
compact disk support, 61
configuring an ASCII terminal, 4
connecting
a modem, 9
ASCII terminal, 4
peripheral devices, 20

D

default printer, changing, 41
device files, 21, 109
dial and button box, 1
disable printer, 35

E

error messages
lp, 95
tape drive, 107

F

floppy disk filesystems, 61
floppy disk support, 61

I

infocmp command, 5
IRIX administration
documentation, xi-xii

L

lp
commands, 32
maintenance, 41
lp error messages, 95
lpmove, 39
lpsched, 37
lp spooler, 46
lp troubleshooting, 84

M

MAKEDEV, tape options, 71
mediad(1M), 61
modems
 cables, 22
 connecting, 9

N

network printing, 47

P

printers
 adding, 41
 commands, 32
 disabling, 35
 enabling, 36
 removing, 46
 status, 47
printing
 canceling, 34
 over the network, 47
print requests
 allowing, 38
 checking remote, 47, 48
 rejecting, 38

R

removing printers, 46

S

scheduler
 starting, 37
 stopping, 37
sending files, 33
serial interface, 21
serial ports, 20
spaceball, 1
spooler, *lp*, 46
stop printing, 34
system administration
 documentation, xi-xii

T

tape
 adding a drive, 63, 70
 capacities, 71
 supported types, 71
 using *MAKEDEV*, 71
troubleshooting
 hardware, 84
 lp, 84
 network printers, 86
 software, 85
 the print system, 84
tty file, 6

W

workstation, serial support, 20

Tell Us About This Manual

As a user of Silicon Graphics products, you can help us to better understand your needs and to improve the quality of our documentation.

Any information that you provide will be useful. Here is a list of suggested topics:

- General impression of the document
- Omission of material that you expected to find
- Technical errors
- Relevance of the material to the job you had to do
- Quality of the printing and binding

Please send the title and part number of the document with your comments. The part number for this document is 007-2861-001.

Thank you!

Three Ways to Reach Us

- To send your comments by **electronic mail**, use either of these addresses:
 - On the Internet: techpubs@sgi.com
 - For UUCP mail (through any backbone site): *[your_site]!sgi!techpubs*
- To **fax** your comments (or annotated copies of manual pages), use this fax number: 650-932-0801
- To send your comments by **traditional mail**, use this address:

Technical Publications
Silicon Graphics, Inc.
2011 North Shoreline Boulevard, M/S 535
Mountain View, California 94043-1389

