Chapter 2

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 IRIS 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.

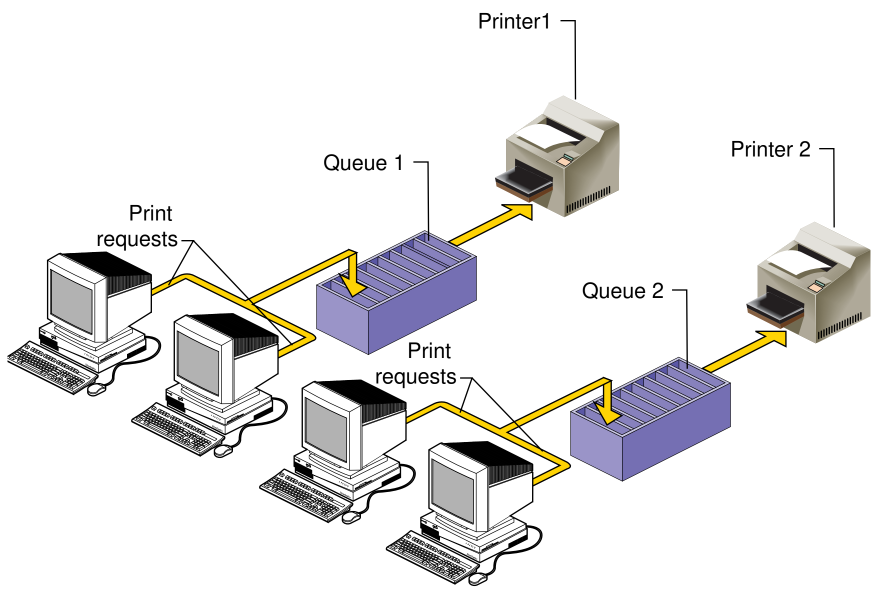

Figure 2-1 Print Spooling

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.

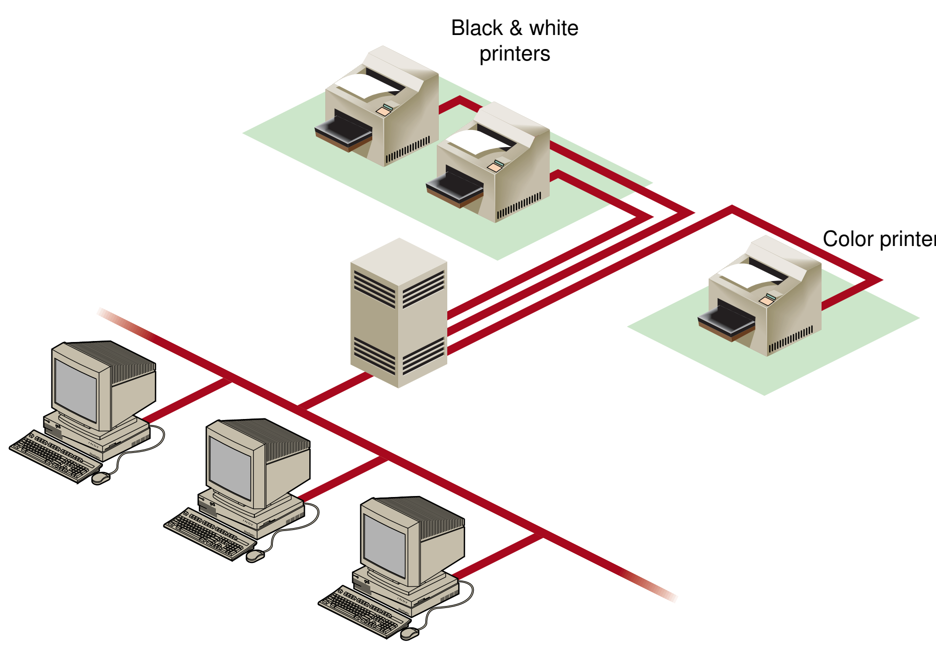

Figure 2-2 Printer Classes

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.
- Queueing 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.

lp Routes jobs to a destination and places them in a queue. The destination may be either a single printer or a class of printers.

cancel Cancels spooled print requests.

disable Prevents a printer from printing jobs that are in the queue.

enable Allows a printer to print jobs in the queue.

lpstat Reports the status of the *lp* 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 –n*number*

  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 –d*destination 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.

lpsched	Starts the *lp* scheduler
lpshut	Stops the *lp* scheduler
reject	Prevents jobs from queueing at a particular destination
accept	Permits job requests to queue at a particular destination
lpmove	Moves job requests from one destination to another
lpadmin	Configures the *lp* 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/plp* 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 -d`*printer-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.

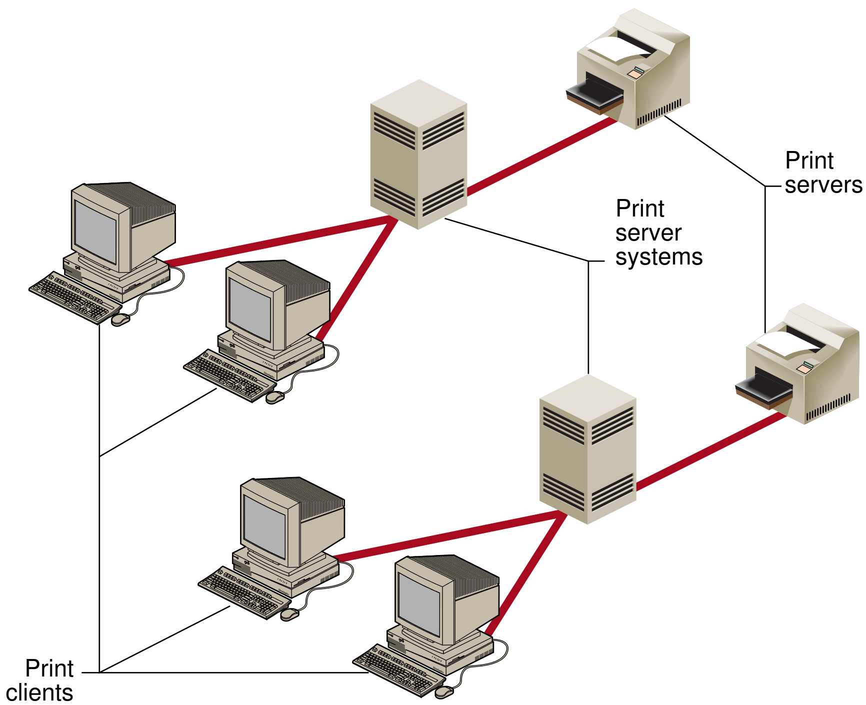

Figure 2-3 Print Servers, Print Clients, and Print Server Systems

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 -d`*printer-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.

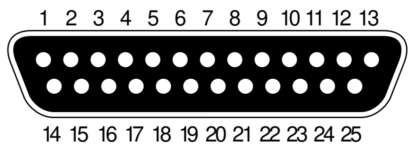

Figure 2-4 Parallel Port Pin Assignment

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.

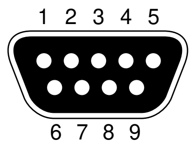

Figure 2-5 DB-9 Pin Assignment

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/ttym** 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/ttyf** devices. This cable also supports */dev/ttym** devices. Table 2-3 lists the correct pinout for these devices, as shown in Figure 2-6.

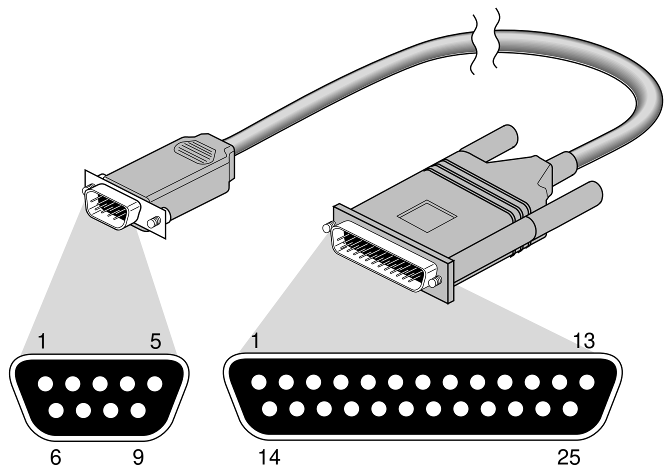

Figure 2-6 DB-9 to DB-25 RTS/CTS Flow Control Cable

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.

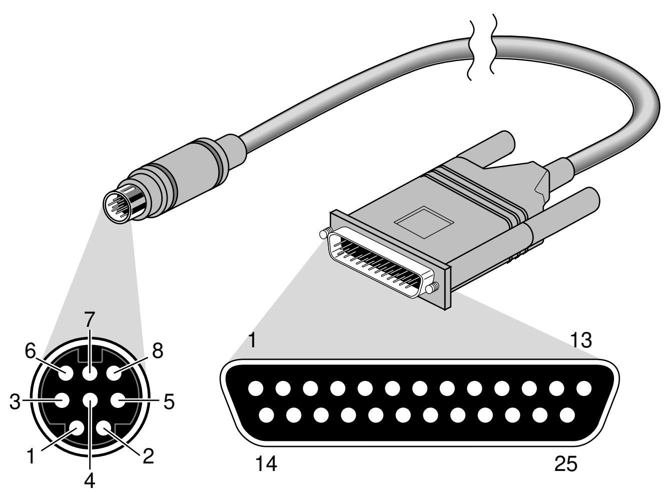

Figure 2-7 Mini-DIN8 to DB-25 Serial Cable

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/ttym** 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.