System Administration Guide:
Advanced Administration

Adobe PostScript™

011025@2471

Contents

Preface

System Administration Guide: Advanced Administration is part of a set that covers a significant part of the Solaris™ system administration information. It includes information for both SPARC™ based and IA based systems.

This book assumes that you have already installed the SunOS™ 5.9 operating system, and you have set up any networking software that you plan to use. The SunOS 5.9 operating system is part of the Solaris 9 product family, which also includes many features, including the Solaris Common Desktop Environment (CDE). The SunOS 5.9 operating system is compliant with AT&T's System V, Release 4 operating system.

For the Solaris 9 release, new features interesting to system administrators are covered in sections called *What's New in ... ?* in the appropriate chapters.

Note – The Solaris operating environment runs on two types of hardware, or platforms—SPARC and IA. The Solaris operating environment runs on both 64–bit and 32–bit address spaces. The information in this document pertains to both platforms and address spaces unless called out in a special chapter, section, note, bullet, figure, table, example, or code example.

Who Should Use This Book

This book is intended for anyone responsible for administering one or more systems running the Solaris 9 release. To use this book, you should have 1-2 years of UNIX® system administration experience. Attending UNIX system administration training courses might be helpful.

How the System Administration Volumes Are Organized

Here is a list of the topics covered by the System Administration Guides.

System Administration Guide: Basic Administration

- "Solaris Management Tools (Roadmap)" in *System Administration Guide: Basic Administration*
- "Managing Users and Groups Topics" in *System Administration Guide: Basic Administration*
- "Managing Server and Client Support Topics" in *System Administration Guide: Basic Administration*
- "Shutting Down and Booting a System Topics" in *System Administration Guide: Basic Administration*
- "Managing Removable Media Topics" in *System Administration Guide: Basic Administration*
- "Managing Software Topics" in *System Administration Guide: Basic Administration*
- "Managing Devices Topics" in *System Administration Guide: Basic Administration*
- "Managing Disks Topics" in *System Administration Guide: Basic Administration*
- "Managing File Systems Topics" in *System Administration Guide: Basic Administration*
- "Backing Up and Restoring Data Topics" in *System Administration Guide: Basic Administration*

System Administration Guide: Advanced Administration

- Chapter 1
- Chapter 9
- Chapter 13
- Chapter 21
- Chapter 25

Ordering Sun Documents

Fatbrain.com, an Internet professional bookstore, stocks select product documentation from Sun Microsystems, Inc.

For a list of documents and how to order them, visit the Sun Documentation Center on Fatbrain.com at http://www1.fatbrain.com/documentation/sun.

Accessing Sun Documentation Online

The docs.sun.com^{SM} Web site enables you to access Sun technical documentation online. You can browse the docs.sun.com archive or search for a specific book title or subject. The URL is `http://docs.sun.com`.

What Typographic Conventions Mean

The following table describes the typographic conventions used in this book.

TABLE P–1 Typographic Conventions

Typeface or Symbol	Meaning	Example
AaBbCc123	The names of commands, files, and directories; on-screen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`
AaBbCc123	What you type, contrasted with on-screen computer output	`machine_name% `**`su`** `Password:`
AaBbCc123	Command-line placeholder: replace with a real name or value	To delete a file, type **rm** *filename*.

TABLE P–1 Typographic Conventions   *(Continued)*

Typeface or Symbol	Meaning	Example
AaBbCc123	Book titles, new words or terms, or words to be emphasized.	Read Chapter 6 in *User's Guide*. These are called *class* options. Do *not* save changes yet.

Shell Prompts in Command Examples

The following table shows the default system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P–2 Shell Prompts

Shell	Prompt
C shell prompt	`machine_name%`
C shell superuser prompt	`machine_name#`
Bourne shell and Korn shell prompt	`$`
Bourne shell and Korn shell superuser prompt	`#`

General Conventions

Be aware of the following conventions used in this book.

- When following steps or using examples, be sure to type double-quotes ("), left single-quotes (`), and right single-quotes (') exactly as shown.

- The key referred to as Return is labeled Enter on some keyboards.

- It is assumed that the root path includes the `/sbin`, `/usr/sbin`, `/usr/bin`, and `/etc` directories, so the steps in this book show the commands in these directories without absolute path names. Steps that use commands in other, less common, directories show the absolute path in the example.

- The examples in this book are for a basic SunOS 5.9 software installation without the Binary Compatibility Package installed and without `/usr/ucb` in the path.

Caution – If `/usr/ucb` is included in a search path, it should always be at the end of the search path. Commands like `ps` or `df` are duplicated in `/usr/ucb` with different formats and options from the SunOS 5.9 commands.

Managing Printing Services Topics

This section provides instructions for managing printing services in the Solaris environment. This section contains these chapters.

Chapter 2	Provides overview information for managing printing services on a network. This chapter provides information on print servers, print clients, and the LP print service.
Chapter 3	Provides overview information for planning printing services on a network, which includes information on allocating system resources and defining printers on a network.
Chapter 4	Provides step-by-step instructions for setting up a printer on a system and making it available to other systems on the network.
Chapter 5	Provides step-by-step instructions for administering printers, such as deleting printers, setting print policies, and managing print requests.
Chapter 6	Provides step-by-step instructions for setting up and maintaining character sets, print filters, forms, and fonts.
Chapter 7	Provides step-by-step instructions for customizing the LP print service, such as adjusting printer port characteristics or adding a `terminfo` entry for a unsupported printer.
Chapter 8	Provides background information on the LP print service.

Managing Printing Services (Overview)

This chapter provides information about managing printers in the Solaris environment. This is a list of the overview information in this chapter.

- "What's New in Printing?" on page 27
- "Printing in the Solaris Operating Environment" on page 29
- "The LP Print Service" on page 32
- "The Solaris Print Client-Server Process" on page 34

For step-by-step instructions on print management tasks, see:

- Chapter 4
- Chapter 5
- Chapter 6
- Chapter 7

What's New in Printing?

This section describes new printing features in the Solaris 9 release.

USB Printer Support

You can use Solaris Print Manager to set up a USB printer that is attached to a SPARC or IA system with USB ports.

The new logical device names for USB printers are:

`/dev/printers/[0...N]*`

Therefore, when you add a USB printer to a printer server, select one of these devices for a USB printer under Printer Port on the Add New Attached Printer screen. See Chapter 4 for more information on using Solaris Print Manager to set up printers.

Although the new Solaris USB printer driver supports all USB printer-class compliant printers, a list of recommended PostScript™ printers is in the `usbprn(7D)` man page.

The `usbprn` driver is compliant with non-PostScript printers that utilize third-party PostScript conversion packages like GhostScript. You can obtain conversion packages from the Solaris Software Companion CD, available at http://www.sun.com/software/solaris/binaries/package.html.

Refer to the Notes and Diagnostics sections of the `usbprn(7D)` man page for information and cautions about hot-plugging USB printers.

Managing Printer Information With LDAP

You can manage printer information in the following name services in this Solaris release:

- LDAP
- NIS
- NIS+
- NIS+ with Federated Naming Service (xfn)
- files

Keep the following in mind when managing printer information in the LDAP name service:

- If the LDAP server is the Netscape™ Directory Server (NSDS), the default distinguished name is `cn=Directory Manager`. If the LDAP server is Sun™ Directory Server, the distinguished name might be something like `cn=admin, dc=xyz, dc=com`. Solaris Print Manager uses `ldapclient` to determine the default LDAP server name. If there is more than one server specified, the first one is automatically selected. See `ldapclient(1M)` for more information.

- Solaris Print Manager always displays printer entries from the current LDAP server. If this is not the domain's LDAP master server, the list of printers displayed may not be the current list of printers. This is because the LDAP replica server might not have been updated by the master server, and so, is out of sync with the master. Replica servers can have various update replication agreements. For example, when there is a change on the master, the replica servers can be updated immediately or updated once a day.

- If the selected LDAP server is a LDAP replica server, any updates are referred to the master server and done there. This again means the printer list could be out of sync with the master. For example, a deleted printer might still appear in the displayed printer list until the replica is updated from the master.

- Users can use the LDAP command line utilities, `ldapadd` and `ldapmodify`, to update printer entries in the directory, but this is not recommended. If these utilities are used, the user must ensure that the *printer-name* attribute value is unique within the `ou=printers` container. If it is not unique, the result of modifications done by Solaris Print Manager, or `lpset`(1M), might not be predictable.

Printing in the Solaris Operating Environment

The Solaris printing software provides an environment for setting up and managing client access to printers on a network.

The Solaris printing software contains these components:

- Solaris Print Manager, a graphic user interface, provides the ability to manage printing configuration on a local system or in a name service.
- The LP print service commands, a command line interface used to set up and manage printers on a local system or in a name service. They also provide functionality above and beyond the other print management tools.

Even if you do use Solaris Print Manager to set up printing, you will have to use some of the LP commands to completely manage printing in the Solaris environment. See Chapter 5 for more information.

The Solaris print software limitations include:

- No support for print servers defined as `s5` (the System V print protocol) in previous Solaris releases.
- No print filtering on print clients.

Solaris Print Manager

Solaris Print Manager is a Java-based graphical user interface that enables you to manage local and remote printer configuration. This tool can be used in the following name service environments: NIS, NIS+, NIS+ with Federated Naming Service (xfn), and files. You must be logged in as superuser to use this tool.

Using Solaris Printer Manager is the preferred method for managing printer configuration information. It is preferred over Admintool: Printers because it centralizes printer information when used in conjunction with a name service. Using a

name service for storing printer configuration information is desirable because it makes printer information available to all systems on the network, making printing administration easier.

Solaris Print Manager recognizes existing printer information on the printer servers, print clients, and in the name service databases. There are no conversion tasks required to use Solaris Print Manager as long as the print clients are running either the Solaris 2.6, Solaris 7, Solaris 8, or Solaris 9 releases.

The Solaris Print Manager package is SUNWppm.

Go to Chapter 4 for step-by-step instructions on setting up new printers with Solaris Print Manager.

Printing Support in the Name Service Switch

The printers database in /etc/nsswitch.conf, the name service switch file, provides centralized printer configuration information to print clients on the network.

By including the printers database and corresponding sources of information in the name service switch file, print clients automatically have access to printer configuration information without having to add it to their own systems.

The default printers entry in the /etc/nsswitch.conf file for files, LDAP, NIS, and NIS+ environments are described in the following table. The nisplus keyword represents the printers.org_dir table. The xfn keyword represents the FNS printer contexts.

If Your Name Service Is ...	The Default printers Entry Is ...
files	printers: user files
ldap	printers: user files ldap
nis	printers: user files nis
nis+	printers: user nisplus files xfn

For example, if your name service is NIS, printer configuration information on print clients is looked up in the following sources in this order:

- user - Represents the user's $HOME/.printers file
- files - Represents the /etc/printers.conf file
- nis - Represents the printers.conf.byname table

See nsswitch.conf(4) and *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)* for more information.

Choosing a Method to Manage Printers

Adding printer information to a name service makes access to printers available to all systems on the network and generally makes printer administration easier because all the information about printers is centralized.

If You ...	To Centralize Printer Information, Then ...
Use a name service	Adding the printer to the LDAP, NIS, NIS+, or NIS+ (xfn) database makes the printer available to all systems on the network.
Don't use a name service	Adding the printer adds the printer information to the printer server's configuration files only. Print clients will not know about the printer automatically.
	You will have to add the printer information to every print client that needs to use the printer.

The following table describes the major printer-related tasks and the tools available to perform the printing tasks.

TABLE 2–1 Solaris Printing Component Features

Component	Availability	Graphical User Interface?	Configures Network Printers?	Manages Print Clients and Servers?	Uses LDAP, NIS, NIS+, or NIS+ (xfn)?
Solaris Print Manager	Solaris 9 and compatible versions and Solaris Easy Access Server 3.0	Yes	Yes	Yes	Yes
Admintool	Solaris 9 and compatible versions	Yes	No	Yes	No
LP commands	Solaris 9 and compatible versions	No	Yes	Yes	Yes

After using the table above to determine which printing tool is best for your network environment, see Chapter 4 for printer setup information.

Most printing configuration tasks can be accomplished with Solaris Print Manager. However, if you have special needs, such as writing interface scripts or adding your

own filters, you can use the LP print service commands, which underlie Solaris Print Manager. Performing printing administration tasks with LP commands are described in Chapter 5.

The LP Print Service

The *LP print service* is a set of software utilities that allows users to print files while they continue to work. Originally, the print service was called the LP spooler. (LP stood for line printer, but its meaning now includes many other types of printers, such as laser printers. Spool is an acronym for system peripheral operation off-line.)

The print service consists of the LP print service software and spooler, which includes Solaris Print Manager; any print filters you might provide; and the hardware (the printer, system, and network connections).

See Chapter 8 for background information about the LP print service.

Other LP print service topics covered in this part and their chapter references are described below.

Managing Network Printers

A *network printer* is a hardware device that is connected directly to the network. It transfers data directly over the network to the output device. The printer or network connection hardware has its own system name and IP address.

Network printers often have software support provided by the printer vendor. If your printer has printer vendor supplied software it is strongly advised that the printer vendor software be utilized. If the network printer vendor does not provide software support, Sun supplied software is available. This software provides generic support for network attached printers but is not capable of providing full access to all possible printer capabilities.

See Chapter 4 for step-by-step instructions on setting up a network printer.

Administering Printers

After you set up print servers and print clients, there are a number of administration tasks you might need to perform frequently:

- Deleting a printer and remote printer access
- Checking the status of printers
- Restarting the print scheduler

See Chapter 5 for step-by-step instructions on how to perform the printer administration tasks.

Setting Definitions for Printers

Establishing definitions for the printers on your network is an ongoing task that lets you provide a more effective print environment for users. For example, you can assign printer descriptions for all your site's printers to help users find where a printer is located, or you can define a class of printers to provide the fastest turnaround for print requests.

See Chapter 3 for information on setting up printer definitions.

Administering Character Sets, Filters, Forms, and Fonts

Depending on your site's requirements and the types of printers you have on the network, you might have to set up and administer printer-specific features of the LP print service. For example, you can assign different print wheels, filters, and forms to different printers. See Chapter 6 for background information and step-by-step instructions on how to set up and administer character sets, print filters, forms, and fonts.

Customizing the LP Print Service

Although the LP print service is designed to be flexible enough to handle most printers and printing needs, it does not handle every possible situation. You might have a printing request that is not accommodated by the standard features of the LP print service. Or you can have a printer that does not quite fit into the way the LP print service handles printers.

You can customize the LP print service in the following ways:

- Adjust the printer port characteristics
- Adjust the `terminfo` database
- Customize the printer interface program
- Create a print filter
- Define a form

See Chapter 7 for detailed descriptions and step-by-step instructions to customize the LP print service.

The Solaris Print Client-Server Process

This section provides an overview of how Solaris printing works.

The Print Client Process

The figure below illustrates the path of a print request from the time the user initiates the request until it is printed.
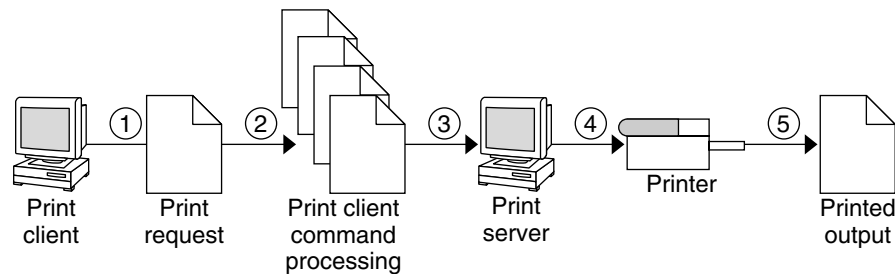
FIGURE 2–1 Overview of the Print Client Process

1. A user submits a print request from a print client.

2. The print command checks a hierarchy of print configuration resources to determine where to send the print request.

3. The print command sends the print request directly to the appropriate print server. A print server can be any server that accepts BSD printing protocol, including SVR4 (LP) print servers and BSD lpd-based print servers.

4. The print server sends the print request to the appropriate printer.

5. The print request is printed.

Using Print Clients

This section of the overview focuses on the *print client*, a system that can send print requests to a print server, and print commands, which enable the print client to initiate print requests.

The figure below highlights the part of the print process in which the user submits a print request from a print client.
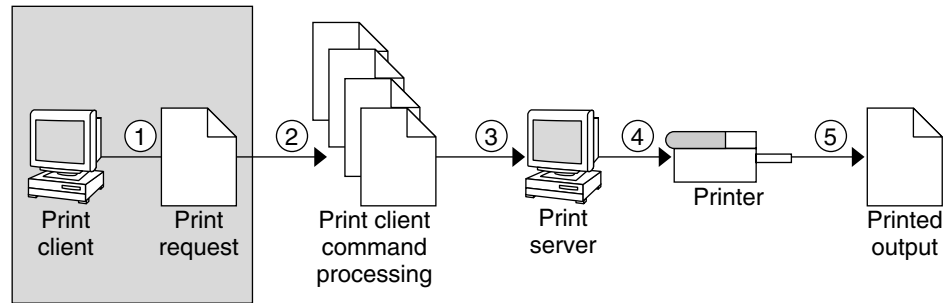
FIGURE 2–2 The User Submits a Print Request from a Print Client

What Is a Print Client?

A system becomes a print client when you install the Solaris print software and enable access to remote printers on the system.

The Solaris print software checks the following resources to locate printers and printer configuration information:

- The command-line interface using the `lp -d` command (atomic or POSIX format)
- A user's `LPDEST` or `PRINTER` variables
- The `_default` variable in the sources configured for the `printers` database in the `/etc/nsswitch.conf` file
- The user's `$HOME/.printers` file
- The local `/etc/printers.conf` file for the NIS name service
- The LDAP or NIS name service database

The print client sends its requests to the print server's queue; the client does not have a local queue. The client writes the print request to a temporary spooling area only if the print server is not available or if an error occurs. This streamlined path to the server decreases the print client's use of resources, reduces the chances for printing problems, and improves performance.

Printer Configuration Resources

This section describes the resources that the print software use to locate printer names and printer configuration information.

The print software can use a name service, which is a network (shared) resource for storing printer configuration information for all printers on the network. The name service (LDAP, NIS, NIS+, or NIS+ (xfn)) simplifies printer configuration maintenance: When you add a printer in the name service, all print clients on the network can access it.

The figure below highlights the part of the print process in which the print software checks a hierarchy of printer configuration resources to determine where to send the print request.
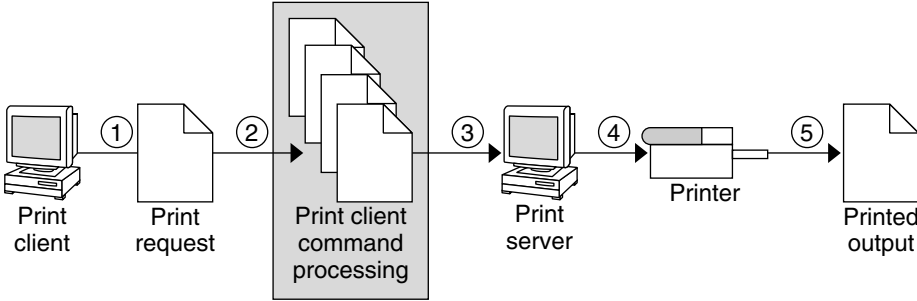
FIGURE 2–3 The Print Client Checks Resources to Locate Printers

How the Print Software Locates Printers

As shown in the figure below, the print software use more options to locate printers and printer configuration information.
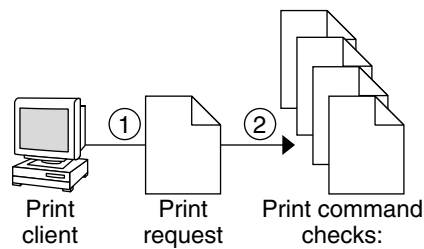
A. Atomic, POSIX, or Context-Based
   Printer Name or Class

B. User's `PRINTER` or `LPDEST`
   Environment Variable for Default Printer

C. `_default` Variable in Printers
   Database in `/etc/nsswitch.conf`

D. User's `$HOME/.printers` File

E. Local `/etc/printers.conf` File

F. LDAP directory information tree
   for printers in the domain's
   'ou=printers' container

G. NIS `printers.conf.byname` Map

H. NIS+ `printers.org_dir` Table

I. `NIS+ (xfn) FNS` Printing Contexts

FIGURE 2–4 How the Print Client Software Locates Printers

1. A user submits a print request from a print client by using the `lp` or `lpr`
   command. The user can specify a destination printer name or class in any of three
   styles:

   ■ Atomic style, which is the print command and option followed by the printer
     name or class, as shown in this example.

     ```
     % lp -d neptune filename
     ```

   ■ POSIX style, which is the print command and option followed by *server:printer*,
     as shown in the following example.

     ```
     % lpr -P galaxy:neptune filename
     ```

   ■ Context-based style shown in this example.

     ```
     % lpr -d thisdept/service/printer/printer-name filename
     ```

2. The print command locates a printer and printer configuration information as
   follows:

   ■ It checks to see if the user specified a destination printer name or printer class in
     one of the three valid styles.

   ■ If the user didn't specify a printer name or class in a valid style, the command
     checks the user's `PRINTER` or `LPDEST` environment variable for a default
     printer name.

   ■ If neither environment variable for the default printer is defined, the command
     checks the sources configured for the `printers` database in the
     `/etc/nsswitch.conf` file.

Using Print Servers

This section of the overview focuses on the print server, a system that has a local printer connected to it and makes the printer available to other systems on the network. The figure below highlights the part of the print process in which the print server sends the print request to the printer.
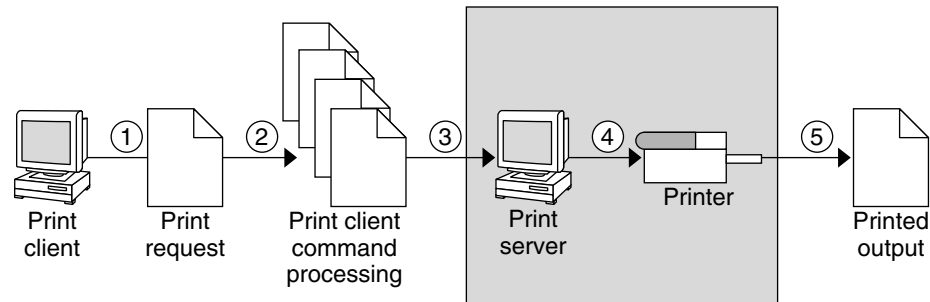
FIGURE 2–5 The Print Server Sends a Print Request to the Printer

The BSD Printing Protocol

The print commands use the BSD printing protocol. One of the big advantages of this protocol is that it can communicate with a variety of print servers:

- `lpd`-based print servers
- SunOS 5.9 and compatible SVR4 (LP) print servers
- Any other print server or printer that accepts the BSD printing protocol

The BSD printing protocol is an industry standard. It is widely used and it provides compatibility between different types of systems from various manufacturers. Sun has chosen to support the BSD printing protocol to provide interoperability in the future.

Where to Go From Here

Go to Chapter 4 for step-by-step instructions on setting up new printers with Solaris Print Manager. If you need printer planning information, see Chapter 3.

Planning Printers on Your Network (Overview)

The goal of setting up printers on a network is to give users access to one or more printers. This section provides information about distributing printers across your network to gain the best efficiency and about planning for printer setup.

- "Distributing Printers on the Network" on page 39
- "Assigning Print Servers and Print Clients" on page 40
- "Print Server Requirements and Recommendations" on page 41

For step-by-step instructions on print management tasks, see:

- Chapter 4
- Chapter 5
- Chapter 6
- Chapter 7

Distributing Printers on the Network

As an administrator, you must determine whether each printer would be best used if it is dedicated to one system or available to many systems. In a network environment, it usually works best to distribute your printers on several print servers. The advantage of setting up several print servers is that when one print server has a problem, you can route print requests to other print servers.

If you use a centralized print configuration, you can still connect printers to users' systems for convenience or for improved response. A printer that is connected to a user's system is still available to other systems on the network.

The figure below shows an example of how you can have a centralized print configuration and still connect printers to users' systems.

Print client   Print client   Local printer connected
                                         to a user's system

Printers connected to a
print server
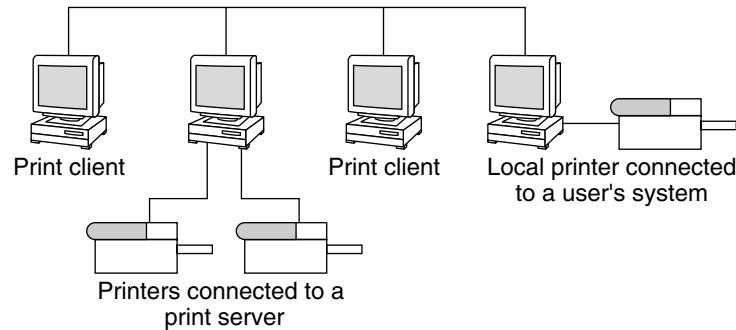
FIGURE 3–1 How to Distribute Printers on a Network

Assigning Print Servers and Print Clients

You must decide which systems will have local printers physically attached to them, and which will systems use printers on other systems. A system that has a local printer attached to it and makes the printer available to other systems on the network is called a *print server*. A system that sends its print requests to a print server is called a *print client*.

The LP print service software provides printing services in the Solaris environment. Besides physically connecting a printer to a system, you must define the printer characteristics to the LP print service and make the system a print server. Once you have print servers set up, you can set up other systems as print clients.

Print servers and print clients can run different versions of the Solaris release and different version of the UNIX operating system. Print clients running the Solaris 9 release and compatible versions can print to print servers running an `lpd`-based print service, such as the SunOS 4.1, BSD UNIX, and Linux releases. In addition, print clients running an `lpd`-based print service can print to print servers running the Solaris 9 release and compatible versions.

The figure below shows a print configuration on a network with systems running the Solaris release and an `lpd`-based print service. See "How Remote Printing Works" on page 174 for details on how Solaris printing works in heterogeneous environments.
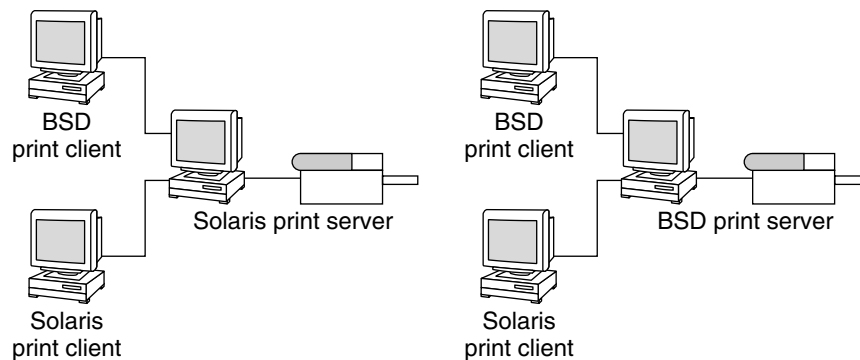
FIGURE 3–2 Solaris and LPD-Based Systems Print Configuration

Print Server Requirements and Recommendations

You can attach a printer to a standalone system or to any system on the network. Any networked system with a printer can be a print server, as long as the system has adequate resources to manage the printing load.

Spooling Space

Spooling space is the amount of disk space that is used to store and process requests in the print queue. Spooling space is the single most important factor to consider when deciding which systems to designate as print servers. When users submit files for printing, the files are stored in the `/var/spool/lp` directory until they have been printed. The size of the `/var` directory depends on the size of the disk and how the disk is partitioned. Spooling space can be allocated in the `/var` directory on the print server hard disk, or mounted from a file server and accessed over the network.

> **Note –** If `/var` is not created as a separate file system, the `/var` directory uses space in the root (/) file system, which is likely to be insufficient.

Disk Space

When evaluating systems as possible print servers, consider their available disk space. A large spool directory can consume 600 Mbytes of disk space. Look at the size and division of disk space on systems that can be designated as print servers.

Also, carefully evaluate the printing needs and use patterns of print client systems. If users in a small group typically print only short email messages—simple ASCII files without sophisticated formatting requirements—a print server with 20 to 25 Mbytes of disk space allocated to `/var` is probably sufficient. If, however, many print client users are printing large documents or bit-mapped or raster images, they will likely fill up the spooling space quite frequently. When users cannot queue their jobs for printing, work flow is interrupted. Requests for more spooling space can force you to either add disk space for spooling or designate a different system as the print server.

If the print server has a `/var` directory that resides in a small partition, and if a large amount of disk space is available elsewhere, you can use that space as spooling space by mounting it on the `/var` directory on the print server. See "Mounting and Unmounting File Systems (Tasks)" in *System Administration Guide: Basic Administration* for information about mounting file systems and editing the `vfstab` file.

Memory

The Solaris environment requires a minimum of 64 Mbytes of memory to run. A print server does not require additional memory. However, you might find that more memory improves performance in filtering print requests.

Swap Space

The swap space allocation on the print server should be sufficient to handle LP print service requirements. See "Configuring Additional Swap Space (Tasks)" in *System Administration Guide: Basic Administration* for information about how to increase swap space.

Hard Disk

For optimal performance, the print server should have a hard disk and a local `/var` directory. You should mount spooling space for a print server on a local hard disk. If a print server has its own hard disk and a local `/var` directory, printing is much faster, and you can more accurately predict the time needed to process print requests.

Planning for Printer Setup

This section provides an overview of planning for printing in the Solaris environment that includes:

- Setting definitions for printers such a printer name, printer description, printer port
- Selecting a printer type and file content type
- Setting up fault notification and default printer destination
- Determining whether you want to print banner pages or limit user access to a printer
- Setting up printer classes and fault recovery

Setting Definitions for Printers

Establishing definitions for the printers on your network is an ongoing task that lets you provide a more effective print environment for users. For example, you can assign parameters for all your site's printers to help users find where a printer is located, or you can define a class of printers to provide the fastest turnaround for print requests.

The `lpadmin` command lets you set all of the print definitions, while Solaris Print Manager lets you set only some of them when you install or modify a printer. The table below lists the print definitions and shows whether you can assign the definition with Solaris Print Manager.

TABLE 3–1 Print Definitions Set With Solaris Print Manager

Print Definition	Can You Set It With Solaris Print Manager?
Printer name	Yes
Printer description	Yes
Printer port	Yes

TABLE 3–1 Print Definitions Set With Solaris Print Manager   *(Continued)*

Print Definition	Can You Set It With Solaris Print Manager?
Printer type	Yes
File contents	Yes, but with less functionality than the `lpadmin` command
Fault notification	Yes, but with less functionality than the `lpadmin` command
Default printer destination	Yes
Printing banner pages	Yes, but with less functionality than the `lpadmin` command
Limiting user access to a printer	Yes, but with less functionality than the `lpadmin` command
Printer class	No
Fault recovery	No

Printer Name

When adding a printer to a system, you specify a *printer name* for the printer. A printer name must be:

- Unique among all printers within the bounds of an administrative domain
- A maximum of 14 alphanumeric characters, which can include dashes and underscores
- Easy to remember and can identify the type of printer, its location, or the print server name

Establish a naming convention that works for your site. For example, if you have different types of printers on the network, including the printer type as part of the printer name can help users choose an appropriate printer. For instance, you could identify PostScript™ printers with the letters PS. If, however, all of the printers at your site are PostScript printers, you would not need to include the initials PS as part of the printer name.

Printer Description

You can assign a description to a printer by using the `lpadmin -D` command or Solaris Print Manager. The printer's description should contain information to help users identify the printer. You might include the room number where the printer is located, the type of printer, the manufacturer, or the name of the person to call if there are printing problems.

Users can look at a printer description by using the following command:

$ **lpstat -D -p** *printer-name*

Printer Port

When you install a printer or later change its setup, you can specify the device, or the *printer port*, to which the printer is connected, by using Solaris Print Manager or the lpadmin -p *printer-name* -v *device-name* command.

Most systems have two serial ports and a parallel port or USB ports. Unless you add ports, you cannot directly connect more than two serial printers and a parallel or two USB printers to one system.

You can select the following printer port types with Solaris Print Manager. These options give you as much flexibility as the lpadmin command.

Printer Port Type	Corresponding Device Name Options
Serial	/dev/term/a
Serial	/dev/term/b
Parallel	/dev/printers/0 –> /dev/ecpp0
USB	/dev/printers/[1–9]
Specify any port name that the print server recognizes	Other

The LP print service initializes the printer port using the settings from the standard printer interface program. See "Managing Print Filters" on page 114 for more information about printer interface programs. If you have a parallel printer or a serial printer for which the default settings do not work, see "Adjusting Printer Port Characteristics" on page 133 for information about customizing the port settings.

IA only – If you use multiple ports on an IA based system, only the first port is enabled by default. The second and any subsequent ports are disabled by default. To use more than one port, you must manually edit the device driver port configuration file for each additional `asy` (serial) port or `lp` (parallel) port. The pathnames for the IA port configuration files are:

`/platform/i86pc/kernel/drv/asy.conf`

`/platform/i86pc/kernel/drv/lp.conf`

See the *Solaris 9 (Intel Platform Edition) Hardware Compatibility List* for information about configuring serial and parallel ports on IA based systems.

Printer Type

The printer type is a generic name for a type of printer. It identifies the `terminfo` database entry that contains various control sequences for the printer. By convention, printer type is usually derived from the manufacturer's model name. For example, the printer type name for the DECwriter™ printer is `decwriter`. However, the common printer type `PS` does not follow this convention. `PS` is used as the printer type for many models of PostScript™ printers, such as Apple LaserWriter®I and Apple LaserWriterII printers.

You can specify the printer type by using the `lpadmin -T` command or Solaris Print Manager.

Solaris Print Manager lets you select a printer type from a menu or choose `Other` and specify any printer type in the `terminfo` database. This provides you as much capability as the `lpadmin` command.

Printer Names in the `terminfo` *Database*

Information about each printer type is stored in the `terminfo` database (`/usr/share/lib/terminfo`). This information includes the printer capabilities and initialization control data. The printer you install must correspond to an entry in the `terminfo` database.

```
$ pwd
/usr/share/lib/terminfo
$ ls
1  3  5  7  9  B  H  P  a  c  e  g  i  k  m  o  q  s  u  w  y
2  4  6  8  A  G  M  S  b  d  f  h  j  l  n  p  r  t  v  x  z
$
```

Each subdirectory contains compiled database entries for terminals or printers. The entries are organized by the first letter of the printer or terminal type. For example, if

you have an Epson® printer, look in `/usr/share/lib/terminfo/e` to find your particular model of Epson printer.

```
$ cd /usr/share/lib/terminfo/e
$ ls
emots          ep2500+high   ep48            ergo4000      exidy2500
env230         ep2500+low    epson2500       esprit
envision230    ep40          epson2500-80    ethernet
ep2500+basic   ep4000        epson2500-hi    ex3000
ep2500+color   ep4080        epson2500-hi80  exidy
$
```

The entries for Epson printers are included in the preceding example.

If you have a NEC® printer, look in the `/usr/share/lib/terminfo/n` directory for your NEC printer model.

```
$ cd /usr/share/lib/terminfo/n
$ ls
ncr7900        ncr7900iv     netronics       network       nuc
ncr7900-na     ncr7901       netty           netx          nucterm
ncr7900i       nec           netty-Tabs      newhp
ncr7900i-na    net           netty-vi        newhpkeyboard
$
```

The entry in this directory for NEC is included in the preceding example.

Selecting a Printer Type

For a local PostScript printer, use a printer type of either PostScript (`PS`) or Reverse PostScript (`PSR`). If your printer supports PostScript, choose `PS` or `PSR` even if the specific printer type is listed in the `terminfo` database.

If your PostScript printer prints pages face up, documents appear to be printed backwards—the first page is at the bottom of the stack and the last page is on the top. If you specify the printer's type as `PSR`, the LP print service reverses the order of the pages before sending them to the printer; the last page is printed first, and the pages are stacked in forward order. However, the LP print service can reliably change the page order only for PostScript files that conform to the Adobe® Document Structuring Conventions in Appendix C of the *PostScript Language Reference Manual* (written by Adobe Systems Incorporated, and published by Addison-Wesley, 1990).

If a printer can emulate more than one kind of printer, you can assign it several types by using the `lpadmin -T` command. If you specify more than one printer type, the LP print service uses the type that is appropriate for each print request.

You might not find the printer type in the appropriate `terminfo` directory. The type of printer is not necessarily linked to the manufacturer's name on the printer. For example, for any type of PostScript printer, you can use the `PS` or `PSR` entry (found in

the `/usr/share/lib/terminfo/P` directory) instead of an entry specific to manufacturer or product names.

If you have an unusual type of printer, you might need to try different entries before you can determine whether a particular `terminfo` entry works for your model of printer. If possible, find an entry in the `terminfo` database that works for your printer. It will be much easier than trying to create an entry. If you have to create your own entry, "Adding a `terminfo` Entry for an Unsupported Printer" on page 135 contains some useful tips.

Selecting a File Content Type

Print filters convert the content type of a file to a content type that is acceptable to the destination printer. The *file content type* tells the LP print service the type of file contents that can be printed directly, without filtering. To print without filtering, the necessary fonts must also be available in the printer. (You must set up and use filtering for other types of files.)

You can specify the file content type for a printer by using the `lpadmin -I` command or Solaris Print Manager. With Solaris Print Manager, you can select a file content type from a menu. Not all available file content types are listed on the menu. You must use the `lpadmin` command to specify file content types that are not included on the Solaris Print Manager menu.

Many printers can print two types of files directly:

- The same type as the printer type (for example, `PS` for a PostScript printer)
- The type `simple` (an ASCII text file)

When submitting a file for printing, the user can indicate the content type of the file (`lp -T` *content-type*). If no file content type is supplied when the request is submitted, the LP server looks at the first file in the request to determine the content type. If the file begins with `^D%!` or `%!`, the request is considered to contain PostScript™ data. Otherwise, the request is assumed to contain `simple` (ASCII) text. The LP print service uses the file content type to determine which filters to use to convert the file contents into a type the printer can handle.

Solaris Print Manager provides a list of file content types from which you can choose when installing or modifying a local printer. The choices are translated to the names that the LP print service uses. The table below describes the file content types you can choose with Solaris Print Manager.

TABLE 3–2 Choosing File Content Type With Solaris Print Manager

File Contents Choice	LP Print Service Name	Description
PostScript	`postscript`	PostScript files do not require filtering.
ASCII	`simple`	ASCII files do not require filtering.
Both PostScript and ASCII	`simple, postscript`	PostScript files and ASCII files do not require filtering.
None	`""`	All files require filtering, except those matching the printer's type.
Any	`any`	No filtering required. If the printer cannot handle a file content type directly, the file will not be printed.

Choose the file content type that best matches the printer's capabilities. PostScript (which means filtering is not needed for PostScript files) is the default choice in Solaris Print Manager and is probably correct most of the time.

Frequently Used Printers

This section provides the printer type and file content type for the printers most commonly used with Solaris software. Although not shown, many of these printers can also directly print files with `simple` content type.

If you have a PostScript printer, use a printer type of PS or PSR and a content type of `postscript`. PSR reverses the pagination and prints the banner page last.

The table below lists additional non-PostScript printers and shows the printer type to use for configuring each printer. For all these printers, the file content type is `simple`.

Note – Sun Microsystems does not supply filtering software for the printers listed in the table below, among others. However, you can use unsupported printers if you supply filtering or if the printer can directly print the file content type. If you have questions about any printer for which Sun Microsystems does not supply filters, contact the printer manufacturer.

TABLE 3–3 Some Non-PostScript Printers for Which Sun Does Not Supply Filters

Printer	Printer Type
Daisy	`daisy`
Datagraphix	`datagraphix`

TABLE 3–3 Some Non-PostScript Printers for Which Sun Does Not Supply Filters *(Continued)*

Printer	Printer Type
DEC LA100	`la100`
DEC LN03	`ln03`
DECwriter	`decwriter`
Diablo	`diablo`
	`diablo-m8`
Epson 2500 variations	`epson2500`
	`epson2500-80`
	`epson2500-hi`
	`epson2500-hi80`
Hewlett-Packard HPCL printer	`hplaser`
IBM Proprinter	`ibmproprinter`

If you want to set up a printer that is not in the `terminfo` database, see "How to Add a `terminfo` Entry for an Unsupported Printer" on page 138.

Setting Up Printers (Tasks)

This chapter describes how to set up a printer and make it accessible to systems on the network with Solaris Print Manager. This is a list of the step-by-step instructions in this chapter.

- "How to Start Solaris Print Manager" on page 54
- "How to Add a New Attached Printer With Solaris Print Manager" on page 57
- "How to Add Printer Access With Solaris Print Manager" on page 59
- "How to Set Up a `.printers` File" on page 61
- "How to Add a Network Printer Using Printer Vendor Supplied Tools" on page 66
- "How to Add A Network Printer Using LP Commands" on page 67

For overview information about printers, see Chapter 2.

Setting Up Printing With Solaris Print Manager

Solaris Print Manager is a Java-based graphical user interface that enables you to manage local and remote printer configuration. This tool can be used in the following name service environments: LDAP, NIS, NIS+, NIS+ with Federated Naming Service (xfn), and files. You must be logged in as superuser to use this tool.

The table below provides an overview of the tasks necessary to set up print servers (New Attached Printer) and print clients (Add Access to Printer). A local or attached printer is one which is physically cabled to the print server; a network printer is physically attached to the network. Adding access to a printer, or adding remote access, is the process of giving print clients (all those machines which are not the server) access to the printer.

Setting Up Printing Task Map

TABLE 4–1 Task Map: Setting Up Printing

Task	Description	For Instructions, Go To
1. Add New Attached Printer	*Using Solaris Print Manager* After physically attaching the printer to a system, make the printer available for printing.	"How to Add a New Attached Printer With Solaris Print Manager" on page 57
2. Add Access to a Printer	*Using Solaris Print Manager* Add printer access on the print client.	"How to Add Printer Access With Solaris Print Manager" on page 59
3. Set Up a `.printers` File	*Optional.* Using a `$HOME/.printers` file enables users to establish their own custom printer aliases.	"How to Set Up a `.printers` File" on page 61
4. Add a New Network Printer	*Using Printer Vendor Supplied Tools* After physically connecting the printer to the network, use vendor-supplied software to configure the network printer. *Using LP Commands* After physically connecting the printer to the network, use Solaris software commands to configure the network printer.	"How to Add a Network Printer Using Printer Vendor Supplied Tools" on page 66 "How to Add A Network Printer Using LP Commands" on page 67
5. Turn Off Banner Pages	*Optional.* You can turn off banner pages so they are never printed.	"How to Turn Off Banner Pages" on page 83
6. Set Up Fault Alerts	*Optional.* You can set up more specific fault alerts for the printer than Solaris Print Manager provides.	"How to Set Fault Alerts for a Printer" on page 86
7. Set Up Fault Recovery	*Optional.* Solaris Print Manager does not enable you to set up how a printer should recover after it faults.	"How to Set Printer Fault Recovery" on page 88
8. Limit Access to the Printer	*Optional.* Solaris Print Manager enables you to set up an allow list, but if you want to limit a few users' access to the printer, you might want to set up a deny list.	"How to Limit User Access to a Printer" on page 90

Setting Up a Printer With Solaris Print Manager

The following table describes each printer attribute to help you determine the information needed to set up a printer with Solaris Print Manager.

Printer Attribute	Description	Example	Default Setting	Required or Optional?
Printer Name	Name of printer	`laser1`	N/A	Required to install an attached or network printer and to add access to a printer
Printer server	Name of printer server	`venus`	The local system	Required to install an attached or network printer and to add access to a printer
Description	User defined string	`laser printer near breakroom`	N/A	Optional
Printer Port	Device printer is attached to	`/dev/term/a`	`/dev/term/a`	Required to install an attached printer
Printer Type	Type of printer	`unknown`	`PostScript`	Required to install an attached or network printer
File Contents	Content to be printed	`any`	`PostScript`	Required to install an attached or network printer
Destination	Destination name for network printers	See "Selecting the Destination (or Network Printer Access) Name" on page 64 for examples	N/A	Required to install a network printer
Protocol	Protocol used to communicate with printer	`TCP`	`BSD`	Required to install a network printer
Fault Notification	How to notify user of errors	`Mail to superuser`	`Write to superuser`	Optional
Default Printer	Identifies the default printer	N/A	N/A	Optional
Always Print Banner	Print banner with print job?	N/A	Banner is printed	Optional
User Access List	List of users allowed to print	`rimmer,lister`	All users can print	Optional

Solaris Print Manager and Special Characters

Solaris Print Manager checks user input for the various text fields in the input screens. There are two types of checking: general illegal input and input that is illegal for specific fields.

Solaris Print Manager does not accept the following characters as input, except for the help screens:

- Shell meta characters, such as "\$^&*(){}''|;:?<>, except for the destination field on the network printer screen, which accepts colons (:)
- Multibyte characters
- Pound signs (#), spaces, or tabs, except the description field, which accepts tabs

Starting Solaris Print Manager

To use Solaris Print Manager to set up your printers, start Solaris Print Manager either by selecting Printer Administrator from the CDE Workspace menu or by starting it from the command line. See the following section for details.

▼ How to Start Solaris Print Manager

1. **Verify that the following prerequisites are met. To use Solaris Print Manager, you must:**

   - Have a bit-mapped display monitor. Solaris Print Manager can be used only on a system with a console that is a bit-mapped screen, such as a standard display monitor that comes with a Sun workstation.
   - Be running an X Window System, such as the CDE environment, or be using the remote display feature on a system running an xhost environment.
   - Be logged in as superuser on the printer server to install an attached or network printer, or on the print client to add access to a printer.
   - Have the required access privileges for managing the LDAP, NIS, NIS+, or NIS+ (xfn) database:
     - If your name service is LDAP, you must have the following:
       - The distinguished name (DN) of a printer administrator and password in the directory.
       - The name or IP address of an LDAP server hosting the administered domain. This is usually displayed and selected automatically.

See "Managing Printer Information With LDAP" on page 28 for more information about using LDAP to manage printer information.

- If your name service is NIS, you must have the root password for the NIS master.

- If you name service is NIS+, you might need to do the following:

   1. Log in to the NIS+ master as superuser.

   2. Identify the group that owns the printers table:

      ```
      # niscat -o printers.org_dir.domain_name.com
                     .
                     .
                     .
                 Group        : "admin.domain_name.com"
      ```

   3. If necessary, add the system that runs Solaris Print Manager to the NIS+ admin group authorized to update the printers.org_dir.<domain> file.

      ```
      # nisgrpadm -a admin.domain_name.com host_name
      ```

   4. Log in to the system that runs Solaris Print Manager as superuser. Depending on your NIS+ configuration, you might also need to run the /usr/bin/keylogin command. See keylogin(1) for more information.

- If your name service is NIS+ (xfn), you might need to do the following:

   1. Log in to the NIS+ master as superuser.

   2. Identify the group that owns the federated naming table:

      ```
      # niscat -o fns.ctx_dir.domain_name.com
                     .
                     .
                     .
                 Group        : "admin.domain_name.com"
      ```

   3. If necessary, add the system that runs Solaris Print Manager to the NIS+ admin group authorized to update the fns.ctx_dir.<domain> file.

      ```
      # nisgrpadm -a admin.domain_name.com host_name
      ```

   4. Log in to the system that runs Solaris Print Manager as superuser. Depending on your NIS+ configuration, you might also need to run the /usr/bin/keylogin command. See keylogin(1) for more information.

- Have the SUNWppm package installed.

   ```
   # pkginfo | grep SUNWppm
   system      SUNWppm          Solaris Print Manager
   ```

2. **Start Solaris Print Manager by selecting Printer Administrator from the Tools option of the CDE Workspace menu. Or, select the Applications menu from the CDE front panel, and click the Printer Administrator icon in the Application Manager's System_Admin window. You can also start Solaris Print Manager by using the following command.**

`# /usr/sadm/admin/bin/printmgr &`

The Select Naming Service window overlays the Solaris Print Manager main window.

If you want to use Solaris Print Manager from a remote system, do the following:

Use the `xhost` command on the local system to give the remote system display access, like this:

`# xhost +`*remote-system*

Then login to the remote system, set the `DISPLAY` environment variable, and start Solaris Print Manager:

```
# DISPLAY=local-system:display_number
# export DISPLAY
# /usr/sadm/admin/bin/printmgr &
```

Note – If Solaris Print Manager fails to start from the CDE menu or from the command line, check the following:

1. Superuser (root) might not have permission to connect to the Xserver process on the local or remote system. If this happens, type this command:

```
$ xhost +hostname
$ su
(Enter root's password)
# /usr/sadm/admin/bin/printmgr &
```

Replace *hostname* with either the local or remote system name before restarting Solaris Print Manager.

2. Verify the `SUNWppm` package is installed on the local or remote system.

`$ pkginfo | grep SUNWppm`

---

3. **Select the name service used in your network from the Select Naming Service window. Choices are: NIS+ (xfn), NIS+, NIS, or files.**

4. **Check that the domain name is correct.**

The Solaris Print Manager main menu is displayed after the name service is loaded successfully.

Setting Up a Print Server

When you install an attached printer and/or a network printer to a system, the printer is made accessible to the local system. The system on which you install the printer becomes the *printer server*.

The following sections describe how to use the new Solaris Print Manager to add an attached printer or a network printer on a printer server. The example that follows each Solaris Print Manager procedure describes how to add a printer with LP commands.

▼ How to Add a New Attached Printer With Solaris Print Manager

1. **Select the system which is to be the printer server.**

2. **Connect the printer to the printer server and turn on the power to the printer.**

   Consult the printer vendor's installation documentation for information about the hardware switches and cabling requirements.

3. **Start Solaris Print Manager on the printer server where you connected the printer.**

   See "How to Start Solaris Print Manager" on page 54 for instructions.

4. **Select New Attached Printer from the Printer menu.**

   The New Attached Printer window is displayed.

5. **Fill in the window.**

   If you need information to complete a field, click the Help button.

6. **Click OK.**

7. **Verify that the printer has been installed by checking for the new printer entry in the Solaris Print Manager main window. Verify the printer can print requests by using the following command:**

   $ `lp -d` *printer-name* *filename*

8. **Exit Solaris Print Manager.**

   Choose Exit from the Print Manager Menu.

Example—Adding a New Attached Printer With LP Commands

This example shows how to make a local PostScript printer available for printing on a print server. The commands in this example must be executed on the print server where the printer is connected. The following information is used as an example. The information you provide will vary:

- Printer name: `luna`
- Port device: `/dev/term/b`
- Printer type: `PS`
- File content type: `postscript`

```
  # chown lp /dev/term/b
1 # chmod 600 /dev/term/b
2 # lpadmin -p luna -v /dev/term/b
3 # lpadmin -p luna -T PS
4 # lpadmin -p luna -I postscript
  # cd /etc/lp/fd
  # for filter in *.fd;do
     > name=`basename $filter .fd`
     > lpfilter -f $name -F $filter
5    > done
  # accept luna
     destination "luna" now accepting requests
6 # enable luna
printer "luna" now enabled
7 # lpadmin -p luna -D "Room 1954 ps"
8 # lpstat -p luna
     printer luna is idle. enabled since Jul 12 11:17 2001. available.
```

1. Gives `lp` ownership and sole access to a port device.
2. Defines the printer name and the port device the printer will use.
3. Sets the printer type of the printer.
4. Specifies the file content types to which the printer can print directly.
5. Adds print filters to the print server.
6. Accepts print requests for the printer and enables the printer.
7. Adds a description for the printer.
8. Verifies that the printer is ready.

Where to Go From Here

Use the following table to determine which tasks to complete next.

If You Need To ...	See ...
Add access to the newly installed printer on the print clients because you did not add the printer information to the name service database	"How to Add Printer Access With Solaris Print Manager" on page 59
Set up a `.printers` file	"How to Set Up a `.printers` File" on page 61

Setting Up a Print Client

A print client is a system that is not the server for the printer, yet has access to the printer. A print client uses the services of the print server to spool, schedule and filter the print jobs. Note that one system can be a print server for one printer and be a print client for another printer.

Access to a printer can be configured on a domain-wide basis or on a per-machine basis depending on whether you add the printer information to the name service database.

The following sections describe how to use the new Solaris Print Manager to add access to a printer on a print client. The example that follows this procedure describes how to add printer access with LP commands.

▼ How to Add Printer Access With Solaris Print Manager

1. **Start Solaris Print Manager on the system where you want to add access to a remote printer.**
   See "How to Start Solaris Print Manager" on page 54 for instructions.

2. **Select Add Access to Printer from the Printer menu.**
   The Add Access to Printer window is displayed.

3. **Fill in the window.**
   If you need information to complete a field, click the Help button.

4. **Click OK.**

5. **Verify that access to the printer is added by checking for the new printer entry in the Solaris Print Manager main window. Verify the printer can print requests by using the following command:**

   $ **lp -d** *printer-name filename*

6. **Exit Solaris Print Manager.**

   Choose Exit from the Print Manager Menu.

Example—Adding Printer Access With LP Commands

If you want to print to a remote printer, you must add access to the remote printer. This example shows how to configure access to a printer named luna, whose print server is saturn. The system saturn becomes a print client of the printer luna.

```
1  # lpadmin -p luna -s saturn
2  # lpadmin -p luna -D "Room 1954 ps"
3  # lpadmin -d luna
4  # lpstat -p luna
        printer luna is idle. enabled since Jul 12 11:17 2001. available.
```

1. Identifies the printer and the print server.
2. Adds a description for the printer.
3. Sets the printer as the system's default printer destination.
4. Verifies that the printer is ready.

Setting Up a .printers File

There is no need to set up a .printers file in your users' home directories if they don't need customized printer information. However, the .printers file enables users to establish their own custom printer aliases. You can use the _default alias to make a printer the default. You can also set up a special _all alias to define a list of printers affected when you cancel a print request or check the status of printers.

Keep in mind use of the .printers file by the LP print service is controlled by the name service switch (/etc/nsswitch.conf). The default configuration is that the print service checks a user's home directory to locate printer configuration information before its checks the other name services. This means you can tailor a user's printer configuration file to use custom printer information rather than the shared information in the name service.

See `printers`(4) for more information about the `.printers` file. See `nsswitch.conf`(4) for more information about the name service switch.

▼ How to Set Up a `.printers` File

1. **Log in to the user's system as superuser.**

2. **Start the file editor you want to use to create a** `.printers` **file in the user's home directory.**

3. **(Optional) Set up the** `_default` **alias to make a specific printer your default printer, using an entry similar to the one shown in the following example.**

   `_default` *printer_name*

4. **(Optional) Set up the** `_all` **alias to define the printers affected when you cancel a print request or check the status of printers, using an entry similar to the one shown in the next example.**

   `_all` *printer1,printer2,printer3*

5. **Save the file as** `.printers`**.**

Adding a Network Printer

A *network printer* is a hardware device this is directly connect to the network. This means it can be accessed from a print server without actually connecting it the print server with a cable. It has its own system name and IP address. Even though a network printer is not connected to a print server, it is necessary to set up a print server for it. The print server provides queuing capabilities, filtering, and printing administration for the network printer.

Network printers might use one or more special protocols that require a vendor-supplied printing program. The procedures to set up the vendor-supplied printing program can vary. If the printer does not come with vendor supplied support, the Solaris network printer support can be used with most devices; it is strongly advised to use the print vendor supplied software when possible.

The vendor might supply an SVR4 printer interface script to replace the standard printer interface script. If so, their SVR4 interface script will call the vendor-supplied printing program to send the job to the printer. If not, you will need to modify the standard interface script to call the vendor-supplied printing program. You can do this

by editing the per-printer copy of the standard interface script to call the vendor-supplied printing program.

The terms used in network printer configuration are:

- Print server: The machine which spools and schedules the jobs for a printer. This is the machine on which the printer is configured.

- Printer-host device: The printer-host device is the software and hardware supplied by a vendor which provides network printer support for a non-network capable printer. The combination of the printer-host device with one or more printers attached to it creates a *network printer*.

- Printer node: This is either the physical printer or the printer-host device. It is the physical printer when the network support resides in the physical printer. It is the printer-host device when an external box is used to provide the network interface. The printer node name is the machine name given with the IP address. This name is selected by the system administrator and has no default or vendor requirement. The printer nodename, as with all nodes, must be unique.

- Printer name: The name entered on the command line when using any of the printer commands. It is selected by the system administrator at the time of printer configuration. Any one physical printer can have several printer or queue names; each provides access to the printer.

- Destination or network printer access name: The internal name of the printer node port that is used by the printer sub-system to access the printer. It is the name of the printer node, or the name of the printer node with a printer vendor port designation. Any printer vendor port designation is explicitly defined in the printer vendor documentation. It is printer specific. In the case where the printer is a printer-host device and a printer, the port designation is documented in the printer-host device documentation. The format is:

  printer_node_name

  or

  printer_node_name:*port_designation*

- Protocol: the over-the-wire protocol used to communicate with the printer. The printer vendor documentation supplies the information regarding the protocol to select. The network printer support supplies both BSD Printer Protocol and raw TCP. Due to implementation variations, you might want to try both.

- Timeout, or retry interval: Timeout is a seed number representing the number of seconds to wait between attempting connections to the printer. This seed number is the smallest amount of time to wait between attempted connections, and increases with an increase in failed connections. After repeated failures to connect to the printer, a message is returned to the user requesting possible human intervention. Attempts to reconnect continue until successful or the job is cancelled by the job owner.

Printer Vendor Supplied Software for Network Printers

Network printers often have software support provided by the printer vendor. If your printer has printer vendor supplied software it is strongly advised that the printer vendor software be utilized. The software is designed to support the attributes of the printer and can take full advantage of the printer capabilities. Read the printer vendor documentation to install and configure the printer under an LP print system.

Sun Support for Network Printers

If the network printer vendor does not provide software support, the Sun supplied software is available. The software provides generic support for network printers and is not capable of providing full access to all possible printer attributes.

A general discussion of how to add a network printer is provided in Chapter 4. The following is a discussion of printer management using the Sun supplied software.

Invoking the Network Printer Support

The software support for network printers is called through the interface script. Configuring a network printer with the network interface script, `netstandard`, causes the network printer support module to be called. The command to configure the printer with the network support is:

```
lpadmin -p printer_name -m netstandard
```

Selecting the Protocol

The print sub-system uses BSD print protocol and raw TCP to communicate with the printer. The printer vendor documentation will provide the information about which protocol to use. In general, we have found that the TCP protocol is more generic across printers.

The command to select the protocol is:

```
lpadmin -p printer_name -o protocol=bsd
```

or

```
lpadmin -p printer_name -o protocol=tcp
```

If the protocol selected is the BSD print protocol, you can further select the order of sending the control file to the printer. Some printers expect the control file, then the data file; others the reverse. See the printer vendor documentation for this information. The default is to send the control file first.

The command to select the ordering is:

```
lpadmin -p printer_name -o bsdctrl=first
```

or

```
lpadmin -p printer_name -o bsdctrl=last
```

Selecting the Printer Node Name

The system administrator selects the printer node name. This name must be unique, as with any node on the network. The printer node name is associated with the IP address of the printer.

Selecting the Destination (or Network Printer Access) Name

The print subsystem requires access information for the printer. This is the name that the subsystem uses when making the network connection to the printer. This name is supplied by the system administrator to the print sub-system through the lpadmin command. It becomes part of the printer configuration database. The printer access name is the name of the printer node, sometimes qualified by a port name. Port designation varies across printer vendors. You will find information about port designation in the documentation that is provided with the printer by the printer vendor. The format of printer access name is:

printer_node-name[:*port_designation*]

Example 1—Destination (or Network Printer Access Name) With Port Designation (Number)

A common port designation with TCP is 9100. If the printer node name is pn1, and the printer vendor defines the port as 9100, then the printer access name is: `pn1:9100`. To configure a printer in this case use:

```
lpadmin -p printer_name -o dest=pn1:9100
```

Example 2—Destination (or Network Printer Access Name) With Port Designation (Name)

When using the BSD protocol, the port designation might not be a number, but some name defined by the printer vendor, for example: *xxx*_parallel_1. If the printer node name is *cardboard,* then the printer access name is: cardboard:xxx_parallel_1. To configure a printer in this case use:

```
lpadmin -p printer_name -o dest=cardboard:xxx_parallel_1
```

Example 3—Destination (or Network Printer Access Name) With No Port Designation

If there is no port designation, and the printer node name is newspaper, the printer access name is the printer node name: newspaper. To configure a printer in this case use:

```
lpadmin -p printer_name -o dest=newspaper
```

Setting the Timeout Value

The timeout option is provided to allow for individual selection of the amount of time (in seconds) to wait between successive attempts to connect to the printer. Some printers have a long warm up time and a longer timeout value is advised. The default is 10 seconds.

The timeout value does not impact the success or failure of the print process. It is a seed value which the software uses as the initial timeout count; on repeated failures, this count is increased. A message is sent to the spooler when repeated attempts to connect to the printer fail. This alerts the user that intervention might be required. This could be anything from the printer being turned off, to out of paper. Should these messages be produced too often, for example when the printer is warming up, increasing the timeout value will eliminate spurious messages.

The system administrator can experiment to find the optimal timeout value. The command to set the timeout is:

```
lpadmin -p printer_name -o timeout=n
```

Managing Network Printer Access

Each network printer should have one and only one server that provides access to it. This enables the server to manage the access to the printer and keep jobs coherent.

The default device for the network printer is /dev/null. This is sufficient when there is only one queue for the printer. Should more queues be required, set the device to a file. This enables the print system to restrict access to the printer across queues. The following commands create a device file and configure it as the network printer device.

```
touch /path/filename
chmod 600 /path/filename
lpadmin -p printer_name -v /path/filename
```

The following is an example of creating a device file called devtreedown.

```
# touch /var/tmp/devtreedown
# chmod 600 /var/tmp/devtreedown
# lpadmin -p treedown -v /var/tmp/devtreedown
```

▼ How to Add a Network Printer Using Printer Vendor Supplied Tools

1. **Connect the printer to the network and turn on the power to the printer.**

   Consult the printer vendor's installation documentation for information about the hardware switches and cabling requirements. Get an IP address and select a name for the printer node. This is equivalent to adding any node to the network.

2. **Follow the printer vendor instructions to add the network printer.**

   Use the printer vendor instructions to configure the network printer. These will be specific to the vendor and printer.

3. **Add client access to the new printer.**

   Now that the printer has been added, create access to the printer for the clients. See "Setting Up a Print Client" on page 59.

4. **Optional tasks to complete.**

   There are several optional tasks you might want to complete when setting up a network printer. See "Setting Up Printing Task Map" on page 52 for pointers to the remaining tasks.

▼ How to Add A Network Printer Using LP Commands

Note – This describes the steps necessary to setup a network printer using the network printer support software. The use of this software is intended for those printers that do not come with vendor supplied software.

---

1. **Connect the printer to the network and turn on the power to the printer.**

   Consult the printer vendor's installation documentation for information about the hardware switches and cabling requirements. Get an IP address and select a name for the printer node. This is equivalent to adding any node to the network.

2. **Collect the information required to configure a network printer.**

   - Printer name
   - Printer server
   - Network printer access name
   - Protocol
   - Timeout

   See the terms described in "Adding a Network Printer" on page 61 for more information.

3. **Define the printer name, the device, the printer type and content type by using the** lpadmin**(1M) command.**

   a. **Define the printer name and the port device the printer will use.**

      # **lpadmin -p** *printer-name* **-v /dev/null**

      The device to use is /dev/null.

   b. **Identify the interface script the printer will use.**

      # **lpadmin -p** *printer-name* **-m netstandard**

      The interface script that is supplied with the network printer support software is /usr/lib/lp/model/netstandard.

   c. **Set the printer destination, protocol, and timeout values.**

      # **lpadmin -p** *printer-name* **-o dest=***access-name:port* **-o protocol=***protocol*
      -o timeout=*value*

-p *printer-name*	Specifies the network printer name.

`-o dest=`*access-name:port*	Sets the printer destination to the network printer access name and a designated printer vendor port, if it is defined in the printer vendor documentation. See "Adding a Network Printer" on page 61 for more information.
`-o protocol=`*protocol*	Sets the over-the-wire protocol used to communicate with the printer. Both BSD and raw TCP are supported.
`-o timeout=`*value*	Sets a retry timeout value that represents a number of seconds to wait between attempting connections to the printer. See "Adding a Network Printer" on page 61 for more information.

    d. Specify the file content types of the printer and the printer type.

        # **lpadmin -p** *printer-name* **-I** *content-type* **-T** *printer-type*

4. Add filters to the print server.

```
# cd /etc/lp/fd
# for filter in *.fd;do
    > name=`basename $filter .fd`
    > lpfilter -f $name -F $filter
    > done
```

5. Enable the printer to accept printer requests and to print the requests.

    # **accept** *printer-name*
    # **enable** *printer-name*

6. Verify the printer is correctly configured.

    # **lpstat -p** *printer-name*

7. Add client access to the new printer.

Now that the printer has been added, create access to the printer for the clients. See "Setting Up a Print Client" on page 59.

8. Optional tasks to complete.

There are several optional tasks you might want to complete when setting up a printer. See "Setting Up Printing Task Map" on page 52 for pointers to the remaining tasks.

The commands in this example must be executed on the print server. The following information is used as an example. The information you provide will vary.

- Printer name: `luna1`
- Server: `saturn`

- Network printer access name: `nimquat:9100`
- Protocol: `tcp`
- Timeout: 5
- Interface: `/usr/lib/lp/model/netstandard`
- Printer type: `PS`
- Content types: `postscript`
- Device: `/dev/null`

```
1 # lpadmin -p luna1 -v /dev/null
2 # lpadmin -p luna1 -m netstandard
3 # lpadmin -p luna1 -o dest=nimquat:9100 -o protocol=tcp
-o timeout=5
4 # lpadmin -p luna1 -I postscript -T PS
 # cd /etc/lp/fd
 # for filter in *.fd;do
       > name=`basename $filter .fd`
       > lpfilter -f $name -F $filter
   5   > done
 # accept luna1
   destination "luna1" now accepting requests
6 # enable luna1
printer "luna1" now enabled
7 # lpadmin -p luna1 -D "Room 1954 ps"
8 # lpstat -p luna1
       printer luna1 is idle. enabled since Jul 12 11:17 2001.  available.
```

1. Defines printer name and sets the device to `/dev/null`.
2. Defines the interface script for network printers.
3. Sets the destination, protocol and timeout.
4. Specifies the file content types to which the printer can print directly, and the printer type.
5. Adds print filters to the print server.
6. Accepts print requests for the printer and enables the printer.
7. Adds a description for the printer.
8. Verifies that the printer is ready.

Converting Printer Configuration Information

This section explains how to convert the printer configuration information from systems running the SunOS 5.5.1 release and copy this information to print clients so they can access existing printers.

Note – If you have only a few existing printers, it might be easier to add access to the printers by using Solaris Print Manager rather than convert the printer configuration information and distribute it to print clients. See Table 4–1 information on adding access to printers.

Converting Printer Configuration Information Task Map

The table below provides an overview of converting printer configuration information.

TABLE 4–2 Converting Printer Configuration Information Task Map

Task	Description	For Instructions, Go To
Convert Existing Printer Configuration Information	*Convert Printer Configuration Information For Systems Running the SunOS 5.5.1 Release* If your site uses SunOS 5.5.1 release, convert the printer configuration information in the `/etc/lp/printers` directory to the `/etc/printers.conf` configuration file. This is usually a one-time task.	"How to Convert Printer Information For a System Running the Solaris 2.5.1 Release" on page 71
	Convert Printer Configuration Information For a System Running an lpd-based Print Server If your site uses `lpd`-based software, convert the printer configuration information in the `/etc/printcap` file to the `/etc/printers.conf` configuration file. This is usually a one-time task.	"How to Convert Printer Information For a System Running an `lpd`-based Print Server" on page 72

Converting Existing Printer Configuration Information

Existing printer configuration information will automatically be converted when installing or upgrading to the Solaris 9 release. This section explains how to convert the printer configuration information for a system running SunOS 5.5.1 release or a system running a lpd-based printer server to the /etc/printers.conf printer configuration file. You'll use one of two print administration commands to automate the conversion task:

- The conv_lp(1M) command enables you to convert information in the /etc/lp/printers directory on a SunOS 5.9 system to entries in the system's /etc/printers.conf file. See "How to Convert Printer Information For a System Running the Solaris 2.5.1 Release" on page 71 for instructions.

- The conv_lpd(1M) command enables you to convert information in a /etc/printcap configuration file from a system running a lpd-based printer server to entries in a /etc/printers.conf file. See "How to Convert Printer Information For a System Running an lpd-based Print Server" on page 72 for instructions.

If you are not using a name service, you should create a master /etc/printers.conf file that includes the existing printers at your site. You can then copy the master file to all the print clients or by loading it into the NIS or NIS+ name service. This is a good way to initially enable all the new print clients access to the existing printers at your site.

/!\ **Caution –** If you are using the NIS or NIS+ name service to configure printer information, do not use a /etc/printers.conf file on your print clients. A print client uses the /etc/printers.conf file first to locate a printer; however, the /etc/printers.conf file might conflict with the printer information in the NIS or NIS+ maps and cause unexpected results. To avoid this problem, remove the /etc/printers.conf file on print clients when you want them to use NIS or NIS+ for printer information.

▼ How to Convert Printer Information For a System Running the Solaris 2.5.1 Release

1. **Log in as superuser on a system running the Solaris 9 release.**

2. **Convert the printer configuration information in the system's** /etc/lp/printers **directory to the** /etc/printers.conf **file.**

   # **/usr/lib/print/conv_lp**

▼ How to Convert Printer Information For a System Running an `lpd`-based Print Server

1. **Copy the** `/etc/printcap` **file from a system running an** `lpd`**-based print server to a system running the Solaris 9 release.**

2. **Log in as superuser on the system running the Solaris 9 release to which you copied the** `/etc/printcap` **file.**

3. **Convert the printer configuration information in the** `/etc/printcap` **file to the** `/etc/printers.conf` **file.**

   ```
   # /usr/lib/print/conv_lpd
   ```

Administering Printers (Tasks)

This chapter provides the procedures to administer printers. This is a list of the step-by-step instructions in this chapter.

For overview information about printing and the LP print service, see Chapter 2.

Managing Printers and the Print Scheduler

This section provides instructions for day-to-day tasks you perform to manage printers and the print scheduler.

Deleting Printers and Printer Access

If a printer needs to be replaced or you want to move the printer to a different location, you must delete the printer information from the LP print service before you physically remove it from the print server. You should also make sure that all the current print requests on the printer are printed or moved to another printer to be printed.

Not only does the printer information need to be deleted from the print server, but it also needs to be deleted from the print clients or network name service. If you delete a local printer from a print server, you should delete the remote printer entry from the print clients or network name service. If you move a printer to another print server, you need to delete the old remote print entry from the print clients or network name service and add access to the remote printer in its new location.

See "How to Delete a Printer and Remote Printer Access" on page 74 for detailed information on how to delete a local and remote printer. You can use Solaris Print Manager to delete a local or remote printer; however, Solaris Print Manager does not enable you to move queued print requests to another printer.

▼ How to Delete a Printer and Remote Printer Access

1. **Log in as superuser or lp on a print client that has access to the printer you want to delete.**

2. **Delete information about the printer from the print client.**

   print-client# **lpadmin -x** *printer-name*

-x	Deletes the specified printer.
printer-name	Name of the printer you want to delete.

Information for the specified printer is deleted from the print client's `/etc/lp/printers` directory.

3. **If the print client does not use another printer on the same print server, delete information about the print server from the print client.**

   print-client# **lpsystem -r** *print-server*

-r	Removes the specified print server.
print-server	Name of the print server you want to delete.

   The print server is deleted from the print client's `/etc/lp/Systems` file.

4. **Repeat step 2 through step 3 on each print client that has access to the printer.**

5. **Log in as superuser or lp on the print server.**

6. **Stop accepting print requests on the printer.**

   print-server# **reject** *printer-name*

reject *printer-name*	Rejects print requests for the specified printer.

   This step prevents any new requests from entering the printer's queue while you are in the process of removing the printer. See "How to Accept or Reject Print Requests for a Printer" on page 94 for a detailed description.

7. **Stop the printer.**

   print-server# **disable** *printer-name*

   This step stops print requests from printing. See "How to Enable or Disable a Printer" on page 96 for a detailed description on how to stop printing.

8. **Move any print requests that are still in the queue to another printer.**

   See "How to Move Print Requests to Another Printer" on page 100 for a detailed description on how to move print requests to another printer.

9. **Delete the printer from the print server.**

   print-server# **lpadmin -x** *printer-name*

   Configuration information for the printer is deleted from the print server's `/etc/lp/printers` directory.

10. **Delete information about the print clients that were using the printer you just deleted, unless they are still using another printer on the print server.**

    print-server# **lpsystem -r** *print-client1* [, *print-client2* . . .]

-r	Removes the specified print client.
print-client	Name of the print client you want to delete from the print server. You can specify multiple print clients in this command. Use a space or a comma to separate print client names. If you use spaces, enclose the list of print clients in quotes.

    The specified print clients are deleted from the print server's /etc/lp/Systems file.

11. **Verify the printer information has been deleted.**

    a. **Check the printer information has been deleted on the print client.**

       print-client$ **lpstat -p** *printer-name* **-l**

       You should receive an error indicating that the printer does not exist in the output of the above command.

    b. **Check the printer information has been deleted on the print server.**

       print-server$ **lpstat -p** *printer-name* **-l**

       You should receive an error indicating that the printer does not exist in the output of the above command.

Example—Deleting a Printer and Remote Printer Access

In the following example, the commands delete the printer luna from the print client terra and from the print server jupiter, and also delete the print client terra from the print server.

```
terra# lpadmin -x luna
Removed "luna".
terra# lpstat -p luna -l
jupiter# lpadmin -x luna
jupiter# lpsystem -r terra
Removed "terra".
jupiter# lpstat -p luna -l
```

Checking Printer Status

Many routine printer administration tasks require information about the status of the LP print service or a specific printer. For example, you can determine which printers

are available for use and examine the characteristics of those printers. You can use the `lpstat` command to find out status information about the LP print service or a specific printer.

▼ How to Check the Status of Printers

1. **Log in on any system on the network.**

2. **Check the status of printers by using the `lpstat` command.**

   Only the most commonly used options are shown here. See `lpstat(1)` for other options.

   $ **lpstat** [**-d**] [**-p** *printer-name* [**-D**] [**-l**]] [**-t**]

`-d`	Shows the system's default printer.
`-p` *printer-name*	Shows if a printer is active or idle, when it was enabled or disabled, and whether it is accepting print requests.
	You can specify multiple printer names with this command. Use a space or a comma to separate printer names. If you use spaces, enclose the list of printer names in quotes. If you don't specify *printer-name*, the status of all printers is displayed.
`-D`	Shows the description of the specified *printer-name*.
`-l`	Shows the characteristics of the specified *printer-name*.
`-t`	Shows status information about the LP print service, including the status of all printers: whether they are active and whether they are accepting print requests.

Examples—Checking the Status of Printers

In the following example, the command requests the name of the system's default printer.

```
$ lpstat -d
system default destination: luna
```

In the following example, the command requests the status of the printer `luna`.

```
$ lpstat -p luna
printer luna is idle. enabled since Jul 12 11:17 2001. available.
```

In the following example, the command requests a description of the printers `asteroid` and `luna`.

```
$ lpstat -p "asteroid luna" -D
printer asteroid faulted. enabled since Jul 12 11:35 2001. available.
unable to print: paper misfeed jam

Description: Printer by break room
printer luna is idle. enabled since Jul 12 11:36 2001. available.
Description: Printer by server room.
```

In the following example, the command requests the characteristics of the printer luna.

```
$ lpstat -p luna -l
printer luna is idle. enabled since Thu Jul 12 15:02:32 ...
        Form mounted:
        Content types: postscript
        Printer types: PS
        Description:
        Connection: direct
        Interface: /usr/lib/lp/model/standard
        After fault: continue
        Users allowed:
                (all)
        Forms allowed:
                (none)
        Banner not required
        Character sets:

        Default pitch:
        Default page size: 80 wide 66 long
        Default port settings:
```

Restarting the Print Scheduler

The print scheduler, lpsched, handles print requests on print servers. However, there might be times when the print scheduler stops running on a system, so print requests stop being accepted or printed.

To restart the print scheduler, you can use the /usr/lib/lp/lpsched command. If a print request was printing when the print scheduler stopped running, the print request will be printed in its entirety when you restart the print scheduler.

▼ How to Stop the Print Scheduler

1. **Log in as superuser or lp on the print server.**

2. **Check to see if the print scheduler is running.**

   ```
   # lpstat -r
   ```

If the print scheduler is not running, the message `scheduler is not running` is displayed.

3. **If the print scheduler is running, stop it.**

   # **/usr/lib/lp/lpshut**

▼ How to Restart the Print Scheduler

1. **Log in as superuser or lp on the print server.**

2. **Check to see if the print scheduler is running.**

   # **lpstat -r**

   If the print scheduler is not running, the message `scheduler is not running` is displayed.

3. **If the print scheduler is not running, start it.**

   # **/usr/lib/lp/lpsched**

Setting or Resetting Miscellaneous Printer Definitions

This section provides step-by-step instructions on setting or resetting printer definitions. Some of the following printer definitions can be set using Solaris Print Manager. The procedures below use the LP commands to quickly set or reset printer definitions.

▼ How to Add a Printer Description

1. **Log in as superuser or lp on the print server.**

2. **Add a printer description by using the** lpadmin**(1M) command.**

   # **lpadmin -p** *printer-name* **-D** *"comment"*

   -p *printer-name*   Name of the printer for which you are adding a description.

-D *"comment"*	Specifies the characteristics of the printer, such as location or administrative contact. Enclose characters that the shell might interpret (like `*`, `?`, `\`, `!`, `^`) in single quotation marks.

The printer description is added in the print server's
`/etc/lp/printers/`*printer-name*`/comment` file.

3. **Verify the** `Description` **information is correct.**

   `$ `**`lpstat -p`** *printer-name* **`-l`**

Example—Adding a Printer Description

In the following example, the command adds a printer description for the printer
`luna`.

`# `**`lpadmin -p luna -D "Nathans office"`**

Setting Up a Default Printer Destination

You can specify a default printer destination for a user so the user doesn't need to type
the printer name when using the print commands. Before you can designate a printer
as the default, the printer must be known to the print service on the system. You can
set a user's default printer destination by setting any of the following:

- `LPDEST` environment variable
- `PRINTER` environment variable
- The `_default` variable in the user's `.PRINTERS` file
- System's default printer (by using the `lpadmin -d` command or Solaris Print
  Manager

When an application provides a printer destination, that destination is used by the
print service, regardless of whether you have set a system's default printer
destination. If an application doesn't provide a printer destination or if you don't
provide a printer name when using a print command, the print command searches for
the default printer in a specific order. The table below shows the search order for a
system's default printer destination.

TABLE 5–1 Search Order for Default Printer Destinations

Search Order	Using `/usr/bin/lp` Command	Using SunOS/BSD Compatibility Commands (`lpr`, `lpq`, and `lprm`)
First	`LPDEST` variable	`PRINTER` variable

TABLE 5–1 Search Order for Default Printer Destinations   *(Continued)*

Search Order	Using /usr/bin/lp **Command**	**Using SunOS/BSD Compatibility Commands (**lpr, lpq, **and** lprm**)**
Second	PRINTER variable	LPDEST variable
Third	System's default printer	System's default printer

▼ How to Set a System's Default Printer

1. **Log in as superuser or lp on the system for which you want to set a default printer.**

2. **Set the system's default printer by using the** lpadmin **command.**

   # **lpadmin -d** [*printer-name*]

-d *printer-name*	Name of the printer you are assigning as the system's default printer. If you don't specify *printer-name*, the system is set up with no default printer.

   The default printer name is entered in the system's /etc/lp/default file.

3. **Check the system's default printer by using the** lpstat **command.**

   $ **lpstat -d**

Example—Setting a System's Default Printer

In the following example, the command sets the printer luna as the system's default printer. This means that luna will be used as the system's default printer if the LPDEST or PRINTER environment variables are not set.

```
# lpadmin -d luna
# lpstat -d
system default destination: luna
```

Printing Banner Pages

A banner page identifies who submitted the print request, the print request ID, and when the request was printed. A banner page will also have a modifiable title to help users identify their printouts.

Banner pages make identifying the owner of a print job easy, which is especially helpful when many users submit jobs to the same printer. Printing banner pages uses

more paper, however, and might not be necessary if a printer has only a few users. In some cases, printing banner pages is undesirable. For example, if a printer has special paper or forms mounted, like paycheck forms, printing banner pages might cause problems.

By default, the print service forces banner pages to be printed. However, you can give users a choice to turn off printing of a banner page when they submit a print request. You can set this choice through the lpadmin command or through Solaris Print Manager. If you give the users a choice, they have to use the -o nobanner option to turn off printing of a banner page.

Also, you can turn off banner pages for a printer so they are never printed. This is important if you have a situation where you don't need or want banner pages. You can turn off banner page printing by using the lpadmin command.

TABLE 5–2 Banner Page Printing

With This Command	Banner Page Printing Is ...	Override?
lpadmin -p *printer* -o banner or lpadmin -p *printer* -o banner=always	Required and printed	If you are a regular user and use p -o nobanner, the request is printed, but the nobanner argument is ignored. If you are root or lp, the nobanner argument is honored.
lpadmin -p *printer* -o nobanner lpadmin -p *printer* -o banner=optional	On by default, but can be disabled on a per request basis with the lp -o nobanner command.	N/A
lpadmin -p *printer* -o banner=never	Disabled	No

For step-by-step command-line instructions, see "How to Turn Off Banner Pages" on page 83.

▼ How to Make Banner Pages Optional

1. **Log in as superuser or lp on the print server.**

2. **Make banner pages optional by using the** `lpadmin` **command.**

   # **`lpadmin -p`** *printer-name* **`-o nobanner=optional`**

`-p` *printer-name*	Name of the printer for which you are making banner pages optional.
`-o nobanner=optional`	Enables users to specify no banner page when they submit a print request.

   If you want to force a banner page to print with every print request, specify the `-o banner=always` option.

   The banner page setting is entered in the print server's `/etc/lp/printers/`*printer-name*`/configuration` file.

3. **Verify the output from the following command contains the line** `Banner not required`**.**

   $ **`lpstat -p`** *printer-name* **`-l`**

   ## Example—Making Banner Pages Optional

   In the following example, the command enables users to request no banner page on the printer `luna`.

   # **`lpadmin -p luna -o nobanner=optional`**

▼ How to Turn Off Banner Pages

1. **Log in as superuser or lp on the print server.**

2. **Turn off banner printing by using the** `lpadmin` **command.**

   `lpadmin -p` *printer-name* **`-o nobanner=never`**

`-p` *printer-name*	Name of the printer for which you are making banner pages optional.
`-o nobanner=never`	Disables banner page printing under all circumstances.

   The banner page setting is entered in the print server's `/etc/lp/printers/`*printer-name*`/configuration` file.

3. **Verify the output from the following command contains the line** `Banner not printed`**.**

   $ **`lpstat -p`** *printer-name* **`-l`**

4. **Submit a print request to the printer to make sure a banner page does not print.**

Example—Turning Off Printing Banner Pages

In the following example, the command disables printing banner pages on the printer `luna`.

`lpadmin -p luna -o nobanner=never`

Setting Up Printer Classes

The print service enables you to group several locally attached printers into one class. You can perform this task only by using the `lpadmin -c` command.

When you have set up a printer class, users can then specify the class (rather than individual printers) as the destination for a print request. The first printer in the class that is free to print is used. The result is faster turnaround because printers are kept as busy as possible.

There are no default printer classes known to the print service; printer classes exist only if you define them. Here are some ways you could define printer classes:

- By printer type (for example, PostScript)
- By location (for example, 5th floor)
- By work group or department (for example, Accounting)

Alternatively, a class might contain several printers that are used in a particular order. The LP print service always checks for an available printer in the order in which printers were added to a class. Therefore, if you want a high-speed printer to be accessed first, you would add it to the class before you add a low-speed printer. As a result, the high-speed printer would handle as many print requests as possible. The low-speed printer would be reserved as a backup printer when the high-speed printer is in use.

Note – Print requests are balanced between printers in a class only for local printers.

Class names, like printer names, must be unique and can contain a maximum of 14 alphanumeric characters and underscores.

You are not obligated to define printer classes. You should add them only if you determine that using printer classes would benefit users on the network.

▼ How to Define a Class of Printers

1. **Log in as superuser or lp on the print server.**

2. **Define a class of printers by using the `lpadmin` command.**

   # **`lpadmin -p`** *printer-name* **`-c`** *printer-class*

`-p` *printer-name*	Name of the printer you are adding to a class of printers.
`-c` *printer-class*	Name of a class of printers.

   The specified printer is added to the end of the list in the class in the print server's `/etc/lp/classes/`*printer-class* file. If the printer class does not exist, it is created.

3. **Verify the printers in a printer class by using the `lpstat` command.**

   $ **`lpstat -c`** *printer-class*

Example—Defining a Class of Printers

In the following example, the command adds the printer `luna` in the class `roughdrafts`.

   # **`lpadmin -p luna -c roughdrafts`**

Setting Up Printer Fault Alerts

If you choose, the print service can notify you when it detects a printer fault. You can select any of the following methods to receive printer fault notification with the `lpadmin -A` command or with Solaris Print Manager:

- Write a message to the terminal on which root is logged in
- Electronic mail to root
- No notification

However, the `lpadmin -A` command offers you an additional option of receiving a message specified by the program of your choice. It also enables you to selectively turn off notification for an error that you already know about.

Unless you specify a program to deliver fault notification, the content of the fault alert is a predefined message that says the printer has stopped printing and needs to be fixed.

The table below lists the alert values that you can set for a printer with the `lpadmin -A` command. These alert values can also be set for print wheels, font cartridges, and forms.

TABLE 5–3 Values for Printing Problem Alerts

Value for `-A` alert	Description
`'mail` [*user-name*]`'`	Send the alert message by email to root or `lp` on the print server, or the specified *user-name*, which is a name of a user.
`'write` [*user-name*]`'`	Send the alert message to the root or `lp` console window on the print server, or to the console window of the specified *user-name*, which is a name of a user. The specified user must be logged in to the print server to get the alert message.
`'`*command*`'`	Run the *command* file for each alert. The environment variables and current directory are saved and restored when the file is executed.
`quiet`	Stop alerts until the fault is fixed. Use this when you (root or specified user) receive repeated alerts.
`none`	Do not send any alerts. This is the default if you don't specify fault alerts for a printer.

▼ How to Set Fault Alerts for a Printer

1. **Log in as superuser or lp on the print server.**

2. **Set fault alerts for a printer with the** `lpadmin` **command.**

   # **`lpadmin -p`** *printer-name* **`-A`** *alert* [**`-W`** *minutes*]

`-p` *printer-name*	Name of the printer for which you are specifying an alert for printer faults.
`-A` *alert*	Specifies what kind of alert will occur when the printer faults. See Table 5–3 for detailed information about the valid values for *alert*. Some valid values are `mail`, `write`, and `quiet`.
`-W` *minutes*	Specifies how often (in minutes) the fault alert will occur. If you don't specify this option, the alert is sent once.

The fault alert setting is entered in the print server's
`/etc/lp/printers/`*printer-name*`/alert.sh` file.

3. **Check the information following the** `On fault` **heading from the output of the following command.**

   $ **lpstat -p** *printer-name* **-l**

Examples—Setting Fault Alerts for a Printer

In the following example, the command sets up the printer `mars` to send fault alerts by email to a user named `joe`, with reminders every 5 minutes.

lpadmin -p mars -A 'mail joe' -W 5

In the following example, the command sets up the printer `venus` to send fault alerts to the console window, with reminders every 10 minutes.

lpadmin -p venus -A write -W 10

In the following example, the command stops fault alerts for the printer `mercury`.

lpadmin -p mercury -A none

In the following example, the command stops fault alerts until the printer `venus` has been fixed.

lpadmin -p venus -A quiet

Setting Up Printer Fault Recovery

If you choose not to send any fault notification, you can find out about printing faults so you can correct the problem. The LP print service will not continue to use a printer that has a fault. In addition to alerts for printer faults, you can also provide alerts that tell the system administrator to mount print wheels, font cartridges, and forms when print requests require them.

You can define the fault recovery options for a printer only by using the `lpadmin -F` command. This task is not available in Solaris Print Manager.

Printer faults can be as simple as running out of paper or needing to replace a toner cartridge. Other more serious problems can include complete printer failure or power failure. After you fix a printer fault, the print request that was active when the fault occurred begins printing in one of three ways:

- Starts printing from the beginning
- Continues printing from the top of the page where printing stopped

- After you enable the printer, continues printing from the top of the page where the printing stopped

A print filter is required to continue printing from the top of a page where the printing stopped. A print filter records the control sequences used by the printer to track page boundaries, which the default filters used by the print service cannot do. You will be notified by the print service if recovery cannot proceed with the specified print filter. For information about writing filters, see "How to Create a New Print Filter" on page 152.

If you want printing to resume immediately after a printer fault is fixed, enable the printer by using the `enable` command.

The table below lists the fault recovery values you can set for a printer with the `lpadmin -F` command.

TABLE 5–4 Values for Printer Fault Recovery

Value for `-F` *recover-options*	Description
`beginning`	After a fault recovery, printing restarts from the beginning of the file.
`continue`	After a fault recovery, printing starts at the top of the page where the printing stopped. This recovery option requires a print filter.
`wait`	After a fault recovery, printing stops until you enable the printer. After you enable the printer (`enable` command), printing starts at the top of the page where printing stopped. This recovery option requires a print filter.

▼ How to Set Printer Fault Recovery

1. **Log in as superuser or lp on the print server.**

2. **Set up fault recovery for the printer with the `lpadmin`(1M) command.**

   # **lpadmin -p** *printer-name* **-F** *recovery-options*

`-p` *printer-name*	Name of the printer for which you are specifying fault recovery.
`-F` *recovery-options*	One of the three valid recovery options:
	`beginning`, `continue`, or `wait`.
	See Table 5–4 for detailed information about the valid values for *recovery-options*.

The fault recovery setting is entered in the print server's
/etc/lp/printers/*printer-name*/configuration file.

3. **Check the information following the** `After fault` **heading in the output of the following command.**

   $ **`lpstat -p`** *printer-name* **`-l`**

Example—Setting Printer Fault Recovery

In the following example, the command sets up the printer `luna` to continue printing at the top of the page where printing stopped.

`lpadmin -p luna -F continue`

Limiting User Access to a Printer

You can control which users can access some or all of the available printers. For example, you can prevent some users from printing on a high-quality printer to minimize expense. To restrict user access to printers, you can create `allow` and `deny` lists using the `lpadmin -u` command on the print server. (Solaris Print Manager enables you to create only allow lists.) If you create neither, a printer is available to all users who can access the printer.

An allow list contains the names of users allowed access to the specified printer; a deny list contains the names of users denied access to the specified printer.

The rules for allow and deny lists are:

When You ...	Then ...
Do not create allow and deny lists, or if you leave both lists empty	All users can access the printer.
Specify `all` in the allow list	All users can access the printer.
Specify `all` in the deny list	All users, except root and lp (on the server), are denied access to the printer.
Make any entry in the allow list	The deny list is ignored. Only those users who are listed can access the printer.
Create a deny list, but you do not create an allow list or you leave the allow list empty	Users who are listed in the deny list are denied access to the printer.

Because the print server is actually controlling access to the printer, allow and deny lists can only be created on the print server itself. If you create allow and deny lists, the print server will exclusively control user access to printers.

The table below lists the values you can add to an allow or deny list to limit user access to a printer.

TABLE 5–5 Values for Allow and Deny Lists

Value for *user-list*	Description
user	*User* on any system
all	All users on all systems
none	No user on any system
system!*user*	*User* on *system* only
!*user*	*User* on local system only
all!*user*	*User* on any system
all!all	All users on all systems
system!all	All users on *system*
!all	All users on local system

▼ How to Limit User Access to a Printer

1. **Log in as superuser or lp on the print server.**

2. **Allow or deny users access to a printer.**

   # **lpadmin -p** *printer-name* **-u allow:***user-list* [**deny:***user-list*]

-p *printer-name*	Name of the printer to which the allow or deny user access list applies.
-u allow:*user-list*	User names to be added to the allow user access list. You can specify multiple user names with this command. Use a space or a comma to separate names. If you use spaces, enclose the list of names in quotes.
	Table 5–5 provides the valid values for *user-list*.

-u deny:*user-list*	User names to be added to the deny user access list. You can specify multiple user names with this command. Use a space or a comma to separate names. If you use spaces, enclose the list of names in quotes.
	Table 5–5 provides the valid values for *user-list*.

The specified users are added to the allow or deny user access list for the printer in one of the following files on the print server:

/etc/lp/printers/*printer-name*/users.allow

/etc/lp/printers/*printer-name*/users.deny

Note – If you specify none as the value for *user-list* in the allow user access list, the following files are not created for the print server:

/etc/lp/printers/*printer-name*/alert.sh

/etc/lp/printers/*printer-name*/alert.var

/etc/lp/printers/*printer-name*/users.allow

/etc/lp/printers/*printer-name*/users.deny

---

3. **Check the information following the** Users allowed **or** Users denied **heading in the output of the following command.**

   $ **lpstat -p** *printer-name* **-l**

Examples—Limiting User Access to a Printer

In the following example, the command allows only the users nathan and george access to the printer luna.

lpadmin -p luna -u allow:nathan,george

In the next example, the command denies the users nathan and george access to the printer asteroid.

lpadmin -p asteroid -u deny:"nathan george"

Managing Print Requests

When a user submits a print request from a print client, the print request is added to a queue on the print server before it is sent to the printer. While a print request is in the queue, you can cancel or gain status information on the request from a client system. You must login to the print server to move, hold, resume, or change the priorities of print requests with LP commands. These actions can help you keep printing services operating smoothly.

The table below lists the values for changing the priority of a print request with the `lp -H` command.

TABLE 5–6 Values for Changing the Priority of a Print Request

Value for `-H` *change-priority*	Description
hold	Places the print request on hold until you cancel it or instruct the LP print service to resume printing the request.
resume	Places a print request that has been on hold back in the queue. It will be printed according to its priority and placement in the queue. If you put a hold on a print job that is already printing, `resume` puts the print request at the head of the queue so it becomes the next request printed.
immediate	Places a print request at the head of the queue. If a request is already printing, you can put it on hold to allow the next request to print immediately.

▼ How to Check the Status of Print Requests

1. **Log in on any system on the network.**

2. **Check the status of printers and print requests**

   Only the most commonly used options are shown here. See `lpstat(1)` for other valid options.

   ```
   $ lpstat -o [list] | -u [user-list]
   ```

-o *list*	Shows the status of print requests on a specific printer. *list* can be one or more printer names, printer class names, or print request IDs.
	You can specify multiple printer names, class names, and IDs for *list*. Use a space or a comma to separate values. If you use spaces, enclose the list of values in quotes.
	If you don't specify *list*, the status of print requests to all printers is displayed.
-u *user-list*	Shows the status of print requests for a specific user. *user-list* can be one or more user names.
	You can specify multiple users with this command. Use a space or a comma to separate user names. If you use spaces, enclose the list of names in quotes.
	If you don't specify *user-list*, the status of print requests for all users is displayed.

When used to check the status of print requests, the lpstat command displays one line for each print request. From left to right, the line shows the request ID, the user, the output size in bytes, the date and time of the request, and information about the request, such as "being filtered."

Examples—Checking the Status of Print Requests

In the following example, the command shows that user fred has one print request queued to the printer luna.

```
$ lpstat
luna-1    fred     1261    Jul 12 17:34
```

In the following example, the command shows that the user paul currently has no print requests in queue.

```
$ lpstat -u paul
```

In the following example, the command shows that there are two print requests on the printer moon.

```
$ lpstat -o moon
moon-78    root     1024    Jul 14 09:07
moon-79    root     1024    Jul 14 09:08
```

Starting or Stopping Printing

The `enable` and `disable` commands control whether a printer prints or stops printing requests that are in the print queue. When you disable a printer, the printer stops printing requests in queue; however, requests are still added to the queue. (You must set the printer to reject print requests so requests are not added to the queue. See "Accepting or Rejecting Print Requests" on page 95 for information about rejecting print requests.)

A printer is enabled to print and accepts print requests when it is added using Solaris Print Manager. Solaris Print Manager doesn't provide any additional printer processing management.

You must enable the printer whenever it has been disabled, which can happen when a printer fault occurs. When you enable a printer, it prints requests from the print queue until the queue is empty, even if the print service rejects additional requests for the print queue.

The figure below shows the point at which processing of print requests is interrupted when a printer is disabled.
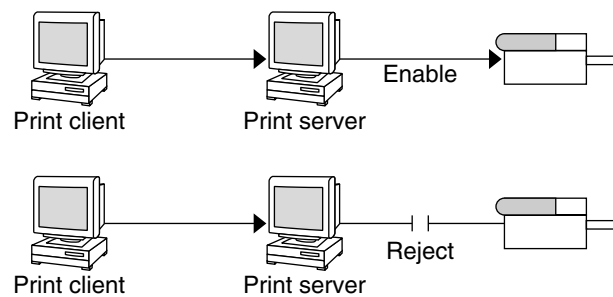
FIGURE 5–1 What Happens When a Printer Is Enabled or Disabled

▼ How to Accept or Reject Print Requests for a Printer

1. **Log in as superuser or lp on the print server.**

2. **Stop accepting print requests for the printer command.**

   # **reject** [**-r** "*reason*"] *printer-name*

   -r "*reason*"   Provides users a reason why the printer is rejecting print
                                 requests. The reason is stored and displayed whenever a user
                                 checks on the status of the printer (lpstat -p).

   printer-name   Name of the printer that will stop accepting print requests.

   The queued requests will continue printing as long as the printer is enabled. For
   instructions on disabling a printer so it stops printing, see "How to Enable or Disable a
   Printer" on page 96.

3. **Start accepting print requests for the printer.**

   # **accept** *printer-name*

4. **Check the status of the printer to see whether it is accepting or rejecting print
   requests.**

   $ **lpstat -p** *printer-name*

Examples—Accepting or Rejecting Print Requests for a Printer

In the following example, the command stops the printer luna from accepting print
requests.

```
# reject -r "luna is down for repairs" luna
destination "luna" will no longer accept requests
```

In the following example, the command sets the printer luna to accept print requests.

```
# accept luna
destination "luna" now accepting requests
```

Accepting or Rejecting Print Requests

The accept and reject commands enable you to turn on or turn off a print queue
that stores requests to be printed.

When you use the reject command, the print queue for a specified printer is turned
off—no new print requests can enter the queue on the print server. All print requests
that are in the queue are still printed. You must disable the printer if you want it to

stop printing requests that are already in the queue. Table 5–7 compares the functions of the accept, reject, enable, and disable commands.

TABLE 5–7 Functions of accept/reject and enable/disable Commands

Command	Function
accept	Accept print requests that are sent to the print queue.
enable	Print the requests that are in the print queue.
reject	Reject print requests that are sent to the print queue.
disable	Stop printing requests that are currently in the print queue.

See "Starting or Stopping Printing" on page 94 for information about disabling a printer.

If a print request is rejected, the print service writes or mails a message to the user who submitted the request, saying that print requests are not being accepted for the specified printer.

You can also specify a reason for not accepting requests through the command line. The reason is displayed on users' systems when one tries to check the printer's queue. The figure below shows the point at which processing of print requests is interrupted when a print queue rejects print requests.
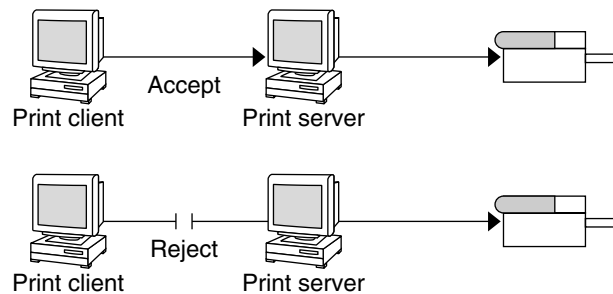
FIGURE 5–2 What Happens When a Print Queue Accepts or Rejects Requests

▼ How to Enable or Disable a Printer

1. **Log in as superuser or** lp **on the print server.**

2. **Stop printing print requests on the printer by using the** `disable` **command.**

   # **disable** [**-c** | **-W**] [**-r** **"***reason***"**] *printer-name*

`disable`	Cancels the current job, then disables the printer. The current job is saved to reprint when the printer is enabled.
`-c`	Cancels the current job, then disables the printer. The current job is not printed later.
`-W`	Waits until the current job is finished before disabling the printer.
`-r` *"reason"*	Provides users with a reason why the printer is disabled. The reason is stored and displayed whenever a user checks on the status of the printer (`lpstat -p`).
printer-name	Name of the printer that will stop printing print requests.

   Note – You cannot enable or disable classes of printers. Only individual printers can be enabled or disabled.

   ---

3. **Start printing print requests on the printer by using the** `enable` **command.**

   # **enable** *printer-name*

4. **Verify the printer is enabled.**

   $ **lpstat -p** *printer-name*

Examples—Enabling or Disabling a Printer

In the following example, the command stops the current job on the printer `luna`, saves it to print later, and provides a reason why the printer has stopped printing print requests.

disable -r "changing the form" luna

In the following example, the command starts printing print requests on the printer luna.

```
# enable luna
printer "luna" enabled
```

Canceling a Print Request

You can use the cancel(1) to cancel print requests from printer queues or to cancel jobs that are printing. There are three ways to use the cancel command:

- Cancel requests by request identification number (request ID)
- Cancel requests from a specific user on all, or specified, printers
- Cancel the job currently printing

When you use cancel, a message tells you the request(s) are canceled, and the next request in queue is printed. You can cancel a print request only if you are:

- The user who submitted the request and you are logged in on the system from which you submitted the request

- The user who submitted the request on any client system and the print server has the "user-equivalence" option configured for the printer in it's /etc/printers.conf file.

- Logged in as superuser or lp on the print server.

To cancel a specific request, you need to know its request ID. The request ID is comprised of the name of the printer, a dash, and the number of the print request—for example, luna-185. When you submit the print request, the request ID is displayed. If you do not remember the print request ID, you can find it by using the lpstat command with the -o printer option.

▼ How to Cancel a Print Request

1. **If you are going to cancel print requests of other users, become superuser or lp.**

2. **Determine the request IDs of the print requests to cancel by using the lpstat command.**
   See "How to Check the Status of Print Requests" on page 92 for more details.

3. **Cancel a print request by using the cancel command.**

   $ **cancel** *request-id* | *printer-name*

request-id	Request ID of a print request to be canceled. You can specify multiple request IDs with this command. Use a space or a comma to separate request IDs. If you use spaces, enclose the list of request IDs in quotes.

printer-name	Specifies the printer for which you want to cancel the currently printing print request.
	You can specify multiple printer names with this command. Use a space or a comma to separate printer names. If you use spaces, enclose the list of printer names in quotes.

4. **Verify the print requests are canceled.**

   $ **lpstat -o** *printer-name*

Examples—Canceling a Print Request

In the following example, the command cancels the `luna-3` and `luna-4` print requests.

```
$ cancel luna-3 luna-4
request "luna-3" cancelled
request "luna-4" cancelled
```

In the following example, the command cancels the print request that is currently printing on the printer `luna`.

```
# cancel luna
request "luna-9" cancelled
```

▼ How to Cancel a Print Request From a Specific User

1. **(Optional) Become superuser or lp if you are going to cancel print requests of other users.**

2. **Cancel a print request from a specific user with the** `cancel` **command.**

   $ **cancel -u** *user-list* [*printer-name*]

-u *user-list*	Cancels the print request for a specified user.
	user-list can be one or more user names. Use a space or a comma to separate user names. If you use spaces, enclose the list of names in quotes.

printer-name	Specifies the printer for which you want to cancel the specified user's print requests.
	printer-name can be one or more printer names. Use a space or a comma to separate printer names. If you use spaces, enclose the list of printer names in quotes.
	If you don't specify *printer-name*, the user's print requests will be canceled on all printers.

Examples—Canceling a Print Request From a Specific User

In the following example, the command cancels all the print requests submitted by the user george on the printer luna.

```
# cancel -u george luna
request "luna-23" cancelled
```

In the following example, the command cancels all the print requests submitted by the user george on all printers.

```
# cancel -u george
request "asteroid-3" cancelled
request "luna-8" cancelled
```

Moving a Print Request

If you plan to change the way a printer is used or decide to take a printer out of service, you should set up the LP print service to reject additional print requests, and then move or cancel any requests that are currently queued to the printer. You can use the lpmove(1M) command to move individual or all print requests to another local printer.

Request IDs are not changed when you move print requests, so users can still find their requests. Print requests that have requirements (such as file content type or forms) that cannot be met by the newly specified printer cannot be moved; they must be canceled.

▼ How to Move Print Requests to Another Printer

To move all print requests from one printer to another, you do not need to know the request IDs; however, it is a good idea to see how many print requests are affected before you move them.

1. **Log in as superuser or lp on the print server.**

2. **(Optional) Check the request IDs of the print requests on the original printer.**

   # **lpstat -o** *printer-name1*

3. **(Optional) Check if the destination printer is accepting print requests.**

   # **lpstat -p** *printer-name2*

   -p *printer-name2*   Name of the printer to which you are moving the print requests.

4. **Move all the print requests from the original printer to the destination printer.**

   # **lpmove** *printer-name1 printer-name2*

   printer-name1   Name of the printer from which all print requests will be moved.

   printer-name2   Name of the printer to which all print requests will be moved.

   If some requests cannot be printed on the destination printer, they are left in the original printer's queue. By using request IDs, you can also move specific print requests to another printer with the lpmove command.

5. **Start accepting print requests on the original printer.**
   If you move all the print requests to another printer, the lpmove command automatically stops accepting print requests for the printer. This step is necessary if you want to begin accepting new print requests for the printer.

   # **accept** *printer-name1*

6. **Check for any remaining print requests in the original printer's queue by using the following command.**

   $ **lpq -P** *printer-name1*

   Make sure all specified print requests were moved to the destination printer's queue by using the following command.

   $ **lpq -P** *printer-name2*

Example—Moving Print Requests to Another Printer

In the following example, the lpmove command moves print requests from the printer luna to the printer terra, and the accept command tells the original printer luna to resume accepting print requests.

```
# lpmove luna terra
# accept luna
```

Changing the Priority of Print Requests

After a user has submitted a print request, you can change its priority in the print server's queue by:

- Putting any print request on hold if it has not finished printing. Putting a request on hold stops it, if it is currently printing, and keeps it from printing until you resume printing it. Other print requests go ahead of the on-hold request.

- Moving any print request to the head of the queue, where it will be the next job eligible for printing. If you want a job to start printing immediately, you can interrupt the job that is currently printing by putting it on hold.

- Changing the priority of a job still waiting to be printed, moving it in the queue so it is ahead of lower priority requests and behind requests at the same level or at a higher priority.

▼ How to Change the Priority of a Print Request

1. **Log in as superuser or lp on the print server that is holding the print request.**

2. **Determine the request IDs of the print requests whose priority you want to change by using the** lpstat **command.**
   See "How to Check the Status of Print Requests" on page 92 for more information.

3. **Change the priority of a print request by using the** lp **command.**

   ```
   # lp -i request-id -H change-priority
   ```

-i *request-id*	Request ID of a print request you want to change.
	You can specify multiple request IDs with this command. Use a space or a comma to separate request IDs. If you use spaces, enclose the list of request IDs in quotes.
-H *change-priority*	One of the three ways to change the priority of a print request: hold, resume, immediate.
	See Table 5–6 for detailed information about valid values for *change-priority*.

   You can also use the lp -q command to change the priority level of a specified print request. You can change the priority level from 0, the highest priority, to 39, the lowest

priority.

Example—Changing the Priority of a Print Request

In the following example, the command changes a print request with the request ID
`asteroid-79`, to priority level 1.

lp -i asteroid-79 -q 1

Managing Character Sets, Filters, Forms, and Fonts (Tasks)

This chapter provides background information and step-by-step instructions for setting up and administering character sets, print filters, forms, and fonts.

This is a list of the step-by-step instructions in this chapter.

For overview information about printing, see Chapter 2.

Managing Character Sets

Printers differ in the method they use to print text in various font styles. For example, PostScript printers treat text as graphics. These printers can generate text in different

fonts, and place the text in any position, size, or orientation on the page. Other types of printers support a more limited number of font styles and sizes, using either print wheels, font cartridges, or preprogrammed selectable character sets. Usually, only one of these printing methods applies to a given printer type.

Print wheels and font cartridges, from the perspective of the LP print service, are similar, because someone must intervene and mount the hardware on the printer, when needed. Character sets that require you to physically mount a wheel or cartridge are referred to as *hardware character sets*. Character sets that do not require hardware mounting, that come preprogrammed with the printer, and can be selected by a print request, are referred to as *software character sets*.

When you set up a non-PostScript printer, you need to tell the LP print service which print wheels or selectable character sets are available to users. When users submit print requests, the `lp -S` command enables them to specify a print wheel or selectable character set to use for the print job. Users do not have to know which type of character set applies; they just refer to the font style by the name you have defined. For example, you can define a print wheel as `gothic`. To request the `gothic` print wheel, the user would enter `lp -S gothic`.

Selectable Character Sets

The selectable character sets supported by a printer are listed in the `terminfo` entry for that printer. For example, the entry for the `ln03` printer is `/usr/share/lib/terminfo/l/ln03`. You can find the names of selectable character sets for any printer type in the `terminfo` database by using the `tput` command. The syntax for the `tput` command is:

```
tput -T printer-type csn
```

The cs*n* option is an abbreviation for character set number. The number starts with 0, which is always the default character set number after the printer is initialized. You can repeat the command, using `-1, -2 , -3 ,` and so on in place of the `-0` , to display the names of the other character sets. For each selectable character set, a `terminfo` name (for example, `usascii, english, finnish,` and so forth) is returned.

In general, the `terminfo` character set names should closely match the character set names used in the manufacturer's documentation for the printer. Because manufacturers do not all use the same character set names, the `terminfo` character set names can differ from one printer type to the next.

You do not have to register the selectable character set names with the LP print service. However, you can give them more meaningful names or aliases.

Note – If you do not specify the selectable character sets that can be used with a printer, the LP print service assumes that the printer can accept any character set name (such as cs0, cs1, or cs2) or the terminfo name known for the printer.

Users can use the lpstat -p -l command to display the names of the selectable character sets that you have defined for each printer on a print server.

Note – Character sets for PostScript printers are not listed when you use the lpstat -p -l command because the PostScript fonts are controlled by PostScript filters, not by entries in the terminfo database. See "Managing Fonts" on page 127 for information about how to administer PostScript fonts.

Hardware-Mounted Character Sets

Another method to obtain alternative character sets is to use removable print wheels or font cartridges that you physically attach, or mount, in a printer.

To administer hardware-mounted character sets, you inform the LP print service of the names you want to use for the available print wheels, and how you want to be alerted when a printer needs a different print wheel. Then, when a user requests a particular character set with the lp -S command, the scheduler sends an alert to mount the print wheel, and the print request is placed in the print queue. When you mount the correct print wheel and tell the LP print service that the print wheel is mounted, the job is printed. See "How to Unmount and Mount a Print Wheel or Font Cartridge" on page 109 for more information.

If you do not specify multiple print wheels or cartridges for a printer, the LP print service assumes that the printer has a single, fixed print wheel or cartridge, and users cannot specify a special print wheel or cartridge when using the printer.

Unlike selectable character sets, the names you use for print wheels or cartridges are not tied to entries in the terminfo database. Print wheel or cartridge names are used only for the purpose of communicating with the LP print service and its users.

The names you choose for print wheels or cartridges, however, should have meaning to the users; the names should refer to font styles. In addition, the names should be the same across printers that have similar print wheels or cartridges, or selectable character sets. That way, users can ask for a font style (character set) without regard to which printer—or even whether a print wheel or cartridges—or selectable character set will be used.

Of course, you and the printer users should agree on the meanings of print wheel or cartridge names. Otherwise, what a user asks for and what you mount, might not be the same character set.

Tracking Print Wheels

The procedure for tracking print wheels is similar to the procedure for tracking forms. Some printers (usually letter-quality printers) have removable print heads, such as print wheels or print cartridges, that provide a particular font or character set. A user can request a named character set. If that character set is not available, the LP print service notifies root of the request. The job is stored in the print queue until the print wheel is changed.

Alerts for Mounting Print Wheels or Cartridges

You request alerts for mounting print wheels or cartridges in the same way you request other alerts from the LP print service. See "Setting Up Printer Fault Alerts" on page 85 for general information about alerts.

▼ How to Define a Print Wheel or Font Cartridge

1. **Log in as superuser or lp on the print server.**

2. **Define a print wheel or font cartridge that can be used with the printer.**

   print-server# **lpadmin -p** *printer-name* **-S** *hard-charset1* [**,** *hard-charset2* **. . .**]

-p *printer-name*	Name of the printer for which you are defining a print wheel or font cartridge.
-S *hard-charset*	Hardware character set name of the print wheel or font cartridge.
	You can specify multiple hardware character sets with this command. Use commas or spaces to separate character set names. If you use spaces, enclose the list of character set names in quotes.
	Define names that are meaningful to users, and inform the users of the names.

   The print wheel or font cartridge definition is added in the print server's /etc/lp/printers/*printer-name*/configuration file.

3. **Log in as superuser or lp on a system that is a print client of the print server.**

4. **Define the same print wheel or font cartridge for the print client.**

   print-client# **lpadmin -p** *printer-name* **-S** *hard-charset1[,hard-charset2...]*

   In this command, the variables are the same as those in step 2.

   The print wheel or font cartridge definition is added in the print client's
   /etc/lp/printers/*printer-name*/configuration file.

5. **Repeat step 3 and step 4 for each print client that might need to use the print wheel or font cartridge.**

6. **Verify the information following the** Character sets **heading in the following output is correct on both the print server and the print client.**

   $ **lpstat -p** *printer-name* **-l**

Example—Defining a Print Wheel

In the following example, the command defines the pica print wheel on the printer luna for a print client named asteroid.

asteroid# **lpadmin -p luna -S pica**

▼ How to Unmount and Mount a Print Wheel or Font Cartridge

1. **Log in as superuser or lp on the print server.**

2. **Unmount the print wheel or font cartridge that is in the printer by using the** lpadmin **command.**

   # **lpadmin -p** *printer-name* **-M -S none**

-p *printer-name*	Printer on which you are unmounting a print wheel or font cartridge.
-M -S none	Specifies unmounting the current print wheel or font cartridge.

   The current print wheel or font cartridge is deleted from the print server's
   /etc/lp/printers/*printer-name*/configuration file.

3. **Remove the print wheel or font cartridge from the printer.**

4. **Put the new print wheel or font cartridge in the printer.**

5. **Mount the new print wheel or font cartridge by using the** `lpadmin` **command.**

   `# ` **`lpadmin -p`** *printer-name* **`-M -S`** *hard-charset*

`-p` *printer-name*	Printer on which you are mounting a print wheel or font cartridge.
`-M -S` *hard-charset*	Hardware character set name of the print wheel or font cartridge you want to mount.

   The print wheel or font cartridge is added in the print server's
   `/etc/lp/printers/`*printer-name*`/configuration` file. The mounted print wheel
   or font cartridge remains active until it is unmounted or until a new print wheel or
   font cartridge is mounted.

6. **Check the information under the** `Print wheels` **or** `Character set` **heading in the
   output of the following command. You should see the name of the print wheel or
   character set and the notation** `(mounted)`

   `$ ` **`lpstat -p`** *printer-name* **`-l`**

   ## Example—Unmounting and Mounting a Print Wheel

   In the following example, the commands unmount the current print wheel on the
   printer `luna` and mount the `pica` print wheel.

   ```
   # lpadmin -p luna -M -S none
   # lpadmin -p luna -M -S pica
   ```

▼ How to Set an Alert to Mount a Print Wheel or Font Cartridge

1. **Log in as superuser or lp on the print server.**

2. **Set an alert to mount a print wheel or font cartridge by using the** `lpadmin`**(1M)
   command.**

   `# ` **`lpadmin -S`** *hard-charset* `-A` *alert* [**`-Q`** *requests*] [**`-W`** *minutes*]

`-S` *hard-charset*	Hardware character set name of the print wheel or font cartridge for which you want to set an alert.

-A *alert*	Specifies what kind of alert will occur when a print wheel or font cartridge is requested. See Table 5–3 for detailed information about the valid values for *alert*. Some valid values are `mail`, `write`, and `quiet`.
	If you specify `mail` or `write`, a predefined alert message says to mount the specified print wheel or font cartridge and includes the names of one or more printers that have been set up to use such a print wheel or cartridge.
-Q *requests*	Specifies the number of print requests that require the print wheel or font cartridge that must be in the queue before an alert occurs. If you don't specify this option, only one print request in the queue triggers an alert.
-W *minutes*	Specifies how often (in minutes) the alert will occur. If you don't specify this option, the alert is sent only once.

The alert is added in the print server's /etc/lp/pwheels/*charset-name*/alert.sh file.

3. **Verify that the alert has been added for the print wheel or font cartridge by checking the output of the following command.**

   # **lpadmin -S** *hard-charset* **-A list**

   Otherwise, if you have set a low number of print requests to trigger the alert, submit enough print requests to meet the minimum requirement and make sure you receive an alert to mount the print wheel or font cartridge.

Examples—Setting an Alert to Mount a Print Wheel or Font Cartridge

In the following example, the command sets email alerts to occur every five minutes for the `elite` print wheel when there are ten print requests for `elite` in the print queue.

   # **lpadmin -S elite -A mail -Q 10 -W 5**

In the following example, the command sets email alerts to occur every minute for the `finnish` font cartridge when there are five print requests for `finnish` in the print queue.

   # **lpadmin -S finnish -A mail -Q 5 -W 1**

In the following example, the command sets console-window alerts to occur every 10 minutes for the `elite` print wheel when there are five print requests for `elite` in the print queue.

   # **lpadmin -S elite -A write -Q 5 -W 10**

In the following example, the command sets no alerts to occur for the elite print wheel.

```
# lpadmin -S elite -A none
```

▼ How to Set Up an Alias for a Selectable Character Set

Note – You do not need to perform this procedure if the terminfo(4) names for the selectable character sets are adequate. See "Adding a terminfo Entry for an Unsupported Printer" on page 135 for more information on using the terminfo database.

---

1. **Log in as superuser or lp on the print server.**

2. **Display the names of the selectable character sets for the specified printer type by using the** tput**(1) command.**

   ```
   # tput -T printer-type csn
   ```

-T *printer-type*	Printer type found in the terminfo database. See "Printer Type" on page 46 for information on entries in the terminfo database.
n	Number (0, 1, 2, 3, 4, 5, and so on) that represents a selectable character set for the specified printer type. The system displays the selectable character set name followed by the prompt symbol. For example, cs1 could cause the system to display english#.

3. **Set up an alias for a selectable character set.**

   ```
   # lpadmin -p printer-name -S select-charset1=alias1[,select-charset2=alias2...]
   ```

-p *printer-name*	Printer on which you are setting up aliases for selectable character sets.
-S *select-charset*	Selectable character set name for which to set an alias. The name can be found in step 2.

alias	Alias for the specified selectable character set. This alias can be used in addition to the selectable character set name.
	You can set up more than one alias with this command. Use commas or spaces to separate the aliases. If you use spaces, enclose the list of aliases in quotes.

The alias is added in the print server's
`/etc/lp/printers/`*printer-name*`/configuration` file.

4. **Log in as superuser or lp on a system that is a print client of the print server.**

5. **Set up an alias for the selectable character set.**

`# `**`lpadmin -p`** *printer-name* **`-S`** *select-charset1=alias1[,select-charset2=alias2...]*

In this command, the variables are the same as those in step 3.

The alias is added in the print client's
`/etc/lp/printers/`*printer-name*`/configuration` file.

6. **Repeat step 4 and step 5 for each print client that might need to use the alias.**

7. **Verify that the selectable character set alias is listed in the output of the following command on the print server and print clients.**

   `$ `**`lpstat -p`** *printer-name* **`-l`**

   Otherwise, submit a print request that uses the alias for the selectable character set and check for output.

Example—Setting Up an Alias for a Selectable Character Set

In the following example, the commands display the names of selectable character sets and specify `text` as an alias for the `usascii` selectable character set on the printer `luna`, which is an `ln03` printer type.

```
# tput -T ln03 cs0
usascii# tput -T ln03 cs1
english# tput -T ln03 csn2
finnish# tput -T ln03 csn3
japanese# tput -T ln03 cs4
norwegian#
# lpadmin -p luna -S usascii=text
```

Managing Print Filters

Print filters are programs that convert the content type of a file to a content type that is acceptable to the destination printer. The LP print service uses filters to:

- Convert a file from one data format to another so it can be printed properly on a specific type of printer

- Handle the special modes of printing, like two-sided printing, landscape printing, or draft- and letter-quality printing

- Detect printer faults and notify the LP print service of them so the print service can alert users and system administrators

Not every print filter can perform all these tasks. Because each task is printer-specific, the tasks can be implemented separately.

The LP print service provides the PostScript filters listed in Table 6–1. The filter programs are located in the `/usr/lib/lp/postscript` directory. For PostScript printing, you usually do not need to do anything beyond installing the filter programs when setting up a print server. Solaris Print Manager automatically enables the supplied filters. However, if you administer other printers, you might need to administer print filters for them.

Creating Print Filters

To create a new print filter, you must write a print filter program and create a print filter definition. Filters contain input types, output types, and complex options that provide a language to process command-line arguments within the filter. See "Creating a New Print Filter" on page 143 for background information and step-by-step instructions.

Adding, Changing, Removing, and Restoring Print Filters

Print filters are added, changed, or removed on the print server only.

You use the `lpfilter`(1M) command to manage the list of available filters. System information about filters is stored in the `/etc/lp/filter.table` file. The `lpfilter` command gets the information about filters to write to the table from filter descriptor files. The filter descriptor files supplied (PostScript only) are located in the `/etc/lp/fd` directory. The actual filter programs are located under `/usr/lib/lp`.

The LP print service imposes no fixed limit on the number of print filters you can define. You can remove filters that are no longer used to avoid extra processing by the LP print service. (LP examines all filters to find one that works for a specific print request.) If in doubt, do not remove a filter.

As you add, change, or delete filters, you can overwrite or remove some of the original filters provided by the LP print service. You can restore the original set of filters, if necessary, and remove any filters you have added.

SunOS software provides a default set of PostScript filters, which Solaris Print Manager automatically adds to a print server. Some of the TranScript filters used with `lpd`-based print services have SunOS equivalents, but others do not. The table below lists the default PostScript filters and identifies the TranScript filters, where applicable.

TABLE 6–1 Default PostScript Filters

Filter	Action	TranScript Equivalent
download	Download fonts	
dpost	`ditroff` to PostScript	psdit
postdaisy	`daisy` to PostScript	
postdmd	`dmd` to PostScript	
postio	Serial interface for PostScript printer	pscomm
postior	Communicate with printer	
postmd	Matrix gray scales to PostScript	
postplot	`plot` to PostScript	psplot
postprint	`simple` to PostScript	enscript
postreverse	Reverse or select pages	psrev
posttek	`TEK4014` to PostScript	ps4014

The SunOS software does *not* provide the following filters:

- `TEX`
- `oscat` (NeWSprint `opost`)
- Enscript

The `postreverse`, `postprint`, `postio`, and `dpost` filters are provided in place of Enscript.

Solaris Print Manager adds the default PostScript filters to a print server. If you have printing needs that are not met by these filters, see "How to Create a New Print Filter" on page 152 for information about writing a custom print filter.

▼ How to Add a Print Filter

1. **Log in as superuser or lp on the print server.**

2. **Add a print filter that is based on a print filter definition by using the** `lpfilter`
   command.

   ```
   # lpfilter -f filter-name -F filter-def
   ```

`-f` *filter-name*	Name you choose for the print filter.
`-F` *filter-def*	Name of the print filter definition.

   The print filter is added in the print server's `/etc/lp/filter.table` file.

3. **Verify that the print filter was added by checking for information about the print
   filter in the output of the following command.**

   ```
   # lpfilter -f filter-name -l
   ```

Example—Adding a Print Filter

In the following example, the command adds the `daisytroff` print filter that has the
`daisytroff.fd` print filter definition.

```
# lpfilter -f daisytroff -F /etc/lp/fd/daisytroff.fd
```

▼ How to Delete a Print Filter

1. **Log in as superuser or lp on the print server.**

2. **Delete the print filter by using the** `lpfilter` **command.**

   ```
   # lpfilter -f filter-name -x
   ```

`-f` *filter-name*	Name of the print filter to be deleted.
`-x`	Deletes the specified filter.

   The print filter is deleted from the print server's `/etc/lp/filter.table` file.

3. **Verify that filter was deleted by using the following command. You should receive
   an error indicating that no filter by the specified name exists.**

   ```
   # lpfilter -f filter-name -l
   ```

Example—Deleting a Print Filter

In the following example, the command deletes the `daisytroff` print filter.

```
# lpfilter -f daisytroff -x
```

▼ How to View Information About a Print Filter

1. **Log in as superuser or lp on the print server.**

2. **Request information about a print filter by using the** `lpfilter` **command.**

   ```
   # lpfilter -f filter-name -l
   ```

-f *filter-name*	Print filter for which you want to view information. Specify `all` for *filter-name* to view information about all the available print filters.
-l	Displays information about the specified filter.

   Information about the specified print filter(s) is displayed.

Examples—Viewing Information About a Print Filter

In the following example, the command requests information for the `postdaisy` print filter, and the information that is displayed in response.

```
# lpfilter -f postdaisy -l
Input types: daisy
Output types: postscript
Printer types: any
Printers: any
Filter type: slow
Command: /usr/lib/lp/postscript/postdaisy
Options: PAGES * = -o*
Options: COPIES * = -c*
Options: MODES group = -n2
Options: MODES group\=\([2-9]\) = -n\1
Options: MODES portrait = -pp
Options: MODES landscape = -pl
Options: MODES x\=\(\-*[\.0-9]*\) = -x\1
Options: MODES y\=\(\-*[\.0-9]*\) = -y\1
Options: MODES magnify\=\([\.0-9]*\) = -m\1
```

In the following example, the command redirects information about the `daisytroff` filter to a file (creates the filter definition for that filter). This is useful if a filter definition is removed unintentionally.

```
# lpfilter -f daisytroff -l > daisytroff.fd
```

In the following example, the command displays all the print filters that have been added to the system, and the information that is displayed in response.

```
# lpfilter -f all -l | grep Filter
(Filter "download")
Filter type: fast
(Filter "postio")
Filter type: fast
(Filter "postior")
Filter type: fast
(Filter "postreverse")
Filter type: slow
```

Managing Forms

A *form* is a sheet of paper on which information is printed in a predetermined format. Unlike plain paper stock, forms usually have text or graphics preprinted on them. Common examples of forms are company letterhead, invoices, blank checks, receipts, and labels.

The term *form* has two meanings: the physical medium (the paper) and the software that defines a form to the LP print service.

The LP print service allows you to control the use of forms. This section provides information about adding, changing, removing, mounting, and controlling access to forms.

Adding, Changing, or Deleting Forms

When you add a form, you tell the LP print service to include the form in its list of available forms. You also have to supply the information required to describe or define the form. Although you can enter such definitions when you add the form, it helps to create the definitions first and save them in files. You can then change the form definition by editing the file. See the table below for information about how to create form definitions.

Note – No form definitions are supplied with the LP print service.

To change a form, you must re-add the form with a different definition.

The LP print service imposes no limit on the number of forms you can define. However, you should delete forms that are no longer appropriate. Obsolete forms can result in unnecessary processing by the print service.

Mounting Forms

To print a form, you must load the paper in the printer and use a command to *mount* the form, which notifies the LP print service that print requests submitted to the printer are to be printed using the form definition. If you use one printer for different types of printing, including forms, you should:

- Disable the printer before you load the paper and mount the form.
- Re-enable the printer when the form is ready; otherwise, the LP print service will continue to print files that do not need the form on the printer.

When you mount a form, make sure it is aligned properly. If an alignment pattern has been defined for the form, you can request that the pattern print repeatedly after you have mounted the form, until you have adjusted the printer so the alignment is correct.

When you want to change or discontinue using a form on a printer, you must notify the LP print service by unmounting the form.

Tracking Forms

The LP print service helps you track which forms are mounted on each printer and notifies you when it cannot find a description it needs to print a form. You are responsible for creating form descriptions and mounting and unmounting form paper in each printer, either as part of setting up a printer or in response to alerts from the LP print service.

Users can specify the form on which they want a job to print. As root, you can mount a specific form, then tell the LP print service that the form is available and on which printer it is mounted. Users can submit print requests specifying a particular form. When the LP print service receives the request, it sends an alert message to root requesting that you mount the form.

Defining Alerts for Mounting Forms

You request alerts for mounting forms in the same way you request other alerts from the LP print service. See "Setting Up Printer Fault Alerts" on page 85 for general information about alerts.

Checking Forms

When you have defined a form for the LP print service, you can check it with either of two commands, depending on the type of information you want to check.

- Show the attributes of the form by using the lpforms(1M) command. You can also redirect the output of the command into a file to save it for future reference.
- Display the current status of the form by using the lpstat command. To protect potentially sensitive content, the alignment pattern is not shown.

If you are not sure about the name of an existing form, you can list the contents of the /etc/lp/forms directory to see the names of the forms there.

Limiting Access to Forms

You can control which printers and users have access to some or all of the forms available on the network. For example, you might want only the people in the payroll or accounts payable department to be able to print check forms. In addition, you might want the check forms to be available only on certain printers.

To limit user access to forms, see "How to Limit User Access to a Form" on page 126. To limit printer access to a form, see "How to Limit Printer Access to a Form" on page 126.

▼ How to Add a Form

1. **Log in as superuser or lp on the print server.**

2. **Add a form that is based on a form definition by using the** lpforms **command.**

   # **lpforms -f** *form-name* **-F /etc/lp/forms/***form*

   | -f *form-name* | Name you choose for the form. |
   | -F /etc/lp/forms/*form* | Name of the form definition. |

   The form is added in the print server's /etc/lp/forms/*form-name*/describe file.

3. **Verify that the form was added by checking for a listing of information about the form in the output of the following command.**

   # **lpforms -f** *form-name* **-l**

Example—Adding a Form

In the following example, the command adds the medical form that uses the medical.fmd form definition.

```
# lpforms -f medical -F /etc/lp/forms/medical.fmd
```

Note – Before the form can be used, one or more printers must be given access to the form. See "How to Limit Printer Access to a Form" on page 126.

▼ How to Delete a Form

1. **Log in as superuser or lp on the print server.**

2. **Delete the form by using the** lpforms **command.**

   ```
   # lpforms -f form-name -x
   ```

-f form-name	Form to be deleted.
-x	Deletes the specified form.

   The form is deleted from /etc/lp/forms/form-name file.

3. **Verify that form was deleted by using the following command. You should receive an error indicating that a form by the specified name does not exist.**

   ```
   # lpforms -f form-name -l
   ```

Example—Deleting a Form

In the following example, the command deletes the medical form.

```
# lpforms -f medical -x
```

▼ How to Unmount and Mount a Form

1. **Log in as superuser or lp on the print server.**

2. **Stop accepting print requests on the printer on which you are unmounting the current form by using the** `reject` **command.**

   # **`reject`** *printer-name*

printer-name	Name of the printer on which you are unmounting a form.

   New print requests (which might not require the form) are not allowed to enter the printer's queue.

3. **Unmount the current form by using the** `lpadmin` **command.**

   # **`lpadmin -p`** *printer-name* **`-M -f none`**

   In this command, the variable *printer-name* is the same as in step 2.

   The current form is deleted from the print server's
   `/etc/lp/printers/`*printer-name*`/configuration` file.

4. **Remove the form paper from the printer.**

5. **Load the form paper for the next print request.**

6. **Mount the form by using the** `lpadmin` **command.**

   # **`lpadmin -p`** *printer-name* **`-M -f`** *form-name* [**`-a -o filebreak`**]

`-p` *printer-name*	Printer on which you are mounting a form.
`-M -f` *form-name*	Name of the form to be mounted.
`-a -o filebreak`	Optionally enables you to print a copy of the alignment pattern defined for the form, if it has one.

   The specified form is added in the print server's
   `/etc/lp/printers/`*printer-name*`/configuration` file.

7. **Start accepting print requests on the printer.**

   # **`accept`** *printer-name*

   The printer is ready to print the form you just mounted.

8. **Verify that the form has been mounted by checking for the form name under the** `Form mounted` **heading in the output of the following command.**

   $ **`lpstat -p`** *printer-name* **`-l`**

   Otherwise, submit a print request that requires the new form and check the printer for output.

Examples—Unmounting and Mounting a Form

The following example shows the process of unmounting the currently mounted form on the printer luna.

```
# reject luna
destination "luna" will no longer accept requests
# lpadmin -p luna -M f none
# accept luna
destination "luna" now accepting requests
```

The following example shows the process of mounting the medical form on the printer luna.

```
# reject luna
destination "luna" will no longer accept requests
# lpadmin -p luna -M f medical -a -o filebreak
# accept luna
destination "luna" now accepting requests
```

▼ How to Set an Alert to Mount a Form

1. **Log in as superuser or lp on the print server.**

2. **Set a request alert for mounting a form by using the** lpadmin **command.**

   ```
   # lpforms -f form-name -A alert [-Q requests] [-W minutes]
   ```

-f *form-name*	Form for which you want to set a request alert.
-A *alert*	Specifies what kind of alert will occur when a form is requested. See Table 5–3 for detailed information about the valid values for *alert*. Some valid values are mail, write, and quiet. If you choose mail or write, a predefined alert message says to mount the specified form and includes the names of one or more printers that have been set up to use the form.
-Q *requests*	Specifies how many print requests that require the form must be in the queue to trigger an alert. If you don't specify this option, an alert occurs with just one print request in the queue.
-W *minutes*	Specifies how often (in minutes) the alert will occur. If you don't specify this option, the alert is sent once.

   The request alert is added in the print server's
   /etc/lp/forms/*form-name*/alert.sh file.

3. **Verify that the alert has been added for the form by checking the output of the following command.**

   ```
   # lpforms -f form-name -A list
   ```

   Otherwise, if you have set a low number of print requests to trigger the alert, submit print requests to meet the minimum requirement and make sure you receive an alert to mount the form.

Examples—Setting an Alert to Mount a Form

In the following example, the command sets email alerts to occur every five minutes for the `letterhead` form when there are 10 print requests for `letterhead` in the print queue.

```
# lpforms -f letterhead -A mail -Q 10 -W 5
```

In the following example, the command sets console window alerts to occur every 10 minutes for the `letterhead` form when there are five requests for `letterhead` in the print queue.

```
# lpforms -f letterhead -A write -Q 5 -W 10
```

In the following example, the command sets no request alerts for the `invoice` form.

```
# lpforms -f invoice -A none
```

▼ How to View Information About a Form

1. **Log in as superuser or lp on the print server.**

2. **Request information about a form by using the** `lpforms` **command.**

   ```
   # lpforms -f form-name -l
   ```

-f *form-name*	Form for which you want to view information. Specify `all` for *form-name* to view information about all the available forms.
-l	Lists the specified form.

   Information about the specified form(s) is displayed.

Examples—Viewing Information About a Form

In the following example, the command displays information about the medical form.

```
# lpforms -f medical -l
Page length: 62
Page width: 72
Number of pages: 2
Line pitch: 6
Character pitch: 12
Character set choice: pica
Ribbon color: black
Comment:
Medical claim form
```

In the following example, the command redirects the information about the medical form to a file. (This command creates the form definition for the form.) This is useful if a form definition gets removed unintentionally.

```
# lpforms -f medical -l > medical.fmd
```

▼ How to View the Current Status of a Form

1. **Log in on the print server.**

2. **Request information about the current status of a form by using the** lpstat**(1) command.**

   ```
   $ lpstat -f form-name
   ```

-f *form-name*	Form for which you want to view the current status. Specify all for *form-name* to view the current status of all the forms.

   Information about the current status of the specified form(s) is displayed.

Example—Viewing the Current Status of a Form

In the following example, the command displays the status of the medical form.

```
$ lpstat -f medical
form medical is available to you
```

▼ How to Limit User Access to a Form

1. **Log in as superuser or lp on the print server.**

2. **Allow or deny users access to a form by using the `lpforms` command.**

   ```
   # lpforms -f form-name -u allow:user-list | deny:user-list
   ```

`-f` *form-name*	Name of the form for which the allow or deny user access list is being created.
`-u allow:`*user-list*	Represents users to be added to the allow access list. Use a comma or a space to separate users' login IDs. If you use spaces, enclose the list of IDs in quotes.
	Table 5–5 provides the valid values for *user-list*.
`deny:`*user-list*	Represents users to be added to the deny user access list. Use a comma or a space to separate users' login IDs. If you use spaces, enclose the list of IDs in quotes. Table 5–5 provides the valid values for *user-list*.

   The specified user(s) are added to the allow or deny user access list for the specified form in one of the following files on the print server:

   `/etc/lp/forms/`*form-name*`/allow` or

   `/etc/lp/forms/`*form-name*`/deny`

3. **Verify the allow and deny user access lists by using the `lpforms` command.**

   ```
   # lpforms -f form-name -l
   ```

Examples—Limiting User Access to a Form

In the following example, the command allows only the users `nathan` and `marcia` access to the `check` form.

```
# lpforms -f check -u allow:nathan,marcia
```

In the following example, the command denies users `jones` and `smith` access to the `dental` form.

```
# lpforms -f dental -u deny:"jones,smith"
```

▼ How to Limit Printer Access to a Form

1. **Log in as superuser or lp on the print server.**

2. **Allow or deny use of forms on a printer by using the** `lpadmin` **command.**

   # **`lpadmin -p`** *printer-name* **`-f allow:`***form-list* **`| deny:`***form-list*

`-p` *printer-name*	Name of the printer for which the allow or deny forms list is being created.
`-f allow:`*form-list* `|` `deny:`*form-list*	Form names to be added to the allow or deny list. Use a space or a comma to separate multiple form names. If you use spaces to separate form names, enclose the list of form names in quotes.

   The specified form(s) are added to the allow or deny forms list in one of the following files on the print server:

   `/etc/lp/printers/`*printer-name*`/form.allow`

   `/etc/lp/printers/`*printer-name*`/form.deny`

3. **Verify the allow and deny forms lists by using the following command.**

   # **`lpstat -p`** *printer-name* **`-l`**

Examples—Limiting Printer Access to a Form

In the following example, the command allows the printer `luna` to access only the `medical`, `dental`, and `check` forms.

`lpadmin -p luna -f allow:medical,dental,check`

In the following example, the command denies the printer `luna` from accessing the `medical`, `dental`, and `check` forms.

`lpadmin -p luna -f deny:"medical dental check"`

Managing Fonts

If you have a laser printer, you might need to install and maintain PostScript fonts. You might also have to decide where to install PostScript fonts and how to manage them. For many printers, the fonts are set up as part of the printer installation process.

PostScript fonts are stored in outline form, either on the printer or on a system that communicates with the printer. When a document is printed, the PostScript interpreter generates each character as needed (in the appropriate size) from the outline description of it. If a font required for a document is not stored on the printer being

used, it must be transmitted to that printer before the document can be printed. This transmission process is called *downloading fonts*.

Fonts are stored and accessed in several ways:

- *Printer-resident fonts* are stored permanently on a printer. These fonts are installed in read-only memory (ROM) on the printer by the manufacturer. If the printer has a disk, you can install fonts on that disk. Most PostScript printers are shipped with 35 standard fonts.

- A *permanently downloaded font* is transmitted to a printer with a PostScript `exitserver` program. A permanently downloaded font remains in printer memory until the printer is turned off. Memory allocated to a downloaded font reduces the memory available on the server for PostScript print requests. Use of an `exitserver` program requires the printer system password and can be reserved for the printer administrator. You should permanently download a font if most print requests serviced by the printer use that font.

- Fonts that are used infrequently or for special purposes can be stored on a user's system. The user can specify these fonts when submitting the print request. The fonts are appended to the print request and transmitted to the printer. When the print request is processed, the space allocated for the font is freed for other print requests.

- Host-resident *fonts* are stored on a system shared by many users. The system that stores the fonts can be a print server or a print client. Each user can request fonts in the document to be printed. This method is useful when there are numerous available fonts, or when these fonts are not used by all print requests. If the fonts will be used only on printers attached to a print server, they should be stored on the print server. If the fonts are to be used by the users on one system and the users can submit requests to multiple printers on a network, the fonts should be stored on the users' system.

  The LP print service provides a special download filter to manage host-resident fonts. It also supplies `troff` width tables for the 35 standard PostScript fonts which reside on many PostScript printers, for use by the `troff`(1) program.

Managing Printer-Resident Fonts

Most PostScript printers come equipped with fonts resident in the printer ROM. Some printers have a disk on which additional fonts are stored. When a printer is installed, you should add the list of printer-resident fonts to the font list for that printer. By identifying printer-resident fonts, you prevent fonts from being transmitted unnecessarily across a network. Each printer has its own list of resident fonts, which is contained in the file:

`/etc/lp/printers/`*printer-name*`/residentfonts`

When the printer is attached to a print server, make sure the list in the `residentfonts` file includes fonts that are on the print server and which are available for downloading to the printer.

You must edit the files containing the list of printer-resident fonts by using a text editor such as `vi`.

Downloading Host-Resident Fonts

When a PostScript document contains a request for fonts not loaded on the printer, the download filter manages this request. The download filter uses PostScript document structuring conventions to determine which fonts to download.

LP print filters are either fast or slow. A fast filter quickly prepares a file for printing, and it must have access to the printer while the filter is processing. A slow filter takes longer to convert a file, and it does not need to access the printer while the filter is processing. An example of a slow filter is ASCII to PostScript.

The download filter is a fast filter; it downloads fonts automatically if the fonts are on the print server. The download filter can also be used to send fonts to a print server. To do this, you can create a new filter table entry that calls the download filter as a slow filter by using the `lp -y` command. Alternatively, you can force selection of this filter by changing the input type.

The download filter performs five tasks:

1. It searches the PostScript document to determine which fonts are requested. These requests are documented with the following PostScript structuring comments: `%%DocumentFonts:` *font1 font2 …* in the header comments.
2. It searches the list of printer-resident fonts to determine if the requested font must be downloaded.
3. If the font is not resident on the printer, the download filter searches the host-resident font directory (by getting the appropriate file name from the map table) to determine if the requested font is available.
4. If the font is available, the filter takes the file for that font and appends it to the file to be printed.
5. It sends the font definition file and the source file (the file to be printed) to the PostScript printer.

Installing and Maintaining Host-Resident Fonts

Some fonts reside on the host system and are transmitted to the printer as needed for particular print requests. As the administrator, you make PostScript fonts available to

all users on a system. To do so, you must know how and where to install these fonts. Because fonts are requested by name and stored in files, the LP print service keeps a map file that shows the correspondence between the names of fonts and the names of the files containing those fonts. Both the map and the font list must be updated when you install host-resident fonts.

The fonts available for use with PostScript printers are stored in directories you create called /usr/share/lib/hostfontdir/*typeface*/*font*, where *typeface* is replaced by a name like palatino or helvetica, and *font* is replaced by a name like bold or italic.

▼ How to Install Downloaded PostScript Fonts

1. **Log in as superuser or lp on the print server or print client.**

2. **Change directory to the** /etc/lp/printers/*printer-name* **directory.**

   # **cd /etc/lp/printers/***printer-name*

   printer-name   Name of the printer on which you want to install downloaded PostScript fonts.

3. **Create the** residentfonts **file, if it does not already exist.**

   # **touch residentfonts**

   This file might not exist if this is the first time you are adding permanently downloaded fonts.

4. **Edit the** residentfonts **file and add all the printer-resident fonts and fonts to be permanently downloaded.**

▼ How to Install Host-Resident PostScript Fonts

1. **Log in as superuser or lp on the print server or print client.**

2. **Create the** hostfontdir **directory, if it does not already exist.**

   # **cd /usr/share/lib**
   # **mkdir hostfontdir**
   # **chmod 775 hostfontdir**

3. **Create a directory for a new typeface, if the directory does not already exist.**

   # **mkdir** *typeface*

4. **Copy the font file to the appropriate directory.**

   `# ` **`cp`** *filename* **`/usr/share/lib/hostfontdir`**/*typeface*/*font*

5. **Add the name of the font and the name of the file in which it resides to the map table.**

   a. **Change to the** `/usr/share/lib/hostfontdir` **directory.**

   b. **Edit the** `map` **file using a text editor such as vi.**

      Add a one-line entry for each font you want to add to the table, with the font name first, followed by a space, followed by the name of the file where the font resides. For example:

      ```
      Palatino-Bold /usr/share/lib/hostfontdir/palatino/bold
      ```

   c. **Save the file.**

      When an example entry exists in the map table on the appropriate system, users will be able to apply the font (for example, Palatino Bold) in their print jobs. When they submit a print request containing this font, the LP print service appends a copy of the file `/usr/share/lib/hostfontdir/palatino/bold` to that file before sending it to the printer.

6. **If you are using** `troff`**, you must create new width tables for this font in the standard** `troff` **font directory.**

Customizing the LP Print Service (Tasks)

This chapter provides background information and procedures for customizing the LP print service.

This is a list of the step-by-step instructions in this chapter.

- "How to Adjust the Printer Port Characteristics" on page 134
- "How to Add a `terminfo` Entry for an Unsupported Printer" on page 138
- "How to Set Up a Custom Printer Interface Program" on page 142
- "How to Create a New Print Filter" on page 152
- "How to Create a New Form Definition" on page 156

For overview information about printers, see Chapter 2.

Adjusting Printer Port Characteristics

The printer port characteristics set by the LP print service must be compatible with the printer communication settings. If the default printer port settings provided by the LP print service do not work with a printer, refer to the printer manual from the manufacturer to find out what settings the printer requires from the LP print service. Use the `stty` command to set and display printer communication settings.

The table below shows the default `stty` settings used by the LP print service.

TABLE 7–1 `stty` Default Settings Used by the LP Print Service

Option	Meaning
`-9600`	Set baud to 9600

TABLE 7–1 stty Default Settings Used by the LP Print Service   *(Continued)*

Option	Meaning
-cs8	Set 8-bit bytes
-cstopb	Send one stop bit per byte
-parity	Do not generate parity
-ixon	Enable XON/XOFF (also known as START/STOP or DC1/DC3)
-opost	Do "output post-processing" using all the settings that follow in this table
-olcuc	Do not map lowercase to uppercase
-onlcr	Change line feed to carriage return/line feed
-ocrnl	Do not change carriage returns into line feeds
-onocr	Output carriage returns even at column 0
-nl0	No delay after line feeds
-cr0	No delay after carriage returns
-tab0	No delay after tabs
-bs0	No delay after backspaces
-vt0	No delay after vertical tabs
-ff0	No delay after form feeds

▼ How to Adjust the Printer Port Characteristics

1. **Log in as superuser or lp on the print server.**

2. **Adjust the printer port characteristics by using the** lpadmin **command.**

   # **lpadmin -p** *printer-name* **-o "stty=***options***"**

-p *printer-name*	Name of the printer for which you are adjusting the port characteristics.
-o "stty=*options*"	Sets the port characteristic (stty option) specified by *options*.You can change more than one stty option setting with this command. Enclose each option in single quotation marks and use a space to separate the options. See stty(1) for a complete list of options. Table 7–1 shows the default stty settings used by the LP print service.

3. **Verify that the printer port characteristics have been changed by using the following command.**

```
# stty -a
```

Examples—Adjusting the Printer Port Characteristics

In the following example, the command sets the port characteristics for the printer `luna`. The `parenb` option enables parity checking/generation, `parodd` sets odd parity generation, and `cs7` sets the character size to 7 bits.

```
# lpadmin -p luna -o "stty='parenb parodd cs7'"
```

In the following example, the command sets the terminal baud rate to 19200 for the printer `venus`.

```
# lpadmin -p venus -o "stty=19200"
```

Adding a `terminfo` Entry for an Unsupported Printer

The LP print service uses an interface program and the `terminfo` database to initialize each printer and establish a selected page size, character pitch, line pitch, and character set.

Each printer is identified in the `terminfo` database with a short name. The name required by the `terminfo` database is identical to the name used to set the `TERM` shell variable. This name is also the printer type you specify when setting up a printer. For example, the entries for different types of PostScript printers are in `/usr/share/lib/terminfo/P`. The default entries provided with the SunOS release are `PS` (for PostScript) and `PSR` (for PostScript Reverse).

If you cannot find a `terminfo` entry for your printer, you still might be able to use the printer with the LP print service without the automatic selection of page size, pitch, and character sets. However, you might have trouble keeping the printer set in the correct modes for each print request.

If there is no `terminfo` entry for your type of printer and you want to keep the printer set in the correct modes, you can either customize the interface program used with the printer or add an entry to the `terminfo` database. A terminal or printer entry in the `terminfo` database contains and defines hundreds of items. The LP print service, however, uses fewer than 50 of these items. The table below lists the required `terminfo` items for a printer.

TABLE 7–2 Required `terminfo` Items for a Printer

Item	Meaning
Booleans:	
cpix	Changing character pitch changes resolution
daisy	Printer requires an operator to change character set
lpix	Changing line pitch changes resolution
Numbers:	
bufsx	Number of bytes buffered before printing
cols	Number of columns in a line
cps	Average print rate in characters per second
it	Tabs initially every n spaces
lines	Number of lines on a page
orc	Horizontal resolution, in units per character
orhi	Horizontal resolution, in units per inch
orl	Vertical resolution, in units per line
orvi	Vertical resolution, in units per inch
Strings:	
chr	Change horizontal resolution
cpi	Change number of characters per inch
cr	Carriage return
csnm	List of character set names
cudl	Down one line
cud	Move carriage down n lines
cuf	Move carriage right n columns
cvr	Change vertical resolution
ff	Page eject
hpa	Horizontal position absolute
ht	Tab to next 8-space tab stop
if	Name of initialization file
iprog	Path name of initialization program

TABLE 7–2 Required `terminfo` Items for a Printer   *(Continued)*

Item		Meaning
	is1	Printer initialization string
	is2	Printer initialization string
	is3	Printer initialization string
Strings:		
	lpi	Change number of lines per inch
	mgc	Clear all margins (top, bottom, and sides)
	rep	Repeat a character *n* times
	rwidm	Disable double-wide printing
	scs	Select character set
	scsd	Start definition of a character set
	slines	Set page length to *n* lines per page
	smgl	Set left margin at current column
	smglp	Set left margin
	smgr	Set right margin at current column
	smgrp	Set right margin
	smglr	Set both left and right margins
	msgt	Set top margin at current line
	smgtp	Set top margin
	smgb	Set bottom margin at current line
	smgbp	Set bottom margin
	smgtb	Set both top and bottom margins
	swidm	Enable double-wide printing
	vpa	Vertical position absolute

▼ How to Add a `terminfo` Entry for an Unsupported Printer

> **Note –** Before you create a `terminfo` entry for a printer, you should first make sure none of the existing `terminfo` entries will support the printer. To do so, try to set up the printer with an entry for a similar printer, if there is one.

1. **Log in as superuser or lp on the print server.**

2. **Determine a** `terminfo` **entry name for the printer.**

   The directories in the `/usr/share/lib/terminfo` directory contain all the valid `terminfo` entries. Use them as a guide for choosing a name for the printer.

3. **Create a** `terminfo` **entry file for the printer.**

   Table 7–2 shows the items you must define in the `terminfo` entry to add a new printer to the LP print service. For more details about the structure of the `terminfo` database, see `terminfo`(4).

   To help you start writing a new `terminfo` entry, use the `infocmp` command to save an existing `terminfo` entry to a file. This is helpful if there is a `terminfo` entry that is similar to one you want to create. For example, the following command saves the `ps` entry to the `ps_cust` file, which will become the new `terminfo` entry.

   ```
   infocmp ps > ps_cust
   ```

4. **Compile the** `terminfo` **entry file into the** `terminfo` **database.**

   ```
   # tic terminfo_entry
   ```

      terminfo_entry   The `terminfo` entry file you created.

5. **Check for the new** `terminfo` **entry file in the** `/usr/share/lib/terminfo` **directory.**

Customizing the Printer Interface Program

If you have a printer that is not supported by the standard printer interface program, you can furnish your own printer interface program. You can copy the standard

program and then tell the LP print service to use it for a specified printer. But first you need to understand what is in the standard program. The following section describes the standard program.

A printer interface program should:

- Initialize the printer port, if necessary. The standard printer interface program uses the `stty` command to initialize the printer port.
- Initialize the printer hardware. The standard printer interface program gets the control sequences from the `terminfo` database and the `TERM` shell variable.
- Print a banner page, if necessary.
- Print the number of copies specified by the print request.

Caution – If you have a printer interface program from a release of UNIX System V prior to Release 3.2, it will probably work with the SunOS 5.9 or compatible LP print service. However, several `-o` options have been standardized in the SunOS 5.9 or compatible LP print service and will be passed to every printer interface program. These options might interfere with similarly named options used by the old interface.

The LP print service, not a printer interface program, is responsible for opening the printer port. The printer port is given to the printer interface program as standard output, and the printer is identified as the "controlling terminal" for the printer interface program so that a "hang-up" of the port will cause a SIGHUP signal to be sent to the printer interface program.

The Standard Printer Interface Program

The standard (model) printer interface program, `/usr/lib/lp/model/standard`, is used by the LP print service to set the printing defaults shown in Table 7–3.

TABLE 7–3 Default Printer Port Characteristics

Characteristic	Default Setting
Default filter	None
Character pitch	None
Line pitch	None
Page width	None
Page length	None
Character set	None

TABLE 7–3 Default Printer Port Characteristics   *(Continued)*

Characteristic	Default Setting
stty options	9600 cs8 -cstopb -parenb -parodd ixon -ixany opost -olcuc onlcr -ocrnl -onocr -onlret -ofill nl0 cr0 tab0 bs0 vt0 ff0
Exit code	0

Customizing `stty` Modes

If you need to change the terminal characteristics, like baud rate or output options, look for the section of the standard printer interface program that begins with the following comment:

```
## Initialize the printer port
```

Exit Codes

When printing is complete, your interface program should exit with a code that shows the status of the print job. The exit code is the last entry in the printer interface program.

The table below shows the exit codes and how they are interpreted by the LP print service.

TABLE 7–4 Printer Interface Program Exit Codes

Code	Meaning to the LP Print Service
0	The print request has been successfully completed. If a printer fault occurred, it has been cleared.
1 to 127	A problem was encountered when printing a request (for example, too many nonprintable characters or the request exceeds the printer capabilities). The LP print service notifies the person who submitted the request that there was an error when printing it. This error will not affect future print requests. If a printer fault has occurred, it has been cleared.
128	This code is reserved for internal use by the LP print service. Interface programs must not exit with this code.

TABLE 7–4 Printer Interface Program Exit Codes	*(Continued)*
Code	**Meaning to the LP Print Service**
129	A printer fault was encountered when printing the request. This fault will affect future print requests. If the fault recovery for the printer directs the LP print service to wait for the administrator to correct the problem, the LP print service disables the printer. If the fault recovery is to continue printing, the LP print service will not disable the printer, but it will try printing again in a few minutes.
>129	These codes are reserved for internal use by the LP print service. Interface programs must not exit with codes in this range.

If the program exits with a code of 129, root is alerted of a printer fault. The LP print service must also reprint the request from the beginning, after the fault has been cleared. If you do not want the entire request to be reprinted, you can have the interface program send a fault message to the LP print service, but wait for the fault to be cleared. When the fault is cleared, the interface program can resume printing the file. When printing is finished, the printer interface program can give a zero exit code, just as if the fault had never occurred. An added advantage of this approach is that the interface program can detect when the fault is cleared automatically, so that the administrator does not need to re-enable the printer.

Fault Messages

You can use the `lp.tell` program to send fault messages to the LP print service. This program is referenced by the `LPTELL` shell variable in the standard printer interface code. The program takes standard input and sends it to the LP print service, where it is put into the message that alerts the administrator to the printer fault. If its standard input is empty, `lp.tell` does not initiate an alert. For an example of how the `lp.tell` program is used, examine the standard printer interface code immediately after the following comment:

```
# Set up the $LPTELL program to capture fault messages here
```

If you use the special exit code `129` or the `lp.tell` program, the printer interface program does not need to disable the printer itself. The interface program can disable the printer directly, but doing so will override the fault-alerting mechanism. Alerts are sent only if the LP print service detects that the printer has a fault, and the special exit code and the `lp.tell` program are its main detection tools.

If the LP print service has to interrupt printing of a file at any time, it kills the interface program with a signal TERM (trap number 15). (See `kill`(1) and `signal`(3C).) If the printer interface program dies from receipt of any other signal, the LP print service assumes that future print requests will not be affected, and continues to use the

printer. The LP print service notifies the user who submitted the request that the request has not been finished successfully.

When the interface is first invoked, the signals HUP, INT, QUIT, and PIPE (trap numbers 1, 2, 3, and 13) are ignored. The standard interface changes this so the signals are trapped at appropriate times. The standard interface interprets receipt of these signals as warnings that the printer has a problem; when it receives a signal, it issues a fault alert.

Using a Customized Printer Interface Program

You can create a customized printer interface program and use it in place of the standard printer interface program on the print server. To do so, you use the `lpadmin` command to register the program with the LP print service for a specific printer.

▼ How to Set Up a Custom Printer Interface Program

1. **Log in as superuser or lp on the print server.**

2. **Determine your next step based on whether you have a custom printer interface program.**

If You ...	Then ...
Need to create a custom printer interface program	Go to step 3.
Already have a custom printer interface program	Go to step 5.

3. **Copy the standard printer interface program.**

   # `cp /var/spool/lp/model/standard` *custom-interface*

4. **Change the copy of the standard printer interface program to meet your needs.**
   Refer to the description of the program in "The Standard Printer Interface Program" on page 139 to determine what you need to change.

5. **Set up the custom printer interface program for a specific printer.**

   # `lpadmin -p` *printer-name* `-i` *custom-interface*

   | `-p` *printer-name* | The printer that will use the custom printer interface program. |
   | `-i` *custom-interface* | Name of the custom printer interface program. |

The custom printer interface program is registered with the LP print service, and will be used by that printer when users submit print requests.

6. **Verify that the custom printer interface program has been added in the** /etc/lp/printers/*printer-name*/configuration **file.**

Examples—Setting Up a Custom Printer Interface Program

In the following example, the command sets up a custom printer interface program named custom for the printer luna.

```
# lpadmin -p luna -i custom
```

In the following example, the command sets up a custom printer interface program that the system venus is using on the printer asteroid.

```
# lpadmin -p asteroid -e venus
```

Creating a New Print Filter

A filter is used by the LP print service each time it has to print a type of file that the printer cannot interpret. Creating a new print filter is not easy; it usually requires extensive experimentation. The process of defining a new print filter consists of two steps:

- Writing a print filter program
- Creating a print filter definition

A print filter can be as simple or as complex as needed. Filters contain input types, output types, and complex options that provide a language to process command-line arguments within the filter.

If you have non-PostScript printers, you have to create and add print filters as required. First, you need to understand what print filters are and the requirements that must be met by a filter program.

Writing a Print Filter Program

The LP print service provides filter programs in the /usr/lib/lp/postscript directory. These filters cover most PostScript printing situations—where the

destination printer requires the data to be in PostScript format. A print filter program must be a binary executable.

Types of Filters

There are two types of print filters: fast filters and slow filters.

Fast filters do not require much processing time to prepare a file for printing. They must have access to the printer when they run. To be capable of detecting printer faults, a print filter must be a fast filter. Any filter that uses the PRINTER keyword as a filter option must be installed as a fast filter.

Slow filters require a great deal of processing time to prepare a file for printing. They do not require access to the printer when they run. Slow filters are run in the background so they do not tie up the printer, allowing other files that do not need slow filtering to be printed.

Converting Files

The LP print service uses print filters to convert files from one content type to another. You can specify the accepted file content types for each printer. The user specifies the file content type when submitting a print request, and the LP print service finds a printer that can print files of that content type. Because many applications can generate files for various printers, this is often sufficient. However, some applications can generate files that cannot be printed on any available printers.

Each time the LP print service receives a request to print a type of file that is in a format that cannot be accepted directly by a printer, the LP print service tries to match the content type of the print request with the content type of the available (or specified) printer. If there is a match, the file can be sent directly to the printer without filtering. If no match is found, or if the content type specifies that a filter be used, the LP print service tries to match the content type of the file with the input content type of available filters, and match the output type of the filter with the content type of the printer. When an appropriate filter is found, the print request is passed through the filter.

Handling Special Printing Modes

A print filter handles special modes and requests to print specific pages. A special printing mode is needed to print any characteristics of print requests that require a customized filter. Filters handle the following characteristics:

- Printer type
- Character pitch

- Line pitch
- Page length
- Page width
- Pages to print
- Character set
- Form name
- Number of copies

The LP print service provides default settings for these characteristics; however, a print filter can handle some characteristics more efficiently. For example, some printers can handle multiple copies more efficiently than the LP print service, and, in this case, you can provide a filter for multiple-copy page control.

Detecting Printer Faults

Each printer has its own way of detecting printer faults and transmitting fault signals to the LP print service. The LP print service only checks for hang-ups (loss of carrier) and excessive delays in printing.

Some printers provide good fault coverage and can send a message describing the reason for a fault. Other printers indicate a fault by using signals other than the signals indicating loss of carrier signal or shut off of data flow. A filter is required to interpret this additional printer fault information.

A filter can also put a print request on hold, wait for a printer fault to clear, and then resume printing. With this capability, the print request that was interrupted does not need to be reprinted in its entirety. Only a filter that knows the control sequences used by a printer can determine where to break a file into pages. Consequently, only such a filter can find the place in the file where printing should start after a fault is cleared.

When a print filter generates messages, those messages are handled by the LP print service, and alerts are sent to the system administrator if alerts are enabled. For further information, see "Setting Up Printer Fault Alerts" on page 85.

Requirements for a Print Filter Program

A print filter can be simple or complex, but it has to meet the following requirements:

- The filter should get the contents of a file from its standard input and send the converted file to the standard output.

- A program cannot be used as a filter if it references external files. You might be tempted to use a program like `troff`, `nroff`, or a similar word processing program as a filter. The LP print service does not recognize references to other files, known as include files, from a filter program. Because `troff` and `nroff` allow include files, they can fail when used as filters. If the program needs other files to

complete its processing, it should not be used as a filter.

- The filter should not depend on files that normally would not be accessible to a user. If a filter fails when run directly by a user, it will fail when run by the LP print service.

- A slow filter can send messages about errors in the file to standard error; a fast filter should not. Error messages from a slow filter are collected and sent to the user who submitted the print request.

- If a slow filter dies because it received a signal, the print request is stopped and the user who submitted the request is notified. Likewise, if a slow filter exits with a non-zero exit code, the print request is stopped and the user is notified. The exit codes from fast filters are treated differently.

If you want the filter to detect printer faults, it should also meet the following requirements:

- If possible, the filter should wait for a fault to be cleared before exiting. It should also continue to print at the top of the page where printing stopped after the fault is cleared. If you do not want use the continuation feature, the LP print service will stop the filter before alerting the administrator.

- The filter should send printer fault messages to its standard error as soon as the fault is recognized. It does not have to exit; it can wait for the fault to be cleared.

- The filter should not send messages about errors in the file to standard error. These messages should be included in the standard output, where they can be read by the user.

- The filter should exit with a zero exit code if the file is finished printing (even if errors in the file have prevented it from being printed correctly).

- The filter should exit with a non-zero exit code only if a printer fault has prevented it from finishing a print request.

- When added to the filter table, the filter must be added as a fast filter.

Creating a Print Filter Definition

A print filter definition tells the LP print service about the filter, what print filter program to run, what kind of conversion it does, and so on. A set of filter descriptor files are provided in the `/etc/lp/fd` directory. These files describe the characteristics of the filters (for example, fast or slow filter), and point to the filter programs (for example, `/usr/lib/lp/postscript/postdaisy`).

When defining a new print filter, in addition to writing a filter program, you must create a print filter definition. A print filter definition contains the following information used by the LP print service:

- Name of the filter program to run
- Input types it accepts

- Output types it produces
- Printer types to which it can send jobs
- Names of specific printers to which it can send jobs
- Filter types (either fast or slow)
- Options

You can type the characteristics as direct input to the lpfilter command. You also can create a file that specifies the filter's characteristics, and use the file name as input to the lpfilter command. Such a file is called a *filter descriptor file* and should be located in the /etc/lp/fd directory. These files are not the filters themselves, but rather point to the filters.

Whether you store the information in a file, or enter it directly on the command line, use the following format:

```
Command: command-pathname [options]
Input types: input-type-list
Output types: output-type-list
Printer types: printer-type-list
Printers: printer-list
Filter type: fast or slow
Options: template-list
```

Note – If you provide more than one definition (that is, more than one line) for any filter characteristic other than Options, only the second definition will be used by the print service.

The information can be arranged in any order, and not all the information is required. When you do not specify values, those shown in the table below are assigned by default. They are not very useful, which is why you should specify explicit values.

TABLE 7–5 Default Values for lpfilter Arguments

Item	Default
Input types	any
Output type	any
Printer types	any
Printers	any
Filter type	slow

Command

Use the full path of the filter program. If there are any fixed options that the program always needs, include them here.

Input Types

Input types is a list of file content types that the print filter can process. The LP print service does limit the number of input types, but most filters can accept only one type. Several file types can be similar enough that the filter can deal with them. You can use whatever names you like, with a maximum of 14 alphanumeric characters and dashes. Do not use underscores as part of the input type name.

The LP print service uses these names to match a filter to a file type, so follow a consistent naming convention. For example, if more than one filter can accept the same input type, use the same name for that input type when you specify it for each filter. Inform your users of these names so they know how to identify the file type when submitting a file for printing.

Output Types

Output types is list of file types that the filter can produce as output. For each input type, the filter produces a single output type. The output type can vary, however, from job to job. The name of the output type is restricted to 14 alphanumeric characters and dashes.

The output type names should either match the types of available (local or remote) printers, or match the input types handled by other filters. The LP print service groups filters in a shell pipeline if it finds that several passes by different filters are needed to convert a file. It is unlikely that you will need this level of sophistication, but the LP print service allows it. Try to find a set of filters that takes as input types all the different files the users might want printed, and that converts those files directly into file types the printer can handle.

Printer Types

Printer types is a list of the types of printers into which the print filter can convert files. For most printers and filters, you can leave this part of the filter definition blank, because it is identical to the list of output types. But it can be different. For example, you could have a printer with a single printer type for purposes of initialization, but which can recognize several different file content types. Essentially, this printer has an internal filter that converts the various file types into one that it can handle. Thus, a

filter might produce one of several output types that match the file types that the printer can handle. The print filter should be marked as working with that printer type.

As another example, you might have two different models of printers that are listed as accepting the same file types. Due to slight differences in manufacture, however, one printer deviates in the results it produces. You label the printers as being of different printer types, say A and B, where B is the one that deviates. You create a filter that adjusts files to account for the deviation produced by printers of type B. Because this filter is needed only for those printer types, you would list it as working only on type B printers.

Printers

A print filter is normally able to work with all printers that accept its output, so you can usually skip this part of the filter definition.

You might, however, have some printers that are or inappropriate for the output that the filter produces. For example, you might want to dedicate one printer for fast turnaround, only sending files that require no filtering to that printer. Other printers of identical type can be used for files that need extensive filtering before they can be printed.

Filter Type

The LP print service recognizes fast and slow filters, as described in "Types of Filters" on page 144.

Slow filters that are invoked by printing modes (using the lp -y command) must be run on the system from which the print request originated. The LP print service cannot pass values for modes to print servers. It can, however, match a file content type (specified after the -T option of the lp command) to a content type on a print server. Therefore, if you want to activate special modes on a print server, you must specify content types that permit the LP print service to match input types and output types.

Options

Options specify how different types of information are converted into command-line arguments to the filter command. This information can include specifications from a user (with the print request), the printer definition, and the specifications implemented by any filters used to process the request.

Defining Print Filter Options With Templates

There are 13 sources of information for defining print filter options, each of which is represented by a *keyword*. Each option is defined in a *template*. A template is a statement in a filter definition that defines an option to be passed to the filter command, based on the value of one of the characteristics of the filter.

The options specified in a filter definition can include none, all, or any subset of the 13 keywords. In addition, a single keyword can be defined more than once, if multiple definitions are required for a complete filter definition. The table below contains descriptions of the 13 keywords available for defining Options in a print filter definition.

TABLE 7–6 Print Filter Options Keywords

Characteristic	Keyword	Possible Patterns	Example
Content type (input)	INPUT	*content-type*	troff
Content type (output)	OUTPUT	*content-type*	postscript, impress
Printer type	TERM	*printer-type*	att495
Printer name	PRINTER	*printer-name*	lp1
Character pitch	CPI	*scaled-decimal*	10
Line pitch	LPI	*scaled-decimal*	6
Page length	LENGTH	*scaled-decimal*	66
Page width	WIDTH	*scaled-decimal*	80
Pages to print	PAGES	*page-list*	1-5,13-20
Character set	CHARSET	*character-set*	finnish
Form name	FORM	*form-name*	invoice2
Number of copies	COPIES	*integer*	3
Special modes	MODES	*mode*	landscape

A print filter definition can include more than one template. Multiple templates are entered on a single line and separated with commas, or they are entered on separate lines, preceded by the Options: prefix.

The format of a template is as follows:

keywordpattern = replacement

The keyword identifies the type of option being registered for a particular characteristic of the filter.

The *pattern* is a specific option for the keyword.

The *replacement* is what happens when the keyword has the noted value.

For an example of how an option is defined for a particular filter, suppose you want to have the print service scheduler assign print requests to filters following this criteria:

- If the type of OUTPUT to be produced by the filter is impress, then pass the -I option to the filter.
- If the type of OUTPUT to be produced by the filter is postscript, then pass the -P option to the filter.

To specify these criteria, provide the following templates as options to the lpfilter command:

```
Options: OUTPUT impress=-I, OUTPUT postscript=-P
```

If the Options line becomes too long, put each template on a separate line, as follows:

```
Options: OUTPUT impress=-I
Options: OUTPUT postscript=-P
```

In both templates, the *keyword* is defined as OUTPUT. In the first template, the pattern is impress and the value of the *replacement* is --I. In the second template, the value of *pattern* is postscript and the value of *replacement* is -P.

To find out which values to supply for each type of template (that is, for the *pattern* and *replacement* arguments for each keyword), consider the following:

- The values for the INPUT templates come from the file content type that needs to be converted by the filter.
- The values for the OUTPUT templates come from the output type that has to be produced by the filter.
- The value for the TERM template is the printer type.
- The value for the PRINTER template is the name of the printer that will print the final output.
- The values for the CPI, LPI, LENGTH, and WIDTH templates come from the user's print request, the form being used, or the default values for the printer.
- The value for the PAGES template is a list of pages that should be printed. Typically, it is a list of page ranges separated by commas. Each page range consists of a pair of numbers separated by a dash, or a single number. (For example, 1–5,6,8,10 indicates pages 1 through 5, plus pages 6, 8, and 10.) However, whatever value was given in the -P option to a print request is passed unchanged.
- The value for the CHARSET template is the name of the character set to be used.
- The value for the FORM template is the name of the form requested by the lp -f command (the command used to submit a print request).

- The value of the COPIES template is the number of copies of the file to print. If the filter uses this template, the LP print service will reduce to one the number of copies of the filtered file it prints, since this "single copy" includes the multiple copies produced by the filter.

- The value of the MODES template comes from the lp -y command. Because a user can specify several -y options, there might be several values for the MODES template. The values will be applied in the left-to-right order given by the user.

The *replacement* part of a template shows how the value of a template should be given to the filter program. It is typically a literal option, sometimes with the placeholder asterisk (*) included to show where the value goes. The *pattern* and *replacement* also can use the regular expression syntax of ed(1) for more complex conversion of user input options into filter options. All regular expression syntax of ed(1) is supported, including the \(... \) and \n constructions, which can be used to extract portions of the *pattern* for copying into the *replacement*, and the &, which can be used to copy the entire *pattern* into the *replacement*.

Note – If a comma or an equal sign (=) is included in a *pattern* or a *replacement*, precede it with a backslash (\). A backslash in front of any of these characters is removed when the *pattern* or *replacement* is used.

▼ How to Create a New Print Filter

1. **Log in as superuser or lp on the print server.**

2. **Create a print filter program.**

   See "Writing a Print Filter Program" on page 143 for information on print filter programs. By convention, filter programs for PostScript printers are located in the /usr/lib/lp/postscript directory. You should put programs you create under /usr/lib/lp in a directory of your choosing.

3. **Create a print filter definition.**

   See "Creating a Print Filter Definition" on page 146 for information on print filter definitions. You should save the printer filter definition in a text file. By convention, filter definitions are located in the /etc/lp/fd directory and are identified with the .fd suffix.

4. **Add the print filter to a print server.**

   For instructions, see "How to Add a Print Filter" on page 116.

Examples—Creating a New Print Filter

The following example shows a print filter definition to convert N37 or Nlp to simple.

```
Input types: N37, Nlp, simple
Output types: simple
Command: /usr/bin/col
Options: MODES expand = -x
Options: INPUT simple = -p -f
```

In this example, the print filter program is named col. Once you add the new print filter to a print server, a user's print requests will be handled as follows:

- When a user enters the following command:

  ```
  $ lp -y expand report.doc
  ```

  The print filter program is run with the following arguments to convert the file:

  ```
  /usr/bin/col -x -p -f
  ```

- When a user enters the following command:

  ```
  $ lp -T N37 -y expand report.doc
  ```

  The print filter program is run with the following arguments to convert the file:

  ```
  /usr/bin/col -x
  ```

The following example shows a print filter definition to convert from troff to PostScript.

```
Input types: troff

Output types: postscript

Printer types: PS

Filter type: slow

Command: /usr/lib/lp/postscript/dpost

Options: LENGTH * = -l*

Options: MODES port = -pp, MODES land = -pl
Options: MODES group \=\([1-9]\) = -n\l
```

In this example, the filter program is named dpost. It takes one input type, troff, produces a postscript output, and works with any printer of type PS (PostScript). Users need to give just the abbreviation port or land when they ask for the paper orientation to be in portrait mode or landscape mode. Because these options are not intrinsic to the LP print service, users must specify them using the lp -y command.

After you add the new print filter to a print server, print requests will be handled as follows:

- When a user enters the following command to submit a `troff` file type for printing on a PostScript printer (type PS), with requests for landscape orientation and a page length of 60 lines:

  ```
  $ lp -T troff -o length=60 -y land -d luna ch1.doc
  ```

  The print filter program `dpost` is run with the following arguments to convert the file:

  ```
  /usr/lib/lp/postscript/dpost -l60 -pl luna ch1.doc
  ```

- When a user enters the following command:

  ```
  $ lp -T troff -y group=4 -d luna ch1.doc
  ```

  The print filter program `dpost` is run with the following arguments to convert the file:

  ```
  /usr/lib/lp/postscript/dpost -n4
  ```

Creating a New Printer Form

When you want to provide a new form, you must define its characteristics by entering information about nine required characteristics (such as page length and page width) as input to the `lpforms` command. The LP print service uses this information to:

- Initialize the printer so that printing is done properly on the form
- Send reminders to the system administrator about how to handle the form

The form name can be anything you choose, as long as it does not contain more than 14 alphanumeric characters and underscores. The information must be in the following format:

```
Page length: scaled number
Page width: scaled number
Number of pages: integer
Line pitch: scaled number
Character pitch: scaled number
Character set choice: character-set-name [,mandatory]
Ribbon color: ribbon-color
Comment:
informal notes about the form
Alignment pattern: [content-type] alignment pattern
```

The optional phrase [,mandatory] means that the user cannot override the character set choice in the form. The *content-type* can be given, although this is optional, with an alignment pattern. If this attribute is given, the print service uses it to determine, as necessary, how to filter and print the file.

With two exceptions, the information can appear in any order. The exceptions are the Alignment pattern (which must always be last), and the *comment* (which must always follow the line with the Comment: prompt). If the comment contains a line beginning with a key phrase (like Page length, Page width , and so on), precede that line with a > character so the key phrase is not at the beginning of the line. The initial > character is stripped from the comment and is not displayed.

Not all of the information must be given. When you do not specify values for the items listed in the table below the default values are assigned. Before running the lpforms command, gather the following information about the new form:

TABLE 7–7 Default Form Values

Item	Default	Description
Page length	66 lines	The length of the form, or the length of each page in a multipage form. This information can be the number of lines, or the size in inches or centimeters.
Page width	80 columns	The width of the form, in characters, inches, or centimeters.
Number of pages	1	The number of pages in a multipage form. The LP print service uses this number with a print filter (if available) to restrict the alignment pattern to a length of one form. See the description of alignment pattern below. If no filter is available, the LP print service does not truncate the output.
Line pitch	6 lines per inch	A measurement of how close lines appear on the form. This is also called leading. It is the distance between two lines, from baseline to baseline, measured by either lines per inch or lines per centimeter.
Character pitch	10 characters per inch	A measurement of how close together characters appear on the form. It is the distance between characters, measured by either characters per inch or characters per centimeter.

TABLE 7–7 Default Form Values   *(Continued)*

Item	Default	Description
Character set choice	Any	The character set, print wheel, or font cartridge that should be used when this form is used. Users can choose a different character set for their own print requests when using this form, or you can require that only one character set be used.
Ribbon color	Any	If the form should always be printed using a certain color ribbon, the LP print service can give a mount alert message indicating which color to use.
Comment	(No default)	Any remarks that might help users understand the form. For example, the remarks could indicate the name of the form, its revision, its purpose, or restrictions on its use.
Alignment pattern	(No default)	A sample file that the LP print service uses to fill one blank form. When mounting the form, you can print this pattern on the form to align it properly. You can also define a content type for this pattern so that the print service knows how to print it.

Note – The LP print service does not try to mask sensitive information in the alignment pattern. If you do not want sensitive information printed on sample forms—for example when you align checks—then you should mask the appropriate data. The LP print service keeps the alignment pattern stored in a safe place, where only those logged in as root or lp can read it.

When you have gathered the information for the form, you enter it as input to the `lpforms` command. You should record this information first in a separate file so you can edit it before entering it with `lpforms`. You can then use the file as input instead of typing each piece of information separately after a prompt.

▼ How to Create a New Form Definition

1. **Log in as superuser or lp on the print server.**

2. **Create a form definition file.**

   See "Creating a New Printer Form" on page 154 for a description on creating print forms. You should save the printer definition in a text file.

3. **Add the form to the LP print service by using the** `lpadmin` **command.**

   # **lpadmin -p** *printer-name* **-M -f** *form-name*

4. **Add the form to a print server.**

   For instructions, see "How to Add a Form" on page 120.

LP Print Service (Reference)

This chapter provides background information on the LP print service.

For step-by-step instructions on print management tasks, see:

- Chapter 4
- Chapter 5
- Chapter 6
- Chapter 7

The LP Print Service

The *LP print service* is a set of software utilities that allows users to print files while they continue to work. Originally, the print service was called the LP spooler. (LP stood for line printer, but its meaning now includes many other types of printers, such as laser printers. Spool is an acronym for system peripheral operation off-line.)

The print service consists of the LP print service software, any print filters you might provide, and the hardware (the printer, system, and network connections).

The Structure of the LP Print Service

This section describes the directory structure, files, logs, and commands of the LP print service.

LP Print Service Directories

The files of the LP print service are distributed among seven directories, as shown in the table below.

TABLE 8–1 Directories for the LP Print Service

Directory	Contents
/usr/bin	The LP print service user commands
/etc/lp	A hierarchy of LP server configuration files
/usr/share/lib	The terminfo database directory
/usr/lib/print	The lp conversion scripts, in.lpd daemon and the printd daemon. The printd daemon transfers all pending jobs in the/var/spool/print directory once per minute. When there are no jobs remaining to transfer, it exits
/usr/sbin	The LP print service administrative commands
/usr/lib/lp	The LP daemons; directories for binary files and PostScript filters; and the model directory (which contains the standard printer interface program)
/var/lp/logs	The logs for LP activities: lpsched.n – Messages from lpsched and requests.n – Information about completed print requests
/var/spool/lp	The spooling directory where files are queued for printing
/var/spool/print	The LP print service client-side request staging area

LP Print Service Configuration Files

The scheduler stores configuration information in LP configuration files located in the /etc/lp directory, as described in the table below.

Caution – The configuration files listed in the table below are private interfaces, and are subject to change in future releases. You should not build software that relies on these files being in their current locations or that relies on the data being in the format currently used.

TABLE 8–2 Contents of the `/etc/lp` Directory

File	Type	Description
classes	Directory	Files identifying classes provided by the `lpadmin -c` command.
fd	Directory	Description of existing filters.
filter.table	File	Print filter lookup table.
forms	Directory	Location to put files for each form. Initially, this directory is empty.
interfaces	Directory	Printer interface program files.
logs	Link to /var/lp/logs	Log files of printing activities.
model	Link to /usr/lib/lp/model	The standard printer interface program.
printers	Directory	Directories for each local printer. Each directory contains configuration information and alert files for an individual printer.
pwheels	Directory	Print wheel or cartridge files.

These configuration files serve a similar function to the `/etc/printcap` file on `lpd`-based print servers.

Note – You can check the contents of the configuration files, but you should not edit them directly. Instead, use the `lpadmin(1M)` command to make configuration changes. Your changes will be written to the configuration files in the `/etc/lp` directory. The `lpsched` daemon administers and updates the configuration files.

The `/etc/lp/printers` directory has a subdirectory for each local printer known to the system. The following example shows the `/etc/lp/printers` subdirectories of printers `sparc1` and `luna`.

```
$ ls -l /etc/lp/printers
drwxrwxr-x 2 lp lp 512 Jan 23 23:53 luna
drwxrwxr-x 2 lp lp 512 Jan 11 17:50 sparc1
```

The following table describes the files within each of the printer-specific directories.

File Name	Description
alert.sh	Shell to execute in response to alerts
alert.vars	Alert variables
configuration	Configuration file
users.deny	List of users to whom printer access is denied
comment	Printer description

The configuration file for the printer luna, /etc/lp/printers/luna/configuration, would typically appear as follows:

```
Banner: on: Always
Content types: PS
Device: /dev/term/b
Interface: /usr/lib/lp/model/standard
Printer type: PS
Modules: default
```

The terminfo Database

The /usr/share/lib directory contains the terminfo database directory, which contains definitions for many types of terminals and printers. The LP print service uses information in the terminfo database to initialize a printer, to establish a selected page size, character pitch, line pitch, and character set, as well as to communicate the sequence of codes to a printer.

Each printer is identified in the terminfo database with a short name. See "Printer Type" on page 46 for a description of the structure of the terminfo database. If necessary, you can add entries to the terminfo database, but it is a tedious and time-consuming process. See "Adding a terminfo Entry for an Unsupported Printer" on page 135.

Daemons and LP Internal Files

The /usr/lib/lp directory contains daemons and files used by the LP print service, as described in the table below.

TABLE 8–3 Contents of the `/usr/lib/lp` Directory

File	Type	Description
bin	Directory	Contains files for generating printing alerts, slow filters, and queue management programs.
lpsched	Daemon	Manages scheduling of LP print requests.
model	Directory	Contains the standard printer interface program.
postscript	Directory	Contains all PostScript filter programs provided by the LP print service. These filters come with descriptor files in the `/etc/lp/fd` directory that tell the LP print service the characteristics of the filters and where to locate them.

LP Print Service Log Files

The LP print service maintains two sets of log files described in the following table.

Log File Name	Description
syslogd(1M)	Set `lpr.debug` in `/etc/syslog.conf` to enable LP print service logging
/var/spool/lp	A list of current requests that are in the print queue
/var/lp/logs/requests	An ongoing history of print requests

Print Queue Logs

The scheduler for each system keeps a log of print requests in the directories `/var/spool/lp/tmp/`*system* and `/var/spool/lp/requests/`*system*. Each print request has two files (one in each directory) that contain information about the request. The information in the `/var/spool/lp/requests/`*system* directory can be accessed only by root or lp. The information in the `/var/spool/lp/tmp/`*system* can be accessed only by the user who submitted the request, root, or lp.

The following example shows the contents of the `/var/spool/lp/tmp/terra` directory:

```
$ ls /var/spool/lp/tmp/terra
20-0 21-0
terra$ cat 21-0
```

```
C 1
D slw2
F /etc/default/login
P 20
t simple
U tamiro
s 0x1000
```

These files remain in their directories only as long as the print request is in the queue. Once the request is finished, the information in the files is combined and appended to the file /var/lp/logs/requests, which is described in the next section.

Use the information in the /var/spool/lp/logs if you need to track the status of a print request that is currently in the queue.

History Logs

The LP print service records a history of printing services in two log files: lpsched and requests. These log files are located in the /var/lp/logs directory. You can use the information in these logs to diagnose and troubleshoot printing problems. This is an example of the contents of the /var/lp/logs directory:

```
# cd /var/lp/logs
# ls
lpsched.1    requests     requests.2
lpsched      lpsched.2    requests.1
#
```

The two most important log files for troubleshooting is the lpsched log, which contains information about local printing requests

The requests log contains information about print requests that are completed and no longer in the print queue. Once a request is finished printing, the information in the /var/spool/lp log files is combined and appended to the /var/lp/logs/requests log.

The requests log has a simple structure, so that you can extract data using common UNIX shell commands. Requests are listed in the order they are printed, and are separated by lines showing their request IDs. Each line below the separator line is marked with a single letter that identifies the kind of information contained in that line. Each letter is separated from the data by a single space.

The following example shows the contents of a requests log:

```
# pwd
/var/lp/logs
# tail requests.2
= slw2-20, uid 200, gid 200, size 5123, Tue Jun 17 10:16:10 MDT
1998
z slw2
```

```
C 1
D slw2
F /etc/motd
P 20
t simple
U irving
s 0x0100
#
```

The table below shows the letter codes and the content of their corresponding lines in the LP `requests` log.

TABLE 8–4 Letter Codes in the LP `requests` Log

Letter	Content of Line
=	The separator line. It contains the following items: request ID, user ID (UID), and group IDs (GIDs) of the user, the total number of bytes in the original (unfiltered) file size, and the time when the request was queued.
C	The number of copies printed.
D	The printer or class destination or the word any.
F	The name of the file printed. The line is repeated for each file printed; files were printed in the order shown.
f	The name of the form used.
H	One of three types of special handling: resume, hold, and immediate.
N	The type of alert used when the print request was successfully completed. The type is the letter M if the user was notified by email or W if the user was notified by a message to the terminal.
O	The printer-dependent -o options (for example, nobanner).
P	The priority of the print request.
p	The list of pages printed.
r	A single-letter line that is included if the user asked for "raw" processing of the files (the lp -r command).
S	The character set, print wheel, or cartridge used.
s	The outcome of the request, shown as a combination of individual bits expressed in hexadecimal form. Several bits are used internally by the print service. The bits and what they mean are describe in the table below.
T	The title placed on the banner page.
t	The type of content found in the files.
U	The name of the user who submitted the print request.

TABLE 8–4 Letter Codes in the LP `requests` Log   *(Continued)*

Letter	Content of Line
x	The slow filter used for the print request.
Y	The list of special modes for the print filters used to print the request.
z	The printer used for the request. This printer differs from the destination (the D line) if the request was queued for any printer or a class of printers, or if the request was moved to another destination.

The table below shows the outcome codes in the LP `requests` log and their descriptions.

TABLE 8–5 Outcome Codes in the LP requests Log

Outcome Code	Description
0x0001	The request was held pending resume.
0x0002	Slow filtering is running.
0x0004	Slow filtering finished successfully.
0x0008	The request is on the printer.
0x0010	Printing finished successfully.
0x0020	The request was held pending user change.
0x0040	The request was canceled.
0x0080	The request will print next.
0x0100	The request failed filtering or printing.
0x0200	The request is in transit to a remote printer. (obsolete)
0x0400	The user will be notified.
0x0800	A notification is running.
0x1000	A remote system has accepted the request. (obsolete)
0x2000	The administrator placed a hold on the request.
0x4000	The printer had to change filters.
0x8000	The request is temporarily stopped.

Spooling Directories

Files queued for printing are stored in the `/var/spool/lp` directory until they are printed, which might be only seconds. The table below shows the contents of the `/var/spool/lp` directory.

TABLE 8–6 Contents of the `/var/spool/lp` Directory

File	Type	Description
SCHEDLOCK	File	Lock file for the scheduler. Check for this file if the scheduler dies and will not restart.
admins	Directory	Link to `/etc/lp`.
bin	Directory	Link to `/usr/lib/lp/bin`.
logs	Link	Link to `../lp/logs` where completed print requests are logged.
model	Link	Link to `/usr/lib/lp/model`.
requests	Directory	Directory that contains subdirectories for each configured printer where print requests are logged until printed. Users cannot access this log.
system	Directory	A print status file for the system.
temp	Link	Link to `/var/spool/lp/tmp/`*hostname*, which contains the spooled requests.
tmp	Directory	Directory for each configured printer where print requests are logged until printed. Changes to existing print requests are also recorded in this log.

LP Print Service Commands

The table below lists frequently used LP print service commands. You must be root or `lp` to use the `1M` commands.

TABLE 8–7 Quick Reference to LP Print Service Commands

Command	Task
enable(1)	Activate a printer
cancel(1)	Cancel a print request
lp(1)	Send one or more file(s) to a printer
lpstat(1)	Report the status of the LP print service

TABLE 8–7 Quick Reference to LP Print Service Commands   *(Continued)*

Command	Task
disable	Deactivate one or more printers
accept(1M)	Permit print requests to be queued for a specific destination
reject	Prevent print requests from being queued for a specific destination
lpadmin(1M)	Set up or change printer configuration
lpfilter(1M)	Set up or change filter definitions
lpforms(1M)	Set up or change preprinted forms
lpadmin(1M)	Mount a form
lpmove(1M)	Move output requests from one destination to another
lpsched(1M)	Start the LP print service scheduler
lpshut(1M)	Stop the LP print service scheduler
lpusers(1M)	Set or change the default priority and priority limits that can be requested by users of the LP print service

Functions of the LP Print Service

The LP print service performs the following functions:

- Administers files and schedules local print requests
- Receives and schedules network requests
- Filters files (if necessary) so they print properly
- Starts programs that interface with the printers
- Tracks the status of jobs
- Tracks forms mounted on the printer
- Tracks print wheels currently mounted
- Delivers alerts to mount new forms or different print wheels
- Delivers alerts about printing problems

The table below describes the directory structure and commands.

How LP Administers Files and Schedules Local Print Requests

The LP print service has a scheduler daemon called lpsched. The scheduler daemon updates the LP system files with information about printer setup and configuration.

The `lpsched` daemon schedules all local print requests on a print server, as shown in the figure below, whether users issue the requests from an application or from the command line. Also, the scheduler tracks the status of printers and filters on the print server. When a printer finishes a request, the scheduler schedules the next request, if there is one, in the queue on the print server.
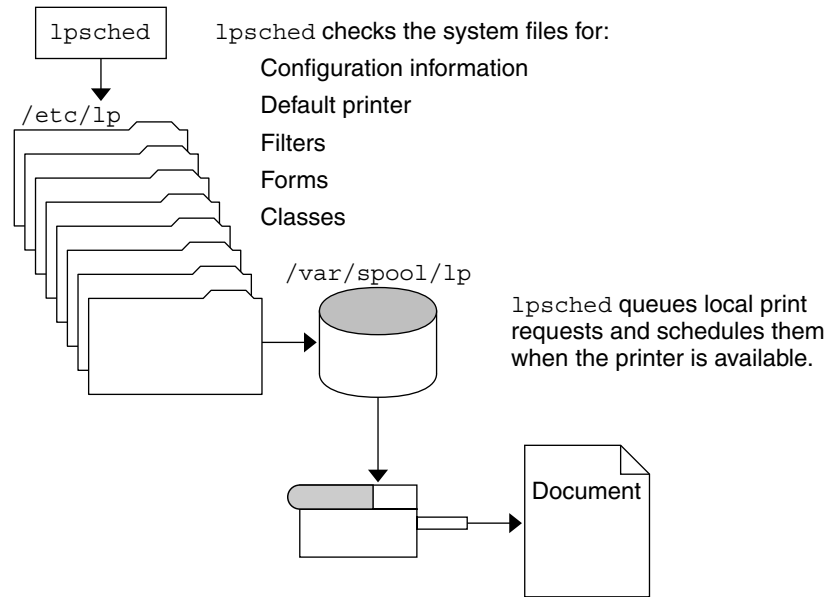
FIGURE 8–1 The `lpsched` Daemon Schedules Local Print Requests

Each print server must have *only* one LP scheduler running. The scheduler is started when a system is booted (or enters run level 2) by the control script `/etc/rc2.d/S80lp`. Without rebooting the systems, you can stop the scheduler with the `/usr/lib/lp/lpshut` command and restart the scheduler with the `lpsched` command. The scheduler for each system manages requests issued to the system by the `lp` commands.

Scheduling Network Print Requests

Each print client communicates directly with a print sever over the network. The communication is done between the requesting command (`lp`, `lpstat`, `cancel`, `lpr`, `lpq`, or `lprm`) and the print service on the print server. Doing so, reduces the print system overhead on client only systems, improving scalability, performance and accuracy of data.

Print servers now listen for print request with the Internet services daemon (inetd). Upon hearing a request for print service from the network, inetd starts a program called the "protocol adaptor" (in.lpd). The protocol adaptor translates the print request and communicates it to the print spooler, returning the results to the requester. It starts on demand and exits when it has serviced the network request. This eliminates idle system overhead for printing. It also eliminates any additional system configuration for networked printing support as was the case in previous versions of Solaris printing.

Filtering Print Files

Print filters are programs on the print server that convert the content of a queued file from one format to another.

A print filter can be as simple or as complex as needed. The SunOS release provides print filters in the /usr/lib/lp/postscript directory that cover most situations where the destination printer requires the data to be in PostScript format. If you need filters for non-PostScript printers, you have to create the filters and add them to the systems that need them.

A set of *print filter descriptor files* are provided in the /etc/lp/fd directory. These descriptor files describe the characteristics of the filter (for example, fast or slow filter), and point to the filter program (for example, /usr/lib/lp/postscript/postdaisy).

What the Printer Interface Program Does

The LP print service interacts with other parts of the operating system. It uses a standard printer interface program to:

- Initialize the printer port, if necessary. The standard printer interface program uses the stty command to initialize the printer port.
- Initialize the printer. The standard printer interface program uses the terminfo database and the TERM shell variable to find the appropriate control sequences.
- Print a banner page, if necessary.
- Print the correct number of copies specified by the print request.

The LP print service uses the standard interface program (found in the /usr/lib/lp/model directory) unless you specify a different one. You can create custom interface programs, but you must make sure that the custom program does not terminate the connection to the printer or interfere with proper printer initialization.

How the `lpsched` Daemon Tracks the Status of Print Requests

The `lpsched` daemon on both the print server and print client keeps a log of each print request that it processes and notes any errors that occur during the printing process. This log is kept in the `/var/lp/logs/lpsched` file. Every night, the `lp` cron job renames `/var/lp/logs/lpsched` to a new `lpsched.n` file and starts a new log file. If errors occur or jobs disappear from the print queue, you can use the log files to determine what `lpsched` has done with a printing job.

Cleaning Out Log Files

The `lpsched` and `requests` log files in the `/var/lp/logs` directory grow as information is appended. The LP print service uses a default `cron` job to clean out the log files. The `lp` cron job is located in the `/var/spool/cron/crontabs/lp` file. It periodically moves the contents of the log files. The contents of *log* are moved to *log*.1, and the contents of *log*.1 are moved to *log*.2. The contents of *log*.2 are lost (that is, replaced by the former contents of *log*.1) when *log*.2 gets overwritten.

```
# pwd
/var/lp/logs
# tail requests
s 0x1010
= slw2-20, uid 200, gid 200, size 5123, Mon Jun 16 12:27:33 MDT
1997
z slw2
C 1
D slw2
F /etc/motd
P 20
t simple
U irving
s 0x1010
#
```

▼ How to Change Frequency of Printer Request Log Rotation

Starting with the Solaris 2.6 release, the `requests` log file on the printer server is rotated weekly rather than daily. You can change the rotation interval back to daily if the printer server is busy.

1. **Become superuser or `lp` on the printer server.**

2. **Set the** EDITOR **environment variable.**

   ```
   # EDITOR=vi
   # export EDITOR
   ```

3. **Edit the** lp crontab **file.**

   ```
   # crontab -e lp
   ```

4. **Change the first line of the file which rotates the** requests **log files every Sunday (**0**) to an asterisk (*) for daily rotation:**

   ```
   13 3 * * * cd /var/lp/logs; if [ -f requests ]; then if
   [ -f requests.1 ]; then /bin/mv requests.1 requests.2; fi; /usr/bin/cp
   requests requests.1; >requests; fi
   ```

5. **Save the file and exit.**

How Local Printing Works

The figure below shows what happens when a user submits a request to print a PostScript file on a *local* printer, which is a printer connected to the user's system. The local system does all processing; however, the print request follows the same path it would if the client and server were separate systems. Requests always flow from client to server following the same path.

A user submits
a print request.

Local
system

lpsched

/var/spool/lp

lp sends the request to
the lpsched daemon.

lpsched spools
the print request.

lpsched matches
the printer type and the
file content type, and
identifies the default
printer for the system.

Requests

lpsched filters the
job. (It also can put the
filtered output back into
the spooling area.)

file.ps

file.ps

When the printer is
free, lpsched starts the
printer's specified interface
program on the serial port.

Interface
program

The Interface program:

Prints the banner page

Catches faults

Depending on the fault
policy, it waits to be reset,
continues, or begins job over.

The interface program
uses the lpcat program
to download the file into
the serial port.

Document

FIGURE 8–2 The Local Printing Process

How Remote Printing Works

The figure below shows what happens when a user on a Solaris print client submits a print request to a lpd-based print server. The command opens a connection and handles it's own communications with the print server directly.
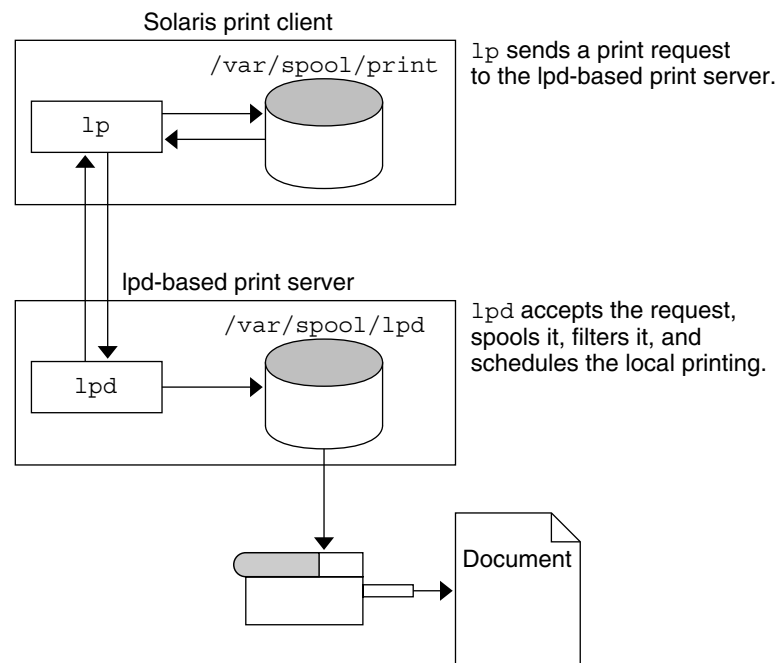
FIGURE 8–3 Printing Between a Solaris Print Client and a lpd-based Print Server

The figure below shows a lpd-based print client submitting a print request to a Solaris print server. The `lpd` daemon handles the local part of the print request and the connection to the print server. On the print server, the network listen process, `inetd`, waits for network printing requests and starts a protocol adaptor to service the request. The protocol adaptor communicates with the `lpsched` daemon, which processes the request on the print server.

The figure below shows what happens when a user on a Solaris print client submits a print request to a Solaris print server. The print command on the print client handles the local part of each print request by communicating directly with the print server.

The `inetd` process on the print server monitors network printing requests and starts a protocol adaptor to communicate with the `lpsched` daemon on the print server, which processes the print request.

Solaris print client

/var/spool/print

lp

Solaris print server

inetd

inetd listens for a request
and starts in.lpd. in.lpd
looks at the request and
loads bsd_lpsched.so.

/var/spool/lp

in.lpd

bsd_lpsched.so

in.lpd passes the request
through bsd_lpsched.so
to lpsched for local printing.

lpsched

Document
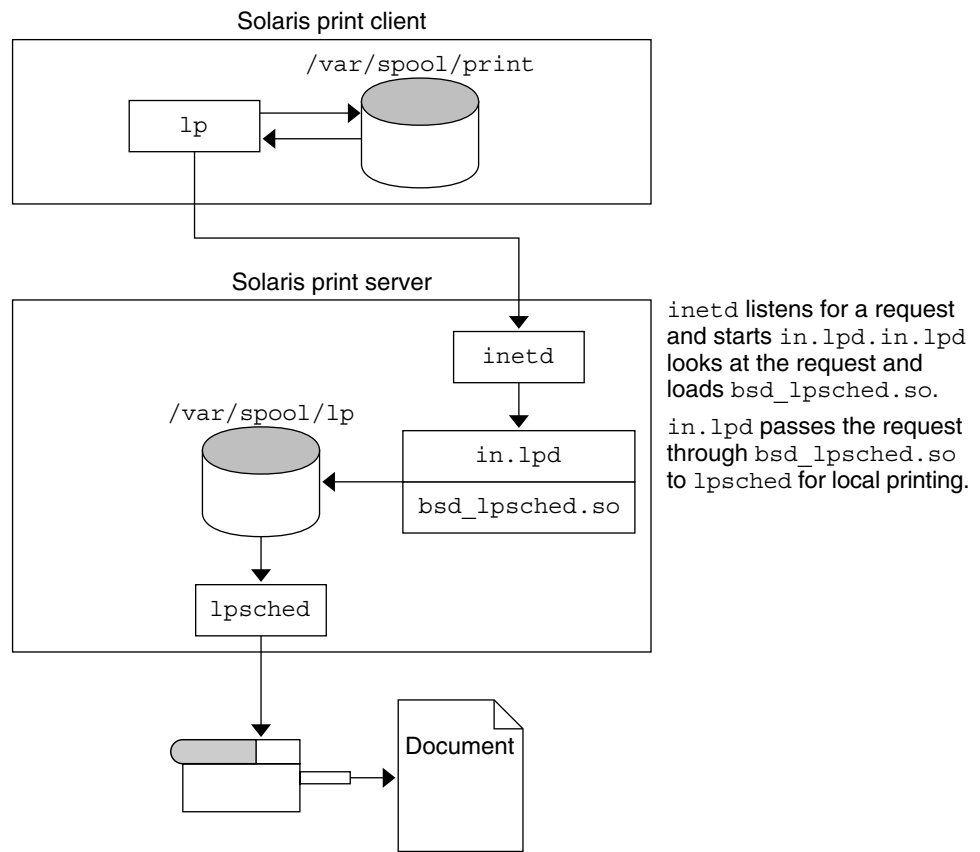
FIGURE 8–4 Printing Between a Solaris Print Client and a Solaris Print Server

Managing Terminals and Modems Topics

This section provides instructions for managing terminals and modems. This section contains these chapters.

Chapter 10	Provides overview information about terminals and modems.
Chapter 11	Provides step-by-step instructions for setting up terminals and modems.
Chapter 12	Provides step-by-step instructions for using SAF commands to set up terminals and modems.

Managing Terminals and Modems (Overview)

This chapter provides the overview information for managing terminals and modems. This is a list of the overview information in this chapter.

- "Terminals, Modems, Ports, and Services" on page 180
- "Tools for Managing Terminals and Modems" on page 182
- "Admintool" on page 183
- "Service Access Facility" on page 183

For step-by-step instructions about how to set up terminals and modems with Admintool, see Chapter 11.

For step-by-step instructions about how to set up terminals and modems with the Service Access Facility (SAF), see Chapter 12.

What's New in Managing Terminals and Modems?

The Solaris Management Console provides a Serial Ports tool for setting up terminals and modems. For information on starting the Solaris Management Console, see "Starting the Solaris Management Console" in *System Administration Guide: Basic Administration*. For information on step-by-step instructions for setting up a terminal or modem, see the console online help.

Terminals, Modems, Ports, and Services

Terminals and modems provide both local and remote access to system and network resources. Setting up terminals and modem access is an important responsibility of a system administrator. This section explains some of the concepts behind modem and terminal management in the Solaris environment.

Terminals

Your system's bit-mapped graphics display is not the same as an alphanumeric terminal, which connects to a serial port and displays only text. You don't have to perform any special steps to administer the graphics display.

Modems

Modems can be set up in three basic configurations:

- Dial-out
- Dial-in
- Bidirectional

A modem connected to your home computer might be set up to provide *dial-out* service, meaning you can access other computers from your own home, but nobody outside can gain access to your machine.

Dial-in service is just the opposite. It allows people to access a system from remote sites, but it does not permit calls to the outside world.

Bidirectional access, as the name implies, provides both dial-in and dial-out capabilities.

Ports

A *port* is a channel through which a device communicates with the operating system. From a hardware perspective, a port is a "receptacle" into which a terminal or modem cable might be plugged.

However, a port is not strictly a physical receptacle, but an entity with hardware (pins and connectors) and software (a device driver) components. A single physical receptacle often provides multiple ports, allowing connection of two or more devices.

Common types of ports include serial, parallel, small computer systems interface (SCSI), and Ethernet.

A *serial* port, using a standard communications protocol, transmits a byte of information bit-by-bit over a single line.

Devices that have been designed according to RS-232-C or RS-423 standards (this includes most modems, alphanumeric terminals, plotters, and some printers) can be plugged interchangeably (using standard cables) into serial ports of computers that have been similarly designed.

When many serial port devices must be connected to a single computer, it might be necessary to add an *adapter board* to the system. The adapter board, with its driver software, provides additional serial ports for connecting more devices than could otherwise be accommodated.

Services

Modems and terminals gain access to computing resources via the serial port software. The serial port software must be set up to provide a particular "service" for the device attached to the port. For example, you can set up a serial port to provide bidirectional service for a modem.

Port Monitors

The main mechanism for gaining access to a service is through a *port monitor*. A port monitor is a program that continuously monitors for requests to log in or access printers or files.

When a port monitor detects a request, it sets whatever parameters are required to establish communication between the operating system and the device requesting service. Then the port monitor transfers control to other processes that provide the services needed.

The table below describes the two types of port monitors included in the Solaris environment.

TABLE 10–1 Port Monitor Types

Port Monitor	Description
listen(1M)	Controls access to network services, such as handling remote print requests prior to the Solaris 2.6 release. The default Solaris operating environment no longer uses this port monitor type.

TABLE 10–1 Port Monitor Types   *(Continued)*

Port Monitor	Description
ttymon(1M)	Provides access to the login services needed by modems and alphanumeric terminals. Admintool automatically sets up a ttymon port monitor to process login requests from these devices.

You might be familiar with an older port monitor called getty(1M). The new ttymon is more powerful; a single ttymon can replace multiple occurrences of getty. Otherwise, these two programs serve the same function.

Tools for Managing Terminals and Modems

The table below lists the recommended tools for managing terminals and modems. Table 10–3 lists specific differences in functionality between the Service Access Facility (SAF) and Admintool: Serial Ports.

TABLE 10–2 Recommended Tools For Managing Terminals and Modems

If You Want The Tool That Is ...	Then Use ...	To Start This Tool See ...
The most comprehensive	Service Access Facility (SAF) commands	"Service Access Facility" on page 183
The quickest setup	Solaris Management Console Serial Ports tool	Solaris Management Console online help
The quickest setup	Admintool graphical user interface (for local systems only)	Chapter 11

TABLE 10–3 Functionality Differences Between Admintool and Service Access Facility

If You Need To ...	Then Use ...	Comment
Inform users that a port is disabled	Service Access Facility ttyadm -i	ttyadmin -i specifies the inactive (disabled) response message. The message is sent to a terminal or modem when a user attempts to log in when the port is disabled. This functionality is not provided when a port is disabled using Admintool.

TABLE 10–3 Functionality Differences Between Admintool and Service Access Facility
(Continued)

If You Need To ...	Then Use ...	Comment
Keep the modem connection when a user logs off a host	Service Access Facility `ttyadm -h`	`ttyadm -h` specifies that the system will not hang up on a modem before setting or resetting to the default or specified value. If `ttyadm -h` is not used, when the user logs out of a host, the host will hang up the modem.
Require the user to type a character before the system displays a prompt	Service Access Facility `ttyadm -r`	`ttyadm -r` specifies that `ttymon` should require the user to type a character or press Return a specified number of times before the login prompt appears. When `-r` is not specified, pressing Return one or more times will print the prompt anyway. This option prevents a terminal server from issuing a welcome message that the Solaris host might misinterpret to be a user trying to log in. Without the `-r` option, the host and terminal server might begin looping and printing prompts to each other.

Admintool

Admintool: Serial Ports sets up the serial port software to work with terminals and modems by calling the `pmadm` command with the appropriate information. It also provides:

- Templates for common terminal and modem configurations
- Multiple port setup, modification, or deletion
- Quick visual status of each port

Service Access Facility

The SAF is the tool used for administering terminals, modems, and other network devices. In particular, SAF enables you to set up:

- `ttymon` and `listen` port monitors (using the `sacadm` command)

- `ttymon` port monitor services (using the `pmadm` and `ttyadm` commands)

- `listen` port monitor services (using the `pmadm` and `nlsadmin` commands)

- And troubleshoot `tty` devices

- And troubleshoot incoming network requests for printing service

- And troubleshoot the Service Access Controller (using the `sacadm` command)

The SAF is an open-systems solution that controls access to system and network resources through `tty` devices and local-area networks (LANs). SAF is not a program. It is a hierarchy of background processes and administrative commands.

Setting Up Terminals and Modems (Tasks)

This chapter provides step-by-step instructions for setting up terminals and modems using Admintool. This is a list of the step-by-step instructions in this chapter.

- "How to Start Admintool" on page 190
- "How to Set Up a Terminal" on page 191
- "How to Set Up a Modem" on page 192
- "How to Set Up a Modem for Use With UUCP" on page 193
- "How to Initialize a Port" on page 195
- "How to Disable a Port" on page 196
- " How to Remove a Port Service" on page 197

For overview information about terminals and modems, see Chapter 10.

Setting Up Terminals and Modems

When setting up serial port information, start Admintool, and select Serial Ports from the Browse menu. Select a serial port from the Serial Ports window and then choose Modify from the Edit menu to bring up the Modify Serial Port window. This window provides access to the port templates and provides information on the port in three levels of detail—Basic, More, and Expert.
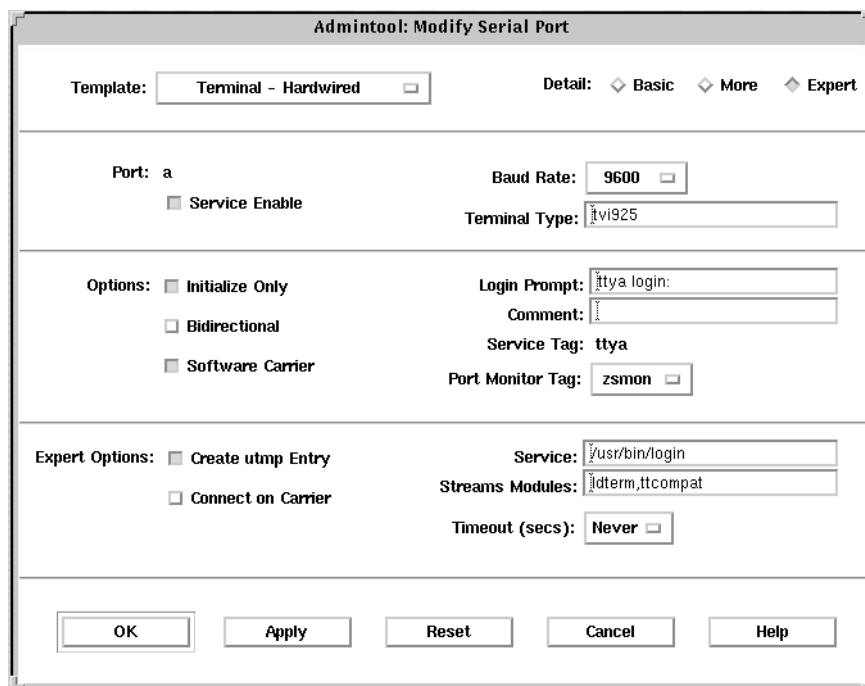
Note – The Modify Serial Port window appears in the Basic detail mode. To view More or Expert details, select the More or Expert option from the Detail menu.

The descriptions of each item in the Modify Serial window are listed in the table below.

TABLE 11–1 Modify Serial Port Window Items

Detail	Item	Description
Basic	Port	Lists the port or ports you selected from Serial Ports main window.
	Service	Specifies that the service for the port is turned on (enabled).
	Baud Rate	Specifies the line speed used to communicate with the terminal. The line speed represents an entry in the /etc/ttydefs file.

TABLE 11–1 Modify Serial Port Window Items   *(Continued)*

Detail	Item	Description
	Terminal Type	Shows the abbreviation for the type of terminal, for example, ansi or vt100. Similar abbreviations are found in `/etc/termcap`. This value is set in the `$TERM` environment variable.
More	Option: Initialize Only	Specifies that the port software is initialized but not configured.
	Option: Bidirectional	Specifies that the port line is used in both directions.
	Option: Software Carrier	Specifies that the software carrier detection feature is used. If the option is *not* checked, the *hardware* carrier detection signal is used.
	Login Prompt	Shows the prompt displayed to a user after a connection is made.
	Comment	Shows the comment field for the service.
	Service Tag	Lists the service tag associated with this port—typically an entry in the `/dev/term` directory.
	Port Monitor Tag	Specifies the name of the port monitor to be used for this port. Note: The default monitor is typically correct.
Expert	Create `utmpx` Entry	Specifies that a `utmpx` entry is created in the accounting files upon login. Note: This item must be selected if a login service is used. See the Service item.
	Connect on Carrier	Specifies that a port's associated service is invoked immediately when a connect indication is received.
	Service	Shows the program that is run upon connection.
	Streams Modules	Shows the STREAMS modules that are pushed before the service is invoked.
	Timeout (secs)	Specifies the number of seconds before a port is closed if the open process on the port succeeds and no input data is received.

Setting Up Terminals

The table below describes the menu items (and their default values) when setting up a terminal using Serial Ports.

TABLE 11–2 Terminal - Hardwired Default Values

Detail	Item	Default Value
Basic	Port	—
	Service	Enabled
	Baud Rate	`9600`
	Terminal Type	—
More	Option: Initialize Only	no
	Option: Bidirectional	no
	Option: Software Carrier	yes
	Login Prompt	`login:`
	Comment	Terminal - Hardwired
	Service Tag	—
	Port Monitor Tag	`zsmon`
Expert	Create `utmpx` Entry	yes
	Connect on Carrier	no
	Service	`/usr/bin/login`
	Streams Modules	`ldterm,ttcompat`
	Timeout (secs)	Never

Setting Up Modems

The table below describes the three modem templates available when setting up a modem using Serial Ports.

TABLE 11–3 Modem Templates

Modem Configuration	Description
Dial-In Only	Users can dial in to the modem but cannot dial out.
Dial-Out Only	Users can dial out from the modem but cannot dial in.
Bidirectional	Users can either dial in or out from the modem.

The table below describes the default values of each template.

TABLE 11–4 Modem Template Default Values

Detail	Item	Modem - Dial-In Only	Modem - Dial-Out Only	Modem - Bidirectional
Basic	Port	—	—	—
	Service	Enabled	Enabled	Enabled
	Baud Rate	9600	9600	9600
	Terminal Type	—	—	—
More	Option: Initialize Only	yes	no	no
	Option: Bidirectional	no	no	yes
	Option: Software Carrier	no	no	no
	Login Prompt	login:	login:	login:
	Comment	Modem - Dial-In Only	Modem - Dial-Out Only	Modem - Bidirectional
	Service Tag	—	—	—
	Port Monitor Tag	zsmon	zsmon	zsmon
Expert	Create utmpx Entry	yes	yes	yes
	Connect on Carrier	no	no	no
	Service	/usr/bin/login	/usr/bin/login	/usr/sbin/login
	Streams Modules	ldterm,ttcompat	ldterm,ttcompat	ldterm,ttcompat
	Timeout (secs)	Never	Never	Never

The table below describes the default values for the Initialize Only template.

TABLE 11–5 Initialize Only - No Connection Default Values

Detail	Item	Default Value
Basic	Port	—
	Service	Enabled
	Baud Rate	9600

TABLE 11–5 Initialize Only - No Connection Default Values   *(Continued)*

Detail	Item	Default Value
	Terminal Type	—
More	Option: Initialize Only	yes
	Option: Bidirectional	no
	Option: Software Carrier	no
	Login Prompt	`login:`
	Comment	Initialize Only - No Connection
	Service Tag	—
	Port Monitor Tag	`zsmon`
Expert	Create `utmpx` Entry	yes
	Connect on Carrier	no
	Service	`/usr/bin/login`
	Streams Modules	`ldterm,ttcompat`
	Timeout (secs)	Never

▼ How to Start Admintool

1. **Verify that the following prerequisites are met. To use Admintool, you must:**

   - Have a bit-mapped display monitor. The Admintool software can be used only on a system with a console that is a bit-mapped screen such as a standard display monitor that comes with a Sun workstation.

   - Be running an X Window System, such as the CDE environment.

   - Be a member of the `sysadmin` group (group 14).

   If you want to perform administration tasks on a system with an ASCII terminal as the console, use Solaris commands instead.

2. **Start Admintool.**

   ```
   $ admintool &
   ```

   The Users main window is displayed.

▼ How to Set Up a Terminal

1. **Start Admintool, if it's not already running.**

   See "How to Start Admintool" on page 190 for more information on starting Admintool.

2. **Select Serial Ports from the Browse menu.**

   The Serial Ports menu is displayed.

3. **Select the port or ports that will be used with a terminal.**

4. **Choose Modify from the Edit menu.**

   The Modify Serial Port window appears in the Basic Detail mode. To enter additional details, select either the More or Expert Detail modes.

5. **Choose Terminal-Hardwired from the Use Template menu.**

   See Table 11–2 for a description of the Terminal–Hardware menu items.

6. **Change values of template entries if desired.**

7. **Click on OK to configure the port.**

8. **Use the** `pmadm` **command to verify the terminal service has been added.**

   ```
   $ pmadm -l -s ttya
   ```

Example—Completed Modify Window to Set Up a Terminal

```
                     Admintool: Modify Serial Port

   Template:    Terminal - Hardwired  ▭        Detail:  ◇ Basic  ◇ More  ◆ Expert

       Port:  a                          Baud Rate:   9600  ▭
              ☐ Service Enable         Terminal Type: │tvi925

   Options:  ☐ Initialize Only        Login Prompt: │ttya login:
             ☐ Bidirectional              Comment:  │
             ☐ Software Carrier        Service Tag: ttya
                                       Port Monitor Tag: zsmon ▭

 Expert Options: ☐ Create utmp Entry       Service: │/usr/bin/login
                 ☐ Connect on Carrier  Streams Modules: │ldterm,ttcompat
                                       Timeout (secs): Never ▭

        OK          Apply        Reset         Cancel         Help
```

▼ How to Set Up a Modem

1. Start Admintool, if it's not already running.

See "How to Start Admintool" on page 190 for more information on starting Admintool.

2. Select Serial Ports from the Browse menu.

The Serial Ports menu is displayed.

3. Select the port or ports that will be used with a modem.

4. Choose Modify from the Edit menu.

The Modify Serial Port window appears in the Basic Detail mode. To enter additional details, select either the More or Expert Detail modes.

5. Choose the modem configuration template from the Use Template menu that meets or most closely matches your modem service.

See Table 11–3 for a description of each template.

See Table 11–4 for the default values of each template. If a UUCP service will be used to dial in to your modem on a Solaris system, see "How to Set Up a Modem for Use
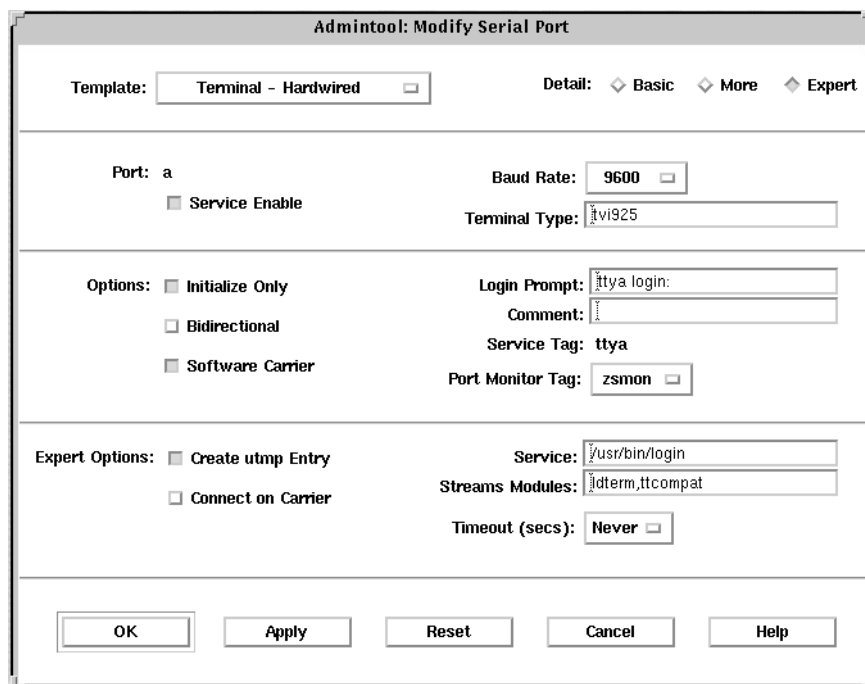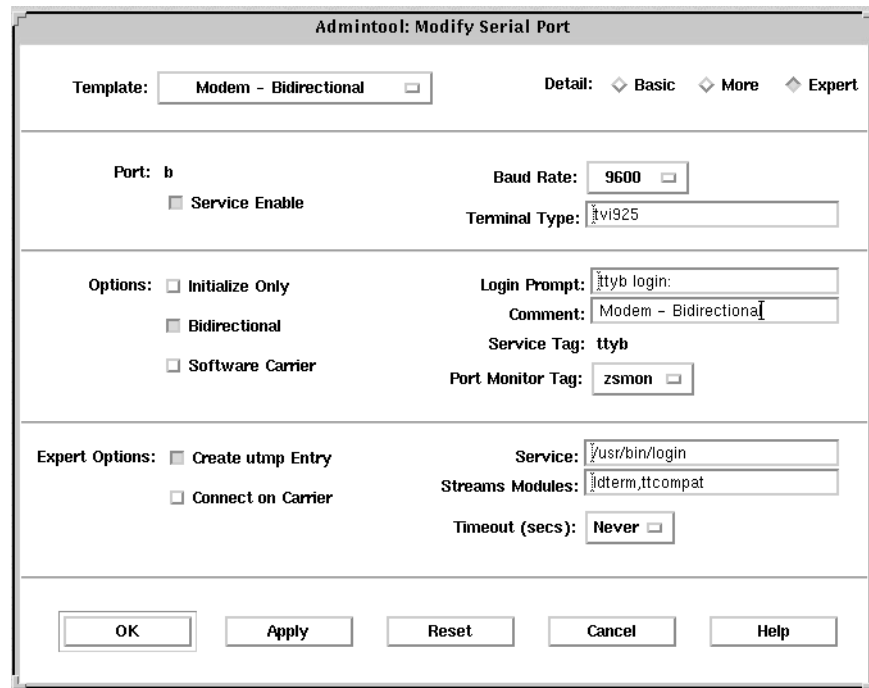
With UUCP" on page 193 for the rest of the procedure.

6. **Change values of template entries if desired.**

7. **Click on OK to configure the port.**

8. **Use the** pmadm **command to verify the modem service has been configured for use with UUCP.**

   ```
   $ pmadm -l -s ttyb
   ```

Example—Completed Modify Window to Set Up a Modem

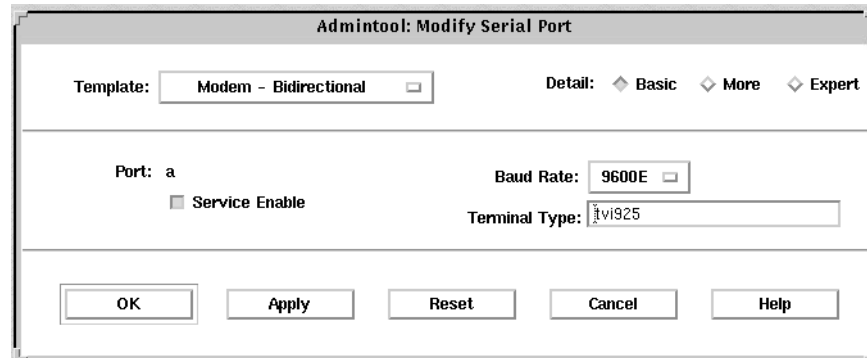## ▼ How to Set Up a Modem for Use With UUCP

UUCP sends information to a service using seven bits and even parity. Solaris modem configurations use eight bits and no parity for internationalization purposes. To set up your modem service to work with UUCP, follow these instructions.

1. **Start Admintool, if it's not already running.**

   See "How to Start Admintool" on page 190 for more information on starting Admintool.

2. **Select Serial Ports from the Browse menu.**

   The Serial Ports menu is displayed.

3. **Select the port or ports that will be used with a modem.**

4. **Choose Modify from the Edit menu.**

   The Modify Serial Port window appears in the Basic Detail mode. For additional details, select either the More or Expert Detail modes.

5. **Select Other from the Baud Rate menu.**

   A window listing baud rates from the /etc/ttydefs file is displayed.

6. **Enter a baud rate that provides seven bit, even parity service. Click on OK.**

7. **Change values of other template entries if desired.**

8. **Click on OK to configure the port.**

9. **Use the pmadm command to verify the modem service has been configured for use with UUCP.**

   ```
   $ pmadm -l -s ttya
   ```

Example—Completed Modify Window to Set Up a Modem for Use With UUCP

In this example, the 9600E baud rate was selected. This provides a service with a 9600 baud rate, seven bits, and even parity.
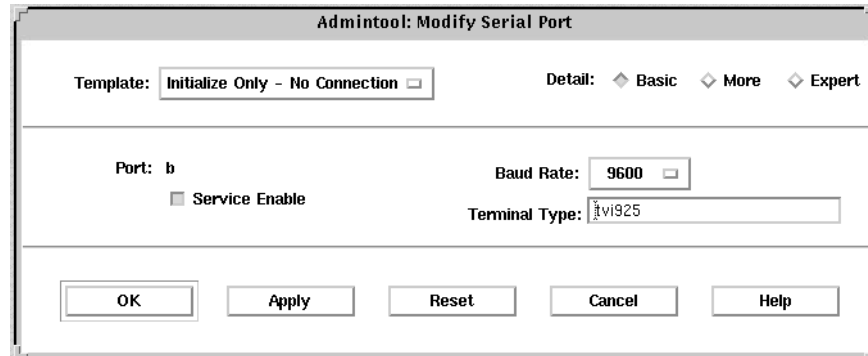
```
┌─────────────────────────────────────────────────────────────────┐
│                    Admintool: Modify Serial Port                  │
│                                                                   │
│   Template:      Modem – Bidirectional  ☐    Detail: ◇ Basic  ◇ More  ◇ Expert │
│  ───────────────────────────────────────────────────────────     │
│                                                                   │
│      Port:  a                            Baud Rate:  9600E  ☐     │
│            ☐ Service Enable             Terminal Type: │tvi925    │
│  ───────────────────────────────────────────────────────────     │
│                                                                   │
│    ┌────────┐  ┌────────┐  ┌────────┐  ┌────────┐  ┌────────┐     │
│    │   OK   │  │ Apply  │  │ Reset  │  │ Cancel │  │  Help  │     │
│    └────────┘  └────────┘  └────────┘  └────────┘  └────────┘     │
└─────────────────────────────────────────────────────────────────┘
```
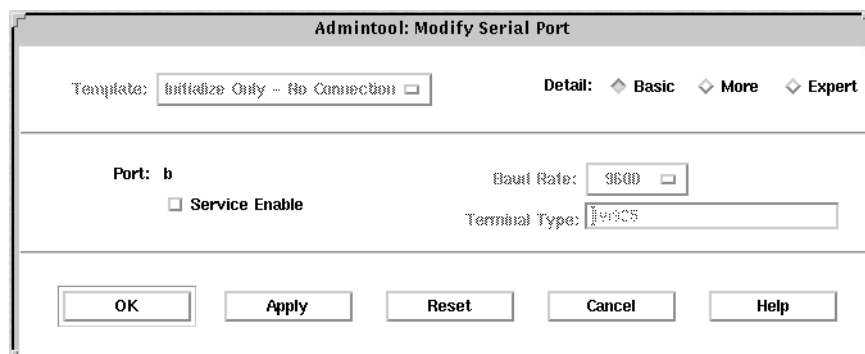
▼ How to Initialize a Port

1. **Start Admintool, if it's not already running.**

   See "How to Start Admintool" on page 190 for more information on starting Admintool.

2. **Select Serial Ports from the Browse menu.**

   The Serial Ports menu is displayed.

3. **Select the port or ports that you want to initialize.**

4. **Choose Modify from the Edit menu.**

   The Modify Serial Port window appears in the Basic Detail mode. To enter additional details, select either the More or Expert Detail modes.

5. **Choose Initialize Only - No Connection from the Use Template menu.**

   See Table 11–5 for a description of the Initialize Only - No Connection template.

6. **Click on OK to initialize the port.**

7. **Use the** pmadm **command to verify the port has been disabled.**

   ```
   $ pmadm -l -s ttyb
   ```

Example—Completed Modify Window to Initialize a Port

```
┌─────────────────────────────────────────────────────────────────────┐
│                    Admintool: Modify Serial Port                      │
├─────────────────────────────────────────────────────────────────────┤
│                                                                       │
│   Template: │ Initialize Only – No Connection □ │   Detail: ◇ Basic  ◇ More  ◇ Expert │
│                                                                       │
│  ─────────────────────────────────────────────────────────────────   │
│                                                                       │
│         Port:  b                          Baud Rate:  │ 9600  □ │     │
│              □ Service Enable                                          │
│                                        Terminal Type: │ tvi925      │  │
│                                                                       │
│  ─────────────────────────────────────────────────────────────────   │
│                                                                       │
│    │   OK   │    │  Apply  │    │  Reset  │    │ Cancel │    │  Help  │ │
│                                                                       │
└─────────────────────────────────────────────────────────────────────┘
```

▼ How to Disable a Port

1. **Start Admintool, if it's not already running.**

   See "How to Start Admintool" on page 190 for more information on starting Admintool.

2. **Select Serial Ports from the Browse menu.**

   The Serial Ports menu is displayed.

3. **Select the port or ports that you want to disable.**

4. **Choose Modify from the Edit menu.**

5. **Click on the Service Enable button to disable the port service in the Modify window.**

   This button acts as a toggle switch to enable or disable a port service.

6. **Click on OK to disable the port.**

7. **Use the** pmadm **command to verify the port service has been disabled.**

   ```
   $ pmadm -l -s ttya
   ```

Example—Completed Modify Window to Disable a Port

Admintool: Modify Serial Port

Template: Initialize Only — No Connection ▭   Detail: ◇ Basic ◇ More ◇ Expert

Port: b
☐ Service Enable

Baud Rate: 9600 ▭
Terminal Type: vt100

| OK | Apply | Reset | Cancel | Help |

▼ How to Remove a Port Service

1. **Start Admintool, if it's not already running.**

   See "How to Start Admintool" on page 190 for more information on starting Admintool.

2. **Select the port or ports that has a service you want to delete.**

3. **Choose Delete from the Edit menu.**

   You are asked if you really want to delete the service for the specified port or ports. You can cancel the delete operation or continue with it.

4. **Use the** pmadm **command to verify the port service has been deleted.**

   ```
   $ pmadm -l -s ttya
   ```

Troubleshooting Terminal and Modem Problems

If users are unable to log in over serial port lines after you have added a terminal or modem and set up the proper services, consider the following possible causes of failure.

- Check with the user.

  Malfunctions in terminals and modem use are typically reported by a user who has failed to log in or dial in. For this reason, it is best to begin troubleshooting by

checking for a problem on the desktop.

Some common reasons for login failure include:

- Login ID or password is incorrect.
- Terminal is waiting for X-ON flow control key (Control-q).
- Serial cable is loose or unplugged.
- Terminal configuration is incorrect.
- Terminal is shut off or otherwise has no power.

■ Check the terminal.

Continue to troubleshoot by checking the configuration of the terminal or modem. Determine the proper *ttylabel* for communicating with the terminal or modem. Verify that the terminal or modem settings match those of the *ttylabel*.

■ Check the terminal server.

If the terminal checks out, continue to search for the source of the problem on the terminal or modem server. Use the pmadm command to verify that a port monitor has been configured to service the terminal or modem and that it has the correct *ttylabel* associated with it.

```
$ pmadm -l -t ttymon
```

Examine /etc/ttydefs and double check the label definition against the terminal configuration. Use sacadm to check the port monitor's status. Use pmadm to check the service associated with the port the terminal uses.

■ Check the serial connection.

If the Service Access Controller is *starting* the TTY port monitor and pmadm reports that the service for the terminal's port is *enabled*, and if the terminal's configuration matches the port monitor's, then continue to search for the problem by checking the serial connection. A serial connection comprises serial ports, cables, and terminals. Test each of these parts by using it with two other parts that are known to be reliable.

Test all of the following:

- Serial ports
- Modems
- Cables
- Connectors

■ Do not use Admintool to modify serial port settings if the serial port is being used as a console. The correct procedure for changing console settings is by modifying the following line in the /etc/inittab file:

```
co:234:respawn:/usr/lib/saf/ttymon -g -h -p "'uname -n' console
login: " -T terminal_type -d /dev/console -l console -m
ldterm,ttcompat
```

■ If you are connecting a modem to an IA based system, verify the modem is supported by viewing the *Solaris 9 (Intel Platform Edition) Hardware Compatibility List*.

Setting Up Terminals and Modems With the Service Access Facility (Tasks)

This chapter explains in detail what a system or network administrator needs to know about the Service Access Facility (SAF) in the Solaris environment.

If you want to see examples of specific SAF commands, skip the first section, "Using the Service Access Facility" on page 199, and use the following list to find the instructions you need.

- "Using the Service Access Facility" on page 199
- "Overall Administration: `sacadm` Command" on page 200
- "Port Monitor Service Administrator: `pmadm` Command" on page 201
- "Port Monitors: TTY Monitor and Network Listener" on page 204
- "Administering `ttymon` Port Monitors" on page 207
- "Administering `ttymon` Services" on page 210
- "Service Access Facility Administration (Reference)" on page 214

For overview information about terminals and modems, see Chapter 10.

Using the Service Access Facility

The SAF is the tool used for administering terminals, modems, and other network devices. The top-level SAF program is the Service Access Controller (SAC). The SAC controls port monitors which you administer through the `sacadm` command. Each port monitor can manage one or more ports.

You administer the services associated with ports through the `pmadm` command. While services provided through SAC can differ from network to network, SAC and the administrative programs `sacadm` and `pmadm` are network independent.

The table below describes the SAF control hierarchy. The `sacadm` command is used to administer the SAC which controls the `ttymon` and `listen` port monitors.

The services of `ttymon` and `listen` are in turn controlled by `pmadm`. One instance of `ttymon` can service multiple ports and one instance of `listen` can provide multiple services on a network interface.

TABLE 12–1 SAF Control Hierarchy

Function	Program	Description
Overall Administration	sacadm	Command for adding and removing port monitors
Service Access Controller	sac	SAF's master program
Port Monitors	ttymon	Monitors serial port login requests
	listen	Monitors requests for network services
Port Monitor Service Administrator	pmadm	Command for controlling port monitors services
Services	logins; remote procedure calls; other	Services to which SAF provides access
Console Administration	console login	The console is automatically set up via an entry in the /etc/inittab file using ttymon-express mode. Do not use the pmadm or sacadm to manage the console directly. See "ttymon and the Console Port" on page 205 for more information.

Overall Administration: `sacadm` Command

The `sacadm` command is the top level of the SAF. The `sacadm` command primarily is used to add and remove port monitors such as `ttymon` and `listen`. Other `sacadm` functions include listing the current status of port monitors and administering port monitor configuration scripts.

Service Access Controller: SAC Program

The Service Access Controller program (SAC) oversees all port monitors. A system automatically starts SAC upon entering multiuser mode.

When SAC is invoked, it first looks for, and interprets, each system's configuration script, by which SAC customizes its environment. The modifications made to the SAC environment are inherited by all the "children" of the SAC. This inherited environment might be modified by the children.

After it has interpreted the per-system configuration script, the SAC program reads its administrative file and starts the specified port monitors. For each port monitor, SAC runs a copy of itself (SAC forks a child process). Each child then interprets its per-port monitor configuration script, if such a script exists.

Any modifications to the environment specified in the per-port monitor configuration script affect the port monitor and will be inherited by all its children. Finally, the child process runs the port monitor program using the command found in the SAC administrative file.

SAC Initialization Process

The following steps summarize what happens when SAC is first started:

1. The SAC program is spawned by `init` at run level two.
2. The SAC program reads `/etc/saf/_safconfig`, the per-system configuration script.
3. The SAC program reads `/etc/saf/_sactab`, the SAC administrative file.
4. The SAC program forks a child process for each port monitor it starts.
5. Each port monitor reads `/etc/saf/pmtag/_config`, the per-port monitor configuration script.

Port Monitor Service Administrator: `pmadm` Command

The `pmadm` command enables you to administer port monitors' services. In particular, you use the `pmadm` command to add or remove a service and to enable or disable a service. You can also install or replace per-service configuration scripts, or print information about a service.

Each instance of a service must be uniquely identified by a port monitor and a port. When you use the `pmadm` command to administer a service, you specify a particular port monitor via the *pmtag* argument, and a particular port via the *svctag* argument.

For each port monitor type, the SAF requires a specialized command to format port monitor-specific configuration data. This data is used by the `pmadm` command. For `ttymon` and `listen` type port monitors, these specialized commands are `ttyadm` and `nlsadmin`, respectively.

A Port Monitor at Work: `ttymon`

Whenever you attempt to log in via a directly connected modem or alphanumeric terminal, `ttymon` goes to work, as follows.

As shown in the figure below, the `init` program is the first process to be started at boot time. Consulting its administrative file (`/etc/inittab`), `init` starts other processes as they are needed. Listed among those processes is the SAC.

SAC, in turn, automatically starts up the port monitors designated in its administrative file (`/etc/saf/_sactab`). The figure below shows only a single `ttymon` port monitor.

After `ttymon` has been started, it monitors the serial port lines for service requests.
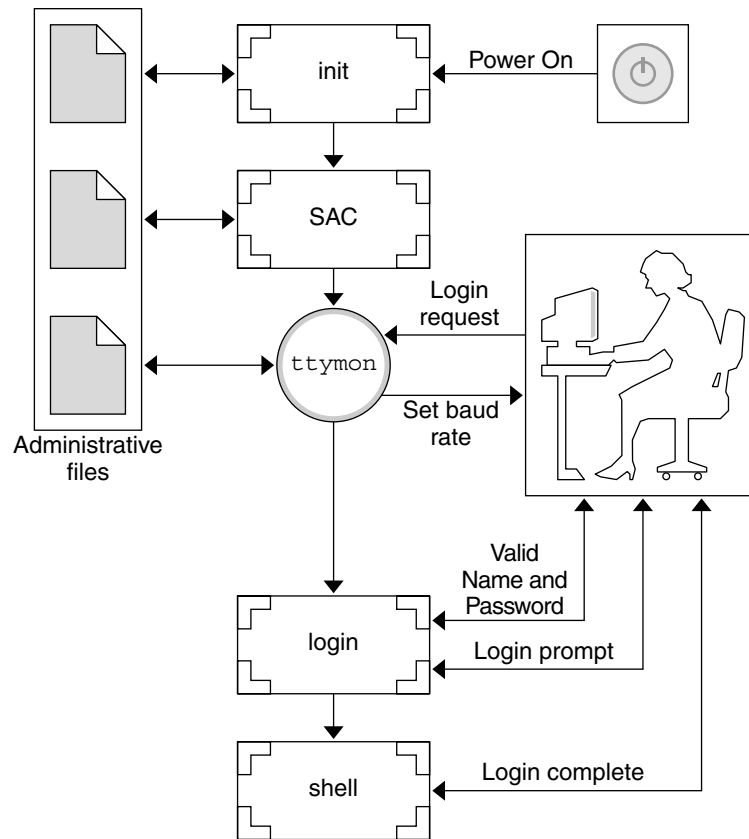
FIGURE 12–1 How `ttymon` Helps Process a Login Request

When someone attempts to log in via an alphanumeric terminal or a modem, the serial port driver passes the activity to the operating system. The `ttymon` port monitor notes the serial port activity, and attempts to establish a communications link. `ttymon` determines what data transfer rate, line discipline, and handshaking protocol are required to communicate with the device.

Having established the proper parameters for communication with the modem or terminal, `ttymon` passes these parameters to the login program and transfers control to it.

Port Initialization Process

When an instance of `ttymon` is invoked by SAC, `ttymon` starts to monitor its ports. For each port, `ttymon` first initializes the line disciplines, if they are specified, and the

speed and terminal settings. The values used for initialization are taken from the appropriate entry in /etc/ttydefs.

The ttymon port monitor then writes the prompt and waits for user input. If the user indicates that the speed is inappropriate by pressing the Break key, ttymon tries the next speed and writes the prompt again.

If *autobaud* is enabled for a port, ttymon will try to determine the baud rate on the port automatically. Users must press Return before ttymon can recognize the baud rate and print the prompt.

When valid input is received, ttymon interprets the per-service configuration file for the port, creates a /etc/utmpx entry if required, establishes the service environment, and invokes the service associated with the port.

After the service terminates, ttymon cleans up the /etc/utmpx entry, if one exists, and returns the port to its initial state.

Bidirectional Service

If a port is configured for bidirectional service, ttymon will:

- Allow users to connect to a service
- Allow uucico, cu, or ct to use the port for dialing out (if the port's free)
- Wait to read a character before printing a prompt
- Invoke the port's associated service—without sending the prompt message—when a connection is requested (if the connect-on-carrier flag is set)

Port Monitors: TTY Monitor and Network Listener

Though SAF provides a generic means for administering any future or third-party port monitors, only two are implemented in the Solaris environment—ttymon and listen.

TTY Port Monitor: `ttymon`

The `ttymon` port monitor is STREAMS-based. It monitors ports; sets terminal modes, baud rates, and line disciplines; and invokes the login process. (It provides Solaris users the same services that `getty` did under previous versions of SunOS 4.1 software.)

The `ttymon` port monitor runs under the SAC program. It is configured using the `sacadm` command. Each instance of `ttymon` can monitor multiple ports. These ports are specified in the port monitor's administrative file. The administrative file is configured using the `pmadm` and `ttyadm` commands.

`ttymon` and the Console Port

Console services are not managed by the Service Access Controller nor any explicit `ttymon` administration file. An entry in the `/etc/inittab` file is used to manage the console port using `ttymon` in *express* mode, which is a special `ttymon` mode that is invoked directly by a command that requires login service.

The default console entry in the `/etc/inittab` file looks like this:

```
co:234:respawn:/usr/lib/saf/ttymon -g -h -p "`uname -n` console login: "
 -T terminal_type -d /dev/console -l console -m ldterm,ttcompat
```

`co:234:respawn:`	`co` identifies the entry as the console; `234` identifies the run levels for the action, `respawn`, which means the console entry should be restarted if it fails or doesn't exist at run levels 2, 3, and 4.
`/usr/lib/saf/ttymon -g -h`	The `-g` option is used so the correct baud rate and terminal setting can be set on a port and connect to a login service without being preconfigured by the SAC. The `-h` option forces a line hangup by setting the line speed to zero before setting the default or specified speed.
`-p "`uname -n` console login:`	Identifies the prompt string for the console port.
`-t terminal_type`	Identifies the terminal type of the console.
`-d /dev/console -l console -m ldterm,ttcompat`	The `-d` option identifies the console device; the `-l` option identifies the `ttylabel` in the `/etc/ttydefs` file; and the `-m` option identifies the STREAMS modules to be pushed.

Special `ttymon`-Specific Administrative Command: `ttyadm`

The `ttymon` administrative file is updated by `sacadm` and `pmadm`, as well as by the `ttyadm` command. The `ttyadm` command formats `ttymon`-specific information and writes it to the standard output, providing a means for presenting formatted `ttymon`-specific data to the `sacadm` and `pmadm` commands.

Thus, `ttyadm` does not administer `ttymon` directly; rather, it complements the generic administrative commands, `sacadm` and `pmadm`. See `ttyadm`(1M) for more details.

Network Listener Service: `listen`

The `listen` port monitor runs under SAC. It monitors the network for service requests, accepts requests when they arrive, and invokes servers in response to those service requests.

The `listen` port monitor is configured using the `sacadm` command. Each instance of `listen` can provide multiple services. These services are specified in the port monitor's administrative file. This administrative file is configured using the `pmadm` and `nlsadmin` commands.

The network listener process can be used with any connection-oriented transport provider that conforms to the Transport Layer Interface (TLI) specification. In the Solaris environment, `listen` port monitors can provide additional network services not provided by `inetd`.

Special `listen`-Specific Administrative Command: `nlsadmin`

The `listen` port monitor's administrative file is updated by `sacadm` and `pmadm`, as well as by the `nlsadmin` command. The `nlsadmin` command formats `listen`-specific information and writes it to the standard output, providing a means of presenting formatted `listen`-specific data to the `sacadm` and `pmadm` commands.

Thus, `nlsadmin` does not administer `listen` directly; rather, it complements the generic administrative commands, `sacadm` and `pmadm`.

Each network can have at least one instance of the network listener process associated with it. Each network is configured separately. The `nlsadmin` command controls the operational states of `listen` port monitors.

The `nlsadmin` command can establish a `listen` port monitor for a given network, configure the specific attributes of that port monitor, and *start* and *kill* the monitor. The `nlsadmin` command can also report on the `listen` port monitors on a machine.

See `nlsadmin`(1M) for more details.

Administering `ttymon` Port Monitors

Use the `sacadm` command to add, list, remove, kill, start, enable, disable, enable, and remove a `ttymon` port monitor.

Note – You must be superuser to perform the following procedures.

▼ How to Add a `ttymon` Port Monitor

To add a `ttymon` port monitor, type:

```
# sacadm -a -p mbmon -t ttymon -c /usr/lib/saf/ttymon -v 'ttyadm
-V' -y "TTY Ports a & b"
```

-a	The *add* port monitor flag
-p	Specifies the *pmtag* mbmon as the port monitor tag
-t	Specifies the port monitor *type* as `ttymon`
-c	Defines the *command* string used to start the port monitor
-v	Specifies the *version* number of the port monitor
-y	Defines a comment to describe this instance of the port monitor

▼ How to View `ttymon` Port Monitor Status

To see the status of a `ttymon` port monitor, type:

```
# sacadm -l -p mbmon
```

-l	The *list* port monitor status flag

-p   Specifies the *pmtag* mbmon as the port monitor tag

Example—Viewing `ttymon` Port Monitor Status

```
# sacadm -l -p mbmon
PMTAG   PMTYPE  FLGS RCNT STATUS     COMMAND
mbmon   ttymon  -    0    STARTING   /usr/lib/saf/ttymon #TTY Ports a & b
```

PMTAG mbmon	Identifies the port monitor name, mbmon.
PMTYPE ttymon	Identifies the port monitor type, ttymon.
FLGS -	Indicates whether the following two flags are set: d, do not enable the new port monitor, or x, do not start the new port monitor. There are no flags set in this example.
RCNT 0	Indicates the return count value. A return count of 0 indicates that the port monitor is not to be restarted if it fails.
STATUS STARTING	Indicates the current status of the port monitor.
COMMAND /usr/lib/saf ...	Identifies the command used to start the port monitor.
#TTY Ports a & b	Identifies any comment used to describe the port monitor.

▼ How to Stop a `ttymon` Port Monitor

To kill a `ttymon` port monitor, type:

```
# sacadm -k -p mbmon
```

-k	The *kill* port monitor status flag
-p	Specifies the *pmtag* mbmon as the port monitor tag

▼ How to Start a `ttymon` Port Monitor

To start a killed `ttymon` port monitor, type:

```
# sacadm -s -p mbmon
```

-s   The *start* port monitor status flag

-p   Specifies the *pmtag* mbmon as the port monitor tag

▼ How to Disable a `ttymon` Port Monitor

Disabling a port monitor prevents new services from starting, without affecting existing services.

To disable a `ttymon` port monitor, type:

```
# sacadm -d -p mbmon
```

-d   The *disable* port monitor status flag

-p   Specifies the *pmtag* mbmon as the port monitor tag

▼ How to Enable a `ttymon` Port Monitor

Enabling a `ttymon` port monitor allows it to service new requests.

To enable a `ttymon` port monitor, type:

```
# sacadm -e -p mbmon
```

-e   The *enable* port monitor status flag

-p   Specifies the *pmtag* mbmon as the port monitor tag

▼ How to Remove a `ttymon` Port Monitor

To remove a `ttymon` port monitor, type:

```
# sacadm -r -p mbmon
```

`-r`	The *remove* port monitor status flag
`-p`	Specifies the *pmtag* `mbmon` as the port monitor tag

Note – Removing a port monitor deletes all the configuration files associated with it. Port monitor configuration files cannot be updated or changed using `sacadm`. To reconfigure a port monitor, *remove* it and *add* a new one.

Administering `ttymon` Services

Use `pmadm` to add services, list the services of one or more ports associated with a port monitor, and enable or disable a service.

Note – You must be superuser to perform the following procedures.

▼ How to Add a Service

To add a standard terminal service to the `mbmon` port monitor, type:

```
# pmadm -a -p mbmon -s a -i root -v `ttyadm -V` -m "`ttyadm -i 'Terminal disabled'
 -l contty -m ldterm,ttcompat -S y -d /dev/term/a -s /usr/bin/login'"
```

Note – In this example, the input wraps to the next line. Do not put a Return or line feed after `contty`.

`-a`	The *add* port monitor status flag
`-p`	Specifies the *pmtag* `mbmon` as the port monitor tag
`-s`	Specifies the *svctag* a as the port monitor *service* tag
`-i`	Specifies the *identity* to be assigned to *svctag* when it runs

-v	Specifies the *version* number of the port monitor
-m	Specifies the ttymon-specific configuration data formatted by ttyadm

The above pmadm command contains an embedded ttyadm command. The options in this embedded command are as follows:

-b	The *bidirectional* port flag
-i	Specifies the *inactive* (disabled) response message
-l	Specifies which TTY *label* in /etc/ttydefs to use
-m	Specifies the STREAMS *modules* to push before invoking this service
-d	Specifies the full path name to the *device* to use for the TTY port
-s	Specifies the full path name of the *service* to invoke when a connection request is received; if arguments are required, enclose the command and its arguments in quotation marks (")

▼ How to View the Status of a TTY Port Service

Use the pmadm command as shown to list the status of a TTY port, or all the ports associated with a port monitor.

Listing One Service

To list one service of a port monitor, type:

```
# pmadm -l -p mbmon -s a
```

-l	Lists service information
-p	Specifies the *pmtag* mbmon as the port monitor tag
-s	Specifies the *svctag* a as the port monitor *service* tag

Listing All Services of All Port Monitors

To list all services of all port monitors, type:

```
# pmadm -l
```

-l   Lists service information

Listing All Services of a Port Monitor

To list all services of a port monitor, type:

```
# pmadm -l -p mbmon
```

-l   Lists service information

-p   Specifies the *pmtag* mbmon as the port monitor tag

Example—Viewing the Status of a TTY Port Monitor Service

```
# pmadm -l -p mbmon
PMTAG  PMTYPE  SVCTAG FLAGS ID    <PMSPECIFIC>
mbmon ttymon  a      -     root  /dev/term/a - - /usr/bin/login - contty
ldterm,ttcompat login:  Terminal disabled - y  #
```

mbmon	Identifies the port monitor name, mbmon, set by using the pmadm -p command.
ttymon	Identifies the port monitor type, ttymon.
a	Indicates the service tag value set by using the pmadm -s command.
-	Identifies whether the following flags are set by using the pmadm -f command:
	x, which means do not enable the service;
	u, which means create a utmpx entry for the service. No flags are set in this example.
root	Identifies the ID assigned to the service when its started. This value is set by using the pmadm -i command.

    <PMSPECIFIC> *Information*

/dev/term/a	Indicates the TTY port pathname set by using the ttyadm -d command.
-	Indicates whether the following flags are set by using the ttadm -c -b -h -I -r command: c, sets the connect on carrier flag for the port b, sets the port as bidirectional, allowing both incoming and outgoing traffic h, suppresses an automatic hangup immediately after an incoming call is received I, initializes the port r, forces ttymon to wait until it receives a character from the port before it prints the login: message.
-	Indicates a value set by using the ttyadm -r option. This option determines when ttymon displays a prompt after receiving data from a port. If count is 0, ttymon will wait until it receives any character. If count is greater than 0, ttymon will wait until count new lines have been received. No value is set in this example.
/usr/bin/login	Identifies the full pathname of the service to be invoked when a connected is received. This value is set by using ttyadm -s command.
-	Identifies the ttyadm -t command's (timeout) value. This option specifies that ttymon should close a port if the open on the port succeeds, and no input data is received in timeout seconds. There is no timeout value in this example.
contty	Identifies the TTY label in the /etc/ttydefs file. This value is set by using the ttyadm -l command.
ldterm,ttcompat	Identifies the STREAMS modules to be pushed. These modules are set by using the ttyadmin -m command.
login: Terminal disabled	Identifies an inactive message to be displayed when the port is disabled. This message is set by using the ttyadm -i command.
tvi925	Identifies the terminal type, if set, by using the ttyadm -Tcommand. The terminal type is tvi925 in this example.
y	Identifies the software carrier value set by using the ttyadm -S command; n will turn software carrier off, y will turn software carrier on. Software carrier is turned on in this example.
#	Identifies any comment specified with the pmadm -y command. (There is no comment in this example).

▼ How to Enable a Port Monitor Service

To enable a disabled port monitor service, type:

```
# pmadm -e -p mbmon -s a
```

-e	The *enable* flag
-p	Specifies the *pmtag* mbmon as the port monitor tag
-s	Specifies the *svctag* a as the port monitor *service* tag

▼ How to Disable a Port Monitor Service

To disable a port monitor service, type:

```
# pmadm -d -p mbmon -s a
```

-d	The *disable* flag
-p	Specifies the *pmtag* mbmon as the port monitor tag
-s	Specifies the *svctag* a as the port monitor *service* tag

Service Access Facility Administration (Reference)

Files Associated With SAF

SAF uses configuration files which can be modified by using the sacadm and pmadm commands. You should not need to edit them manually.

File Name	Description
/etc/saf/_sysconfig	Per-system configuration script

File Name	Description
/etc/saf/_sactab	SAC's administrative file; contains configuration data for the port monitors that the SAC controls
/etc/saf/*pmtag*	Home directory for port monitor *pmtag*
/etc/saf/*pmtag*/_config	Per-port monitor configuration script for port monitor pmtag if it exists
/etc/saf/*pmtag*/_pmtab	Port monitor *pmtag*'s administrative file; contains port monitor-specific configuration data for the services *pmtag* provides
/etc/saf/*pmtag*/*svctag*	Per-service configuration script for service *svctag*
/var/saf/log	SAC's log file
/var/saf/*pmtag*	Directory for files created by *pmtag*, for example, log files

The /etc/saf/_sactab File

The /etc/saf/_sactab looks like this:

```
# VERSION=1
zsmon:ttymon::0:/usr/lib/saf/ttymon      #
```

# VERSION=1	Indicates the Service Access Facility version number.
zsmon	Is the name of the port monitor.
ttymon	Is the type of port monitor.
::	Indicates whether the following two flags are set:
	d, do not enable the port monitor
	x, do not start the port monitor. No flags are set in this example.
0	Indicates the return code value. A return count of 0 indicates that the port monitor is not be restarted if it fails.
/usr/lib/saf/ttymon	Indicates the port monitor pathname

The /etc/saf/*pmtab/_pmtab* File

The /etc/saf/*pmtab/_pmtab* file, such as /etc/saf/zsmon/_pmtab, looks like this:

```
# VERSION=1
ttya:u:root:reserved:reserved:reserved:/dev/term/a:I::/usr/bin/login::9600:ldterm,
ttcompat:ttya login\: ::tvi925:y:#
```

# VERSION=1	Indicates the Service Access Facility version number.
ttya	Indicates the service tag.
x,u	Identifies whether the following flags are set:
	x, which means do not enable the service.
	u, which means create a utmpx entry for the service.
root	Indicates the identity assigned to the service tag.
reserved	This field is reserved.
reserved	This field is reserved.
reserved	This field is reserved.
/dev/term/a	Indicates the TTY port pathname.
/usr/bin/login	Identifies the full pathname of the service to be invoked when a connection is received.
:c,b,h,I,r:	Indicates whether the following flags are set:
	c, sets the connect on carrier flag for the port
	b, sets the port as bidirectional, allowing both incoming and outgoing traffic
	h, suppresses an automatic hangup immediately after an incoming call is received
	I, initializes the port
	r, forces ttymon to wait until it receives a character from the port before it prints the login: message.
9600	Identifies the TTY label defined in /etc/ttydefs file.
ldterm,ttcompat	Identifies the STREAMS modules to be pushed.
ttya login\:	Identifies the prompt to be displayed.
:y/n:	
message	Identifies any inactive (disabled) response message
tvi925	Identifies the terminal type.

y	Indicates whether software carrier is set (y/n).

Service States

The sacadm command controls the states of services. The possible states are shown below.

State	Notes
Enabled	*Default state* – When the port monitor is added, the service operates.
Disabled	*Default state* – When the port monitor is removed, the service stops.

To determine the state of any particular service, use the following:

pmadm -l -p *portmon_name* **-s** *svctag*

Port Monitor States

The sacadm command controls the states of ttymon and listen port monitors. The possible states are shown below.

State	Notes
Started	*Default state* – When the port monitor is added, it is automatically started.
Enabled	*Default state* – When the port monitor is added, it is automatically ready to accept requests for service.
Stopped	*Default state* – When the port monitor is removed, it is automatically stopped.
Disabled	*Default state* – When the port monitor is removed, it automatically continues existing services and refuses to add new services.
Starting	*Intermediate state* – The port monitor is in the process of starting.
Stopping	*Intermediate state* – The port monitor has been manually terminated, but it has not completed its shutdown procedure. It is on the way to becoming stopped.
Notrunning	*Inactive state* – The port monitor has been killed. All ports previously monitored are inaccessible. An external user cannot tell whether a port is disabled or notrunning.

State	Notes
Failed	*Inactive state* – The port monitor is unable to start and remain running.

To determine the state of any particular port monitor, use the following:

```
# sacadm -l -p portmon_name
```

Port States

Ports can be enabled or disabled depending on the state of the port monitor that controls them.

State	Notes
Serial (`ttymon`) Port States	
Enabled	The `ttymon` port monitor sends a prompt message to the port and provides login service to it.
Disabled	Default state of all ports if `ttymon` is killed or disabled. If you specify this state, `ttymon` will send out the `disabled` message when it receives a connection request.

CHAPTER **13**

Managing System Resources Topics

This topic map identifies the chapters related to managing system resources.

Chapter 14	Provides an overview of Solaris commands and utilities that help you manage system resources by using disk quotas, the `cron` and `at` commands, and accounting programs.
Chapter 15	Provides step-by-step instructions for displaying and changing system information.
Chapter 16	Provides step-by-step instructions for optimizing disk space by locating unused files and large directories.
Chapter 17	Provides step-by-step instructions for setting up and administering disk quotas.
Chapter 18	Provides step-by-step instructions for scheduling routine or one-time system events by using the `crontab` and `at` commands..
Chapter 19	Provides step-by-step instructions for setting up and maintaining system accounting.
Chapter 20	Provides reference information for system accounting software, including how the different accounting features operate and how to use the various accounting programs.

Managing System Resources (Overview)

This chapter provides a brief description of new features and a map of features to help you manage system resources.

Using these features, you can display general system information, monitor disk space, set disk quotas, use accounting programs, and schedule the `crontab` and `at` commands to automatically run routine commands.

For information on the procedures associated with managing system resources, see "Managing System Resources (Task Map)" on page 223.

What's New in Managing System Resources?

New `df` and `du` Options

The `df` and `du` commands have two new options, `-h` and `-H`, that display disk space in powers of 1024 and powers of 1000, respectively. These options make it easier to interpret the output of the `df` and `du` command by providing disk space in Mbytes and Gbytes, when possible.

See "How to Display Information About Files and Disk Space" on page 234 for more information.

Changing System Parameters

In previous versions of this book, there was a section on changing system parameters such as increasing the number of processes per user and shared memory segments. This section has been removed. See *Solaris Tunable Parameters Reference Manual* for information on changing system tunable parameters.

In addition, information on setting up a network time server has been moved to "Time Related Services" in *System Administration Guide: Resource Management and Network Services*.

Extended Accounting Features

Extended accounting introduces a new, variable length general-purpose accounting file format that represents general groups of accounting data. It provides the ability to configure resource utilization recorded by the kernel in the various accounting files.

Extended accounting features include:

- *Tasks* – New process collectives for tracking resource usage.
- *Projects* – New administrative databases for charging resource usage. Resource usage can be charged to a project based on tasks performed.
- acctadm – A new tool for configuring various attributes of the extended accounting feature. For example, the resources tracked by the system accounting can be configured on a system-wide basis.

The new default accounting configuration requires no administration. However, if you use the extended accounting features, do not remove the /etc/project file. This file contains important information about the extended accounting configuration.

Use the following table to find more information about the extended accounting features in this release.

Extended Accounting Features	For More Information
Turning on and off extended accounting	"Extended Accounting" in *System Administration Guide: Resource Management and Network Services* and acctadm(1M)
Description of the projects database	"Projects and Tasks" in *System Administration Guide: Resource Management and Network Services* and project(4)
Directly collecting extended accounting data	libexacct(3LIB), getacct(2), putacct(2), and wracct(2)

Managing System Resources (Task Map)

TABLE 14–1 High-Level View of Managing System Resources (Task Map)

Task	Description	Instructions
Displaying and changing system information	Use various commands to display and changing system information such as general system information, the date and time, and changing the system's host name.	Chapter 15
Managing disk use	Identify how disk space is used and take steps to remove old and unused files.	Chapter 16
Managing quotas	Use UFS file system quotas to manage how much disk space is used by users.	Chapter 17
Scheduling system events	Use cron and at jobs to help schedule system routines that can include clean up of old and unused files.	Chapter 18
Managing system accounting	Use system accounting to identify how users and applications are using system resources.	Chapter 19

Displaying and Changing System Information (Tasks)

This chapter describes tasks required to display and change the most common system information. This is a list of the step-by-step instructions in this chapter.

- "How to Determine If a System Can Run the 64–bit Solaris Operating Environment" on page 226
- "How to Display General System Information" on page 228
- "How to Display a System's Host ID Number " on page 229
- "How to Display the Date and Time" on page 229
- "How to Set a System's Date and Time Manually" on page 230
- "How to Set Up a Message-of-the-Day" on page 231
- "How to Change a System's Host Name" on page 231

Displaying System Information

The following table describes commands that enable you to display general system information.

TABLE 15–1 Commands for Displaying System Information

Command	System Information Displayed	Man Page
psrinfo	Processor type	psrinfo(1M)
isainfo	Supported applications and it reports the number of bits supported by *native* applications on the running system, which can be passed as a token to scripts	isainfo(1)

TABLE 15–1 Commands for Displaying System Information   *(Continued)*

Command	System Information Displayed	Man Page
showrev	Hostname, host ID, release, kernel architecture, application architecture, hardware provider, domain, and kernel version	showrev(1M)
uname	Operating system name, release, and version; node name; hardware name; processor type	uname(1)
hostid	Host ID number	hostid(1)
prtconf	Installed memory	prtconf(1M)
date	Date and time	date(1)

▼ How to Determine If a System Can Run the 64–bit Solaris Operating Environment

Currently, the only platform capable of supporting the 64–bit Solaris operating environment is an UltraSPARC system. To verify if a system is an UltraSPARC system, use the following command:

```
$ uname -m
sun4u
```

If the output of the uname -m command is sun4u, then the machine is an UltraSPARC system.

If you are running the Solaris 9 release, you can verify that the machine is an UltraSPARC system by using the psrinfo command:

```
# psrinfo -v
Status of processor 0 as of: 05/30/01 13:48:46
  Processor has been on-line since 05/18/01 10:02:40.
  The sparcv9 processor operates at 333 MHz,
        and has a sparcv9 floating point processor.
```

If the processor type is sparcv9, the platform is capable of running the 64-bit Solaris operating environment. This test does not work on previous versions of the psrinfo command, where all platforms report the less precise sparc as the processor type.

▼ How to Determine If a System Has 64-bit Solaris Capabilities Enabled

Use the `isainfo` command to determine if a system has 64-bit capabilities enabled, which means that the system is booted with the 64–bit kernel.

Examples—Determining If a System Has 64–bit Solaris Capabilities Enabled

The output for an UltraSPARC system running a 32-bit kernel appears as follows:

```
$ isainfo -v
32-bit sparc applications
```

This output means that this system is capable of supporting only 32-bit applications.

The output for an UltraSPARC system running a 64-bit kernel appears as follows:

```
 $ isainfo -v
64-bit sparcv9 applications
32-bit sparc applications
```

This output means that this system is capable of supporting both 32-bit and 64-bit applications.

Use the `isainfo -b` command to display the number of bits supported by native applications on the running system.

The output from a SPARC, IA, or UltraSPARC system running the 32–bit Solaris operating environment appears as follows:

```
$ isainfo -b
32
```

The output from a 64–bit UltraSPARC system running the 64–bit Solaris operating environment appears as follows:

```
$ isainfo -b
64
```

The command returns `64` only. Even though a 64–bit UltraSPARC system is capable of running both types of applications, 64–bit applications are the best kind of applications to run on a 64–bit system.

▼ How to Display System and Software Release Information

To display system information and software release information, use the showrev command.

```
$ showrev [-a]
```

The -a option displays all system release information available.

Example—Displaying System and Software Release Information

The following example shows showrev command output.

```
$ showrev -a
Hostname: starbug
Hostid: nnnnnnnn
Release: 5.9
Kernel architecture: sun4u
Application architecture: sparc
Hardware provider: Sun_Microsystems
Domain: solar.com
Kernel version: SunOS 5.9 s81_37 February 2002

OpenWindows version:
X11 Version 6.4.2 11 April 2001

No patches are installed
$
```

▼ How to Display General System Information

To display system information, use the uname command.

```
$ uname[-a]
```

The -aoption displays the operating system name as well as the system node name, operating system release, operating system version, hardware name, and processor type.

Example—Displaying General System Information

The following example shows uname command output.

```
$ uname
SunOS
$ uname -a
SunOS starbug 5.9 Generic sun4u sparc SUNW,Ultra-5_10
$
```

▼ How to Display a System's Host ID Number

To display the host ID number in hexadecimal format, use the `hostid` command.

```
$ hostid
```

Example—Displaying a System's Host ID Number

The following example shows sample output from the `hostid` command.

```
$ hostid
80a5d34c
```

▼ How to Display a System's Installed Memory

To display the amount of memory installed on your system, use the `prtconf` command.

```
$ prtconf [| grep Memory]
```

The `grep Memory` command selects output from the `prtconf` command to display memory information only.

Example—Displaying a System's Installed Memory

The following example shows sample output from the `prtconf` command.

```
# prtconf | grep Memory
Memory size: 128 Megabytes
```

▼ How to Display the Date and Time

To display the current date and time according to your system clock, use the `date` command.

```
$ date
```

Example—Displaying the Date and Time

The following example shows sample output from the `date` command.

```
$ date
Thu May 31 17:44:58 MDT 2001
$
```

Changing System Information

This section describes commands that enable you to change general system information.

▼ How to Set a System's Date and Time Manually

1. **Become superuser.**

2. **Enter the new date and time.**

   # **date** *mmddHHMM[[cc]yy]*

mm	Month, using two digits.
dd	Day of the month, using two digits.
HH	Hour, using two digits and a 24-hour clock.
MM	Minutes, using two digits.
cc	Century, using two digits.
yy	Year, using two digits.

   See `date`(1) for more information.

3. **Verify that you have reset your system's date correctly by using the `date` command with no options.**

Example—Setting a System's Date and Time Manually

The following example shows how to use the `date` command to manually set a system's date and time.

```
# date
Thu Jun 21 13:59:15 MDT 2001
# date 0621141001
Thu Jun 21 14:10:00 MDT 2001
```

▼ How to Set Up a Message-of-the-Day

The message-of-the-day (MOTD) facility, located in /etc/motd, enables you to send announcements or inquiries to all users of a system when they log in. Use this facility sparingly, and edit this file regularly to remove obsolete messages.

1. **Become superuser.**

2. **Edit the** /etc/motd **file and add a message of your choice.**

   Edit the text to include the message that will be displayed during user login. Include spaces, Tabs, and Returns.

3. **Verify the changes by displaying the contents of the** /etc/motd **file.**

   ```
   $ cat /etc/motd
   Welcome to the UNIX Universe. Have a nice day.
   ```

Example—Setting Up a Message-of-the-Day

The default message-of-the-day, provided when you install Solaris software, contains SunOS version information:

```
$ cat /etc/motd
Sun Microsystems Inc.   SunOS 5.9      Generic  May 2002
```

The following example shows an edited /etc/motd file that provides information about system availability to each user who logs in.

```
$ cat /etc/motd
The system will be down from 7:00 a.m to 2:00 p.m. on
Saturday, July 7, for upgrades and maintenance.
Do not try to access the system during those hours.
Thank you.
```

▼ How to Change a System's Host Name

A system's host name is specified in several different locations.

Remember that you will need to update your name service database to reflect the new host name.

1. **Become superuser.**

2. **Change the system's host name in the following files:**

   - `/etc/nodename`
   - `/etc/hostname.`*xxy*
   - `/etc/inet/hosts`
   - `/etc/net/ticlts/hosts`
   - `/etc/net/ticots/hosts`
   - `/etc/net/ticotsord/hosts`

3. **(Optional) If using a name service, change the system's host name in the** `host`**file.**

4. **Reboot the system to activate the new host name.**

   ```
   # init 6
   ```

Managing Disk Use (Tasks)

This chapter describes how to optimize disk space by locating unused files and large directories. This is a list of the step-by-step instructions in this chapter.

- "How to Display Information About Files and Disk Space" on page 234
- "How to Display the Size of Files" on page 236
- "How to Find Large Files" on page 237
- "How to Find Files That Exceed a Specified Size Limit" on page 238
- "How to Display the Size of Directories, Subdirectories, and Files" on page 239
- "How to Display the User Ownership of Local UFS File Systems" on page 240
- "How to List the Newest Files" on page 242
- "How to Find and Remove Old or Inactive Files " on page 242
- "How to Clear Out Temporary Directories" on page 244
- "How to Find and Delete `core` Files" on page 244
- "How to Delete Crash Dump Files" on page 245

Displaying Information About Files and Disk Space

This table summarizes the commands available for displaying information about file size and disk space..

Command and Man Page	Description
df df(1M)	Reports the number of free disk blocks and files.

Command and Man Page	Description
du du(1)	Summarizes disk space allocated to each subdirectory.
find -size find(1)	Searches recursively through a directory based on the size specified with the -size option.
ls -l ls(1)	Lists the size of a file in bytes.

▼ How to Display Information About Files and Disk Space

Display information about how disk space is used by using the df command.

$ **df** [*directory*] [**-F** *fstype*] [**-g**] [**-k**] [**-t**]

df	With no options, lists all mounted file systems and their device names, the number of 512-byte blocks used, and the number of files.
directory	Specifies the directory whose file system you want to check.
-F *fstype*	Displays a list of unmounted file systems, their device names, the number of 512-byte blocks used, and the number of files on file systems of type *fstype*.
-h	Displays disk space in the power of 1024.
-H	Displays disk space in the power of 1000.
-t	Displays the total blocks as well as the blocks used for all mounted file systems.

Examples—Displaying Information About File Size and Disk Space

In the following example, all the file systems listed are locally mounted except for /usr/dist, which is mounted remotely from the system venus.

```
$ df
/                  (/dev/dsk/c0t0d0s0 ):  287530 blocks    92028 files
/usr               (/dev/dsk/c0t0d0s6 ): 1020214 blocks   268550 files
/proc              (/proc            ):       0 blocks      878 files
/dev/fd            (fd               ):       0 blocks        0 files
```

```
/etc/mnttab           (mnttab          ):        0 blocks        0 files
/var/run              (swap            ):   396016 blocks     9375 files
/tmp                  (swap            ):   396016 blocks     9375 files
/opt                  (/dev/dsk/c0t0d0s5 ):   381552 blocks    96649 files
/export/home          (/dev/dsk/c0t0d0s7 ):   434364 blocks   108220 files
/usr/dist             (venus:/usr/dist  ):14750510 blocks  2130134 files
```

In the following example, file system information is displayed in 1024 bytes.

```
$ df -h
Filesystem            size   used  avail capacity  Mounted on
/dev/dsk/c0t0d0s0     1.9G    58M   1.8G     4%     /
/dev/dsk/c0t0d0s6     2.5G   765M   1.7G     1%     /usr
/proc                  0K     0K     0K     0%     /proc
fd                     0K     0K     0K     0%     /dev/fd
mnttab                 0K     0K     0K     0%     /etc/mnttab
swap                  2.1G    24K   2.1G     1%     /var/run
swap                  2.1G     0K   2.1G     0%     /tmp
/dev/dsk/c0t0d0s5     1.9G    12M   1.8G     1%     /opt
/dev/dsk/c0t0d0s7     1.9G    10K   1.8G     1%     /export/home
```

The following example shows information about the same system as the previous example, but only UFS file system information is displayed in 1000 bytes.

```
$ df -H -F ufs
Filesystem            size   used  avail capacity  Mounted on
/dev/dsk/c0t0d0s0     2.0G    60M   1.9G     4%     /
/dev/dsk/c0t0d0s6     2.6G   783M   1.8G    31%     /usr
/dev/dsk/c0t0d0s4     2.0G    17M   1.9G     1%     /var
/dev/dsk/c0t0d0s3     2.0G    10K   1.9G     1%     /export
/dev/dsk/c0t0d0s5     2.0G    13M   1.9G     1%     /opt
/dev/dsk/c0t0d0s7     2.0G    10K   1.9G     1%     /export/home
```

Note – Although /proc and /tmp are local file systems, they are not UFS file systems. /proc is a PROCFS file system, /var/run and /tmp are TMPFS file systems, and /etc/mnttab is a MNTFS file system.

The following example shows a list of all mounted file systems, device names, total 512-byte blocks used, and number of files. The second line of each two-line entry displays the total number of blocks and files allocated for the file system.

```
$ df -t
/                  (/dev/dsk/c0t0d0s0 ):   287530 blocks     92028 files
                              total:   385614 blocks     96832 files
/usr               (/dev/dsk/c0t0d0s6 ):  1020214 blocks    268550 files
                              total:  2381102 blocks    300288 files
/proc              (/proc           ):        0 blocks       879 files
                              total:        0 blocks       924 files
/dev/fd            (fd              ):        0 blocks         0 files
                              total:        0 blocks        72 files
/etc/mnttab        (mnttab          ):        0 blocks         0 files
                              total:        0 blocks         1 files
```

```
/var/run          (swap              ):   396112 blocks     9375 files
                                    total:   396112 blocks     9395 files
/tmp              (swap              ):   396112 blocks     9375 files
                                    total:   396128 blocks     9395 files
/opt              (/dev/dsk/c0t0d0s5 ):   381552 blocks    96649 files
                                    total:   385614 blocks    96832 files
/export/home      (/dev/dsk/c0t0d0s7 ):   434364 blocks   108220 files
                                    total:   434382 blocks   108224 files
/usr/dist         (venus:/usr/dist   ): 14750510 blocks  2130134 files
                                    total: 41225162 blocks  2482176 files
```

Checking the Size of Files

You can check the size of files and sort them by using the `ls` command. You can find files that exceed a size limit by using the `find` command. For more information, see `ls`(1) and `find`(1).

▼ How to Display the Size of Files

1. **Change to the directory where the files you want to check are located.**

2. **Display the size of the files.**

   ```
   $ ls [-l] [-s]
   ```

-l	Displays a list of files and directories in long format, showing the sizes in bytes. (See the example below.)
-s	Displays a list of the files and directories, showing the sizes in blocks.

Examples—Displaying the Size of Files

The following example shows that the `lastlog` and `messages` files are larger than the other files in the `/var/adm` directory.

```
$ cd /var/adm
$ ls -l
total 144
drwxrwxr-x   5 adm      adm          512 Sep  1 14:11 acct/
-rw-------   1 uucp     bin            0 Sep  1 14:08 aculog
-r--r--r--   1 root     root      350700 Sep  3 10:37 lastlog
```

```
drwxr-xr-x   2 adm      adm           512 Sep  1 14:08 log/
-rw-r--r--   1 root     root        14619 Sep  2 16:11 messages
-rw-r--r--   1 adm      adm          8200 Sep  3 14:35 pacct
-rw-r--r--   1 adm      adm           920 Sep  3 10:47 pacct1
drwxr-xr-x   2 adm      adm           512 Sep  1 14:08 passwd/
drwxrwxr-x   2 adm      sys           512 Sep  1 14:11 sa/
drwxr-xr-x   2 root     sys           512 Sep  1 14:36 sm.bin/
-rw-rw-rw-   1 root     bin             0 Sep  1 14:08 spellhist
-rw-------   1 root     root          420 Sep  3 14:17 sulog
-rw-r--r--   1 root     bin          4092 Sep  3 10:37 utmpx
-rw-r--r--   1 root     root          122 Sep  1 15:39 vold.log
-rw-r--r--   1 adm      adm         11904 Sep  3 10:47 wtmpx
```

The following example shows that the `lpsched.1` file uses two blocks.

```
$ cd /var/lp/logs
$ ls -s
total 2           0 lpsched        2 lpsched.1
```

▼ How to Find Large Files

1. **Change to the directory that you want to search.**

2. **Display the size of files in blocks from largest to smallest.**

   ```
   $ ls -s | sort -nr | more
   ```

`sort -nr`	Sorts the list of files by block size from largest to smallest.

Example—Finding Large Files

In the following example, the `lastlog` and `messages` files are the largest files in the `/var/adm` directory.

```
$ cd /var/adm
$ ls -s | sort -nr | more
 48 lastlog
 30 messages
 24 wtmpx
 18 pacct
  8 utmpx
  2 vold.log
  2 sulog
  2 sm.bin/
  2 sa/
  2 passwd/
  2 pacct1
  2 log/
```

```
    2 acct/
    0 spellhist
    0 aculog
total 144
```

▼ How to Find Files That Exceed a Specified Size Limit

To locate and display the names of files that exceed a specified size, use the find command.

```
$ find directory -size +nnn
```

directory	Identifies the directory you want to search.
-size +*nnn*	Is a number of 512-byte blocks. Files that exceed this size are listed.

Example—Finding Files That Exceed a Specified Size Limit

The following example shows how to find files larger than 400 blocks in the current working directory.

```
$ find . -size +400 -print
./Howto/howto.doc
./Howto/howto.doc.backup
./Howto/howtotest.doc
./Routine/routineBackupconcepts.doc
./Routine/routineIntro.doc
./Routine/routineTroublefsck.doc
./.record
./Mail/pagination
./Config/configPrintadmin.doc
./Config/configPrintsetup.doc
./Config/configMailappx.doc
./Config/configMailconcepts.doc
./snapshot.rs
```

Checking the Size of Directories

You can display the size of directories by using the du command and options. Additionally, you can find the amount of disk space used by user accounts on local UFS file systems by using the quot command. For more information about these commands, see du(1) and quot(1M).

▼ How to Display the Size of Directories, Subdirectories, and Files

Display the size of one or more directories, subdirectories, and files by using the du command. Sizes are displayed in 512-byte blocks.

$ **du** [**-as**] [*directory ...*]

du	Displays the size of each directory you specify, including each subdirectory beneath it.
-a	Displays the size of each file and subdirectory, and the total number of blocks contained in the specified directory.
-s	Displays the total number of blocks contained in the specified directory.
-h	Displays the size of each directory in 1024 bytes.
-H	Displays the size of each directory in 1000 bytes.
directory ...	Identifies one or more directories you want to check.

Examples—Displaying the Size of Directories, Subdirectories, and Files

The following example shows the total sizes of two directories.

```
$ du -s /var/adm /var/spool/lp
130     /var/adm
40      /var/spool/lp
```

The following example shows the sizes of two directories, all of the subdirectories and files they contain, and the total number of blocks contained in each directory.

```
$ du /var/adm /var/spool/lp
2       /var/adm/log
```

```
2       /var/adm/passwd
2       /var/adm/acct/fiscal
2       /var/adm/acct/nite
2       /var/adm/acct/sum
8       /var/adm/acct
2       /var/adm/sa
2       /var/adm/sm.bin
130     /var/adm
4       /var/spool/lp/admins
2       /var/spool/lp/fifos/private
2       /var/spool/lp/fifos/public
6       /var/spool/lp/fifos
2       /var/spool/lp/requests/starbug
4       /var/spool/lp/requests
2       /var/spool/lp/system
2       /var/spool/lp/tmp/starbug
2       /var/spool/lp/tmp/.net/tmp/starbug
4       /var/spool/lp/tmp/.net/tmp
2       /var/spool/lp/tmp/.net/requests/starbug
4       /var/spool/lp/tmp/.net/requests
10      /var/spool/lp/tmp/.net
14      /var/spool/lp/tmp
40      /var/spool/lp
```

The following example shows directory sizes in 1024 bytes.

```
du -h /usr/share/audio
 796K   /usr/share/audio/samples/au
 797K   /usr/share/audio/samples
 798K   /usr/share/audio
```

▼ How to Display the User Ownership of Local UFS File Systems

1. **Become superuser.**

2. **Display users, directories, or file systems, and the number of 1024-byte blocks used.**

   # **quot** [**-a**] [*filesystem*]

-a	Lists all users of each mounted UFS file system and the number of 1024-byte blocks used.
filesystem	Identifies a UFS file system. Users and the number of blocks used are displayed.

Note – The `quot` command works only on local UFS file systems.

Example—Displaying the User Ownership of Local UFS File Systems

In the following example, users of the root (/) file system are displayed. Then, users of all mounted UFS file systems are displayed.

```
# quot /
/dev/rdsk/c0t0d0s0:
43340    root
 3142    rimmer
   47    uucp
   35    lp
   30    adm
    4    bin
    4    daemon
# quot -a
/dev/rdsk/c0t0d0s0 (/):
43340    root
 3150    rimmer
   47    uucp
   35    lp
   30    adm
    4    bin
    4    daemon
/dev/rdsk/c0t0d0s6 (/usr):
460651  root
206632  bin
  791    uucp
   46    lp
    4    daemon
    1    adm
/dev/rdsk/c0t0d0s7 (/export/home):
    9    root
```

Finding and Removing Old or Inactive Files

Part of the job of cleaning up heavily loaded file systems involves locating and removing files that have not been used recently. You can locate unused files using the `ls` or `find` commands. For more information, see `ls`(1) and `find`(1).

Other ways to conserve disk space include emptying temporary directories such as the ones located in /var/tmp or /var/spool, and deleting core and crash dump files. For more information about these files, refer to Chapter 28.

▼ How to List the Newest Files

List files, displaying the most recently created or changed files first, by using the ls -t command.

```
$ ls -t [directory]
```

-t	Sorts files by latest time stamp first.
directory	Identifies the directory you want to search.

Example—Listing the Newest Files

The following example shows how to use the ls -tl command to locate the most recently created or changed files within the /var/adm directory. The sulog file was created or edited most recently.

```
$ ls -tl /var/adm
total 134
-rw-------   1 root     root          315 Sep 24 14:00 sulog
-r--r--r--   1 root     other      350700 Sep 22 11:04 lastlog
-rw-r--r--   1 root     bin          4464 Sep 22 11:04 utmpx
-rw-r--r--   1 adm      adm         20088 Sep 22 11:04 wtmpx
-rw-r--r--   1 root     other           0 Sep 19 03:10 messages
-rw-r--r--   1 root     other           0 Sep 12 03:10 messages.0
-rw-r--r--   1 root     root        11510 Sep 10 16:13 messages.1
-rw-r--r--   1 root     root            0 Sep 10 16:12 vold.log
drwxr-xr-x   2 root     sys           512 Sep 10 15:33 sm.bin
drwxrwxr-x   5 adm      adm           512 Sep 10 15:19 acct
drwxrwxr-x   2 adm      sys           512 Sep 10 15:19 sa
-rw-------   1 uucp     bin             0 Sep 10 15:17 aculog
-rw-rw-rw-   1 root     bin             0 Sep 10 15:17 spellhist
drwxr-xr-x   2 adm      adm           512 Sep 10 15:17 log
drwxr-xr-x   2 adm      adm           512 Sep 10 15:17 passwd
```

▼ How to Find and Remove Old or Inactive Files

1. **Become superuser.**

2. **Find files that have not been accessed for a specified number of days and list them in a file.**

   # **find** *directory* **-type f**[**-atime +** *nnn*] [**-mtime +** *nnn*] **-print >** *filename*

directory	Identifies the directory you want to search. Directories below this directory are also searched.
-atime +*nnn*	Finds files that have not been accessed within the number of days (nnn) you specify.
-mtime +*nnn*	Finds files that have not been modified within the number of days (nnn) you specify.
filename	Identifies the file that contains the list of inactive files.

3. **Remove the inactive files that you listed in the previous step.**

   # **rm `cat** *filename*`

   Where *filename* identifies the file created in the previous step which contains the list of inactive files.

Example—Finding and Removing Old or Inactive Files

The following example shows files in /var/adm and the subdirectories that have not been accessed in the last 60 days. The /var/tmp/deadfiles file contains the list of inactive files. The rm command removes these inactive files.

```
# find /var/adm -type f -atime +60 -print > /var/tmp/deadfiles &
# more /var/tmp/deadfiles
/var/adm/aculog
/var/adm/spellhist
/var/adm/wtmpx
/var/adm/sa/sa13
/var/adm/sa/sa27
/var/adm/sa/sa11
/var/adm/sa/sa23
/var/adm/sulog
/var/adm/vold.log
/var/adm/messages.1
/var/adm/messages.2
/var/adm/messages.3
# rm `cat /var/tmp/deadfiles`
#
```

▼ How to Clear Out Temporary Directories

1. Become superuser.

2. Change to the /var/tmp **directory.**

```
# cd /var/tmp
```

⚠️ **Caution –** Be sure you are in the right directory before completing step 3. Step 3 deletes all files in the current directory.

3. Delete the files and subdirectories in the current directory.

```
# rm -r *
```

4. Change to other directories containing unnecessary temporary or obsolete subdirectories and files, and delete them by repeating Step 3.

Example—Clearing Out Temporary Directories

The following example shows how to clear out the /var/tmp directory, and how to verify that all files and subdirectories were removed.

```
# cd /var/tmp
# ls
deadfiles          wxconAAAa0003r:0.0  wxconAAAa000NA:0.0
test_dir           wxconAAAa0003u:0.0  wxconAAAa000cc:0.0
wxconAAAa000zs:0.0
# rm -r *
# ls
#
```

▼ How to Find and Delete core Files

1. Become superuser.

2. Change to the directory where you want to search for core files.

3. Find and remove any core files in this directory and its subdirectories.

```
# find . -name core -exec rm {} \;
```

Example—Finding and Deleting `core` Files

The following example shows how to find and remove core files from the jones user account by using the find command.

```
# cd /home/jones
# find . -name core -exec rm {} \;
```

▼ How to Delete Crash Dump Files

Crash dump files can be very large, so if you have enabled your system to store these files, do not retain them for longer than necessary.

1. **Become superuser.**

2. **Change to the directory where crash dump files are stored.**

   ```
   # cd /var/crash/system
   ```

   Where *system* identifies a system that created the crash dump files.

   Caution – Be sure you are in the right directory before completing step 3. Step 3 deletes all files in the current directory.

   ---

3. **Remove the crash dump files.**

   ```
   # rm *
   ```

4. **Verify that the crash dump files are removed.**

   ```
   # ls
   ```

Example—Deleting Crash Dump Files

The following example shows how to remove crash dump files from the system venus, and how to verify that the crash dump files were removed.

```
# cd /var/crash/venus
# rm *
# ls
```

Managing Quotas (Tasks)

This chapter describes how to set up and administer quotas for disk space and inodes. This is a list of the step-by-step instructions in this chapter.

What Are Quotas?

Quotas enable system administrators to control the size of UFS file systems by limiting the amount of disk space and the number of inodes (which roughly corresponds to the number of files) that individual users can acquire. For this reason, quotas are especially useful on the file systems where user home directories reside. As a rule, public and /tmp file systems usually do not benefit as much from the establishment of quotas.

Setting up quotas involves these general steps:

1. Enable file system quotas by issuing a series of commands, ensuring that quotas are enforced each time the system is rebooted and the file system is mounted. Entries must be added to the /etc/vfstab file, and a quotas file must be created

in the root of the file system.

2. After a quota is created for one user, the quota can be copied as a prototype to set up other user quotas.

3. Before quotas are turned on, another command checks for consistency by comparing the proposed quotas to the current disk usage to make sure there are no conflicts.

4. Finally, a command turns on the quotas for one or more file systems.

These steps ensure that quotas are automatically activated on a file system each time it is mounted. See Chapter 17 for step-by-step instructions.

Once quotas are in place, they can be changed to adjust the amount of disk space or number of inodes that users can consume. Additionally, quotas can be added or removed as system needs change. See "Changing and Removing Quotas" on page 258 for instructions on how to change quotas, disable individual quotas, or remove quotas from file systems.

In addition, quota status can be monitored. Quota commands enable administrators to display information about quotas on a file system, or search for users who have exceeded their quotas. For procedures that describe how to use these commands, see "Checking Quotas" on page 255.

Using Quotas

Using quotas enable system administrators to control the size of UFS file systems by limiting the amount of disk space and the number of inodes (which roughly corresponds to the number of files) that individual users can acquire. For this reason, quotas are especially useful on the file systems where user home directories reside.

Once they are in place, quotas can be changed to adjust the amount of disk space or number of inodes that users can consume. Additionally, quotas can be added or removed as system needs change. See "Changing and Removing Quotas" on page 258 for instructions on changing quotas or the amount of time that quotas can be exceeded, disabling individual quotas, or removing quotas from file systems.

In addition, quota status can be monitored. Quota commands enable administrators to display information about quotas on a file system, or search for users who have exceeded their quotas. For procedures that describe how to use these commands, see "Checking Quotas" on page 255.

Setting Soft Limits and Hard Limits for Quotas

You can set both soft and hard limits. The system will not allow a user to exceed his or her hard limit. However, a system administrator may set a soft limit (sometimes referred to as a quota), which the user can temporarily exceed. The soft limit must be less than the hard limit.

Once the user exceeds the soft limit, a timer begins. While the timer is ticking, the user is allowed to operate above the soft limit but cannot exceed the hard limit. Once the user goes below the soft limit, the timer is reset. However, if the user's usage remains above the soft limit when the timer expires, the soft limit is enforced as a hard limit. By default, the soft limit timer is set to seven days.

The `timeleft` field in the `repquota` and `quota` commands shows the value of the timer.

For example, let's say a user has a soft limit of 10,000 blocks and a hard limit of 12,000 blocks. If the user's block usage exceeds 10,000 blocks and the timer is also exceeded (more than seven days), the user will not be able to allocate more disk blocks on that file system until his or her usage drops below the soft limit.

The Difference Between Disk Block and File Limits

A file system provides two resources to the user: blocks (for data) and inodes (for files). Each file consumes one inode. File data is stored in data blocks (usually made up of 1 Kbyte blocks).

Assuming there are no directories, a user can exceed his or her inode quota by creating all empty files (without using any blocks). A user can also use one inode yet exceed his or her block quota by simply creating one file large enough to consume all the data blocks in the user's quota.

Setting Up Quotas

You can set up quotas to limit the amount of disk space and number of inodes (roughly equivalent to the number of files) available to users. These quotas are activated automatically each time a file system is mounted. This section describes how to configure file systems for quotas, and how to set up and activate quotas.

Setting up quotas involves these general steps:

1. A series of commands prepares a file system to accept quotas, ensuring that quotas will be enforced each time the system is rebooted and the file system is mounted. Entries must be added to the /etc/vfstab file, and a quotas file must be created in the top-level directory of the file system.

2. After a quota is created for one user, it can be copied as a prototype to set up other user quotas.

3. Before quotas are actually turned on, another command checks for consistency by comparing the proposed quotas with the current disk usage to make sure that there are no conflicts.

4. Finally, a command turns the quotas on for one or more entire file systems.

These steps ensure that quotas are automatically activated on a file system each time it is mounted. For specific information about these procedures, see "Setting Up Quotas (Task Map)" on page 251.

The following table describes the commands you use to set up disk quotas.

TABLE 17–1 Commands for Setting Up Quotas

Command	Task	Man Page
edquota	Sets the hard limits and soft limits on the number of inodes and the amount of disk space for each user	edquota(1M)
quotacheck	Examines each mounted UFS file system, comparing the file system's current disk usage against information stored in the file system's disk quota file, and resolves inconsistencies	quotacheck(1M)
quotaon	Activates the quotas for the specified file systems	quotaon(1M)
quota	Displays users' disk quotas on mounted file systems to verify that the quotas have been correctly set up	quota(1M)

Guidelines for Setting Up Quotas

Before you set up quotas, you need to determine how much space and how many inodes to allocate to each user. If you want to be sure that the total file system space is never exceeded, you can divide the total size of the file system between the number of users. For example, if three users share a 100-Mbyte slice and have equal disk space needs, you could allocate 33 Mbytes to each user.

In environments where not all users are likely to push their limits, you might want to set individual quotas so that they add up to more than the total size of the file system. For example, if three users share a 100-Mbyte slice, you could allocate 40 Mbytes to each.

When you have established a quota for one user by using the `edquota` command, you can use this quota as a prototype to set the same quota for other users on the same file system.

Before you turn on the quotas, you must first configure the UFS file systems for the quotas, establish quotas for each user, and run the `quotacheck` command to check for consistency between current disk usage and quota files. Also, if systems are rebooted infrequently, it is a good idea to periodically run the `quotacheck` command.

The quotas you set up with the `edquota` command are not enforced until you turn them on by using the `quotaon` command. If you have properly configured the quota files, the quotas are turned on automatically each time a system is rebooted and the file system is mounted.

Setting Up Quotas (Task Map)

Task	Description	For Instructions
1. Configure a file system for quotas	Edit `/etc/vfstab` so that quotas are activated each time the file system is mounted, and create a `quotas` file.	"How to Configure File Systems for Quotas" on page 252
2. Set up quotas for a user	Use the `edquota` command to create disk quotas and inode quotas for a single user account.	"How to Set Up Quotas for a User" on page 252
3. (Optional) Set up quotas for multiple users	Use the `edquota` command to apply prototype quotas to other user accounts.	"How to Set Up Quotas for Multiple Users" on page 253
4. Check for consistency	Use the `quotacheck` command to compare quotas to current disk usage for consistency across one or more file systems.	"How to Check Quota Consistency" on page 254

Task	Description	For Instructions
5. Turn on quotas	Use the `quotaon` command to initiate quotas on one or more file systems.	"How to Turn On Quotas" on page 255

▼ How to Configure File Systems for Quotas

1. **Become superuser.**

2. **Edit the** `/etc/vfstab` **file and add** `rq` **to the** `mount options` **field for each UFS file system that will have quotas.**

3. **Change directory to the root of the file system that will have quotas.**

4. **Create a file named** `quotas`.

   ```
   # touch quotas
   ```

5. **Change permissions to read/write for root access only.**

   ```
   # chmod 600 quotas
   ```

Examples—Configuring File Systems for Quotas

The following `/etc/vfstab` example shows that the `/export/home` directory from the system `pluto` is mounted as an NFS file system on the local system. You can tell quotas are enabled by the `rq` entry under the `mount options` column.

```
#device            device    mount        FS     fsck    mount    mount
#to mount          to fsck   point        type   pass    at boot  options
#
pluto:/export/home -         /export/home nfs    -       yes      rq
```

The following example line from `/etc/vfstab` shows that the local `/work` directory is mounted with quotas enabled, signified by the `rq` entry under the `mount options` column.

```
#device            device             mount  FS    fsck mount    mount
#to mount          to fsck            point  type  pass at boot  options
#
/dev/dsk/c0t4d0s0 /dev/rdsk/c0t4d0s0 /work ufs   3    yes      rq
```

▼ How to Set Up Quotas for a User

1. **Become superuser.**

2. **Use the quota editor to create a temporary file that contains one line of quota information for each mounted UFS file system that has a** `quotas` **file in the file system's root directory.**

   # **edquota** *username*

   Where *username* is the user for whom you want to set up quotas.

3. **Change the number of 1-Kbyte disk blocks (both soft and hard) and the number of inodes (both soft and hard) from 0 (the default) to the quotas you specify for each file system.**

4. **Verify the user's quota by using the** `quota` **command.**

   # **quota -v** *username*

`-v`	Displays the user's quota information on all mounted file systems where quotas exist.
username	Specifies the user name to view quota limits.

Examples—Setting Up Quotas for a User

The following example shows the contents of the temporary file opened by `edquota` on a system where `/files` is the only mounted file system containing a `quotas` file in the root directory.

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

The following example shows the same line in the temporary file after quotas have been set up.

```
fs /files blocks (soft = 50, hard = 60) inodes (soft = 90, hard = 100)
```

▼ How to Set Up Quotas for Multiple Users

1. **Become superuser.**

2. **Use the quota editor to apply the quotas you already established for a prototype user to the additional users you specify.**

   # **edquota -p** *prototype-user username ...*

prototype-user	User name of the account for which you have set up quotas.

username ...	Specifies one or more user names of additional accounts.

Example—Setting Up Prototype Quotas for Multiple Users

The following example applies the quotas established for user bob to users mary and john.

```
# edquota -p bob mary john
```

▼ How to Check Quota Consistency

Note – To ensure accurate disk data, the file systems being checked should be quiescent when you run the quotacheck command manually. The quotacheck command is run automatically when a system is rebooted.

---

1. **Become superuser.**

2. **Run a consistency check on UFS file systems.**

   ```
   # quotacheck [-va] filesystem
   ```

-v	(Optional) Identifies the disk quotas for each user on a particular file system.
-a	Checks all file systems with an rq entry in the /etc/vfstab file.
filesystem	Specifies the file system to check.

   See quotacheck(1M) for more information.

Example—Checking Quota Consistency

The following example checks quotas for the /export/home file system on the /dev/rdsk/c0t0d0s7 slice. The /export/home file system is the only file system with an rq entry in the /etc/vfstab file.

```
# quotacheck -va
*** Checking quotas for /dev/rdsk/c0t0d0s7 (/export/home)
```

▼ How to Turn On Quotas

1. **Become superuser.**

2. **Turn on file system quotas by using the** quotaon **command.**

   # **quotaon** [**-v**] **-a** *filesystem ...*

-v	Displays a message for each file system after quotas are turned on.
-a	Turns on quotas for all file systems with an rq entry in the /etc/vfstab file.
filesystem ...	Turns on quotas for one or more file systems that you specify. More than one file system is specified by separating each file system name with a space.

Example—Turning On Quotas

The following example turns quotas on for the file systems on the /dev/dsk/c0t4d0s7 and /dev/dsk/c0t3d0s7 slices.

```
# quotaon -v /dev/dsk/c0t4d0s7 /dev/dsk/c0t3d0s7
/dev/dsk/c0t4d0s7: quotas turned on
/dev/dsk/c0t3d0s7: quotas turned on
```

Checking Quotas

After you have set up and turned on disk quotas and inode quotas, you can check for users who exceed their quotas. In addition, you can check quota information for entire file systems.

The following table describes the commands you use to check quotas.

TABLE 17–2 Commands for Checking Quotas

Command	Task
quota	Displays user quotas and current disk use, and information about users who are exceeding their quotas
repquota	Displays quotas, files, and the amount of space owned for specified file systems

▼ How to Check for Exceeded Quotas

You can display the quotas and disk use for individual users on file systems on which quotas have been activated by using the `quota` command.

1. **Become superuser.**

2. **Display user quotas for mounted file systems where quotas are enabled.**

   # **quota** [**-v**] *username*

-v	Displays one or more users' quotas on all mounted file systems that have quotas.
username	Is the login name or UID of a user's account.

Example—Checking for Exceeded Quotas

The following example shows that the user account identified by UID 301 has one 1–Kbyte quota but has not used any disk space.

```
# quota -v 301
Disk quotas for bob (uid 301):
Filesystem  usage  quota limit timeleft files quota  limit timeleft
/export/home   0      1     2                0     2      3
```

Filesystem	Is the mount point for the file system.
usage	Is the current block usage.
quota	Is the soft block limit.
limit	Is the hard block limit.
timeleft	Is the amount of time (in days) left on the quota timer.
files	Is the current inode usage.
quota	Is the soft inode limit.
limit	Is the hard inode limit.
timeleft	Is the amount of time (in days) left on the quota timer.

▼ How to Check Quotas on a File System

Display the quotas and disk use for all users on one or more file systems by using the `repquota` command.

1. **Become superuser.**

2. **Display all quotas for one or more file systems, even if there is no usage.**

   # **repquota** [**-v**] **-a** *filesystem*

-v	Reports on quotas for all users, even those users who do not consume resources.
-a	Reports on all file systems.
filesystem	Reports on the specified file system.

Example—Checking Quotas on a File System

The following example shows output from the repquota command on a system that has quotas enabled on only one file system (/export/home).

```
# repquota -va
/dev/dsk/c0t3d0s7 (/export/home):
                  Block limits               File limits
User      used   soft   hard  timeleft  used  soft   hard  timeleft
#301  --            0      1   2.0 days          0     2     3
#341  --    57     50     60   7.0 days  2      90   100
```

Block limits	
used	Is the current block usage.
soft	Is the soft block limit.
hard	Is the hard block limit.
timeleft	Is the amount of time (in days) left on the quota timer.
File limits	
used	Is the current inode usage.
soft	Is the soft inode limit.
hard	Is the hard inode limit.
timeleft	Is the amount of time (in days) left on the quota timer.

Changing and Removing Quotas

You can change quotas to adjust the amount of disk space or the number of inodes users can consume. You can also remove quotas, for individual users or from entire file systems, as needed.

The following table describes the commands you use to change quotas or remove quotas.

TABLE 17–3 Commands for Changing Quotas and Removing Quotas

Command	Task
edquota	Changes the hard limits and soft limits on the number of inodes or amount of disk space for each user. Also, changes the soft quota time limit for each file system with a quota.
quotaoff	Turns off quotas for specified file systems.

▼ How to Change the Soft Time Limit Default

By default users can exceed the soft time limits for their quotas for one week. So, after a week of repeated violations of the soft time limits of either disk space quotas or inode quotas, the system prevents users from using any more inodes or disk blocks.

You can change the length of time that users may exceed their disk space quotas or inode quotas by using the edquota command.

1. **Become superuser.**

2. **Use the quota editor to create a temporary file that contains soft time limits.**

   # **edquota -t**

   Where the -toption specifies the editing of the soft time limits for each file system.

3. **Change the time limits from 0 (the default) to the time limits you specify by numbers and the keywords** month, week, day, hour, min, **or** sec.

Note – This procedure does not affect current quota violators.

Examples—Changing the Soft Time Limit Default

The following example shows the contents of the temporary file opened by edquota on a system where /export/home is the only mounted file system with quotas. The 0 (default) value means that the default time limit of one week is used.

```
fs /export/home blocks time limit = 0 (default), files time limit = 0 (default)
```

The following example shows the same temporary file after the time limit for exceeding the blocks quota has been changed to two weeks, and the time limit for exceeding the number of files has been changed to 16 days.

```
fs /export/home blocks time limit = 2 weeks, files time limit = 16 days
```

▼ How to Change Quotas for a User

1. **Become superuser.**

2. **Use the quota editor to open a temporary file containing one line for each mounted file system that has a** quotas **file in the file system's root directory.**

   # **edquota** *username*

   Where *username* specifies the user name whose quota you want to change.

⚠ **Caution –** Although you can specify multiple users as arguments to the edquota command, the information displayed does not show which user this information belongs to, which could create some confusion.

---

3. **Enter the number of 1-Kbyte disk blocks, both soft and hard, and the number of inodes, both soft and hard.**

4. **Verify that a user's quota has been correctly changed by using the** quota **command.**

   # **quota -v** *username*

   -v   Displays user quota information on all mounted file
                                       systems with quotas enabled.

username	Specifies the user name whose quota you want to check.

Examples—Changing Quotas for a User

The following example shows the contents of the temporary file opened by the
edquota command on a system where /files is the only mounted file system
containing a quotas file in the file system's root directory.

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

The following example shows the same temporary file after quotas have been
changed.

```
fs /files blocks (soft = 0, hard = 500) inodes (soft = 0, hard = 100)
```

The following example shows how to verify that the hard quotas for user smith have
been changed to 500 1-Kbyte blocks, and 100 inodes.

```
# quota -v smith
Disk quotas for smith (uid 12):
Filesystem  usage  quota  limit  timeleft  files  quota  limit  timeleft

  /files     1      0     500               1      0     100
```

▼ How to Disable Quotas for a User

1. **Become superuser.**

2. **Use the quota editor to create a temporary file containing one line for each mounted
   file system that has a** quotas **file in its top-level directory.**

   ```
   # edquota username
   ```
   Where *username* specifies the user name whose quota you want to disable.

Caution – Although you can specify multiple users as arguments to the `edquota` command, the information displayed does not show which user this information belongs with, which could create some confusion.

3. **Change the number of 1-Kbyte disk blocks, both soft and hard, and the number of inodes, both soft and hard, to 0.**

Note – Be sure you change the values to zero. Do *not* delete the line from the text file.

4. **Verify that you have disabled a user's quota by using the** `quota` **command.**

   # **`quota -v`** *username*

`-v`	Displays user quota information on all mounted file systems with quotas enabled.
username	Specifies the user name (UID) whose quota you want to check.

Examples—Disabling Quotas for a User

The following example shows the contents of the temporary file opened by the `edquota` command on a system where `/files` is the only mounted file system that contains a quotas file in the file system's root directory.

```
fs /files blocks (soft = 50, hard = 60) inodes (soft = 90, hard = 100)
```

The following example shows the same temporary file after quotas have been disabled.

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

▼ How to Turn Off Quotas

1. **Become superuser.**

2. **Turn off file system quotas.**

   # **quotaoff** [**-v**] **-a** *filesystem ...*

-v	Displays a message from each file system when quotas are turned off.
-a	Turns off quotas for all file systems.
filesystem	Turns off quotas for one or more file systems you specify. More than one file system is specified by separating each file system name with a space.

Example—Turning Off Quotas

The following example turns off the quotas for the /export/home file system.

```
# quotaoff -v /export/home
/export/home: quotas turned off
```

Scheduling System Tasks (Tasks)

This chapter describes how to schedule routine or single (one-time) systems tasks by using the `crontab` and `at` commands. This chapter also explains how to control access to these commands by using `cron.deny`, `cron.allow`, and `at.deny` files.

This is a list of the step-by-step instructions in this chapter.

- "How to Create or Edit a `crontab` File" on page 268
- "How to Display a `crontab` File" on page 270
- "How to Remove a `crontab` File" on page 271
- "How to Deny `crontab` Access" on page 273
- "How to Limit `crontab` Access to Specified Users" on page 273
- "How to Create an `at` Job" on page 276
- "How to Display the `at` Queue" on page 277
- "How to Display `at` Jobs" on page 277
- "How to Remove `at` Jobs" on page 278
- "How to Deny Access to the `at` Command" on page 279

Ways to Automatically Execute System Tasks

You can set up many system tasks to execute automatically. Some of these tasks should occur at regular intervals. Other tasks need to run only once, perhaps during off hours such as evenings or weekends.

This section contains overview information about two commands, `crontab` and `at`, which enable you to schedule routine tasks to execute automatically. The `crontab` schedules repetitive commands. The `at` command schedules tasks that execute once.

The following table summarizes `crontab` and `at` commands, as well as the files that enable you to control access to these commands.

TABLE 18–1 Command Summary: Scheduling System Tasks

Command	What It Schedules	Location of Files	Files That Control Access
`crontab`	Multiple system tasks at regular intervals	`/var/spool/cron/crontabs`	`/etc/cron.d/cron.allow` and `/etc/cron.d/cron.deny`
`at`	A single system task	`/var/spool/cron/atjobs`	`/etc/cron.d/at.deny`

You can also use the Solaris Management Console's Scheduled Jobs tool to schedule routine tasks. See "Working With the Solaris Management Console (Tasks)" in *System Administration Guide: Basic Administration* for information on using and starting the Solaris Management Console.

For Scheduling Repetitive Jobs: `crontab`

You can schedule routine system administration tasks to execute daily, weekly, or monthly by using the `crontab` command.

Daily `crontab` system administration tasks might include:

- Removing files more than a few days old from temporary directories
- Executing accounting summary commands
- Taking snapshots of the system by using the `df` and `ps` commands
- Performing daily security monitoring
- Running system backups

Weekly `crontab` system administration tasks might include:

- Rebuilding the `catman` database for use by the `man -k` command
- Running the `fsck -n` command to list any disk problems

Monthly `crontab` system administration tasks might include:

- Listing files not used during a specific month
- Producing monthly accounting reports

Additionally, users can schedule `crontab` commands to execute other routine system tasks, such as sending reminders and removing backup files.

For step-by-step instructions on scheduling `crontab` jobs, see Chapter 18.

For Scheduling a Single Job: `at`

The `at` command allows you to schedule a job for execution at a later time. The job can consist of a single command or a script.

Similar to `crontab`, the `at` command allows you to schedule the automatic execution of routine tasks. However, unlike `crontab` files, `at` files execute their tasks once, and then are removed from their directory. Therefore, `at` is most useful for running simple commands or scripts that direct output into separate files for later examination.

Submitting an `at` job involves entering a command and following the `at` command syntax to specify options to schedule the time your job will be executed. For more information about submitting `at` jobs, see "Description of the `at` Command" on page 275.

The `at` command stores the command or script you entered, along with a copy of your current environment variable in the `/var/spool/cron/atjobs` directory. Your `at` job file name is given a long number that specifies its location in the `at` queue, followed by the `.a` extension, such as `793962000.a`.

The `cron` daemon periodically executes the `atrun` program, usually at 15-minute intervals. Then the `atrun` program executes `at` jobs at their scheduled times. After the `cron` daemon executes your `at` job, the `at` job's file is removed from the `atjobs` directory.

For step-by-step instructions on scheduling `at` jobs, see Chapter 15.

——————

Scheduling a Repetitive System Task (`cron`)

The following sections describe how to create, edit, display, and remove `crontab` files, as well as how to control access to them.

Inside a `crontab` File

The `cron` daemon schedules system tasks according to commands found within each `crontab` file. A `crontab` file consists of commands, one per line, that will be executed at regular intervals. The beginning of each line contains date and time information that tells the `cron` daemon when to execute the command.

For example, a crontab file named root is supplied during SunOS software installation. The file's contents include these command lines:

```
10 3 * * 0,4 /etc/cron.d/logchecker
10 3 * * 0   /usr/lib/newsyslog
15 3 * * 0 /usr/lib/fs/nfs/nfsfind
1 2 * * * [ -x /usr/sbin/rtc ] && /usr/sbin/rtc -c > /dev/null 2>&1
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean
```

The first command line instructs the system to run logchecker at 3:10 a.m. every Sunday and Thursday. The second command line instructs the system to run newsyslog at 3:10 a.m. every Sunday. The third command line instructs the system to execute nfsfind every Sunday at 3:15 a.m. The fourth command line instructs the system to check for daylight savings time (and make corrections if necessary) at 2:10 a.m. daily. If there is no RTC time zone nor an /etc/rtc_config file, this entry does nothing. The fifth command line instructs the system to check for (and remove) duplicate entries in the Generic Security Service table, /etc/gss/gsscred_db, at 3:30 a.m. daily.

For more information about the syntax of lines within a crontab file, see "Syntax of crontab File Entries" on page 267.

The crontab files are stored in /var/spool/cron/crontabs. Several crontab files besides root are provided during SunOS software installation (see the following table).

TABLE 18–2 Default crontab Files

crontab **File**	**Function**
adm	Accounting
lp	Printing
root	General system functions and file system cleanup
sys	Performance collection
uucp	General uucp cleanup

Besides the default crontab files, users can create crontab files to schedule their own system tasks. Other crontab files are named after the user accounts in which they are created, such as bob, mary, smith, or jones.

To access crontab files that belong to root or other users, superuser privileges are required.

Procedures explaining how to create, edit, display, and remove crontab files are described in subsequent sections.

How the `cron` Daemon Handles Scheduling

The `cron` daemon manages the automatic scheduling of `crontab` commands. The role of the `cron` daemon is to check the `/var/spool/cron/crontab` directory for the presence of `crontab` files, normally every 15 minutes. The `cron` daemon checks for new `crontab` files or changes to existing ones, reads the execution times listed within the files, and submits the commands for execution at the proper times.

In much the same way, the `cron` daemon controls the scheduling of `at` files, which are stored in the `/var/spool/cron/atjobs` directory.

Syntax of `crontab` File Entries

A `crontab` file consists of commands, one per line, that execute automatically at the time specified by the first five fields at the beginning of each command line. These first five fields, described in the following table, are separated by spaces.

TABLE 18–3 Acceptable Values for `crontab` Time Fields

Time Field	Values
Minute	0-59
Hour	0-23
Day of month	1-31
Month	1-12
Day of week	0-6 (0 = Sunday)

Follow these guidelines for using special characters in `crontab` time fields:

- Use a space to separate each field.
- Use a comma to separate multiple values.
- Use a hyphen to designate a range of values.
- Use an asterisk as a wildcard to include all possible values.
- Use a comment mark (#) at the beginning of a line to indicate a comment or a blank line.

For example, the following `crontab` command entry displays a reminder in the user's console window at 4 p.m. on the first and fifteenth of every month.

```
0 16 1,15 * * echo Timesheets Due > /dev/console
```

Each command within a crontab file must consist of one line, even if that line is very long, because crontab does not recognize extra carriage returns. For more detailed information about crontab entries and command options, refer to crontab(1).

Creating and Editing crontab Files

The simplest way to create a crontab file is to use the crontab -e command. This command invoke the text editor set up for your system environment, which is defined by the EDITOR environment variable. If this variable has not been set, crontab uses the default editor ed.Preferably, you should choose a text that you know well.

The following example shows how to determine if an editor has been defined, and how to set up vi as the default.

```
$ which $EDITOR
$
$ EDITOR=vi
$ export EDITOR
```

When you create a crontab file, it is automatically placed in the /var/spool/cron/crontabs directory and is given your user name. You can create or edit a crontab file for another user, or root, if you have superuser privileges.

▼ How to Create or Edit a crontab File

1. **(Optional) Become superuser to create or edit a** crontab **file that belongs to root or another user.**

2. **Create a new** crontab **file, or edit an existing one.**

   ```
   $ crontab -e [username]
   ```

   Where *username* specifies the name of the user's account for which you want to create or edit a crontab file. To create or edit crontab files requires superuser privileges.

> **Caution –** If you accidentally type the `crontab` command with no option, press the interrupt character for your editor. This character allows you to quit without saving changes. If you instead saved changes and exited the file, the existing `crontab` file is overwritten with an empty file.

3. **Add command lines to the file.**

   Follow the syntax described in "Syntax of `crontab` File Entries" on page 267. The `crontab` file will be placed in `/var/spool/cron/crontabs`.

4. **Verify your** `crontab` **file changes by using the** `crontab -l` **command.**

   ```
   # crontab -l [username]
   ```

Example—Creating or Editing a `crontab` File

The following example shows how to create a `crontab` file for another user.

```
# crontab -e jones
```

The following command entry added to a new `crontab` file automatically removes any log files from the user's home directory at 1:00 a.m. every Sunday morning. Because the command entry does not redirect output, redirect characters are added to the command line after `*.log` to make sure that the command executes properly.

```
# This command helps clean up user accounts.
1 0 * * 0 rm /home/jones/*.log > /dev/null 2>&1
```

▼ How to Verify a `crontab` File

To verify that a `crontab` file exists for a user, use the `ls -l` command in the `/var/spool/cron/crontabs` directory. For example, the following display shows that `crontab` files exist for users `smith` and `jones`.

```
$ ls -l /var/spool/cron/crontabs
-rw-r--r--  1 root     sys            190 Feb 26 16:23 adm
-rw-------  1 root     staff          225 Mar  1  9:19 jones
-rw-r--r--  1 root     root          1063 Feb 26 16:23 lp
-rw-r--r--  1 root     sys            441 Feb 26 16:25 root
-rw-------  1 root     staff           60 Mar  1  9:15 smith
-rw-r--r--  1 root     sys            308 Feb 26 16:23 sys
```

Verify the contents of user's `crontab` file by using `crontab -l` as described in "How to Display a `crontab` File" on page 270.

Displaying `crontab` Files

The `crontab -l` command displays the contents of your `crontab` file much the way the `cat` command displays the contents of other types of files. You do not have to change directories to `/var/spool/cron/crontabs` (where `crontab` files are located) to use this command.

By default, the `crontab -l` command displays your own `crontab` file. To display `crontab` files that belong to other users, you must be superuser.

▼ How to Display a `crontab` File

1. **(Optional) Become superuser to display a** `crontab` **file that belongs to root or another user.**

2. **Display the** `crontab` **file.**

   $ **`crontab -l`** [*username*]

   Where *username* specifies the name of the user's account for which you want to create or edit a `crontab` file. To create or edit `crontab` files requires superuser privileges.

> **Caution –** If you accidentally type the `crontab` command with no option, press the interrupt character for your editor. This character allows you to quit without saving changes. If you instead saved changes and exited the file, the existing `crontab` file is overwritten with an empty file.

Example—Displaying a `crontab` File

The following example shows how to use `crontab -l` to display the contents of the user's default `crontab` file, the default root `crontab` file, and the `crontab` file belonging to another user.

```
$ crontab -l
13 13 * * * chmod g+w /home1/documents/*.book > /dev/null 2>&1
$ su
Password:
# crontab -l
#ident  "@(#)root       1.19    98/07/06 SMI"   /* SVr4.0 1.1.3.1       */
#
# The root crontab should be used to perform accounting data collection.
#
# The rtc command is run to adjust the real time clock if and when
# daylight savings time changes.
```

```
#
10 3 * * 0,4 /etc/cron.d/logchecker
10 3 * * 0   /usr/lib/newsyslog
15 3 * * 0 /usr/lib/fs/nfs/nfsfind
1 2 * * * [ -x /usr/sbin/rtc ] && /usr/sbin/rtc -c > /dev/null 2>&1
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean
# crontab -l jones
13 13 * * * cp /home/jones/work_files /usr/backup/. > /dev/null
2>&1
```

Removing `crontab` Files

By default, `crontab` file protections are set up so that you cannot inadvertently delete a `crontab` file by using the `rm` command. Instead, use the `crontab -r` command to remove `crontab` files.

By default, `crontab -r` removes your own `crontab` file. You must be superuser to remove `crontab` files that belong to superuser or other users.

You do not have to change directories to `/var/spool/cron/crontabs` (where `crontab` files are located) to use this command.

▼ How to Remove a `crontab` File

1. **(Optional) Become superuser to remove a** `crontab` **file that belongs to root or another user.**

2. **Remove the** `crontab` **file.**

   $ **`crontab -r`** [*username*]

   Where *username* specifies the name of the user's account for which you want to create or edit a `crontab` file. To create or edit `crontab` files requires superuser privileges.

⚠ **Caution –** If you accidentally type the `crontab` command with no option, press the interrupt character for your editor. This character allows you to quit without saving changes. If you instead saved changes and exited the file, the existing `crontab` file is overwritten with an empty file.

---

3. **Verify that the** `crontab` **file is removed.**

   # **`ls /var/spool/cron/crontabs`**

Example—Removing a `crontab` File

The following example shows how user `smith` uses the `crontab -r` command to remove his `crontab` file.

```
$ ls /var/spool/cron/crontabs
adm     jones    lp    root    smith    sys     uucp
$ crontab -r
$ ls /var/spool/cron/crontabs
adm     jones    lp    root    sys     uucp
```

Controlling Access to the `crontab` Command

You can control access to the`crontab` command by using two files in the `/etc/cron.d` directory: `cron.deny` and `cron.allow`. These files permit only specified users to perform the `crontab` command tasks such as creating, editing, displaying, or removing their own `crontab` files.

The `cron.deny` and `cron.allow` files consist of a list of user names, one per line. These access control files work together as follows:

- If `cron.allow` exists, only the users listed in this file can create, edit, display, or remove `crontab` files.

- If `cron.allow` does not exist, all users can submit `crontab` files, except for users listed in `cron.deny`.

- If neither `cron.allow` nor `cron.deny` exists, superuser privileges are required to run the `crontab` command.

Superuser privileges are required to edit or create the `cron.deny` and `cron.allow` files.

The `cron.deny` file, created during SunOS software installation, contains the following user names:

```
$ cat /etc/cron.d/cron.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
```

None of the user names in the default `cron.deny` file can access the `crontab` command. You can edit this file to add other user names that will be denied access to the `crontab` command.

No default `cron.allow` file is supplied. So, after Solaris software installation, all users (except the ones listed in the default `cron.deny` file) can access the `crontab`command. If you create a `cron.allow` file, only these users can access the `crontab` command.

▼ How to Deny `crontab` Access

1. **Become superuser.**

2. **Edit the** `/etc/cron.d/cron.deny` **file and add user names, one per line, who will be prevented from using** `crontab` **commands.**

   ```
   daemon
   bin
   smtp
   nuucp
   listen
   nobody
   noaccess
   username1
   username2
   username3
   .
   .
   .
   ```

3. **Verify the** `/etc/cron.d/cron.deny` **file.**

   ```
   # cat /etc/cron.d/cron.deny
   ```

▼ How to Limit `crontab` Access to Specified Users

1. **Become superuser.**

2. **Create the** `/etc/cron.d/cron.allow` **file.**

3. **Enter the** `root` **user name into the** `cron.allow` **file.**

   If you do not add `root` to the file, superuser access to `crontab` commands will be denied.

4. **Enter the user names, one per line, that will be allowed to use the** `crontab` **command.**

root
username1
username2
username3
.
.
.

Examples—Limiting `crontab` Access to Specified Users

The following example shows a `cron.deny` file that prevents user names `visitor`, `jones`, and `temp` from accessing the `crontab`command.

```
$ cat /etc/cron.d/cron.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
jones
temp
visitor
```

The following example shows a `cron.allow` file. The users `smith`, `jones`, `lp`, and `root` are the only ones who can access the `crontab` command.

```
$ cat /etc/cron.d/cron.allow
root
jones
lp
smith
```

▼ How to Verify Limited `crontab` Access

To verify if a specific user can access `crontab`, use the `crontab -l` command while you are logged into the user account.

```
$ crontab -l
```

If the user can access `crontab`, and already has created a `crontab` file, the file is displayed. Otherwise, if the user can access `crontab` but no `crontab` file exists, a message such as the following is displayed:

```
crontab: can't open your crontab file
```

This user either is listed in `cron.allow` (if the file exists), or the user is not listed in `cron.deny`.

If the user cannot access the `crontab` command, the following message is displayed whether or not a previous `crontab` file exists:

```
crontab: you are not authorized to use cron. Sorry.
```

This message means that either the user is not listed in `cron.allow` (if the file exists), or the user is listed in `cron.deny`.

Scheduling a Single System Task (`at`)

The following sections describe how to use the `at` command to schedule jobs (commands and scripts) for execution at a later time, how to display and remove these jobs, and how to control access to the `at` command.

By default, users can create, display, and remove their own `at` job files. To access `at` files that belong to root or other users, you must have superuser privileges.

When you submit an `at` job, it is assigned a job identification number along with the `.a` extension, which becomes the job's file name.

Description of the `at` Command

Submitting an `at` job file involves:

1. Invoking the `at` utility and specifying a command execution time.
2. Entering a command or script to execute later.

   Note – If output from this command or script is important, be sure to direct the output to a file for later examination.

For example, the following `at` job removes `core` files from the user account `smith` near midnight on the last day of July.

```
$ at 11:45pm July 31
at> rm /home/smith/*core*
at> Press Control-d
commands will be executed using /bin/csh
job 933486300.a at Tue Jul 31 23:45:00 2001
```

Controlling access to the `at` Command

You can set up a file to control access to the `at` command, permitting only specified users to create, remove, or display queue information about their `at` jobs. The file that controls access to the `at` command, `/etc/cron.d/at.deny`, consists of a list of user names, one per line. The users listed in this file cannot access `at` commands.

The `at.deny` file, created during SunOS software installation, contains the following user names:

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
```

With superuser privileges, you can edit the `at.deny` file to add other user names whose `at` access you want to restrict.

▼ How to Create an `at` Job

1. **Start the `at` utility, specifying the time you want your job executed.**

   $ **at** [**-m**] *time* [*date*]

-m	Sends you email after the job is completed.
time	Hour that you want to schedule the job. Add am or pm if you do not specify the hours according to a 24-hour clock. Acceptable keywords are midnight, noon, and now. Minutes are optional.
date	First three or more letters of a month, a day of the week, or the keywords today or tomorrow.

2. **At the `at` prompt, type the commands or scripts you want to execute, one per line.**

   You may enter more than one command by pressing Return at the end of each line.

3. **Exit the `at` utility and save the `at` job by pressing Control-D.**

   Your `at` job is assigned a queue number, which is also the job's file name. This number is displayed when you exit the `at` utility.

Examples—Creating an `at` Job

The following example shows the `at` job that user `jones` created to remove her backup files at 7:30 p.m. She used the `-m` option so that she would receive an email message after her job completed.

```
$ at -m 1930
at> rm /home/jones/*.backup
at> Press Control-D
job 897355800.a at Thu Jul  12 19:30:00 2001
```

She received a mail message which confirmed the execution of her `at` job.

```
Your "at" job "rm /home/jones/*.backup"
completed.
```

The following example shows how `jones` scheduled a large `at` job for 4:00 a.m. Saturday morning. The job output was directed to `big.file`.

```
$ at 4 am Saturday
at> sort -r /usr/dict/words > /export/home/jones/big.file
```

▼ How to Display the `at` Queue

To check your jobs that are waiting in the `at` queue, use the `atq` command. This command displays status information about the `at` jobs that you created.

```
$ atq
```

▼ How to Verify an `at` Job

To verify that you have created an `at` job, use the `atq` command. The `atq` command confirms that `at` jobs that belong to `jones` have been submitted to the queue.

```
$ atq
Rank      Execution Date     Owner     Job          Queue   Job Name
  1st   Jul 12, 2001 19:30   jones   897355800.a     a      stdin
  2nd   Jul 14, 2001 23:45   jones   897543900.a     a      stdin
  3rd   Jul 17, 2001 04:00   jones   897732000.a     a      stdin
```

▼ How to Display `at` Jobs

To display information about the execution times of your `at` jobs, use the `at -l` command.

```
$ at -l [job-id]
```

Where the -l *job-id* option identifies the identification number of the job whose status you want to display.

Example—Displaying at Jobs

The following example shows output from the at -l command, which provides status information on all jobs submitted by a user.

```
$ at -l
897543900.a     Sat Jul 14 23:45:00 2001
897355800.a     Thu Jul 12 19:30:00 2001
897732000.a     Tue Jul 17 04:00:00 2001
```

The following example shows the output displayed when a single job is specified with the at -l command.

```
$ at -l 897732000.a
897732000.a          Tue Jul 17 04:00:00 2001
```

▼ How to Remove at Jobs

1. **(Optional) Become superuser to remove an** at **job that belongs to root or another user.**

2. **Remove the** at **job from the queue before the job is executed.**

   ```
   $ at -r [job-id]
   ```

   Where the -r *job-id* option specifies the identification number of the job you want to remove.

3. **Verify that the** at **job is removed by using the** at -l **(or the** atq**) command.**

   The at -l command displays the jobs remaining in the at queue. The job whose identification number you specified should not appear.

   ```
   $ at -l [job-id]
   ```

Example—Removing at Jobs

In the following example, a user wants to remove an at job that was scheduled to execute at 4 a.m. on July 17th. First, the user displays the at queue to locate the job identification number. Next, the user removes this job from the at queue. Finally, the user verifies that this job has been removed from the queue.

```
$ at -l
897543900.a     Sat Jul 14 23:45:00 2001
897355800.a     Thu Jul 12 19:30:00 2001
```

```
897732000.a    Tue Jul 17 04:00:00 2001
$ at -r 897732000.a
$ at -l 897732000.a
at: 858142000.a: No such file or directory
```

▼ How to Deny Access to the at Command

1. **Become superuser.**

2. **Edit the** /etc/cron.d/at.deny **file and add the names of users, one per line, that will be prevented from using** at **commands.**

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
username1
username2
username3
.
.
.
```

Example—Denying at Access

The following example shows an at.deny file that has been edited so that the users smith and jones cannot access the at command.

```
$ cat at.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
jones
smith
```

▼ How to Verify that `at` Access Is Denied

To verify that a username was added correctly to `/etc/cron.d/at.deny`, use the `at -l` command while logged in as the user. If the user cannot access the `at` command, the following message is displayed.

```
# su smith
Password:
$ at -l
at: you are not authorized to use at.  Sorry.
```

Likewise, if the user tries to submit an `at` job, the following message is displayed:

```
$ at 2:30pm
at: you are not authorized to use at.  Sorry.
```

This message confirms that the user is listed in the `at.deny` file.

If `at` access is allowed, the `at -l` command returns nothing.

Managing System Accounting (Tasks)

This section describes how to set up and maintain system accounting.

This is a list of the step-by-step instructions in this chapter.

- "How to Set Up System Accounting" on page 287
- "How to Bill Users" on page 289
- "How to Fix a Corrupted `wtmpx` File" on page 290
- "How to Fix `tacct` Errors" on page 290
- "How to Restart the `runacct` Script" on page 291
- "How to Set Up System Accounting" on page 287
- "How to Permanently Disable System Accounting" on page 292

For reference information on the various system accounting reports, see Chapter 20.

What is System Accounting?

The SunOS 5.9 system accounting software is a set of programs that enables you to collect and record data about user connect time, CPU time charged to processes, and disk usage. Once you collect this data, you can generate reports and charge fees for system usage.

You can use the accounting programs to:

- Monitor system usage
- Locate and correct performance problems
- Maintain system security

After you set up the system accounting programs, they run mostly on their own.

How System Accounting Works

Automatic accounting is set up by first putting the accounting startup script into root's `crontab` file. The accounting startup script can then be started automatically by the `cron` command.

The following overview illustrates the system accounting process.

1. Between system startup and shutdown, raw data about system use (such as user logins, running processes, and data storage) are collected in accounting files.

2. Periodically (usually once a day), the `/usr/lib/acct/runacct` script processes the various accounting files and produces both cumulative summary files and daily accounting reports. Then the `/usr/lib/acct/prdaily` script prints the daily reports.

   For more information about the `runacct` script, see "The `runacct` Script" on page 295.

3. Monthly, you can process and print the cumulative `runacct` summary files by executing the `monacct` script. The summary reports produced by the `monacct` script provide an efficient means for billing users on a monthly or other fiscal basis.

System Accounting Components

The accounting software provides C language programs and shell scripts that organize data into summary files and reports. These programs reside in the `/usr/lib/acct` directories. The accounting reports reside in the `/var/adm/acct` directory.

Daily accounting can help you perform four types of auditing:

- Connect
- Process
- Disk
- Fee calculations

Connect Accounting

Connect accounting enables you to determine:

- The length of time a user was logged in
- How the `tty` lines are being used
- The number of reboots on your system
- How many times the accounting software was turned off and on

To provide this information, the system stores:

- Records of time adjustments

- Boot times
- Times the accounting software was turned off and on
- Changes in run levels
- The creation of user processes (`login` processes and `init` processes), and the terminations of processes.

These records (produced from the output of system programs such as `date`, `init`, `login`, `ttymon`, and `acctwtmp`) are stored in the `/var/adm/wtmpx` file. Entries in the `wtmpx` file can contain the following information:

- Login name
- Device name
- Process ID
- Entry type
- Tme stamp that denotes when the entry was made.

Process Accounting

Process accounting enables you to keep track of the following data about each process that runs on your system:

- User IDs and group IDs of users using the process
- Beginning times and elapsed times of the process
- CPU time for the process (user time and system time)
- Amount of memory used
- Commands run
- The `tty` controlling the process

Every time a process terminates, the `exit` program collects this information and writes it to the `/var/adm/pacct` file.

Disk Accounting

Disk accounting enables you to gather and format the following data about the files each user has on disks:

- User name and user ID of the user
- Number of blocks used by the user's files

This data is collected by the `/usr/lib/acct/dodisk` shell script at intervals determined by the entry you add to the `/var/spool/cron/crontabs/root` file. In turn, the `dodisk` script invokes the `acctdisk` and `acctdusg` commands, which gather disk usage by login.

Caution – Information gathered by running the dodisk script is stored in the /var/adm/acct/nite/disktacct file. This information is overwritten the next time the dodisk script is run. Therefore, avoid running the dodisk script twice in the same day.

The acctdusg command might overcharge for files that are written randomly, which can create holes in the files. This problem occurs because the acctdusg command does not read the indirect blocks of a file when determining the file size. Rather, the acctdusg command determines the file size by checking the *di_size* value of the inode.

Calculating User Fees

The chargefee utility stores charges for special services provided to a user, such as file restoration, in the /var/adm/fee file. Each entry in the file consists of a user login name, user ID, and the fee. This file is checked by the runacct script every day and new entries are merged into the accounting records. For instructions on running the chargefee script to bill users, see "How to Bill Users" on page 289.

How Daily Accounting Works

Here is a step-by-step summary of how daily accounting works:

1. When the system is switched into multiuser mode, the /usr/lib/acct/startup program is executed. The startup program executes several other programs that invoke daily accounting.

2. The acctwtmp program adds a "boot" record to the /var/adm/wtmpx file. In this record, the system name is shown as the user name in the wtmpx record. The following table summarizes how the raw accounting data is gathered and where it is stored.

TABLE 19–1 Raw Accounting Data

File in /var/adm	Information Stored	Written By	Format
wtmpx	Connect sessions	login, init	binary
	Changes	date	
	Reboots	acctwtmp	
	Shutdowns	shutacct	
pacctn	Processes	Kernel (when the process ends)	binary

TABLE 19–1 Raw Accounting Data   *(Continued)*

File in `/var/adm`	Information Stored	Written By	Format
		`turnacct switch` (which creates a new file when the old one reaches 500 blocks)	
`fee`	Special charges	`chargefee`	ASCII
`acct/nite/disktacct`	Disk space used	`dodisk`	binary

3. The `turnacct` script, invoked with the `-on` option, begins process accounting. Specifically, the `turnacct` script executes the `accton` program with the `/var/adm/pacct` argument.

4. The remove shell script "cleans up" the saved `pacct` and `wtmpx` files left in the `sum` directory by the `runacct` script.

5. The `login` and `init` programs record connect sessions by writing records into the `/var/adm/wtmpx` file. Date changes (using date with an argument) are also written to the `/var/adm/wtmpx` file. Reboots and shutdowns using the `acctwtmp` command are also recorded in the `/var/adm/wtmpx` the.

6. When a process ends, the kernel writes one record per process, using the `acct.h` format, in the `/var/adm/pacct` file.

   Every hour, the `cron` command executes the `ckpacct` script to check the size of the `/var/adm/pacct` file. If the file grows past 500 blocks (default), the `turnacct switch` command is executed. (The program moves the `pacct` file to the `pacct`*n* file and creates a new one.) The advantage of having several smaller `pacct` files becomes apparent when you try to restart the `runacct` script if a failure occurs when processing these records.

7. The `runacct` script is executed by the `cron` command each night. The `runacct`script processes the accounting files: `/var/adm/pacct`*n*, `/var/adm/wtmpx`, `/var/adm/fee`, and `/var/adm/acct/nite/disktacct`, to produce command summaries and usage summaries by user name.

8. The `/usr/lib/acct/prdaily` script is executed on a daily basis by the `runacct` script to write the daily accounting information in the `/var/adm/acct/sum/rprt`*MMDD* files.

9. The `monacct` script should be executed on a monthly basis (or at intervals you determine, such as the end of every fiscal period). The `monacct` script creates a report based on data stored in the `sum` directory that has been updated daily by the `runacct` script. After creating the report, the `monacct` script "cleans up" the `sum` directory to prepare the directory's files for the new `runacct` data.

What Happens if the System Shuts Down

If the system is shut down using the `shutdown` command, the `shutacct` script is executed automatically. The `shutacct` script writes a reason record into the `/var/adm/wtmpx` file and turns off process accounting.

Setting Up System Accounting

You can set up system accounting to run while the system is in multiuser mode (system state 2). Generally, this task involves:

1. Creating the `/etc/rc0.d/K22acct` and `/etc/rc2.d/S22acct` startup scripts

2. Modifying the `/var/spool/cron/crontabs/adm` and `/var/spool/cron/crontabs/root` crontab files

The following table describes the default accounting scripts.

TABLE 19–2 Default Accounting Scripts

Accounting Script	Purpose	Run Frequency
`ckpacct`	Checks the size of the `/usr/adm/pacct` log file and makes sure it does not get too large.	Periodically
`runacct(1M)`	Processes connect, disk, and fee accounting information. You can remove the commands from this script for the accounting features you do not want processed.	Daily
`monacct`	Generates fiscal accounting summary reports on a monthly basis. You can determine how often this script is run. You can remove the commands from this script for the accounting features you do not want generated.	On a fiscal basis

You can choose which accounting scripts run by default. After these entries have been added to the `crontab` files, accounting should run automatically.

▼ How to Set Up System Accounting

1. **Become superuser.**

2. **If necessary, install the** `SUNWaccr` **and** `SUNWaccu` **packages on your system by using the** `pkgadd` **command.**

3. **Install** `/etc/init.d/acct` **as the startup script for Run Level 2.**

   ```
   # ln /etc/init.d/acct /etc/rc2.d/S22acct
   ```

4. **Install** `/etc/init.d/acct` **as the stop script for Run Level 0.**

   ```
   # ln /etc/init.d/acct /etc/rc0.d/K22acct
   ```

5. **Add the following lines to the** `adm crontab` **file to start the** `ckpacct`**,** `runacct`**, and** `monacct` **scripts automatically.**

   ```
   # EDITOR=vi; export EDITOR
   # crontab -e adm
   0 * * * * /usr/lib/acct/ckpacct
   30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
   30 7 1 * * /usr/lib/acct/monacct
   ```

6. **Add the following line to the** `root crontab` **file to start the** `dodisk` **script automatically.**

   ```
   # crontab -e
   30 22 * * 4 /usr/lib/acct/dodisk
   ```

7. **Edit** `/etc/acct/holidays` **to include national and local holidays.**

   See holidays(4) and the example below for more information.

8. **Reboot the system, or start accounting manually by typing:**

   ```
   # /etc/init.d/acct start
   ```

Examples—Setting Up Accounting

This modified `adm crontab` contains entries for the `ckpacct`, `runacct`, and `monacct` scripts.

```
#ident   "@(#)adm      1.5    92/07/14 SMI"    /* SVr4.0 1.2   */
#
# The adm crontab file should contain startup of performance
# collection if the profiling and performance feature has been
# installed.
0 * * * * /usr/lib/acct/ckpacct
30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
30 7 1 * * /usr/lib/acct/monacct
```

This modified `root crontab` contains entries for the `dodisk` program.

```
#ident  "@(#)root       1.16    98/04/28 SMI"   /* SVr4.0 1.1.3.1        */
#
# The root crontab should be used to perform accounting data collection.
#
# The rtc command is run to adjust the real time clock if and when
# daylight savings time changes.
#
10 3 * * 0,4 /etc/cron.d/logchecker
10 3 * * 0   /usr/lib/newsyslog
15 3 * * 0 /usr/lib/fs/nfs/nfsfind
1 2 * * * [ -x /usr/sbin/rtc ] && /usr/sbin/rtc -c > /dev/null 2>&1
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean
30 22 * * 4 /usr/lib/acct/dodisk
```

The following example shows a sample `/etc/acct/holidays` file.

```
* @(#)holidays    January 1, 2001
*
* Prime/Nonprime Table for UNIX Accounting System
*
* Curr    Prime    Non-Prime
* Year    Start    Start
*
  1999    0800    1800
*
* only the first column (month/day) is significant.
*
* month/day    Company
*       Holiday
*
1/1        New Years Day
7/4        Indep. Day
12/25      Christmas
```

Billing Users

If you provide special user services by request, such as restoring files or remote printing, you might want to bill users by running the `chargefee` utility. The `chargefee` utility records charges in the `/var/adm/fee` file . Each time the `runacct` utility is executed, new entries are merged into the total accounting records.

See `acctsh`(1M)for more information.

▼ How to Bill Users

1. **Become superuser.**

2. **Charge a user for special services.**

   # **/usr/lib/acct/chargefee** *username amount*

username	User account you want to bill.
amount	Number of units to bill the user. This is an arbitrary unit that you set to charge users based on some task like printing or restoring a file. You would have to write a script that invokes chargefee and charges a user for a specific task.

Example—Billing Users

The following example charges the user print_customer 10 units.

/usr/lib/acct/chargefee print_customer 10

Maintaining Accounting Information

This section describes how to fix corrupted accounting files and how to restart the runacct script.

Fixing Corrupted Files and wtmpx Errors

Unfortunately, system accounting is not foolproof. Occasionally, a file becomes corrupted or lost. Some of the files can simply be ignored or restored from backup. However, certain files must be fixed to maintain the integrity of system accounting.

The wtmpx files seem to cause the most problems in the daily operation of the system accounting. When the date is changed manually and the system is in multiuser mode, a set of date change records is written into the /var/adm/wtmpx file. The wtmpfix utility is designed to adjust the time stamps in the wtmp records when a date change is encountered. However, some combinations of date changes and reboots slip through the wtmpfix utility and cause the acctcon program to fail.

▼ How to Fix a Corrupted `wtmpx` File

1. **Become superuser.**

2. **Change to the** `/var/adm` **directory.**

3. **Convert the** `wtmpx` **file from binary to ASCII format.**

   `# /usr/lib/acct/fwtmp < wtmpx > wtmpx.ascii`

4. **Edit** `wtmpx.ascii` **to delete the corrupted records.**

5. **Convert the** `wtmpx.ascii` **file back to a binary file.**

   `# /usr/lib/acct/fwtmp -ic < wtmpx.ascii > wtmpx`

   See `fwtmp`(1M) for more information.

Fixing `tacct` Errors

The integrity of the `/var/adm/acct/sum/tacct` file is important if you are charging users for system resources. Occasionally, unusual `tacct` records appear with negative numbers, duplicate user IDs, or a user ID of 65535. First, check the `/var/adm/acct/sum/tacctprev` file by using the `prtacct` script to print it. If the contents look all right, patch the latest `/var/adm/acct/sum/tacct`*MMDD* file, then recreate the `/var/adm/acct/sum/tacct` file. The following steps outline a simple patch procedure.

▼ How to Fix `tacct` Errors

1. **Become superuser.**

2. **Change to the** `/var/adm/acct/sum` **directory.**

3. **Convert the** `tacct`*MMDD* **file from binary to ASCII format.**

   `# /usr/lib/acct/acctmerg -v < tacct`*MMDD* `> xtacct`

   MMDD is pair of two-digit numbers that represent the month and day.

4. **Edit the** `xtacct` **file, removing corrupted records and writing duplicate records to another file.**

5. **Convert the** `xtacct` **file from ASCII format to binary.**

   `# /usr/lib/acct/acctmerg -i < xtacct > tacct`*MMDD*

   MMDD is pair of two-digit numbers that represent the month and day.

6. **Merge the files** `tacctprev` **and** `tacct`.*MMDD* **into the** `tacct` **file.**

   `# /usr/lib/acct/acctmerg < tacctprev tacct`*MMDD*` > tacct`

Restarting the `runacct` Script

The `runacct` script can fail for a variety of reasons. The most common reasons are a system crash, the `/var` directory running out of space, or a corrupted `wtmpx` file. If the `active.`*MMDD* file exists, check it first for error messages. If the `active` and `lock` files exist, check `fd2log` for any relevant messages.

Run without arguments, the `runacct` script assumes that this invocation is the first invocation of the day. The argument *MMDD* is necessary if the `runacct` script is being restarted and specifies the month and day for which the `runacct` script reruns the accounting. The entry point for processing is based on the contents of the `statefile` file. To override the `statefile` file, include the desired state on the command line. See the `runacct`(1M) man page for a description of the available states.

> ⚠ **Caution –** When you run the `runacct` program manually, be sure to run it as user `adm`.

▼ How to Restart the `runacct` Script

1. **Remove the** `lastdate` **file and any** `lock*` **files, if any.**

   ```
   $ cd /var/adm/acct/nite
   $ rm lastdate lock*
   ```

   The `lastdate` file contains the date that the `runacct` program was last run. Restarting `runacct` in the next step recreates this file.

2. **Restart the** `runacct` **script.**

   `$ /usr/lib/acct/runacct` *MMDD* [*state*] `2> /var/adm/acct/nite/fd2log &`

MMDD	Month and day specified by two-digit numbers.
state	Specifies a state, or starting point, where the `runacct` processing should begin.

Stopping and Disabling System Accounting

You can temporarily stop system accounting or disable it permanently.

▼ How to Temporarily Stop System Accounting

1. **Become superuser.**

2. **Edit the** adm crontab **file to stop the** ckpacct, runacct, **and** monacct **programs from running by commenting out the appropriate lines.**

   ```
   # EDITOR=vi; export EDITOR
   # crontab -e adm
   #0 * * * * /usr/lib/acct/ckpacct
   #30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
   #30 7 1 * * /usr/lib/acct/monacct
   ```

3. **Edit the** crontab **file for user** root **to stop the** dodisk **program from running by commenting out the appropriate line.**

   ```
   # crontab -e
   #30 22 * * 4 /usr/lib/acct/dodisk
   ```

4. **Stop the accounting program.**

   ```
   # /etc/init.d/acct stop
   ```

5. **(Optional) Remove the newly added comment symbols from the** crontab **files and restart the accounting program to re-enable system accounting.**

   ```
   # /etc/init.d/acct start
   ```

▼ How to Permanently Disable System Accounting

1. **Become superuser.**

2. **Edit the** adm crontab **file and delete the entries for the** ckpacct, runacct, **and** monacct **programs.**

   ```
   # EDITOR=vi; export EDITOR
   # crontab -e adm
   ```

3. **Edit the** `root crontab` **file and delete the entries for the** `dodisk` **program.**

   # **crontab -e**

4. **Remove the startup script for Run Level 2.**

   # **unlink /etc/rc2.d/S22acct**

5. **Remove the stop script for Run Level 0.**

   # **unlink /etc/rc0.d/K22acct**

6. **Stop the accounting program.**

   # **/etc/init.d/acct stop**

System Accounting (Reference)

This is a list of reference information in this chapter.

The runacct Script

The main daily accounting script, runacct, is normally invoked by the cron command outside of prime business hours. The runacct script processes connect, fee, disk, and process accounting files. This script also prepares daily and cumulative summary files for use by the prdaily and monacct scripts for billing purposes.

The runacct script takes care not to damage files if errors occur. A series of protection mechanisms are used to recognize an error, provide intelligent diagnostics, and complete processing in such a way that the runacct script can be restarted with minimal intervention. It records its progress by writing descriptive messages into the active file. Files used by the runacct script are assumed to be in the /var/adm/acct/nite directory, unless otherwise noted. All diagnostic output during the execution of the runacct script is written into the fd2log file.

When the runacct script is invoked, it creates the lock and lock1 files. These files are used to prevent simultaneous execution of runacct. The runacct program prints an error message if these files exist when it is invoked. The lastdate file contains the month and day the runacct script was last invoked, and is used to prevent more than one execution per day. If the runacct script detects an error, a message is written to the console, mail is sent to root and adm, locks might be

removed, diagnostic files are saved, and execution is ended. For instructions on how to start the runacct script again, see "How to Restart the runacct Script" on page 291.

To allow the runacct script to be restarted, processing is broken down into separate re-entrant states. The statefile file is used to keep track of the last state completed. When each state is completed, the statefile file is updated to reflect the next state. After processing for the state is complete, the statefile file is read and the next state is processed. When the runacct script reaches the CLEANUP state, it removes the locks and ends. States are executed as shown in the following table.

TABLE 20–1 runacct States

State	Description
SETUP	The turnacct switch command is executed to create a new pacct file. The /var/adm/pacct*n* process accounting files (except for the pacct file) are moved to the /var/adm/Spacct*n.MMDD* files. The /var/adm/wtmpx file is moved to the /var/adm/acct/nite/wtmp.*MMDD* file (with the current time record added on the end) and a new /var/adm/wtmp file is created. The closewtmp and utmp2wtmp programs add records to the wtmp.*MMDD* file and the new wtmpx file to account for users currently logged in.
WTMPFIX	The wtmpfix program checks the wtmp.*MMDD* file in the nite directory for accuracy. Because some date changes cause the acctcon program to fail, the wtmpfix program attempts to adjust the time stamps in the wtmpx file if a record of a date change appears. This program also deletes any corrupted entries from the wtmpx file. The fixed version of the wtmp.*MMDD* file is written to the tmpwtmp file.
CONNECT	The acctcon program is used to record connect accounting records in the file ctacct.*MMDD*. These records are in tacct.h format. In addition, acctcon creates the lineuse and reboots files. The reboots file records all the boot records found in the wtmpx file.
PROCESS	The acctprc program is used to convert the /var/adm/Spacct*n.MMDD* process accounting files into total accounting records in the ptacct*n.MMDD* files. The Spacct and ptacct files are correlated by number so that if the runacct script fails, the Spacct files are not processed.
MERGE	The acctmerg program merges the process accounting records with the connect accounting records to form the daytacct file.
FEES	The acctmerg program merges ASCII tacct records from the fee file into the daytacct file.

TABLE 20–1 `runacct` States *(Continued)*

State	Description
DISK	If the `dodisk` script procedure has been run, which produces the `disktacct` file, the DISK program merges the file into the `daytacct` file and moves the `disktacct` file to the `/tmp/disktacct.`*MMDD* file.
MERGETACCT	The `acctmerg` program merges the `daytacct` file with the `sum/tacct` file , the cumulative total accounting file. Each day, the `daytacct` file is saved in the `sum/tacct.`*MMDD*, file so that the `sum/tacct` file can be re-created if it is corrupted or lost.
CMS	The `acctcms` program is run several times. This program is first run to generate the command summary by using the `Spacct`*n* files and write the data to the `sum/daycms` file. The `acctcms` program is then run to merge the `sum/daycms` file with the `sum/cms` cumulative command summary file. Finally, the `acctcms` program is run to produce `nite/daycms` and `nite/cms`, the ASCII command summary files from the `sum/daycms` and `sum/cms` files, respectively. The `lastlogin` program is used to create the `/var/adm/acct/sum/loginlog` log file, the report of when each user last logged in. If the `runacct` script is run after midnight, the dates showing the time last logged in by some users will be incorrect by one day.
USEREXIT	Any installation-dependent (local) accounting program can be included at this point. The `runacct` script expects it to be called the `/usr/lib/acct/runacct.local` program.
CLEANUP	Cleans up temporary files, runs the `prdaily` script and saves its output in the `sum/rpt.`*MMDD* file, removes the locks, and then exits.

Caution – When restarting the `runacct` script in the CLEANUP state, remove the last `ptacct` file because it will not be complete

Daily Accounting Reports

The `runacct` shell script generates five basic reports upon each invocation. The following table describes the five basic reports generated.

TABLE 20–2 Daily Accounting Reports

Report Type	Description
"Daily Report" on page 298	Shows terminal line utilization by `tty` number.
"Daily Usage Report " on page 299	Indicates usage of system resources by users (listed in order of user ID).
"Daily Command Summary " on page 300	Indicates usage of system resources by commands, listed in descending order of memory use. In other words, the command that used the most memory is listed first. This same information is reported for the month with the monthly command summary.
"Monthly Command Summary " on page 302	A cumulative summary that reflects the data accumulated since the last invocation of the `monacct` program.
"Last Login Report " on page 302	Shows the last time each user logged in (arranged in chronological order).

Daily Report

This report gives information about each terminal line used. The following is a sample Daily Report.

```
Oct 16 02:30 2001  DAILY REPORT FOR venus Page 1


from Mon Oct 15 02:30:02 2001
to   Tue Oct 16 02:30:01 2001
1       runacct
1       acctcon

TOTAL DURATION IS 1440 MINUTES
LINE         MINUTES  PERCENT  # SESS  # ON  # OFF
console      868      60       1       1     2
TOTALS       868      --       1       1     2
```

The `from` and `to` lines specify the time period reflected in the report. This time period covers the time the last Daily Report was generated to the time the current Daily Report was generated. The comes a log of system reboots, shutdowns, power failure recoveries, and any other record dumped into the `/var/adm/wtmpx` file by the `acctwtmp` program. For more information, see `acct`(1M).

The second part of the report is a breakdown of terminal line utilization. The TOTAL DURATION tells how long the system was in multiuser mode (accessible through the terminal lines). The following table describes the data provided in the Daily Report.

TABLE 20–3 Daily Report Data

Column	Description
LINE	The terminal line or access port.
MINUTES	The total number of minutes that the line was in use during the accounting period.
PERCENT	The TOTAL DURATION divided by the total number of MINUTES.
# SESS	The number of times this line or port was accessed for a login session.
# ON	Same as SESS. (This column no longer has meaning. Previously, it listed the number of times that a line or port was used to log in a user.)
# OFF	This column reflects the number of times a user logs out and any interrupts that occur on that line. Generally, interrupts occur on a port when ttymon is first invoked after the system is brought to multiuser mode. If the # OFF exceeds the # SESS by a large factor, the multiplexer, modem, or cable is probably going bad, or there is a bad connection somewhere. The most common cause is an unconnected cable dangling from the multiplexer.

During real time, you should monitor the /var/adm/wtmpx file because it is the file from which the connect accounting is derived. If the wtmpx file grows rapidly, execute the following command to see which tty line is the noisiest.

/usr/lib/acct/acctcon -l *file* < /var/adm/wtmpx

If interruption is occurring frequently, general system performance will be affected. Additionally, the wtmp file might become corrupted. To correct this problem, see "How to Fix a Corrupted wtmpx File" on page 290.

Daily Usage Report

The Daily Usage Report gives a breakdown of system resource utilization by user. A sample of this report follows.

```
Oct 16 02:30 2001   DAILY USAGE REPORT FOR skisun Page 1
```

UID	LOGIN NAME	CPU PRIME	(MINS) NPRIME	KCORE-PRIME	MINS NPRIME	CONNECT PRIME	(MINS) NPRIME	DISK BLOCKS	# OF PROCS	# OF SESS	# DISK SAMPLES	FEE
0	TOTAL	72	148	11006173	51168	26230634	57792	539	330	0	2150	1
0	root	32	76	11006164	33664	26230616	22784	0	0	0	127	0
4	adm	0	0	22	51	0	0	0	420	0	0	0
101	rimmer	39	72	894385	1766020	539	330	0	1603	1	0	0

The following table describes the data provided in the Daily Usage Report.

TABLE 20–4 Daily Usage Report Data

Column	Description
UID	User ID number.
LOGIN NAME	Login (or user) name of the user. Identifies a user who has multiple login names.
CPU (MINS)	Amount of time, in minutes, that the user's process used the central processing unit. Divided into PRIME and NPRIME (non-prime) utilization. The accounting system's version of this data is located in the /etc/acct/holidays file.
KCORE-MINS	A cumulative measure of the amount of memory in Kbyte segments per minute that a process uses while running. Divided into PRIME and NPRIME utilization.
CONNECT (MINS)	Amount of timein minutes, that the a user was logged into the system, or "real time." Divided into PRIME and NPRIME use. If these numbers are high while the # OF PROCS is low, you can conclude that the user logs in first thing in the morning and hardly touches the terminal the rest of the day.
DISK BLOCKS	Output from the acctdusg program, which runs the disk accounting programs and merges the accounting records (daytacct). For accounting purposes, a block is 512 bytes.
# OF PROCS	Number of processes invoked by the user. If large numbers appear, a user might have a shell procedure that has run out of control.
# OF SESS	Number of times a user logged on to the system.
# DISK SAMPLES	Number of times disk accounting was run to obtain the average number of DISK BLOCKS.
FEE	Often unused field that represents the total accumulation of units charged against the user by the chargefee script.

Daily Command Summary

The Daily Command Summary report shows the system resource use by command. With this report, you can identify the most heavily used commands and, based on how those commands use system resources, gain insight on how best to tune the system.

These reports are sorted by TOTAL KCOREMIN, which is an arbitrary gauge but often a good one for calculating drain on a system.

A sample daily command summary follows.

```
                       TOTAL COMMAND SUMMARY
COMMAND    NUMBER      TOTAL     TOTAL      TOTAL    MEAN     MEAN     HOG    CHARS       BLOCKS
NAME       CMDS     KCOREMIN  CPU-MIN  REAL-MIN  SIZE-K  CPU-MIN  FACTOR TRNSFD      READ

TOTALS     2150  1334999.75   219.59 724258.50 6079.48    0.10    0.00  397338982 419448

netscape     43  2456898.50    92.03  54503.12 26695.51    2.14    0.00  947774912 225568
adeptedi      7    88328.22     4.03    404.12 21914.95    0.58    0.01   93155160   8774
dtmail        1    54919.17     5.33  17716.57 10308.94    5.33    0.00  213843968  40192
acroread      8    31218.02     2.67  17744.57 11682.66    0.33    0.00  331454464  11260
dtwm          1    16252.93     2.53  17716.57 6416.05    2.53    0.00  158662656  12848
dtterm        5     4762.71     1.30  76300.29 3658.93    0.26    0.00   33828352  11604
dtaction     23     1389.72     0.33      0.60 4196.43    0.01    0.55   18653184    539
dtsessio      1     1174.87     0.24  17716.57 4932.97    0.24    0.00   23535616   5421
dtcm          1      866.30     0.18  17716.57 4826.21    0.18    0.00    3012096   6490
```

The following table describes the data provided in the Daily Command Summary..

TABLE 20–5 Daily Command Summary

Column	Description
COMMAND NAME	Name of the command. Unfortunately, all shell procedures are lumped together under the name sh because only object modules are reported by the process accounting system. You should monitor the frequency of programs called a.out or core or any other unexpected name. You can use the acctcom program to determine who executed an oddly named command and if superuser privileges were used.
NUMBER CMDS	Total number of times this command was run during prime time.
TOTAL KCOREMIN	Total cumulative measurement of the Kbyte segments of memory used by a process per minute of run time.
TOTAL CPU-MIN	Total processing time this program accumulated during prime time.
TOTAL REAL-MIN	Total real-time (wall-clock) minutes this program accumulated.
MEAN SIZE-K	Mean of the TOTAL KCOREMIN over the number of invocations reflected by NUMBER CMDS.
MEAN CPU-MIN	Mean derived between the NUMBER CMDS and TOTAL CPU-MIN.
HOG FACTOR	Total CPU time divided by elapsed time. Shows the ratio of system availability to system use, providing a relative measure of total available CPU time consumed by the process during its execution.
CHARS TRNSFD	Total number of characters pushed around by the read and write system calls. Might be negative due to overflow.

TABLE 20–5 Daily Command Summary   *(Continued)*

Column	Description
BLOCKS READ	Total number of the physical block reads and writes that a process performed.

Monthly Command Summary

The format of the Daily Command Summary and the Monthly Command Summary reports are virtually the same. However, the daily summary reports only on the current accounting period while the monthly summary reports on the start of the fiscal period to the current date. In other words, the monthly report is a cumulative summary that reflects the data accumulated since the last invocation of the monacct program.

A sample report follows.

```
Oct 16 02:30 2001  MONTHLY TOTAL COMMAND SUMMARY Page 1


                              TOTAL COMMAND SUMMARY
COMMAND   NUMBER      TOTAL    TOTAL     TOTAL     MEAN      MEAN    HOG      CHARS      BLOCKS
NAME       CMDS    KCOREMIN  CPU-MIN  REAL-MIN   SIZE-K  CPU-MIN  FACTOR   TRNSFD      READ

TOTALS    42718  4398793.50  361.92  956039.00 12154.09 0.01     0.00  16100942848 825171

netscape    789  3110437.25  121.03   79101.12 25699.58 0.15     0.00   3930527232 302486
adeptedi     84  1214419.00   50.20    4174.65 24193.62 0.60     0.01    890216640 107237
acroread    145   165297.78    7.01   18180.74 23566.84 0.05     0.00   1900504064  26053
dtmail        2    64208.90    6.35   20557.14 10112.43 3.17     0.00    250445824  43280
dtaction    800    47602.28   11.26      15.37  4226.93 0.01     0.73    640057536   8095
soffice.     13    35506.79    0.97       9.23 36510.84 0.07     0.11    134754320   5712
dtwm          2    20350.98    3.17   20557.14  6419.87 1.59     0.00    190636032  14049
```

For a description of the data provided in the Monthly Command Summary, see "Daily Command Summary" on page 300.

Last Login Report

This report gives the date when a particular login was last used. You can use this information to find unused logins and login directories that can be archived and deleted. A sample report appears follows.

```
Oct 16 02:30 2001  LAST LOGIN Page 1


01-06-12  kryten        01-09-08  protoA     01-10-14  ripley
```

```
01-07-14  lister          01-09-08  protoB       01-10-15  scutter1
01-08-16  pmorph          01-10-12  rimmer       01-10-16  scutter2
```

Looking at the `pacct` File With `acctcom`

At any time, you can examine the contents of the /var/adm/pacct*n* files, or any file
with records in the acct.h format, by using the acctcom program. If you do not
specify any files and do not provide any standard input when you run this command,
the acctcom command reads the pacct file. Each record read by the acctcom
command represents information about a terminated process. Active processes can be
examined by running the ps command. The default output of the acctcom command
provides the following information:

Sample acctcom output follows:

```
# acctcom
COMMAND                               START     END       REAL    CPU     MEAN
NAME       USER     TTYNAME          TIME      TIME      (SECS)  (SECS)  SIZE(K)
#accton    root     ?                02:30:01  02:30:01  0.03    0.01    304.00
turnacct   adm      ?                02:30:01  02:30:01  0.42    0.01    320.00
mv         adm      ?                02:30:01  02:30:01  0.07    0.01    504.00
utmp_upd   adm      ?                02:30:01  02:30:01  0.03    0.01    712.00
utmp_upd   adm      ?                02:30:01  02:30:01  0.01    0.01    824.00
utmp_upd   adm      ?                02:30:01  02:30:01  0.01    0.01    912.00
utmp_upd   adm      ?                02:30:01  02:30:01  0.01    0.01    920.00
utmp_upd   adm      ?                02:30:01  02:30:01  0.01    0.01   1136.00
utmp_upd   adm      ?                02:30:01  02:30:01  0.01    0.01    576.00
closewtm   adm      ?                02:30:01  02:30:01  0.10    0.01    664.00
```

- Command name (pound (#) sign if the command was executed with superuser
  privileges)
- User name
- `tty` name (listed as ? if unknown)
- Command starting time
- Command ending time
- Real time (in seconds)
- CPU time (in seconds)
- Mean size (in Kbytes)

You can obtain the following information by using acctcom options:

- State of the `fork`/`exec` flag (1 for `fork` without `exec`)
- System exit status
- Hog factor
- Total `kcore` minutes

- CPU factor
- Characters transferred
- Blocks read

  The following table describes the acctcom options.

TABLE 20–6 acctcom Options

Option	Description
-a	Shows average statistics about the processes selected. The statistics are printed after the output is recorded.
-b	Reads the files backward, showing latest commands first. This option has no effect if reading standard input.
-f	Prints the fork/exec flag and system exit status columns. The output is an octal number.
-h	Instead of mean memory size, shows the hog factor, which is the fraction of total available CPU time consumed by the process during its execution. Hog factor = *total_CPU_time/elapsed_time*.
-i	Prints columns containing the I/O counts in the output.
-k	Shows total kcore minutes instead of memory size.
-m	Shows mean core size. This is the default.
-q	Prints average statistics, not output records.
-r	Shows CPU factor: *user_time*/(*system_time + user_time*).
-t	Shows separate system and user CPU times.
-v	Excludes column headings from the output.
-C *sec*	Shows only processes with total CPU time (system plus user) exceeding *sec* seconds.
-e *time*	Shows processes existing at or before *time*, given in the format *hr[:min[:sec]]*.
-E *time*	Shows processes starting at or before *time*, given in the format *hr[:min[:sec]]*. Using the same time for both -S and -E , shows processes that existed at the time.
-g *group*	Shows only processes that belong to *group*.
-H *factor*	Shows only processes that exceed *factor*, where *factor* is the "hog factor" (see the -h option).
-I *chars*	Shows only processes that transferred more characters than the cutoff number specified by *chars*.

TABLE 20–6 `acctcom` Options   *(Continued)*

Option	Description
-l *line*	Show only processes belonging to the terminal /dev/*line*.
-n *pattern*	Shows only commands matching *pattern* (a regular expression except that "+" means one or more occurrences).
-o *ofile*	Instead of printing the records, copies them in `acct.h` format to *ofile*.
-O *sec*	Shows only processes with CPU system time exceeding *sec* seconds.
-s *time*	Show processes existing at or after *time*, given in the format *hr*[:*min*[:*sec*]].
-S *time*	Show processes starting at or after *time*, given in the format *hr*[:*min*[:*sec*]].
-u *user*	Shows only processes that belong to *user*.

System Accounting Files

The /var/adm directory contains the active data collection files. The following table describes the accounting files in this directory.

TABLE 20–7 Files in the /var/adm Directory

File	Description
dtmp	Output from the `acctdusg` program
fee	Output from the `chargefee` program, which are the ASCII `tacct` records
pacct	Active process accounting file
pacct*n*	Process accounting files switched by running the `turnacct` script.
Spacct*n.MMDD*	Process accounting files for *MMDD* during execution of the `runacct` script

The /var/adm/acct directory contains the nite, sum, and fiscal directories, which contain the actual data collection files. For example, the nite directory contains files that are reused daily by the runacct script. A brief summary of the files in the /var/adm/acct/nite directory follows.

TABLE 20–8 Files in the `/var/adm/acct/nite` Directory

File	Description
active	Used by the `runacct` script to record progress and print warning and error messages
active.*MMDD*	Same as the `active` file after the `runacct` script detects an error
cms	ASCII total command summary used by `prdaily`
ctacct.*MMDD*	Connect accounting records in `tacct.h` format
ctmp	Output of `acctcon1` program, connect session records in `ctmp.h` format (`acctcon1` and `acctcon2` are provided for compatibility purposes)
daycms	ASCII daily command summary used by the `prdaily` script.
daytacct	Total accounting records for one day in `tacct.h` format
disktacct	Disk accounting records in `tacct.h` format, created by the `dodisk` script
fd2log	Diagnostic output during execution of the `runacct` script
lastdate	Last day the `runacct` script executed (in `date +%m%d` format)
lock	Used to control serial use of the `runacct` script
lineuse	`tty` line usage report used by the `prdaily` script
log	Diagnostic output from the `acctcon` program
log.*MMDD*	Same as the `log` file after the `runacct` script detects an error
owtmpx	Previous day's `wtmpx` file
reboots	Beginning and ending dates from the `wtmpx` file and a listing of reboots
statefile	Used to record current state during execution of the `runacct` script
tmpwtmp	`wtmpx` file corrected by the `wtmpfix` program
wtmperror	Contains `wtmpfix` error messages
wtmperror.*MMDD*	Same as the `wtmperror` file after the `runacct` script detects an error
wtmp.*MMDD*	The `runacct` script's copy of the `wtmpx` file

The `sum` directory contains the cumulative summary files updated by the `runacct` script and used by the `monacct` script. The following table summarizes the files in the `/var/adm/acct/sum` directory .

TABLE 20–9 Files in the `/var/adm/acct/sum` Directory

File	Description
cms	Total command summary file for current fiscal period in binary format
cmsprev	Command summary file without latest update
daycms	Command summary file for the day's usage in internal summary format
loginlog	Record of last date each user logged on; created by the `lastlogin` script and used in the `prdaily` script
rprt.*MMDD*	Saved output of `prdaily` script
tacct	Cumulative total accounting file for current fiscal period
tacctprev	Same as `tacct` without latest update
tacct.*MMDD*	Total accounting file for *MMDD*

The fiscal directory contains periodic summary files created by the `monacct` script. The following table summarizes the files in the `/var/adm/acct/fiscal` directory.

TABLE 20–10 Files in the `/var/adm/acct/fiscal` Directory

File	Description
cms*n*	Total command summary file for fiscal period *n* in internal summary format
fiscrpt*n*	Report similar to rprt*n* for fiscal period *n*
tacct*n*	Total accounting file for fiscal period *n*

Files Produced by the `runacct` Script

The following table summarizes most useful files produced by the `runacct` script found in the `/var/adm/acct` directory.

TABLE 20–11 Files Produced by `runacct`

File	Description
`nite/lineuse`	The `runacct`script calls the `acctcon` program to gather data on terminal line usage from the `/var/adm/acct/nite/tmpwtmp` file and writes the data to the `/var/adm/acct/nite/lineuse` file. The `prdaily` script uses this data to report line usage. This report is especially useful for detecting bad lines. If the ratio between the number of logouts to logins is greater than about three to one, the line is very likely failing.
`nite/daytacct`	The total accounting file for the day in `tacct.h` format.
`sum/tacct`	Contains the accumulation of each day's `nite/daytacct` data and is used for billing purposes. The `monacct` script restarts accumulating this data each month or fiscal period.
`sum/daycms`	The `runacct` script calls the `acctcms` program to process the commands used during the day to create the Daily Command Summary report and stores the data in the `/var/adm/acct/sum/daycms` file. The ASCII version is the `/var/adm/acct/nite/daycms` file.
`sum/cms`	This file is the accumulation of each day's command summaries. It is restarted by the execution of the `monacct` script. The ASCII version is the `nite/cms` file.
`sum/loginlog`	The `runacct` script calls `lastlogin` script to update the last date logged in for the logins in the `/var/adm/acct/sum/loginlog` file. The `lastlogin` command also removes from this file any logins that are no longer valid.
`sum/rprt.`*MMDD*	Each execution of the `runacct` script saves a copy of the daily report that was printed by the `prdaily` script.

Managing System Performance Topics

This section provides instructions for managing system performance. This section contains these chapters.

Chapter 22	Provides overview information about performance topics.
Chapter 23	Provides step-by-step instructions for using process commands to enhance system performance.
Chapter 24	Provides step-by-step instructions for using `vmstat`, `sar`, and disk utilization commands to monitor performance.

Managing System Performance (Overview)

Getting good performance from a computer or network is an important part of system administration. This chapter is an overview of some of the factors that contribute to maintaining and managing the performance of the computer systems in your care.

This is a list of the overview information in this chapter.

- "Where to Find System Performance Tasks" on page 314
- "System Performance and System Resources" on page 314
- "Processes and System Performance" on page 315
- "About Monitoring Performance" on page 316

What's New in Managing System Performance?

This section describes new features for managing system performance.

DNLC Improvements

The directory name look-up cache (DNLC) is enhanced to provide improved performance when you access files in large directories with 1000 or more files.

The DNLC is a general file-system service that caches the most recently referenced directory names and their associated vnodes. UFS directory entries are stored linearly on disk. This means that locating an entry requires searching each entry for the name.

Adding a new entry requires searching the entire directory to ensure the name does not exist. To solve this performance problem, entire directories are cached in memory by the DNLC.

Another feature in this release is DNLC caching of file objects that have been looked up, but do not exist. This feature is known as negative caching, and is useful because some applications repeatedly test to check if a file exists.

The section that follows describes the new DNLC tunable parameters. These parameters are set optimally and should not be changed casually.

See *Solaris Tunable Parameters Reference Manual* for more information.

The `pargs` and `preap` commands

Two new commands, `pargs` and `preap`, improve process debugging. The `pargs` command can be used to print the arguments and environment variables associated with a live process or core file. The `preap` command can be used to remove zombie processes.

The `pargs` command solves a long-standing problem of not being able to display all the arguments passed to a process with the `ps` command. Use the `pargs` command in combination with the `pgrep` command to display the arguments passed to a process, as follows:

```
# pargs `pgrep ttymon`
579:    /usr/lib/saf/ttymon -g -h -p system-name console login:
-T sun -d /dev/console -l
argv[0]: /usr/lib/saf/ttymon
argv[1]: -g
argv[2]: -h
argv[3]: -p
argv[4]: system-name console login:
argv[5]: -T
argv[6]: sun
argv[7]: -d
argv[8]: /dev/console
argv[9]: -l
argv[10]: console
argv[11]: -m
argv[12]: ldterm,ttcompat
548:    /usr/lib/saf/ttymon
argv[0]: /usr/lib/saf/ttymon
```

Use `pargs -e` to display the environment variables associated with a process, as in the following example:

```
$ pargs -e 6763
6763: tcsh
envp[0]: DISPLAY=:0.0
```

You can use the `pargs` and `preap` commands to examine any process you have privileges to look at. As superuser, you can examine any process.

You can use the `preap` command to clean up a defunct (also called a *zombie*) process. A zombie process is one that has not yet had its exit status reaped (or claimed) by its parent. These processes are generally harmless, but can consume system resources if they are numerous.

See `preap`(1)and `proc`(1) for information on the using the `pargs` command.

New Fair Share (FSS) and Fixed (FX) Schedulers

The FSS and FX schedulers are new user process schedulers. Both of these schedulers use the same range of priorities (0 to 59) as the time-sharing (TS)/interactive (IA) scheduling classes.

Scheduler Interaction

FX is a fixed-priority scheduler that gives processes running in it an assured priority. FX is not meant to be a default scheduler. TS balances loads. FX does not. You can run FX and TS on the same system, but if you set the priority too high for FX, you could starve the processes running in TS.

The fair share scheduler (FSS) guarantees application performance by explicitly allocating shares of CPU resources to projects. The fair share scheduler (FSS) should not be used with TS or FX on the same system unless processor sets are used. If you use processor sets, you can mix TS, IA, and FX with FSS in one system as long as all the processes running on each processor set are in one scheduling class to prevent them from competing for the same CPUs. The FX scheduler, in particular, should not be used in conjunction with the new FSS scheduling class unless processor sets are used, to prevent applications in the FX class from using priorities high enough to starve applications in the FSS class.

See "Fair Share Scheduler" in *System Administration Guide: Resource Management and Network Services* for more information about using these schedulers. See "Projects and Tasks" in *System Administration Guide: Resource Management and Network Services* for information on using projects.

Where to Find System Performance Tasks

Use these references to find step-by-step instructions for monitoring system performance.

- Chapter 23
- Chapter 24

System Performance and System Resources

The performance of a computer system depends upon how the system uses and allocates its resources. It is important to monitor your system's performance on a regularly so that you know how it behaves under normal conditions. You should have a good idea of what to expect, and be able to recognize a problem when it occurs.

System resources that affect performance are described in the following table.

System Resource	Description
Central processing unit (CPU)	The CPU processes instructions, fetching instructions from memory and executing them.
Input/output (I/O) devices	I/O devices transfer information into and out of the computer. Such a device could be a terminal and keyboard, a disk drive, or a printer.
Memory	Physical (or main) memory is the amount of memory (RAM) on the system.

Chapter 24 describes the tools that display statistics about the activity and the performance of the computer system.

Processes and System Performance

Terms related to processes are described in the table below.

TABLE 22–1 Process Terminology

Term	Description
Process	An instance of program in execution.
Lightweight process (LWP)	Is a virtual CPU or execution resource. LWPs are scheduled by the kernel to use available CPU resources based on their scheduling class and priority. LWPs include a kernel thread, which contains information that has to be in memory all the time and an LWP, which contains information that is swappable.
Application thread	A series of instructions with a separate stack that can execute independently in a user's address space. They can be multiplexed on top of LWPs.

A process can consist of multiple LWPs and multiple application threads. The kernel schedules a kernel-thread structure, which is the scheduling entity in the SunOS environment. Various process structures are described in the table below.

TABLE 22–2 Process Structures

Structure	Description
proc	Contains information that pertains to the whole process and has to be in main memory all the time.
kthread	Contains information that pertains to one LWP and has to be in main memory all the time.
user	Contains the per process information that is swappable.
klwp	Contains the per LWP process information that is swappable.

The figure below illustrates the relationship of these structures.

```
                    Main Memory
                   (non-swappable)
                          |
        process           |       kernel thread
     (proc structure)     |     (kthread structure)
                          |
 per process              |                   per LWP
 ─────────────────────────┼─────────────────────────
                          |
          user            |            LWP
      (user structure)    |       (klwp structure)
                          |
                    Swappable
```
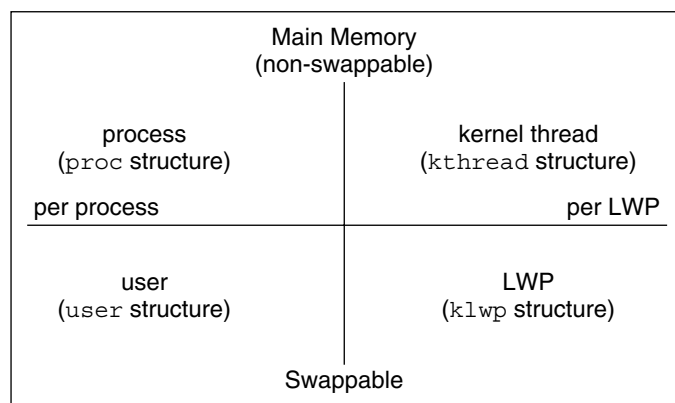
FIGURE 22–1 Process Structures

Most process resources are accessible to all the threads in the process. Almost all process virtual memory is shared. A change in shared data by one thread is available to the other threads in the process.

About Monitoring Performance

While your computer is running, counters in the operating system are incremented to keep track of various system activities. System activities that are tracked are:

- Central processing unit (CPU) utilization
- Buffer usage
- Disk and tape input/output (I/O) activity
- Terminal device activity
- System call activity
- Context switching
- File access
- Queue activity
- Kernel tables
- Interprocess communication
- Paging
- Free memory and swap space
- Kernel Memory Allocation (KMA)

Monitoring Tools

The Solaris software provides several tools to help you keep track of how your system is performing. These include:

TABLE 22–3 Performance Monitoring Tools

The ...	Enable(s) You To ...	For More Information, See ...
`sar` and `sadc` utilities	Collect and report on system activity data	Chapter 24
`ps` and `prstat` commands	Display information about active processes	Chapter 23
`vmstat` and `iostat` commands	Summarize system activity data, such as virtual memory statistics, disk usage, and CPU activity	Chapter 24
`swap` command	Display information about available swap space on your system	"Configuring Additional Swap Space (Tasks)" in *System Administration Guide: Basic Administration*
`netstat` and `nfsstat` commands	Display information about network performance	`netstat`(1M) and `nfsstat`(1M)
Sun Enterprise SyMON	Collect system activity data on Sun's Enterprise™ level systems	*Sun Enterprise SyMON 2.0.1 Software User's Guide*

Managing Processes (Tasks)

This chapter describes the procedures for managing system processes. This is a list of the step-by-step instructions in this chapter.

Commands for Managing Processes (Overview)

The table below describes commands for managing processes.

TABLE 23–1 Commands for Managing Processes

Command and	Man Page	Description
ps pgrep prstat	ps(1), pgrep(1), and prstat(1M)	Check the status of active processes on a system, as well as display detailed information about the processes

TABLE 23–1 Commands for Managing Processes   *(Continued)*

Command and	Man Page	Description
dispadmin	dispadmin(1M)	List default scheduling policies
priocntl	priocntl(1)	Assign processes to a priority class and manage process priorities
nice	nice(1)	Change the priority of a timesharing process
psrset	psrset(1M)	Bind specific process groups to a group of processors rather than to just a single processor

The Solaris Management Console's Processes tool enables you to manage processes with a user-friendly interface. See "Working With the Solaris Management Console (Tasks)" in *System Administration Guide: Basic Administration* for information on using and starting the Solaris Management Console.

The ps Command

The ps command enables you to check the status of active processes on a system, as well as display technical information about the processes. This data is useful for such administrative tasks as determining how to set process priorities.

Depending on which options you use, ps reports the following information:

- Current status of the process
- Process ID
- Parent process ID
- User ID
- Scheduling class
- Priority
- Address of the process
- Memory used
- CPU time used

The table below describes some of the fields reported by the ps command. The fields displayed depend on which option you choose. See ps(1) for a description of all available options.

TABLE 23–2 Summary of Fields in ps Reports

Field	Description
UID	The effective user ID of the process's owner.

TABLE 23–2 Summary of Fields in `ps` Reports   *(Continued)*

Field	Description
PID	The process ID.
PPID	The parent process's ID.
C	The processor utilization for scheduling. This field is not displayed when the `-c` option is used.
CLS	The scheduling class to which the process belongs: real-time, system, or timesharing. This field is included only with the `-c` option.
PRI	The kernel thread's scheduling priority. Higher numbers mean higher priority.
NI	The process's `nice` number, which contributes to its scheduling priority. Making a process "nicer" means lowering its priority.
ADDR	The address of the `proc` structure.
SZ	The virtual address size of the process.
WCHAN	The address of an event or lock for which the process is sleeping.
STIME	The starting time of the process (in hours, minutes, and seconds).
TTY	The terminal from which the process (or its parent) was started. A question mark indicates there is no controlling terminal.
TIME	The total amount of CPU time used by the process since it began.
CMD	The command that generated the process.

▼ How to List Processes (ps)

You can use the `ps` command to list all the processes on a system.

```
$ ps [-ef]
```

ps	Displays only the processes associated with your login session.
-ef	Displays full information about all the processes being executed on the system.

Example—Listing Processes

The following example shows output from the ps command when no options are used.

```
$ ps
  PID TTY       TIME COMD
 1664 pts/4    0:06 csh
 2081 pts/4    0:00 ps
```

The following example shows output from ps -ef. This shows that the first process executed when the system boots is sched (the swapper) followed by the init process, pageout, and so on.

```
$ ps -ef
    UID    PID  PPID  C     STIME TTY       TIME CMD
    root     0     0  0  May 05 ?         0:04 sched
    root     1     0  0  May 05 ?        10:48 /etc/init -
    root     2     0  0  May 05 ?         0:00 pageout
    root     3     0  0  May 05 ?        43:21 fsflush
    root   238     1  0  May 05 ?         0:00 /usr/lib/saf/sac -t 300
    root   115     1  0  May 05 ?         0:10 /usr/sbin/rpcbind
    root   158     1  0  May 05 ?         0:00 /usr/lib/autofs/autom...
    root   134     1  0  May 05 ?         0:12 /usr/sbin/inetd -s
    root   107     1  0  May 05 ?        11:49 /usr/sbin/in.routed -q
    root   117     1  5  May 05 ?       899:32 /usr/sbin/keyserv
    root   125     1  0  May 05 ?         0:00 /usr/sbin/kerbd
    root   123     1  0  May 05 ?         4:17 /usr/sbin/nis_cachemgr
    daemon 137     1  0  May 05 ?         0:00 /usr/lib/nfs/statd
    root   139     1  0  May 05 ?         0:02 /usr/lib/nfs/lockd
    root   159     1 50  May 05 ?      8243:36 /usr/sbin/automount
    root   162     1  0  May 05 ?         0:07 /usr/sbin/syslogd
    root   181     1  0  May 05 ?         0:03 /usr/sbin/nscd...
    root   169     1  0  May 05 ?         5:09 /usr/sbin/cron
    root   207     1  0  May 05 ?         0:00 /usr/lib/lpsched
    root   210     1  0  May 05 ?         0:01 /usr/sbin/vold
    root   200     1  0  May 05 ?         0:08 /usr/lib/sendmail -bd -q1h
    root  4942     1  0  May 17 console 0:00 /usr/lib/saf/ttymon...
    root   208     1  0  May 05 ?         0:00 /usr/lib/utmpd
    root   241   238  0  May 05 ?         0:00 /usr/lib/saf/ttymon
    root  5748   134  0 17:09:49 ?        0:01 in.rlogind
    root  5750  5748  0 17:09:52 pts/0    0:00 -sh
    root  5770  5750  2 17:23:39 pts/0    0:00 ps -ef
```

The /proc File System and Commands

In addition, process tools are available in the /usr/proc/bin directory that display highly detailed information about the processes listed in the /proc directory, also known as the process file system (PROCFS). Images of active processes are stored here by their process ID number.

The process tools are similar to some options of the ps command, except that the output provided by the tools is more detailed. In general, the process tools:

- Display more details about processes, such as fstat and fcntl information, working directories, and trees of parent and child processes
- Provide control over processes, allowing users to stop or resume them

Displaying Information About Processes (/proc Tools)

You can display detailed, technical information about active processes by using some of the process tool commands contained in /usr/proc/bin. The table below lists these process tools. Refer to proc(1) for more information.

TABLE 23–3 /usr/proc/bin Process Tools That Display Information

Process Tool	What It Displays
pcred	Credentials
pfiles	fstat and fcntl information for open files in a process
pflags	/proc tracing flags, pending and held signals, and other status information
pldd	Dynamic libraries linked into a process
pmap	Address space map
psig	Signal actions
pstack	Hex+symbolic stack trace
ptime	Process time using microstate accounting
ptree	Process trees that contain the process
pwait	Status information after a process terminates
pwdx	Current working directory for a process

> **Note –** To avoid typing long command names, add the process tool directory to your PATH variable. This enables you to run process tools by entering only the last part of each file name (for example, pwdx instead of /usr/proc/bin/pwdx).

▼ How to Display Information About Processes

1. **(Optional) Use output from the** pgrep **command to obtain the identification number of the process you want to display more information about.**

   # **pgrep** *process*

   process   Name of the process you want to display more information about.

   The process identification number is in the first column of the output.

2. **Use the appropriate** /usr/proc/bin **command to display the information you need.**

   # **/usr/proc/bin/***pcommand* *pid*

   pcommand   Process tool command you want to run. Table 23–3 lists these commands.

   pid   Identification number of a process.

Examples—Displaying Information About Processes

The following example shows how to use process tool commands to display more information about an lpsched process. First, the /usr/proc/bin path is defined to avoid typing long process tool commands. Next, the identification number for lpsched is obtained. Finally, output from three process tool commands is shown.

```
# PATH=$PATH:/usr/proc/bin
# export PATH ¹
# ps -e | grep lpsched ²
207 ?        0:00 /usr/lib/lpsched
# pwdx 207 ³
207:    /
# ptree 207 ⁴
207   /usr/lib/lpsched
# pfiles 207 ⁵
207:    /usr/lib/lpsched
  Current rlimit: 4096 file descriptors
   0: S_IFIFO mode:0000 dev:179,0 ino:70 uid:0 gid:0 size:0
```

```
        O_RDWR
  1: S_IFIFO mode:0000 dev:179,0 ino:70 uid:0 gid:0 size:0
        O_RDWR
  3: S_IFCHR mode:0666 dev:32,8 ino:11446 uid:0 gid:3 rdev:21,0
        O_WRONLY FD_CLOEXEC
  4: S_IFDOOR mode:0444 dev:183,0 ino:59515 uid:0 gid:0 size:0
        O_RDONLY|O_LARGEFILE FD_CLOEXEC  door to nscd[201]
  5: S_IFREG mode:0664 dev:32,9 ino:1330 uid:71 gid:8 size:0
        O_WRONLY
```

1. Adds the /usr/proc/bin directory to the PATH variable.
2. Obtains the process identification number for lpsched.
3. Displays the current working directory for lpsched.
4. Displays the process tree containing lpsched.
5. Displays fstat and fcntl information.

The following example shows output from the pwait command, which waits until a process terminates, then displays information about what happened. The following example shows output from the pwait command after a Command Tool window was exited.

```
$ ps -e | grep cmdtool
  273 console 0:01 cmdtool
  277 console 0:01 cmdtool
  281 console 0:01 cmdtool
$ pwait -v 281
281: terminated, wait status 0x0000
```

Controlling Processes (/proc Tools)

You can control some aspects of processes by using some of the process tools contained in /usr/proc/bin. The table below lists these process tools. Refer to proc(1) for detailed information about process tools.

TABLE 23–4 Process Tools

Tools That Control Processes	What the Tools Do
/usr/proc/bin/pstop *pid*	Stops the process
/usr/proc/bin/prun *pid*	Restarts the process
/usr/proc/bin/ptime *pid*	Times the process using microstate accounting
/usr/proc/bin/pwait [-v] *pid*	Waits for specified processes to terminate

TABLE 23–4 Process Tools   *(Continued)*

Tools That Control Processes	What the Tools Do
Tools That Display Process Details	What the Tools Display
/usr/proc/bin/pcred *pid*	Credentials
/usr/proc/bin/pfiles *pid*	fstat and fcntl information for open files
/usr/proc/bin/pflags *pid*	/proc tracing flags, pending and held signals, and other status information for each lwp
/usr/proc/bin/pldd *pid*	Dynamic libraries linked into each process
/usr/proc/bin/pmap *pid*	Address space map
/usr/proc/bin/psig *pid*	Signal actions
/usr/proc/bin/pstack *pid*	Hex+symbolic stack trace for each lwp
/usr/proc/bin/ptree *pid*	Process trees containing specified pids
/usr/proc/bin/pwdx *pid*	Current working directory

In these commands, *pid* is a process identification number. You can obtain this number by using the ps -ef command.

Chapter 23 describes how to use the process tool commands to perform selected system administration tasks, such as displaying details about processes, and starting and stopping them. A more detailed description of the process tools can be found in proc(1).

If a process becomes trapped in an endless loop, or if it takes too long to execute, you may want to stop (kill) the process. See Chapter 23 for more information about stopping processes using the pkill command.

The previous flat /proc file system has been restructured into a directory hierarchy that contains additional sub-directories for state information and control functions.

It also provides a watchpoint facility that is used to remap read/write permissions on the individual pages of a process's address space. This facility has no restrictions and is MT-safe.

The new /proc file structure provides complete binary compatibility with the old /proc interface except that the new watchpoint facility cannot be used with the old interface.

Debugging tools have been modified to use /proc's new watchpoint facility, which means the entire watchpoint process is faster.

The following restrictions have been removed when setting watchpoints using the dbx debugging tool:

- Setting watchpoints on local variables on the stack due to SPARC register windows
- Setting watchpoints on multi-threaded processes

See proc(4), core(4), and mdb(1) for more information.

Note – To avoid typing long command names, add the process tool directory to your PATH variable. This allows you to run process tools by entering only the last part of each file name (for example, prun instead of /usr/proc/bin/prun).

▼ How to Control Processes

1. **(Optional) Use output from the ps command to obtain the identification number of the process you want to display more information about.**

   # **pgrep** *process*

process	Name of the process you want to display more information about.

   The process identification number is in the first column of the output.

2. **Use the appropriate /usr/proc/bin command to control the process.**

   # **/usr/proc/bin/***pcommand* *PID*

pcommand	Process tool command you want to run. Table 23–4 lists these commands.
PID	Identification number of a process.

3. **Verify the process status using the ps command.**

   # **pgrep** *PID*

Example—Controlling Processes

The following example shows how to use process tools to stop and restart Print Tool.

```
# PATH=$PATH:/usr/proc/bin
# export PATH 1
# ps -e | grep 'print*' 2
264 console 0:03 printtool
# pstop 264 3
# prun 264 4
# ps | grep 264
264 console 0:03 printtool
#
```

1. Adds the /usr/proc/bin directory to the PATH variable.
2. Obtains the process identification number for Print Tool.
3. Stops the Print Tool process.
4. Restarts the Print Tool process.

Killing a Process (`pkill`)

Sometimes it is necessary to stop (kill) a process. The process may be in an endless loop, or you may have started a large job that you want to stop before it is completed. You can kill any process that you own, and superuser can kill any processes in the system except for those with process IDs 0, 1, 2, 3, and 4.

Refer to pgrep(1)for more detailed information.

▼ How to Kill a Process

1. **(Optional) To kill a process belonging to another user, become superuser.**

2. **(Optional) Use output from the** pgrep **command to obtain the identification number of the process you want to display more information about.**

   $ **pgrep** *process*

   | *process* | Name of the process you want to display more information about. |

   The process identification number is in the first column of the output.

3. **Use the** `pkill` **command to stop the process.**

   $ **`pkill`** [**`-9`**] *PID* ...

`-9`	Ensures that the process terminates promptly.
PID ...	ID of the process or processes to stop.

4. **Use the** `pgrep` **command to verify that the process has been stopped.**

   $ **`pgrep`** *PID* ...

Managing Process Class Information

The listing below shows which classes are configured on your system, and the user priority range for the timesharing class. The possible classes are:

- Fair–share (`FSS`)
- Fixed (`FX`)
- System (`SYS`)
- Interactive (`IA`)
- Real-time (`RT`)
- Timesharing (`TS`)
  - The user-supplied priority ranges from -20 to +20.
  - The priority of a process is inherited from the parent process. This is referred to as the *user-mode* priority.
  - The system looks up the user-mode priority in the timesharing dispatch parameter table and adds in any `nice` or `priocntl` (user-supplied) priority and ensures a 0-59 range to create a *global* priority.

Changing the Scheduling Priority of Processes With `priocntl`

The scheduling priority of a process is the priority it is assigned by the process scheduler, according to scheduling policies. The `dispadmin` command lists the default scheduling policies.

The priocntl(1) command can be used to assign processes to a priority class and to manage process priorities. See the section called "How to Designate a Process Priority" on page 331 for instructions on using the priocntl command to manage processes.

▼ How to Display Basic Information About Process Classes

You can display process class and scheduling parameters with the priocntl -l command.

```
$ priocntl -l
```

Example—Getting Basic Information About Process Classes

The following example shows output from the priocntl -l command.

```
# priocntl -l
CONFIGURED CLASSES
==================

SYS (System Class)

TS (Time Sharing)
        Configured TS User Priority Range: -60 through 60

IA (Interactive)
        Configured IA User Priority Range: -60 through 60

RT (Real Time)
        Maximum Configured RT Priority: 59
```

▼ How to Display the Global Priority of a Process

You can display the global priority of a process by using the ps command.

```
$ ps -ecl
```

The global priority is listed under the PRI column.

Example—Displaying the Global Priority of a Process

The following example shows output from ps -ecl. Data in the PRI column show that pageout has the highest priority, while sh has the lowest.

```
$ ps -ecl
 F S UID PID  PPID CLS PRI  ADDR      SZ  WCHAN    TTY      TIME   COMD
19 T 0   0    0    SYS 96   f00d05a8  0            ?        0:03   sched
 8 S 0   1    0    TS  50   ff0f4678 185  ff0f4848 ?        36:51  init
19 S 0   2    0    SYS 98   ff0f4018  0   f00c645c ?        0:01   pageout
19 S 0   3    0    SYS 60   ff0f5998  0   f00d0c68 ?        241:01 fsflush
 8 S 0   269  1    TS  58   ff0f5338 303  ff49837e ?        0:07   sac
 8 S 0   204  1    TS  43   ff2f6008  50  ff2f606e console  0:02   sh
```

▼ How to Designate a Process Priority

1. **Become superuser.**

2. **Start a process with a designated priority.**

   # **priocntl -e -c** *class* **-m** *userlimit* **-p** *pri command_name*

-e	Executes the command.
-c *class*	Specifies the class within which to run the process. The default classes are TS (timesharing) or RT (real-time).
-m *userlimit*	Specifies the maximum amount you can raise or lower your priority, when using the -p option.
-p *pri command_name*	Lets you specify the relative priority in the RT class, for a real-time thread. For a timesharing process, the -p option lets you specify the user-supplied priority which ranges from -20 to +20.

3. **Verify the process status by using the** ps -ecl **command.**

   # **ps -ecl | grep** *command_name*

Example—Designating a Priority

The following example starts the find command with the highest possible user-supplied priority.

```
# priocntl -e -c TS -m 20 -p 20 find . -name core -print
# ps -ecl | grep find
```

▼ How to Change Scheduling Parameters of a Timeshare Process

1. **Become superuser.**

2. **Change the scheduling parameter of a running timeshare process.**

   # **priocntl -s -m** *userlimit* [-**p** *userpriority*] **-i** *idtype idlist*

-s	Lets you set the upper limit on the user priority range and change the current priority.
-m *userlimit*	Specifies the maximum amount you can raise or lower your priority, when using the -p option.
-p *userpriority*	Allows you to designate a priority.
-i*idtype idlist*	Uses a combination of *idtype* and *idlist* to identify the process. The *idtype* specifies the type of ID, such as PID or UID.

3. **Verify the process status by using the** ps -ecl **command.**

   # **ps -ecl | grep** idlist

Example—Changing Scheduling Parameters of a Timeshare Process

The following example executes a command with a 500-millisecond time slice, a priority of 20 in the RT class, and a global priority of 120.

```
# priocntl -e -c RT -t 500 -p 20 myprog
# ps -ecl | grep myprog
```

▼ How to Change the Class of a Process

1. **(Optional) Become superuser.**

Note – You must be superuser or working in a real-time shell to change processes from, or to, real-time processes.

---

2. **Change the class of a process.**

   # **priocntl -s -c** *class* **-i** *idtype idlist*

-s	Lets you set the upper limit on the user priority range and change the current priority.
-c *class*	Specifies the class, TS or RT, to which you are changing the process.
-i *idtype idlist*	Uses a combination of *idtype* and *idlist* to identify the process. The *idtype* specifies the type of ID, such as PID or UID.

3. **Verify the process status by using the** ps -ecl **command.**

   # **ps -ecl | grep** *idlist*

Example—Changing the Class of a Process

The following example changes all the processes belonging to user 15249 to real-time processes.

```
# priocntl -s -c RT -i uid 15249
# ps -ecl | grep 15249
```

Note – If, as superuser, you change a user process to the real-time class, the user cannot subsequently change the real-time scheduling parameters (using priocntl -s).

Changing the Priority of a Timesharing Process With nice

The nice(1) command is only supported for backward compatibility to previous Solaris releases. The priocntl command provides more flexibility in managing processes.

The priority of a process is determined by the policies of its scheduling class, and by its *nice number*. Each timesharing process has a global priority which is calculated by adding the user-supplied priority, which can be influenced by the `nice` or `priocntl` commands, and the system-calculated priority.

The execution priority number of a process is assigned by the operating system, and is determined by several factors, including its schedule class, how much CPU time it has used, and (in the case of a timesharing process) its `nice` number.

Each timesharing process starts with a default `nice` number, which it inherits from its parent process. The `nice` number is shown in the `NI` column of the `ps` report.

A user can lower the priority of a process by increasing its user-supplied priority. But only the superuser can lower a `nice` number to increase the priority of a process. This is to prevent users from increasing the priorities of their own processes, thereby monopolizing a greater share of the CPU.

Nice numbers range between 0 and +40, with 0 representing the highest priority. The default value is 20. Two versions of the command are available, the standard version, `/usr/bin/nice`, and a version that is part of the C shell.

▼ How to Change the Priority of a Process

You can raise or lower the priority of a command or a process by changing the `nice` number. To lower the priority of a process:

`/usr/bin/nice` *command_name*	Increase the `nice` number by four units (the default)
`/usr/bin/nice +4` *command_name*	Increase the `nice` number by four units
`/usr/bin/nice -10` *command_name*	Increase the `nice` number by ten units

The first and second commands increase the `nice` number by four units (the default); and the third command increases the `nice` by ten units, lowering the priority of the process.

The following commands raise the priority of the command by lowering the `nice` number.

To raise the priority of a process:

`/usr/bin/nice -10` *command_name*	Raises the priority of the command by lowering the `nice` number

`/usr/bin/nice - -10 `*command_name*	Raises the priority of the command by lowering the `nice` number. The first minus sign is the option sign, and the second minus sign indicates a negative number.

The above commands raise the priority of the command, *command_name*, by lowering the `nice` number. Note that in the second case, the two minus signs are required.

Process Troubleshooting

Here are some tips on obvious problems you may find:

- Look for several identical jobs owned by the same user. This may come as a result of running a script that starts a lot of background jobs without waiting for any of the jobs to finish.
- Look for a process that has accumulated a large amount of CPU time. You'll see this by looking at the `TIME` field. Possibly, the process is in an endless loop.
- Look for a process running with a priority that is too high. Type `ps -c` to see the `CLS` field, which displays the scheduler class of each process. A process executing as a real-time (`RT`) process can monopolize the CPU. Or look for a timeshare (`TS`) process with a high `nice` value. A user with superuser privileges may have bumped up the priorities of this process. The system administrator can lower the priority by using the `nice` command.
- Look for a runaway process—one that progressively uses more and more CPU time. You can see it happening by looking at the time when the process started (`STIME`) and by watching the cumulation of CPU time (`TIME`) for awhile.

Monitoring System Performance (Tasks)

This chapter describes procedures for monitoring system performance by using the `vmstat`, `iostat`, `df`, and `sar` commands. This is a list of the step-by-step instructions in this chapter.

What's New in Monitoring System Performance?

The `vmstat` command interrupt reporting now includes all system interrupts in the faults-in column.

In previous Solaris releases, this column did not report clock device interrupts.

See the `vmstat`(1M) man page for more information.

Displaying Virtual Memory Statistics (`vmstat`)

You can use the `vmstat` command to report virtual memory statistics and such information about system events as CPU load, paging, number of context switches, device interrupts, and system calls. The `vmstat` command can also display statistics on swapping, cache flushing, and interrupts.

Refer to `vmstat`(1M) for a more detailed description of this command.

▼ How to Display Virtual Memory Statistics (`vmstat`)

Collect virtual memory statistics using the `vmstat` command with a time interval.

```
$ vmstat n
```

n	Interval in seconds between reports.

The table below describes the fields in the `vmstat` output.

TABLE 24–1 Output From the `vmstat` Command

Category	Field Name	Description
procs		Reports the following states:
	r	The number of kernel threads in the dispatch queue
	b	Blocked kernel threads waiting for resources
	w	Swapped out LWPs waiting for processing resources to finish
memory		Reports on usage of real and virtual memory:
	swap	Available swap space
	free	Size of the free list
page		Reports on page faults and paging activity, in units per second:
	re	Pages reclaimed
	mf	Minor and major faults
	pi	Kbytes paged in
	po	Kbytes paged out
	fr	Kbytes freed
	de	Anticipated memory needed by recently swapped-in processes
	sr	Pages scanned by page daemon (not currently in use). If `sr` does not equal zero, the page daemon has been running.
disk		Reports the number of disk operations per second, showing data on up to four disks
faults		Reports the trap/interrupt rates (per second):
	in	Interrupts per second
	sy	System calls per second
	cs	CPU context switch rate
cpu		Reports on the use of CPU time:
	us	User time
	sy	System time
	id	Idle time

Example—Displaying Virtual Memory Statistics

The following example shows the vmstat display of statistics gathered at five-second intervals.

```
$ vmstat 5
 procs     memory            page            disk          faults      cpu
 r b w   swap free re  mf pi po  fr de sr f0 s3 -- --  in  sy  cs us sy id
 0 0 8 28312  668  0   9  2  0   1  0  0  0  1  0  0  10  61  82  1  2 97
 0 0 3 31940  248  0  10 20  0  26  0 27  0  4  0  0  53 189 191  6  6 88
 0 0 3 32080  288  3  19 49  6  26  0 15  0  9  0  0  75 415 277  6 15 79
 0 0 3 32080  256  0  26 20  6  21  0 12  1  6  0  0 163 110 138  1  3 96
 0 1 3 32060  256  3  45 52 28  61  0 27  5 12  0  0 195 191 223  7 11 82
 0 0 3 32056  260  0   1  0  0   0  0  0  0  0  0  0   4  52  84  0  1 99
```

▼ How to Display System Event Information (vmstat -s)

Run vmstat -s to show the total of various system events that have taken place since the system was last booted.

```
$ vmstat -s
        0 swap ins
        0 swap outs
        0 pages swapped in
        0 pages swapped out
   392182 total address trans. faults taken
    20419 page ins
      923 page outs
    30072 pages paged in
     9194 pages paged out
    65167 total reclaims
    65157 reclaims from free list
        0 micro (hat) faults
   392182 minor (as) faults
    19383 major faults
    85775 copy-on-write faults
    66637 zero fill page faults
    46309 pages examined by the clock daemon
        6 revolutions of the clock hand
    15578 pages freed by the clock daemon
     4398 forks
      352 vforks
     4267 execs
 12926285 cpu context switches
109029866 device interrupts
   499296 traps
 22461261 system calls
   778068 total name lookups (cache hits 97%)
    18739 user    cpu
```

```
        34662 system cpu
     52051435 idle   cpu
        25252 wait   cpu
```

▼ How to Display Swapping Statistics (`vmstat -S`)

Run `vmstat -S` to show swapping statistics.

```
$ vmstat -S
 procs     memory            page            disk          faults      cpu
 r b w   swap  free  si  so pi po fr de sr f0 s0 s6 --   in   sy   cs us sy id
 0 0 0 200968 17936   0   0  0  0  0  0  0  0  0  0  0  109   43   24  0  0 100
```

The fields are described in the table below.

TABLE 24–2 Output From the `vmstat -S` Command

Field Name	Description
`si`	Average number of LWPs swapped in per second
`so`	Number of whole processes swapped out

Note – The `vmstat` command truncates the output of both of these fields. Use the `sar` command to display a more accurate accounting of swap statistics.

▼ How to Display Cache Flushing Statistics (`vmstat -c`)

Run `vmstat -c` to show cache flushing statistics for a virtual cache.

```
$ vmstat -c
usr    ctx    rgn    seg     pag     par
  0  60714      5 134584 4486560 4718054
```

It shows the total number of cache flushes since the last boot. The cache types are described in the table below.

TABLE 24–3 Output From the `vmstat -c` Command

Cache Name	Cache Type
`usr`	User
`ctx`	Context

TABLE 24–3 Output From the `vmstat -c` Command   *(Continued)*

Cache Name	Cache Type
rgn	Region
seg	Segment
pag	Page
par	Partial-page

▼ How to Display Interrupts Per Device (`vmstat -i`)

Run `vmstat -i` to show interrupts per device.

```
$ vmstat -i
```

Example—Displaying Interrupts Per Device

The following example shows output from the `vmstat -i` command.

```
$ vmstat -i
interrupt          total     rate
-------------------------------
clock          52163269      100
esp0            2600077        4
zsc0              25341        0
zsc1              48917        0
cgsixc0             459        0
lec0             400882        0
fdc0                 14        0
bppc0                 0        0
audiocs0              0        0
-------------------------------
Total          55238959      105
```

Displaying Disk Utilization Information (`iostat` *n*)

Use the `iostat` command to report statistics about disk input and output, and produces measures of throughput, utilization, queue lengths, transaction rates, and service time. For a detailed description of this command, refer to `iostat`(1M).

▼ How to Display Disk Utilization Information (`iostat`)

You can display disk activity information by using the `iostat` command with a time interval.

```
$ iostat 5
     tty          fd0           sd3          nfs1          nfs31          cpu
tin tout kps tps serv  kps tps serv  kps tps serv  kps tps serv  us sy wt id
  0    1   0   0 410    3   0  29    0   0   9    3   0  47   4  2  0 94
```

The first line of output shows the statistics since the last boot. Each subsequent line shows the interval statistics. The default is to show statistics for the terminal (`tty`), disks (`fd` and `sd`), and CPU (`cpu`).

The table below describes the fields in the `iostat` command output.

TABLE 24–4 Output From the `iostat` *n* Command

For Each ...	Field Name	Description
Terminal		
	`tin`	Number of characters in the terminal input queue
	`tout`	Number of characters in the terminal output queue
Disk		
	`bps`	Blocks per second
	`tps`	Transactions per second
	`serv`	Average service time, in milliseconds
CPU		
	`us`	In user mode

For Each ...	Field Name	Description
	sy	In system mode
	wt	Waiting for I/O
	id	Idle

Example—Displaying Disk Utilization Information

The following example shows disk statistics gathered every five seconds.

```
$ iostat 5
tty          sd0          sd6          nfs1         nfs49         cpu
tin tout kps tps serv  kps tps serv  kps tps serv  kps tps serv  us sy wt id
  0    0   1   0   49    0   0   0     0   0   0     0   0  15    0  0  0 100
  0   47   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
  0   16   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
  0   16   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
  0   16  44   6  132    0   0   0     0   0   0     0   0   0     0  0  1  99
  0   16   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
  0   16   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
  0   16   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
  0   16   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
  0   16   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
  0   16   3   1   23    0   0   0     0   0   0     0   0   0     0  0  1  99
  0   16   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
  0   16   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
  0   16   0   0    0    0   0   0     0   0   0     0   0   0     0  0  0 100
```

▼ How to Display Extended Disk Statistics (`iostat -xtc`)

Run `iostat -xtc` to get extended disk statistics.

```
$ iostat -xtc
             extended device statistics               tty        cpu
device     r/s    w/s    kr/s   kw/s wait actv  svc_t  %w  %b  tin tout  us sy wt id
fd0        0.0    0.0    0.0    0.0  0.0  0.0    0.0   0   0    0    0    0  0  0 100
sd0        0.0    0.0    0.4    0.4  0.0  0.0   49.5   0   0
sd6        0.0    0.0    0.0    0.0  0.0  0.0    0.0   0   0
nfs1       0.0    0.0    0.0    0.0  0.0  0.0    0.0   0   0
nfs49      0.0    0.0    0.0    0.0  0.0  0.0   15.1   0   0
nfs53      0.0    0.0    0.4    0.0  0.0  0.0   24.5   0   0
nfs54      0.0    0.0    0.0    0.0  0.0  0.0    6.3   0   0
nfs55      0.0    0.0    0.0    0.0  0.0  0.0    4.9   0   0
```

This command displays a line of output for each disk. The output fields are described in the table below.

TABLE 24–5 Output From the `iostat -xtc` Command

Field Name	Description
r/s	Reads per second
w/s	Writes per second
Kr/s	Kbytes read per second
Kw/s	Kbytes written per second
wait	Average number of transactions waiting for service (queue length)
actv	Average number of transactions actively being serviced
svc_t	Average service time, in milliseconds
%w	Percentage of time the queue is not empty
%b	Percentage of time the disk is busy

Displaying Disk Usage Statistics (`df`)

Use the `df` command to show the amount of free disk space on each mounted disk. The *usable* disk space reported by `df` reflects only 90% of full capacity, as the reporting statistics leave a 10% head room above the total available space. This head room normally stays empty for better performance.

The percentage of disk space actually reported by `df` is used space divided by usable space.

If the file system is above 90% capacity, transfer files to a disk that is not as full by using `cp`, or to a tape by using `tar` or `cpio`; or remove the files.

For a detailed description of this command, refer to the `df`(1M) man page.

▼ How to Display File System Information (`df`)

Use the `df -k` command to display file system information in Kbytes.

```
$ df -k
Filesystem            kbytes    used   avail capacity  Mounted on
/dev/dsk/c0t3d0s0     192807   40231  133296    24%    /
```

The table below describes the df -k command output.

TABLE 24–6 Output From the df -k Command

Field Name	Description
kbytes	Total size of usable space in the file system
used	Amount of space used
avail	Amount of space available for use
capacity	Amount of space used, as a percent of the total capacity
mounted on	Mount point

Example—Displaying File System Information

The following example shows output of the df -k command.

```
$ df -k
/dev/dsk/c0t0d0s0     192807   49043  124484    29%    /
/dev/dsk/c0t0d0s6    1190551  680444  450580    61%    /usr
/proc                      0       0       0     0%    /proc
fd                         0       0       0     0%    /dev/fd
mnttab                     0       0       0     0%    /etc/mnttab
swap                  198056       0  198056     0%    /var/run
swap                  198064       8  198056     1%    /tmp
/dev/dsk/c0t0d0s5     192807    2031  171496     2%    /opt
/dev/dsk/c0t0d0s7     217191       9  195463     1%    /export/home
venus:/usr/dist     20612581 13237316 6963015   66%    /usr/dist
```

Monitoring System Activities (sar)

Use the sar command to:

- Organize and view data about system activity
- Access system activity data on a special request basis
- Generate automatic reports to measure and monitor system performance, and special request reports to pinpoint specific performance problems. "Collecting System Activity Data Automatically (sar)" on page 363 describes these tools.

For a detailed description of this command, refer to sar(1).

▼ How to Check File Access (`sar -a`)

Display file access operation statistics with the sar -a command.

```
$ sar -a
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00  iget/s namei/s dirbk/s
01:00:00       0       0       0
02:00:02       0       0       0
03:00:00       0       1       0
04:00:00       0       0       0
05:00:01       0       0       0
06:00:00       0       0       0

Average        0       1       0
```

The operating system routines reported are described in the following table.

TABLE 24–7 Output from the sar -a Command

Field Name	Description
iget/s	The number of requests made for inodes that were not in the directory name lookup cache (dnlc).
namei/s	This is the number of file system path searches per second. If namei does not find a directory name in the dnlc, it calls iget to get the inode for either a file or directory. Hence, most igets are the result of dnlc misses.
dirbk/s	This is the number of directory block reads issued per second.

The larger the values reported, the more time the kernel is spending to access user files. The amount of time reflects how heavily programs and applications are using the file systems. The -a option is helpful for viewing how disk-dependent an application is.

▼ How to Check Buffer Activity (`sar -b`)

Display buffer activity statistics with the sar -b command.

The buffer is used to cache metadata, which includes inodes, cylinder group blocks, and indirect blocks.

```
$ sar -b
00:00:00 bread/s lread/s %rcache bwrit/s lwrit/s %wcache pread/s pwrit/s
01:00:00       0       0     100       0       0      55       0       0
```

The buffer activities displayed by the -b option are described in the table below. The most important entries are the cache hit ratios %rcache and %wcache, which measure the effectiveness of system buffering. If %rcache falls below 90, or if %wcache falls below 65, it may be possible to improve performance by increasing the buffer space.

TABLE 24–8 Output from the sar -b Command

Field Name	Description
bread/s	Average number of reads per second submitted to the buffer cache from the disk
lread/s	Average number of logical reads per second from the buffer cache
%rcache	Fraction of logical reads found in the buffer cache (100% minus the ratio of bread/s to lread/s)
bwrit/s	Average number of physical blocks (512 blocks) written from the buffer cache to disk, per second
lwrite/s	Average number of logical writes to the buffer cache, per second
%wcache	Fraction of logical writes found in the buffer cache(100% minus the ratio of bwrit/s to lwrit/s)
pread/s	Average number of physical reads, per second, using character device interfaces
pwrit/s	Average number of physical write requests, per second, using character device interfaces

Example—Checking Buffer Activity

The following abbreviated example of sar -b output shows that the %rcache and %wcache buffers are not causing any slowdowns, because all the data is within acceptable limits.

```
$ sar -b
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 bread/s lread/s %rcache bwrit/s lwrit/s %wcache pread/s pwrit/s
01:00:00       0       0     100       0       0      55       0       0
02:00:02       0       0     100       0       0      55       0       0
03:00:00       0       0     100       0       0      72       0       0
04:00:00       0       0     100       0       0      56       0       0
05:00:01       0       0     100       0       0      55       0       0
06:00:00       0       0     100       0       0      55       0       0
```

```
Average           0       0      94       0       0      64       0       0
```

▼ How to Check System Call Statistics (`sar -c`)

Display system call statistics by using the `sar -c` command.

```
$ sar -c
00:00:00 scall/s sread/s swrit/s  fork/s  exec/s rchar/s wchar/s
01:00:00      38       2       2    0.00    0.00     149     120
```

The table below describes the following system call categories reported by the `-c` option. Typically, `reads` and `writes` account for about half of the total system calls, although the percentage varies greatly with the activities that are being performed by the system.

TABLE 24–9 Output from the `sar -c` Command

Field Name	Description
`scall/s`	All types of system calls per second (generally about 30 per second on a busy four- to six-user system)
`sread/s`	`read` system calls per second
`swrit/s`	`write` system calls per second
`fork/s`	`fork` system calls per second (about 0.5 per second on a four- to six-user system); this number will increase if shell scripts are running
`exec/d`	`exec` system calls per second; if `exec/s` divided by `fork/s` is greater than three, look for inefficient `PATH` variables
`rchar/s`	Characters (bytes) transferred by `read` system calls per second
`wchar/s`	Characters (bytes) transferred by `write` system calls per second

Example—Checking System Call Statistics

The following abbreviated example shows output from the `sar -c` command.

```
$ sar -c
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 scall/s sread/s swrit/s  fork/s  exec/s rchar/s wchar/s
01:00:00      38       2       2    0.00    0.00     149     120
02:00:02      38       2       2    0.00    0.00     149     120
03:00:00      42       2       2    0.05    0.05     218     147
04:00:00      39       2       2    0.01    0.00     155     123
```

```
05:00:01        38      2       2     0.00    0.00    150     120
06:00:00        38      2       2     0.01    0.00    149     120

Average         50      4       3     0.02    0.02    532     238
```

▼ How to Check Disk Activity (`sar -d`)

Display disk activity statistics with the `sar -d` command.

```
$ sar -d
00:00:00   device        %busy   avque   r+w/s   blks/s   avwait   avserv

01:00:00   fd0               0     0.0       0        0      0.0      0.0
```

The table below describes the disk devices activities reported by the `-d` option. Note that queue lengths and wait times are measured when there is something in the queue. If `%busy` is small, large queues and service times probably represent the periodic efforts by the system to ensure that altered blocks are written to the disk in a timely fashion.

TABLE 24–10 Output from the `sar -d` Command

Field Name	Description
device	Name of the disk device being monitored
%busy	Percentage of time the device spent servicing a transfer request
avque	The sum of the average wait time plus the average service time
r+w/s	Number of read and write transfers to the device per second
blks/s	Number of 512-byte blocks transferred to the device per second
avwait	Average time, in milliseconds, that transfer requests wait idly in the queue (measured only when the queue is occupied)
avserv	Average time, in milliseconds, for a transfer request to be completed by the device (for disks, this includes seek, rotational latency, and data transfer times)

Examples—Checking Disk Activity

This abbreviated example illustrates the `sar -d` output.

```
$ sar -d
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00   device        %busy   avque   r+w/s   blks/s   avwait   avserv
```

```
01:00:00   fd0          0    0.0    0    0    0.0    0.0
           nfs1         0    0.0    0    0    0.0    0.0
           sd0          0    0.0    0    0    0.0   39.6
           sd0,a        0    0.0    0    0    0.0   39.6
           sd0,b        0    0.0    0    0    0.0    0.0
           sd0,c        0    0.0    0    0    0.0    0.0
           sd0,f        0    0.0    0    0    0.0    0.0
           sd0,g        0    0.0    0    0    0.0    0.0
           sd0,h        0    0.0    0    0    0.0    0.0
           sd6          0    0.0    0    0    0.0    0.0
```

▼ How to Check Page-Out and Memory (`sar -g`)

Use the `sar -g` option reports page-out and memory freeing activities (in averages).

```
$ sar -g
00:00:00  pgout/s ppgout/s pgfree/s pgscan/s %ufs_ipf
01:00:00    0.00     0.00     0.00     0.00     0.00
```

The output displayed by `sar -g` is a good indicator of whether more memory may be needed. Use the `ps -elf` command to show the number of cycles used by the page daemon. A high number of cycles, combined with high values for `pgfree/s` and `pgscan/s` indicates a memory shortage.

The `sar -g` command also shows whether inodes are being recycled too quickly, causing a loss of reusable pages.

Output from the `-g` option is described in the following table.

TABLE 24–11 Output From the `sar -g` Command

Field Name	Description
`pgout/s`	The number of page-out requests per second.
`ppgout/s`	The actual number of pages that are paged-out, per second. (A single page-out request may involve paging-out multiple pages.)
`pgfree/s`	The number of pages, per second, that are placed on the free list.
`pgscan/s`	The number of pages, per second, scanned by the page daemon. If this value is high, the page daemon is spending a lot of time checking for free memory. This implies that more memory may be needed.

TABLE 24–11 Output From the `sar -g` Command   *(Continued)*

Field Name	Description
`%ufs_ipf`	The percentage of `ufs` inodes taken off the free list by `iget` that had reusable pages associated with them. These pages are flushed and cannot be reclaimed by processes. Thus, this is the percentage of `iget`s with page flushes. A high value indicates that the free list of inodes is page-bound and the number of `ufs` inodes may need to be increased.

Example—Checking Page-Out and Memory

The following abbreviated example shows output from the `sar -g` command.

```
$ sar -g
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00  pgout/s ppgout/s pgfree/s pgscan/s %ufs_ipf
01:00:00     0.00     0.00     0.00     0.00     0.00
02:00:02     0.00     0.00     0.00     0.00     0.00
03:00:00     0.00     0.01     0.01     0.00     0.00
04:00:00     0.00     0.00     0.00     0.00     0.00
05:00:01     0.00     0.00     0.00     0.00     0.00
06:00:00     0.00     0.00     0.00     0.00     0.00

Average      0.01     0.12     0.21     0.66     0.00
```

▼ How to Check Kernel Memory Allocation (`sar -k`)

Use the `sar -k` command to report on the following activities of the Kernel Memory Allocator (KMA).

The KMA allows a kernel subsystem to allocate and free memory as needed. Rather than statically allocating the maximum amount of memory it is expected to require under peak load, the KMA divides requests for memory into three categories: small (less than 256 bytes), large (512 to 4 Kbytes), and oversized (greater than 4 Kbytes). It keeps two pools of memory to satisfy small and large requests. The oversized requests are satisfied by allocating memory from the system page allocator.

If you are investigating a system that is being used to write drivers or STREAMS that use KMA resources, then `sar -k` will likely prove useful. Otherwise, you will probably not need the information it provides. Any driver or module that uses KMA resources, but does not specifically return the resources before it exits, can create a memory leak. A memory leak causes the amount of memory allocated by KMA to increase over time. Thus, if the `alloc` fields of `sar -k` increase steadily over time,

there may be a memory leak. Another indication of a memory leak is failed requests. If this occurs, a memory leak has probably caused KMA to be unable to reserve and allocate memory.

If it appears that a memory leak has occurred, you should check any drivers or STREAMS that may have requested memory from KMA and not returned it.

```
$ sar -k
00:00:00 sml_mem   alloc  fail  lg_mem   alloc   fail  ovsz_alloc  fail
01:00:00 2523136 1866512     0 18939904 14762364    0      360448    0
02:00:02 2523136 1861724     0 18939904 14778748    0      360448    0
```

Output from the -k option is described in the table below.

TABLE 24–12 Output From the sar -k Command

Field Name	Description
sml_mem	The amount of memory, in bytes, that the KMA has available in the small memory request pool (a small request is less than 256 bytes)
alloc	The amount of memory, in bytes, that the KMA has allocated from its small memory request pool to small memory requests
fail	The number of requests for small amounts of memory that failed
lg_mem	The amount of memory, in bytes, that the KMA has available in the large memory request pool (a large request is from 512 bytes to 4 Kbytes)
alloc	The amount of memory, in bytes, that the KMA has allocated from its large memory request pool to large memory requests
fail	The number of failed requests for large amounts of memory
ovsz_alloc	The amount of memory allocated for oversized requests (those greater than 4 Kbytes); these requests are satisfied by the page allocator—thus, there is no pool
fail	The number of failed requests for oversized amounts of memory

Example—Checking Kernel Memory Allocation (sar)

The following is an abbreviated example of sar -k output.

```
$ sar -k
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 sml_mem   alloc  fail  lg_mem   alloc  fail  ovsz_alloc  fail
01:00:00 2523136 1866512     0 18939904 14762364    0      360448    0
02:00:02 2523136 1861724     0 18939904 14778748    0      360448    0
03:00:00 2523136 1865664     0 18939904 14745884    0      360448    0
```

```
04:00:00 2523136 1867692        0 18939904 14746616        0        360448        0
05:00:01 2523136 1867208        0 18939904 14763700        0        360448        0
06:00:00 2523136 1867772        0 18939904 14779444        0        360448        0

Average  2724096 1791806        0 20089344 15434591        0        360448        0
```

▼ How to Check Interprocess Communication (`sar -m`)

Use the `sar -m` command to report interprocess communication activities.

```
$ sar -m
00:00:00   msg/s   sema/s
01:00:00   0.00    0.00
```

These figures will usually be zero (0.00), unless you are running applications that use messages or semaphores.

The output from the `-m` option is described in the table below.

TABLE 24–13 Output From the `sar -m` Command

Field Name	Description
msg/s	The number of message operations (sends and receives) per second.
sema/s	The number of semaphore operations per second.

Example—Checking Interprocess Communication

The following abbreviated example shows output from the `sar -m` command.

```
$ sar -m
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00   msg/s   sema/s
01:00:00   0.00    0.00
02:00:02   0.00    0.00
03:00:00   0.00    0.00
04:00:00   0.00    0.00
05:00:01   0.00    0.00
06:00:00   0.00    0.00

Average    0.00    0.00
```

▼ How to Check Page-In Activity (`sar -p`)

Use the `sar -p` command to report page-in activity which includes protection and translation faults.

```
$ sar -p
00:00:00  atch/s  pgin/s ppgin/s  pflt/s  vflt/s slock/s
01:00:00   0.07    0.00    0.00    0.21    0.39    0.00
```

The reported statistics from the -p option are described in the table below.

TABLE 24–14 Output from the `sar -p` Command

Field Name	Description
`atch/s`	The number of page faults, per second, that are satisfied by reclaiming a page currently in memory (attaches per second). Instances of this include reclaiming an invalid page from the free list and sharing a page of text currently being used by another process (for example, two or more processes accessing the same program text).
`pgin/s`	The number of times, per second, that file systems receive page-in requests.
`ppgin/s`	The number of pages paged in, per second. A single page-in request, such as a soft-lock request (see `slock/s`), or a large block size, may involve paging-in multiple pages.
`pflt/s`	The number of page faults from protection errors. Instances of protection faults are illegal access to a page and "copy-on-writes." Generally, this number consists primarily of "copy-on-writes."
`vflt/s`	The number of address translation page faults, per second. These are known as validity faults, and occur when a valid process table entry does not exist for a given virtual address.
`slock/s`	The number of faults, per second, caused by software lock requests requiring physical I/O. An example of the occurrence of a soft-lock request is the transfer of data from a disk to memory. The system locks the page that is to receive the data, so that it cannot be claimed and used by another process.

Example—Checking Page-In Activity

The following abbreviated example shows output from `sar -p`.

```
$ sar -p
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00  atch/s  pgin/s ppgin/s  pflt/s  vflt/s slock/s
```

```
01:00:00    0.07    0.00    0.00    0.21    0.39    0.00
02:00:02    0.07    0.00    0.00    0.21    0.39    0.00
03:00:00    0.32    0.00    0.00    1.10    2.48    0.00
04:00:00    0.09    0.00    0.00    0.32    0.57    0.00
05:00:01    0.07    0.00    0.00    0.21    0.39    0.00
06:00:00    0.07    0.00    0.00    0.21    0.39    0.00

Average     0.26    0.20    0.30    0.92    1.78    0.00
```

▼ How to Check Queue Activity (`sar -q`)

Use the `sar -q` command to report the average queue length while the queue is occupied, and the percentage of time that the queue is occupied.

```
$ sar -q
00:00:00 runq-sz %runocc swpq-sz %swpocc
01:00:00
```

Note – The number of LWPs swapped out may greater than zero even if the system has an abundance of free memory. This happens when a sleeping LWP is swapped out and has not been awakened (for example, a process or LWP sleeping, waiting for the keyboard or mouse input).

Output from the `-q` option is described in the table below.

TABLE 24–15 Output From the `sar -q` Command

Field Name	Description
`runq-sz`	The number of kernel threads in memory waiting for a CPU to run. Typically, this value should be less than 2. Consistently higher values mean that the system may be CPU-bound.
`%runocc`	The percentage of time the dispatch queues are occupied.
`swpq-sz`	The average number of swapped out LWPs.
`%swpocc`	The percentage of time LWPs are swapped out.

Example—Checking Queue Activity

The following abbreviated example shows output from the `sar -q` command. If `%runocc` is high (greater than 90 percent) and `runq-sz` is greater than 2, the CPU is heavily loaded and response is degraded. In this case, additional CPU capacity may be required to obtain acceptable system response.

```
$ sar -q
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 runq-sz %runocc swpq-sz %swpocc
01:00:00
02:00:02
03:00:00    1.0        0
04:00:00
05:00:01    1.0        0
06:00:00


Average      1.3        0
```

▼ How to Check Unused Memory (sar -r)

Use the sar -r command to report the number of memory pages and swap-file disk
blocks that are currently unused.

```
$ sar -r
00:00:00 freemem freeswap
01:00:00    2135    401922
```

Output from the -r option is described in the table below.

TABLE 24–16 Output From the sar -r Command

Field Name	Description
freemem	The average number of memory pages available to user processes over the intervals sampled by the command. Page size is machine-dependent.
freeswap	The number of 512-byte disk blocks available for page swapping.

Example—Checking Unused Memory

The following example shows output from the sar -r command.

```
$ sar -r
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 freemem freeswap
01:00:00    2135    401922
02:00:02    2137    401949
03:00:00    2137    402006
04:00:00    2139    401923
05:00:01    2138    402033
06:00:00    2137    401919
```

```
Average     2500   399914
```

▼ How to Check CPU Utilization (`sar -u`)

Display CPU utilization with the `sar -u` command.

```
$ sar -u
00:00:00    %usr    %sys    %wio    %idle
01:00:00       0       0       0     100
```

(The `sar` command without any options is equivalent to `sar -u`.) At any given
moment, the processor is either busy or idle. When busy, the processor is in either user
or system mode. When idle, the processor is either waiting for I/O completion or
"sitting still" with no work to do.

Output from the `-u` option is described in the table below.

TABLE 24–17 Output From the `sar -u` Command

Field Name	Description
`%sys`	Lists the percentage of time that the processor is in system mode
`%user`	Lists the percentage of time that the processor is in user mode
`%wio`	Lists the percentage of time the processor is idle and waiting for I/O completion
`%idle`	Lists the percentage of time the processor is idle and is not waiting for I/O

A high `%wio` generally means a disk slowdown has occurred.

Example—Checking CPU Utilization

The following example shows output from the `sar -u` command.

```
$ sar -u
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00    %usr    %sys    %wio    %idle
01:00:00       0       0       0     100
02:00:02       0       0       0     100
03:00:00       0       0       0     100
04:00:00       0       0       0     100
05:00:01       0       0       0     100
06:00:00       0       0       0     100
07:00:00       0       0       0     100
```

```
08:00:01        0        0        0      100
08:20:00        0        0        0      100
08:40:00        0        0        0      100
09:00:00        0        0        0      100
09:20:00        0        0        0      100
09:40:00        0        0        0      100
10:00:00        0        0        0      100
10:20:00        0        0        0      100
10:40:01        0        0        0      100
11:00:00        5        2       10       82

Average         0        0        0      100
```

▼ How to Check System Table Status (`sar -v`)

Use the `sar -v` command to report the status of the process table, inode table, file table, and shared memory record table.

```
$ sar -v
00:00:00  proc-sz    ov  inod-sz    ov  file-sz    ov   lock-sz
01:00:00   43/922     0 2984/4236    0  322/322     0    0/0
```

Output from the `-v` option is described in the table below.

TABLE 24–18 Output From the `sar -v` Command

Field Name	Description
proc-sz	The number of process entries (`proc` structs) currently being used, or allocated in the kernel.
inod-sz	The total number of inodes in memory verses the maximum number of inodes allocated in the kernel. This is not a strict high water mark; it can overflow.
file-sz	The size of the open system file table. The `sz` is given as `0`, since space is allocated dynamically for the file table.
ov	The number of shared memory record table entries currently being used or allocated in the kernel. The `sz` is given as `0` because space is allocated dynamically for the shared memory record table.
lock-sz	The number of shared memory record table entries currently being used or allocated in the kernel. The `sz` is given as `0` because space is allocated dynamically for the shared memory record table.

Example—Checking System Table Status

The following abbreviated example shows output from the `sar -v` command. This example shows that all tables are large enough to have no overflows. These tables are all dynamically allocated based on the amount of physical memory.

```
$ sar -v
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00  proc-sz    ov  inod-sz    ov  file-sz    ov   lock-sz
01:00:00   43/922     0 2984/4236    0  322/322     0    0/0
02:00:02   43/922     0 2984/4236    0  322/322     0    0/0
03:00:00   43/922     0 2986/4236    0  323/323     0    0/0
04:00:00   43/922     0 2987/4236    0  322/322     0    0/0
05:00:01   43/922     0 2987/4236    0  322/322     0    0/0
06:00:00   43/922     0 2987/4236    0  322/322     0    0/0
```

▼ How to Check Swap Activity (`sar -w`)

Use the `sar -w` command to report swapping and switching activity.

```
$ sar -w
00:00:00 swpin/s bswin/s swpot/s bswot/s pswch/s
01:00:00   0.00     0.0    0.00     0.0      22
```

Target values and observations are described in the table below.

TABLE 24–19 Output From the `sar -w` Command

Field Name	Description
swpin/s	The number of LWP transfers into memory per second.
bswin/s	The average number of processes swapped out of memory per second. If the number is greater than 1, you may need to increase memory.
swpot/s	The average number of processes swapped out of memory per second. If the number is greater than 1, you may need to increase memory.
bswot/s	The number of blocks transferred for swap-outs per second.
pswch/s	The number of kernel thread switches per second.

Note – All process swap-ins include process initialization.

Example—Checking Swap Activity

The following example shows output from the sar -w command.

```
$ sar -w
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 swpin/s bswin/s swpot/s bswot/s pswch/s
01:00:00    0.00    0.0    0.00    0.0     22
02:00:02    0.00    0.0    0.00    0.0     22
03:00:00    0.00    0.0    0.00    0.0     22
04:00:00    0.00    0.0    0.00    0.0     22
05:00:01    0.00    0.0    0.00    0.0     22
06:00:00    0.00    0.0    0.00    0.0     22
07:00:00    0.00    0.0    0.00    0.0     22
08:00:01    0.00    0.0    0.00    0.0     22
08:20:00    0.00    0.0    0.00    0.0     22
08:40:00    0.00    0.0    0.00    0.0     22
09:00:00    0.00    0.0    0.00    0.0     22
09:20:00    0.00    0.0    0.00    0.0     22
09:40:00    0.00    0.0    0.00    0.0     22
10:00:00    0.00    0.0    0.00    0.0     22
10:20:00    0.00    0.0    0.00    0.0     22
10:40:01    0.00    0.0    0.00    0.0     23
11:00:00    0.00    0.0    0.00    0.0    144

Average     0.00    0.0    0.00    0.0     24
```

▼ How to Check Terminal Activity (sar -y)

Use the sar -y command to monitor terminal device activities.

```
$ sar -y
00:00:00 rawch/s canch/s outch/s rcvin/s xmtin/s mdmin/s
01:00:00      0      0      0      0      0      0
```

If you have a lot of terminal I/O, you can use this report to determine if there are any bad lines. The activities recorded are defined in the table below.

TABLE 24–20 Output From the sar -y Command

Field Name	Description
rawch/s	Input characters (raw queue), per second

TABLE 24–20 Output From the `sar -y` Command *(Continued)*

Field Name	Description
`canch/s`	Input characters processed by canon (canonical queue) per second
`outch/s`	Output characters (output queue) per second
`rcvin/s`	Receiver hardware interrupts per second
`xmtin/s`	Transmitter hardware interrupts per second
`mdmin/s`	Modem interrupts per second

The number of modem interrupts per second (`mdmin/s`) should be close to zero, and the receive and transmit interrupts per second (`xmtin/s` and `rcvin/s`) should be less than or equal to the number of incoming or outgoing characters, respectively. If this is not the case, check for bad lines.

Example—Checking Terminal Activity

The following abbreviated example shows output from the `sar -y` command.

```
$ sar -y
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 rawch/s canch/s outch/s rcvin/s xmtin/s mdmin/s
01:00:00       0       0       0       0       0       0
02:00:02       0       0       0       0       0       0
03:00:00       0       0       0       0       0       0
04:00:00       0       0       0       0       0       0
05:00:01       0       0       0       0       0       0
06:00:00       0       0       0       0       0       0
07:00:00       0       0       0       0       0       0
08:00:01       0       0       0       0       0       0
08:20:00       0       0       0       0       0       0
08:40:00       0       0       0       0       0       0
09:00:00       0       0       0       0       0       0
09:20:00       0       0       0       0       0       0
09:40:00       0       0       0       0       0       0
10:00:00       0       0       0       0       0       0
10:20:00       0       0       0       0       0       0
10:40:01       0       0      20       0       0       0

Average        0       0       3       0       0       0
```

▼ How to Check Overall System Performance (`sar -A`)

Use the `sar -A` command to display a view of overall system performance.

This provides a more global perspective. If data from more than one time segment is shown, the report includes averages.

Collecting System Activity Data Automatically (`sar`)

Three commands are involved in automatic system activity data collection: `sadc`, `sa1`, and `sa2`.

The `sadc` data collection utility periodically collects data on system activity and saves it in a file in binary format—one file for each 24-hour period. You can set up `sadc` to run periodically (usually once each hour), and whenever the system boots to multiuser mode. The data files are placed in the directory `/usr/adm/sa`. Each file is named sa*dd*, where *dd* is the current date. The format of the command is as follows:

```
/usr/lib/sa/sadc [t n] [ofile]
```

The command samples *n* times with an interval of *t* seconds (*t* should be greater than 5 seconds) between samples. It then writes, in binary format, to the file *ofile*, or to standard output. If *t* and *n* are omitted, a special file is written once.

Running `sadc` When Booting

The `sadc` command should be run at system boot time in order to record the statistics from when the counters are reset to zero. To make sure that `sadc` is run at boot time, the `/etc/init.d/perf` file must contain a command line that writes a record to the daily data file.

The command entry has the following format:

```
su sys -c "/usr/lib/sa/sadc /usr/adm/sa/sa`date +5d`"
```

Running `sadc` Periodically With `sa1`

To generate periodic records, you need to run `sadc` regularly. The simplest way to do this is by putting a line into the `/var/spool/cron/sys` file, which calls the shell script, `sa1`. This script invokes `sadc` and writes to the daily data files, `/var/adm/sa/sa`*dd*. It has the following format:

```
/usr/lib/sa/sa1 [t n]
```

The arguments *t* and *n* cause records to be written *n* times at an interval of *t* seconds. If these arguments are omitted, the records are written only one time.

Producing Reports With `sa2`

Another shell script, `sa2`, produces reports rather than binary data files. The `sa2` command invokes the `sar` command and writes the ASCII output to a report file.

Setting Up Automatic Data Collection (`sar`)

The `sar` command can be used either to gather system activity data itself or to report what has been collected in the daily activity files created by `sadc`.

The `sar` command has the following formats:

```
sar [-aAbcdgkmpqruvwy] [-o file] t [n]
```

```
sar [-aAbcdgkmpqruvwy] [-s time] [-e time] [-i sec] [-f file]
```

The `sar` command below samples cumulative activity counters in the operating system every *t* seconds, *n* times. (*t* should be 5 seconds or greater; otherwise, the command itself may affect the sample.) You must specify a time interval between which to take the samples; otherwise, the command operates according to the second format. The default value of *n* is 1. The following example takes two samples separated by 10 seconds. If the -o option is specified, samples are saved in *file* in binary format.

```
$ sar -u 10 2
```

Other important information about the `sar` command:

- With no sampling interval or number of samples specified, `sar` extracts data from a previously recorded file, either the one specified by the -f option or, by default, the standard daily activity file, /var/adm/sa/sa*dd*, for the most recent day.

- The -s and -e options define the starting and ending times for the report. Starting and ending times are of the form *hh*[*:mm*[*:ss*]] (where *h*, *m*, and *s* represent hours, minutes, and seconds).

- The -i option specifies, in seconds, the intervals between record selection. If the -i option is not included, all intervals found in the daily activity file are reported.

The table below lists the `sar` options and their actions.

TABLE 24–21 Options for `sar` Command

Option	Actions
-a	Checks file access operations
-b	Checks buffer activity
-c	Checks system calls
-d	Checks activity for each block device
-g	Checks page-out and memory freeing
-k	Checks kernel memory allocation
-m	Checks interprocess communication
-p	Checks swap and dispatch activity
-q	Checks queue activity
-r	Checks unused memory
-u	Checks CPU utilization
-nv	Checks system table status
-w	Checks swapping and switching volume
-y	Checks terminal activity
-A	Reports overall system performance (same as entering all options)

If no option is used, it is equivalent to calling the command with the -u option.

▼ How to Set Up Automatic Data Collection

1. **Become superuser.**

2. **Edit the** /etc/init.d/perf **file and uncomment all lines:**

   This version of the sadc command writes a special record that marks the time when the counters are reset to zero (boot time). The sadc output is put into the file sa*dd* (where *dd* is the current date), which acts as the daily system activity record.

3. **Edit the** /var/spool/cron/crontabs/sys **file (the system** crontab **file) and uncomment the following lines:**

   ```
   # 0 * * * 0-6 /usr/lib/sa/sa1
   # 20,40 8-17 * * 1-5 /usr/lib/sa/sa1
   # 5 18 * * 1-5 /usr/lib/sa/sa2 -s 8:00 -e 18:01 -i 1200 -A
   ```

The first entry writes a record to /var/adm/sa/sa*dd* on the hour, every hour, seven days a week.

The second entry writes a record to /var/adm/sa/sa*dd* twice each hour during peak working hours: at 20 minutes and 40 minutes past the hour, from 8 a.m. to 5 p.m., Monday through Friday.

Thus, these two crontab entries cause a record to be written to /var/adm/sa/sa*dd* every 20 minutes from 8 a.m. to 5 p.m., Monday through Friday, and every hour on the hour otherwise. You can change these defaults to meet your needs.

CHAPTER **25**

Troubleshooting Solaris Software Topics

This section provides instructions for troubleshooting Solaris software problems. This section contains these chapters.

Chapter 26	Provides overview information about troubleshooting common software problems and instructions for troubleshooting a system crash.
Chapter 27	Provides step-by-step instructions for managing core files.
Chapter 28	Provides step-by-step instructions for saving crash dumps and customizing system error logging.
Chapter 29	Provides problem scenarios and possible solutions for general software problems such as a hung system or a system that won't boot.
Chapter 30	Provides solutions for solving common file access problems such as incorrect command search paths and file permissions.
Chapter 31	Provides solutions for solving common printer problems such as no output or incorrect output.
Chapter 32	Provides specific `fsck` error messages and corresponding solutions for solving file system-related problems.
Chapter 33	Provides specific error messages and possible solutions for problems encountered when adding or removing software packages.

Troubleshooting Software Problems (Overview)

This chapter provides a general overview of troubleshooting software problems, including information on troubleshooting system crashes and viewing system messages.

This is a list of information in this chapter.

- "Where to Find Software Troubleshooting Tasks" on page 369
- "Troubleshooting a System Crash" on page 370
- "Troubleshooting a System Crash Checklist" on page 371
- "Viewing System Messages" on page 372
- "Customizing System Message Logging" on page 374

Where to Find Software Troubleshooting Tasks

Use these references to find step-by-step instructions for troubleshooting software problems.

- Chapter 28
- Chapter 29
- Chapter 30
- Chapter 31
- Chapter 32
- Chapter 33

Troubleshooting a System Crash

If a system running the Solaris operating environment crashes, provide your service provider with as much information as possible—including crash dump files.

What to Do if the System Crashes

The most important things to remember are:

1. Write down the system console messages.

   If a system crashes, making it run again might seem like your most pressing concern. However, before you reboot the system, examine the console screen for messages. These messages can provide some insight about what caused the crash. Even if the system reboots automatically and the console messages have disappeared from the screen, you might be able to check these messages by viewing the system error log file that is generated automatically in /var/adm/messages (or /usr/adm/messages). See "How to View System Messages" on page 373 for more information about viewing system error log files.

   If you have frequent crashes and can't determine their cause, gather all the information you can from the system console or the /var/adm/messages files, and have it ready for a customer service representative to examine. See "Troubleshooting a System Crash" on page 370 for a complete list of troubleshooting information to gather for your service provider.

   See Chapter 29 if the system fails to reboot successfully after a system crash.

2. Synchronize the disks and reboot.

   ```
   ok sync
   ```

   See Chapter 29 if the system fails to reboot successfully after a system crash.

3. Attempt to save the crash information written onto the swap area by running the savecore command.

   ```
   # savecore
   ```

See Chapter 28 for information about saving crash dumps automatically.

Gathering Troubleshooting Data

Answer the following questions to help isolate the system problem. Use "Troubleshooting a System Crash Checklist" on page 371 for gathering troubleshooting data for a crashed system.

TABLE 26–1 Identifying System Crash Data

Question	Description
Can you reproduce the problem?	This is important because a reproducible test case is often essential for debugging really hard problems. By reproducing the problem, the service provider can build kernels with special instrumentation to trigger, diagnose, and fix the bug.
Are you using any third-party drivers?	Drivers run in the same address space as the kernel, with all the same privileges, so they can cause system crashes if they have bugs.
What was the system doing just before it crashed?	If the system was doing anything unusual like running a new stress test or experiencing higher-than-usual load, that may have led to the crash.
Were there any unusual console messages right before the crash?	Sometimes the system will show signs of distress before it actually crashes; this information is often useful.
Did you add any tuning parameters to the `/etc/system` *file?*	Sometimes tuning parameters, such as increasing shared memory segments so that the system tries to allocate more than it has, can cause the system to crash.
Did the problem start recently?	If so, did the onset of problems coincide with any changes to the system, for example, new drivers, new software, different workload, CPU upgrade, or a memory upgrade.

Troubleshooting a System Crash Checklist

Use this checklist when gathering system data for a crashed system.

Item	Your Data
Is a core file available?	
Identify the operating system release and appropriate software application release levels.	

Item	Your Data
Identify system hardware.	
Include `prtdiag` output for sun4u systems. Include Explorer (?) output for other systems.	
Are patches installed? If so, include `showrev -p` output.	
Is the problem reproducible?	
Does the system have any third-party drivers?	
What was the system doing before it crashed?	
Were there any unusual console messages right before the system crashed?	
Did you add any parameters to the `/etc/system` file?	
Did the problem start recently?	

Viewing System Messages

System messages display on the console device. The text of most system messages look like this:

[ID *msgid facility.priority*]

For example:

```
[ID 672855 kern.notice] syncing file systems...
```

If the message originated in the kernel, the kernel module name is displayed. For example:

```
Oct 1 14:07:24 mars ufs: [ID 845546 kern.notice] alloc: /: file system full
```

When a system crashes, it may display a message on the system console like this:

panic: *error message*

Less frequently, this message may be displayed instead of the panic message:

```
Watchdog reset !
```

The error logging daemon, `syslogd`, automatically records various system warnings and errors in message files. By default, many of these system messages are displayed

on the system console and are stored in the /var/adm directory. You can direct where these messages are stored by setting up system message logging. See "How to Customize System Message Logging" on page 375 for more information. These messages can alert you to system problems, such as a device that is about to fail.

The /var/adm directory contains several message files. The most recent messages are in /var/adm/messages (and in messages.0), and the oldest are in messages.3. After a period of time (usually every ten days), a new messages file is created. The messages.0 file is renamed messages.1, messages.1 is renamed messages.2, and messages.2 is renamed messages.3. The current /var/adm/messages.3 is deleted.

Because /var/adm stores large files containing messages, crash dumps, and other data, this directory can consume lots of disk space. To keep the /var/adm directory from growing too large, and to ensure that future crash dumps can be saved, you should remove unneeded files periodically. You can automate this task by using crontab. See "How to Delete Crash Dump Files" on page 245 and Chapter 18 for more information on automating this task.

▼ How to View System Messages

Display recent messages generated by a system crash or reboot by using the dmesg command.

```
$ dmesg
```

Or use the more command to display one screen of messages at a time.

```
$ more /var/adm/messages
```

For more information, refer to dmesg(1M).

Example—Viewing System Messages

The following example shows output from the dmesg command.

```
$ dmesg
date starbug genunix: [ID 540533 kern.notice] SunOS Release 5.9 Version 64-bit
date starbug genunix: [ID 223299 kern.notice] Copyright (c) 1983-2002 ...
date starbug genunix: [ID 678236 kern.info] Ethernet address = xx:xx:xx:xx:xx:xx
date starbug unix: [ID 389951 kern.info] mem = 131072K (0x8000000)
date starbug unix: [ID 930857 kern.info] avail mem = 122134528
date starbug rootnex: [ID 466748 kern.info] root nexus = Sun Ultra 5/10 UPA/PCI ...
date starbug rootnex: [ID 349649 kern.info] pcipsy0 at root: UPA 0x1f 0x0
date starbug genunix: [ID 936769 kern.info] pcipsy0 is /pci@1f,0
date starbug pcipsy: [ID 370704 kern.info] PCI-device: pci@1,1, simba0
date starbug genunix: [ID 936769 kern.info] simba0 is /pci@1f,0/pci@1,1
date starbug pcipsy: [ID 370704 kern.info] PCI-device: pci@1, simba1
```

```
date starbug genunix: [ID 936769 kern.info] simba1 is /pci@1f,0/pci@1
date starbug simba: [ID 370704 kern.info] PCI-device: ide@3, uata0
date starbug genunix: [ID 936769 kern.info] uata0 is /pci@1f,0/pci@1,1/ide@3
.
.
.
```

Customizing System Message Logging

You can capture additional error messages that are generated by various system processes by modifying the /etc/syslog.conf file. By default, /etc/syslog.conf directs many system process messages to the /var/adm message files. Crash and boot messages are stored here as well. To view /var/adm messages, see "How to View System Messages" on page 373.

The /etc/syslog.conf file has two columns separated by tabs:

facility.level ...	*action*
facility.level	A *facility* or system source of the message or condition. May be a comma-separated listed of facilities. Facility values are listed in Table 26–2. A *level*, indicates the severity or priority of the condition being logged. Priority levels are listed in Table 26–3.
action	The action field indicates where the messages are forwarded.

The following example shows sample lines from a default /etc/syslog.conf file.

```
user.err                          /dev/sysmsg
user.err                          /var/adm/messages
user.alert                        'root, operator'
user.emerg                        *
```

This means the following user messages are automatically logged:

- User errors are printed to the console and also are logged to the /var/adm/messages file.

- User messages requiring immediate action (alert) are sent to the root and operator users.

- User emergency messages are sent to individual users.

The most common error condition sources are shown in the table below. The most common priorities are shown in Table 26–3 in order of severity.

TABLE 26–2 Source Facilities for `syslog.conf` Messages

Source	Description
kern	The kernel
auth	Authentication
daemon	All daemons
mail	Mail system
lp	Spooling system
user	User processes

Note – Starting in the Solaris 2.6 release, the number of `syslog` facilities that can be activated in the `/etc/syslog.conf` file is unlimited. In previous releases, the number of facilities was limited to 20.

TABLE 26–3 Priority Levels for `syslog.conf` Messages

Priority	Description
emerg	System emergencies
alert	Errors requiring immediate correction
crit	Critical errors
err	Other errors
info	Informational messages
debug	Output used for debugging
none	This setting doesn't log output

▼ How to Customize System Message Logging

1. **Become superuser.**

2. **Using the editor of your choice, edit the** `/etc/syslog.conf` **file, adding or changing message sources, priorities, and message locations according to the syntax described in** `syslog.conf`**(4) .**

3. **Exit the file, saving the changes.**

Example—Customizing System Message Logging

This sample `/etc/syslog.conf user.emerg` facility sends user emergency messages to root *and* individual users.

```
user.emerg                                          'root, *'
```

Enabling Remote Console Messaging

The following new console features improve your ability to troubleshoot remote systems:

- The `consadm` command enables you to select a serial device as an *auxiliary* (or remote) console. Using the `consadm` command, a system administrator can configure one or more serial ports to display redirected console messages and to host `sulogin` sessions when the system transitions between run levels. This feature enables you to dial in to a serial port with a modem to monitor console messages and participate in `init` state transitions. (See `sulogin(1M)` and the step-by-step procedures below for more information.)

  While you can log in to a system using a port configured as an auxiliary console, it is primarily an output device displaying information that is also displayed on the default console. If boot scripts or other applications read and write to and from the default console, the write output displays on all the auxiliary consoles, but the input is only read from the default console. (See "Using the `consadm` Command During an Interactive Login Session" on page 378 for using the `consadm` command during an interactive login session.)

- Console output now consists of kernel and `syslog` messages written to a new pseudo device, `/dev/sysmsg`. In addition, `rc` script startup messages are written to `/dev/msglog`. Previously, all of these messages were written to `/dev/console`.

  Scripts that direct console output to `/dev/console` need to be changed to `/dev/msglog` if you want to see script messages displayed on the auxiliary consoles. Programs referencing `/dev/console` should be explicitly modified to use `syslog()` or `strlog()` if you want messages to be redirected to an auxiliary device.

- The `consadm` command runs a daemon to monitor auxiliary console devices. Any display device designated as an auxiliary console that disconnects—hangs up or loses carrier—is removed from the auxiliary console device list and is no longer active. Enabling one or more auxiliary consoles does not disable message display

on the default console; messages continue to display on `/dev/console`.

Using Auxiliary Console Messaging During Run Level Transitions

Keep the following in mind when using auxiliary console messaging during run level transitions:

- Input cannot come from an auxiliary console if user input is expected for an `rc` script that is run when a system is booting. The input must come from the default console.

- The `sulogin` program, invoked by `init` to prompt for the superuser password when transitioning between run levels, has been modified to send the superuser password prompt to each auxiliary device in addition to the default console device.

- When the system is in single-user mode and one or more auxiliary consoles are enabled using the `consadm` command, a console login session runs on the first device to supply the correct superuser password to the `sulogin` prompt. When the correct password is received from a console device, `sulogin` disables input from all other console devices.

- A message is displayed on the default console and the other auxiliary consoles when one of the consoles assumes single-user privileges. This message indicates which device has become the console by accepting a correct superuser password. If there is a loss of carrier on the auxiliary console running the single-user shell, one of two actions may occur:

  - If the auxiliary console represents a system at run level 1, the system proceeds to the default run level.

  - If the auxiliary console represents a system at run level S, the system displays the `ENTER RUN LEVEL (0-6, s or S):` message on the device where the `init s` or `shutdown` command had been entered from the shell. If there isn't any carrier on that device either, you will have to reestablish carrier and enter the correct run level. The `init` or `shutdown` command will not redisplay the run-level prompt.

- If you are logged in to a system using a serial port, and an `init` or `shutdown` command is issued to transition to another run level, the login session is lost whether this device is the auxiliary console or not. This situation is identical to Solaris releases without auxiliary console capabilities.

- Once a device is selected as an auxiliary console using the `consadm` command, it remains the auxiliary console until the system is rebooted or the auxiliary console is unselected. However, the `consadm` command includes an option to set a device as the auxiliary console across system reboots. (See the procedure below for step-by-step instructions.)

Using the `consadm` Command During an Interactive Login Session

If you want to run an interactive login session by logging in to a system using a terminal that is connected to a serial port, and then using the `consadm` command to see the console messages from the terminal, note the following behavior.

- If you use the terminal for an interactive login session while the auxiliary console is active, the console messages are sent to the `/dev/sysmsg` or `/dev/msglog` devices.

- While you issue commands on the terminal, input goes to your interactive session and not to the default console (`/dev/console`).

- If you run the `init` command to change run levels, the remote console software kills your interactive session and runs the `sulogin` program. At this point, input is accepted only from the terminal and is treated like it's coming from a console device. This allows you to enter your password to the `sulogin` program as described in "Using Auxiliary Console Messaging During Run Level Transitions" on page 377.

  Then, if you enter the correct password on the (auxiliary) terminal, the auxiliary console runs an interactive `sulogin` session, locks out the default console and any competing auxiliary console. This means the terminal essentially functions as the system console.

- From here you can change to run level 3 or go to another run level. If you change run levels, `sulogin` runs again on all console devices. If you exit or specify that the system should come up to run level 3, then all auxiliary consoles lose their ability to provide input. They revert to being display devices for console messages.

  As the system is coming up, you must provide information to `rc` scripts on the default console device. After the system comes back up, the `login` program runs on the serial ports and you can log back into another interactive session. If you've designated the device to be an auxiliary console, you will continue to get console messages on your terminal, but all input from the terminal goes to your interactive session.

▼ How to Enable an Auxiliary (Remote) Console

The `consadm` daemon does not start monitoring the port until after you add the auxiliary console with the `consadm` command. As a security feature, console messages are only redirected until carrier drops, or the auxiliary console device is unselected. This means carrier must be established on the port before you can successfully use the `consadm` command.

See `consadm`(1M) for more information on enabling an auxiliary console.

1. **Log in to the system as superuser.**

2. **Enable the auxiliary console.**

   # **consadm -a** *devicename*

3. **Verify that the current connection is the auxiliary console.**

   # **consadm**

Example—Enabling an Auxiliary (Remote) Console

```
# consadm -a /dev/term/a
# consadm
 /dev/term/a
```

▼ How to Display a List of Auxiliary Consoles

1. **Log in to the system as superuser.**

2. **Select one of the following steps:**

   a. **Display the list of auxiliary consoles.**

   ```
   # consadm
   /dev/term/a
   ```

   b. **Display the list of persistent auxiliary consoles.**

   ```
   # consadm -p
   /dev/term/b
   ```

▼ How to Enable an Auxiliary (Remote) Console Across System Reboots

1. **Log in to the system as superuser.**

2. **Enable the auxiliary console across system reboots.**

   # **consadm -a -p** *devicename*

   This adds the device to the list of persistent auxiliary consoles.

3. **Verify that the device has been added to the list of persistent auxiliary consoles.**

   # **consadm**

Example—Enabling an Auxiliary (Remote) Console Across System Reboots

```
# consadm -a -p /dev/term/a
# consadm
/dev/term/a
```

▼ How to Disable an Auxiliary (Remote) Console

1. **Log in to the system as superuser.**

2. **Select one of the following steps:**

   a. **Disable the auxiliary console.**

      ```
      # consadm -d devicename
      ```

      or

   b. **Disable the auxiliary console and remove it from the list of persistent auxiliary consoles.**

      ```
      # consadm -p -d devicename
      ```

3. **Verify that the auxiliary console has been disabled.**

   ```
   # consadm
   ```

Example—Disabling an Auxiliary (Remote) Console

```
# consadm -d /dev/term/a
# consadm
```

Managing Core Files

This chapter describes how to manage core files with the `coreadm` command.

For information on the procedures associated with managing core files, see "Managing Core Files (Task Map)" on page 381.

Managing Core Files (Task Map)

TABLE 27–1 Managing Core Files (Task Map)

Task	Description	Instructions
1. Display the current core dump configuration	Display the current core dump configuration by using the `coreadm`command.	"How to Display the Current Core Dump Configuration" on page 384
2. Modify the core dump configuration	Modify the core dump configuration to do one of the following:	
	Set a core file name pattern.	"How to Set a Core File Name Pattern" on page 385
	Enable a per-process core file path.	"How to Enable a Per-Process Core File Path" on page 385
	Enable a global core file path.	"How to Enable a Global Core File Path" on page 385
3. Examine a Core Dump File	Use the `proc` tools to view a core dump file.	"Examining Core Files" on page 386

Managing Core Files Overview

Core files are generated when a process or application terminates abnormally. Core files are managed with the `coreadm` command.

The `coreadm` command enables you to manage core files. For example, you can use the `coreadm` command to configure a system so that all process core files are placed in a single system directory. This means it is easier to track problems by examining the core files in a specific directory whenever a Solaris process or daemon terminates abnormally.

Limitations of the previous Solaris process core dump features are:

- Process `core` files are placed in their current working directory, and thus all Solaris daemons, which typically `chdir` to the root (/) directory as part of their initialization, overwrite each other's `core` files.

- Many system daemons, such as `statd`, perform `setuid` operations but do not produce core files in the event of a problem, for security reasons.

Configurable Core File Paths

Two new configurable `core` file paths that can be enabled or disabled independently of each other are:

- A per-process core file path, which defaults to `core` and is enabled by default. If enabled, the per-process core file path causes a `core` file to be produced when the process terminates abnormally. The per-process path is inherited by a new process from its parent process.

  When generated, a per-process core file is owned by the owner of the process with read/write permissions for the owner. Only the owning user can view this file.

- A global core file path, which defaults to `core` and is disabled by default. If enabled, an *additional* core file with the same content as the per-process core file is produced by using the global core file path.

  When generated, a global core file is owned by superuser with read/write permissions for superuser only. Non-privileged users cannot view this file.

When a process terminates abnormally, it produces a `core` file in the current directory as in previous Solaris releases. But if the global core file path is enabled and set to `/corefiles/core`, for example, then each process that expires produce *two* core files: one in the current working directory and one in the `/corefiles` directory.

By default, a `setuid` process does not produce core files using either the global or per-process path.

Expanded Core File Names

If a global core file directory is enabled, `core` files can be distinguished from one another by using the variables described in the following table.

Variable Name	Variable Definition
`%p`	Process ID
`%u`	Effective user ID
`%g`	Effective group ID
`%f`	Executable file name
`%n`	System node name, equivalent to the `uname -n` output
`%m`	Machine name, equivalent to the `uname -m` output
`%t`	Decimal value of time(2) system call
`%%`	Literal %

For example, if the global core file path is set to:

`/var/core/core.%f.%p`

and a `sendmail` process with PID `12345` terminates abnormally, it produces the following `core` file:

`/var/core/core.sendmail.12345`

Setting the Core File Name Pattern

You can set a core file name pattern on a global basis or a per-process basis, and you can specify whether you want these settings saved across a system reboot.

For example, the following `coreadm` command sets the global core file pattern for all processes started by the `init` process. This pattern will persist across system reboots.

`$ `**`coreadm -i /var/core/core.%f.%p`**

Global core values are stored in the `/etc/coreadm.conf` file, which means these settings are saved across a system reboot.

This `coreadm` command sets the per-process core file name pattern for all processes:

`$ `**`coreadm -p /var/core/core.%f.%p $$`**

The $$ symbols represent a placeholder for the process ID of the currently running shell. The per-process core file name pattern is inherited by all child processes.

Once a global or per-process core file name pattern is set, it must be enabled with the `coreadm -e` command. See the procedures below for more information.

You can set the core file name pattern for all processes run during a user's login session by putting the command in a user's `$HOME/.profile` or `.login` file.

Enabling `setuid` Programs to Produce Core Files

You can use the `coreadm` command to enable or disable `setuid` programs to produce core files for all system processes or on a per-process basis by setting the following paths:

- If the global `setuid` option is enabled, a global core file path allows all `setuid` programs on a system to produce `core` files.
- If the per-process `setuid` option is enable, a per-process core file path allows specific `setuid` processes to produce `core` files.

By default, both flags are disabled. For security reasons, the global core file path must be a full pathname, starting with a leading `/`. If superuser disables per-process core files, individual users cannot obtain core files.

The `setuid` core files are owned by superuser with read/write permissions for superuser only. Ordinary users cannot access them even if the process that produced the `setuid` core file was owned by an ordinary user.

See `coreadm`(1M) for more information.

How to Display the Current Core Dump Configuration

Use the `coreadm` command without any options to display the current core dump configuration.

```
$ coreadm
            global core file pattern: /var/core/core.%f.%p
              init core file pattern: core
                   global core dumps: enabled
              per-process core dumps: enabled
             global setid core dumps: enabled
        per-process setid core dumps: disabled
             global core dump logging: disabled
```

How to Set a Core File Name Pattern

1. **Determine whether you want to set a per-process or global core file and select one of the following:**

   a. **Set a per-process file name pattern.**

   ```
   # coreadm -p $HOME/corefiles/%f.%p $$
   ```

   b. **Set a global file name pattern.**
   Become superuser first.

   ```
   # coreadm -g /var/corefiles/%f.%p
   ```

▼ How to Enable a Per-Process Core File Path

1. **Become superuser.**

2. **Enable a per-process core file path.**

   ```
   # coreadm -e process
   ```

3. **Display the current process core file path to verify the configuration.**

   ```
   $ coreadm $$
   1180:   /home/kryten/corefiles/%f.%p
   ```

▼ How to Enable a Global Core File Path

1. **Become superuser.**

2. **Enable a global core file path.**

   ```
   # coreadm -e global -g /var/core/core.%f.%p
   ```

3. **Display the current process core file path to verify the configuration.**

   ```
   # coreadm
        global core file pattern: /var/core/core.%f.%p
          init core file pattern: core
                global core dumps: enabled
           per-process core dumps: enabled
          global setid core dumps: disabled
     per-process setid core dumps: disabled
         global core dump logging: disabled
   ```

Troubleshooting Core File Problems

Error Message

```
NOTICE: 'set allow_setid_core = 1' in /etc/system is obsolete
NOTICE: Use the coreadm command instead of 'allow_setid_core'
```

Cause
  You have an obsolete parameter that allows `setuid` core files in your
  `/etc/system` file.

Solution
  Remove `allow_setid_core=1` from the `/etc/system` file. Then use the
  `coreadm` command to enable global `setuid` core file paths.

Examining Core Files

Some of the proc tools have been enhanced to examine process core files as well as live
processes. The proc tools are utilities that can manipulate features of the `/proc` file
system.

The `/usr/proc/bin/pstack`, `pmap`, `pldd`, `pflags`, and `pcred` tools can now be
applied to core files by specifying the name of the core file on the command line,
similar to the way you specify a process ID to these commands.

For more information on using proc tools to examine core files, see `proc`(1).

Example—Examining Core Files With proc Tools

```
$ ./a.out
Segmentation Fault(coredump)
$ /usr/proc/bin/pstack ./core
core './core' of 19305: ./a.out
 000108c4 main     (1, ffbef5cc, ffbef5d4, 20800, 0, 0) + 1c
 00010880 _start   (0, 0, 0, 0, 0, 0) + b8
```

Managing System Crash Information (Tasks)

This chapter describes how to manage system crash information in the Solaris environment.

For information on the procedures associated with managing system crash information, see "Managing System Crash Information (Task Map)" on page 387.

Managing System Crash Information (Task Map)

The followint table identifies the procedures needed to manage system crash information.

TABLE 28–1 Managing System Crash Information (Task Map)

Task	Description	Instructions
1. Display the current crash dump configuration	Display the current crash dump configuration by using the `dumpadm`command.	"How to Display the Current Crash Dump Configuration" on page 391
2. Modify the crash dump configuration	Use the `dumpadm` command to specify the type of data to dump, whether or not the system will use a dedicated dump device, the directory for saving crash dump files, and the amount of space that must remain available after crash dump files are written.	"How to Modify a Crash Dump Configuration" on page 391
3. Examine a crash dump file	Use the `mdb` command to view crash dump files.	"How to Examine a Crash Dump" on page 393

TABLE 28–1 Managing System Crash Information (Task Map)   *(Continued)*

Task	Description	Instructions
4. Recover from a full crash dump directory	*Optional.* The system crashes but there is no room in the `savecore` directory, and you want to save some critical system crash dump information.	"How to Recover From a Full Crash Dump Directory (Optional)" on page 394
5. Disable or enable the saving of crash dump files	*Optional.* Use the `dumpadm` command to disable or enable the saving the crash dump files. Saving crash dump files is enabled by default.	"How to Disable or Enable Saving Crash Dumps (Optional)" on page 394

System Crashes (Overview)

System crashes can occur due to hardware malfunctions, I/O problems, and software errors. If the system crashes, it will display an error message on the console, and then write a copy of its physical memory to the dump device. The system will then reboot automatically. When the system reboots, the `savecore` command is executed to retrieve the data from the dump device and write the saved crash dump to your `savecore` directory. The saved crash dump files provide invaluable information to your support provider to aid in diagnosing the problem.

System Crash Dump Files

The `savecore` command runs automatically after a system crash to retrieve the crash dump information from the dump device and writes a pair of files called `unix.X` and `vmcore.X`, where *X* identifies the dump sequence number. Together, these files represent the saved system crash dump information.

Crash dump files are sometimes confused with *core* files, which are images of user applications that are written when the application terminates abnormally.

Crash dump files are saved in a predetermined directory, which by default, is `/var/crash/`*hostname*. In previous Solaris releases, crash dump files were overwritten when a system rebooted—unless you manually enabled the system to save the images of physical memory in a crash dump file. Now the saving of crash dump files is enabled by default.

System crash information is managed with the `dumpadm` command. See "The `dumpadm` Command" on page 389 for more information.

Saving Crash Dumps

You can examine the control structures, active tables, memory images of a live or crashed system kernel, and other information about the operation of the kernel by using the mdb utility. Using mdb to its full potential requires a detailed knowledge of the kernel, and is beyond the scope of this manual. See mdb(1M) for more details on using this utility.

Additionally, crash dumps saved by savecore can be useful to send to a customer service representative for analysis of why the system is crashing. If you will be sending crash dump files to a customer service representative, perform the first two tasks listed in "Managing System Crash Information (Task Map)" on page 387.

The dumpadm Command

Use the dumpadm command to manage system crash dump information in the Solaris environment.

■ The dumpadm command, which allows system administrators to configure crash dumps of the operating system. The dumpadm configuration parameters include the dump content, dump device, and the directory in which crash dump files are saved. See "The dumpadm Command" on page 389 for more information about the dumpadm command.

■ Dump data is now stored in compressed format on the dump device. Kernel crash dump images can be as big as 4 Gbytes or more. Compressing the data means faster dumping and less disk space needed for the dump device.

■ Saving crash dump files is run in the background when a dedicated dump device—not the swap area—is part of the dump configuration. This means a booting system does not wait for the savecore command to complete before going to the next step. On large memory systems, the system can be available before savecore completes.

■ System crash dump files, generated by the savecore command, are now saved by default.

■ The savecore -L command is a new feature which enables you to get a crash dump of the live running Solaris operating environment. This command is intended for troubleshooting a running system by taking a snapshot of memory during some bad state—such as a transient performance problem or service outage. If the system is up and you can still run some commands, you can execute the savecore -L to save a snapshot of the system to the dump device, and then immediately write out the crash dump files to your savecore directory. Because the system is still running, you may only use savecore -L if you have configured a dedicated dump device.

The following table describes dumpadm's configuration parameters.

Dump Parameter	Description
dump device	The device that stores dump data temporarily as the system crashes. When the dump device is not the swap area, `savecore` runs in the background, which speeds up the boot process.
savecore directory	The directory that stores system crash dump files.
dump content	Type of data, kernel memory or all of memory, to dump.
minimum free space	Minimum amount of free space required in the `savecore` directory after saving crash dump files. If no minimum free space has been configured, the default is one megabyte.

See dumpadm(1M) for more information.

The dump configuration parameters managed by the `dumpadm` command are stored in the `/etc/dumpadm.conf` file.

Note – Do not `/etc/dumpadm.conf` edit manually. This could result in an inconsistent system dump configuration.

How the `dumpadm` Command Works

During system startup, the `dumpadm` command is invoked by the `/etc/init.d/savecore` script to configure crash dumps parameters based on information in the `/etc/dumpadm.conf` file.

Specifically, it initializes the dump device and the dump content through the `/dev/dump` interface.

After the dump configuration is complete, the `savecore` script looks for the location of the crash dump file directory by parsing the content of `/etc/dumpadm.conf` file. Then, `savecore` is invoked to check for crash dumps. It will also check the content of the `minfree` file in the crash dump directory.

Dump Devices and Volume Managers

Do not configure a dedicated dump device that is under the control of volume management product such as Solaris Volume Manager (SVM) for accessibility and performance reasons. You can keep your swap areas under control of SVM and this is a recommend practice, but keep your dump device separate.

Managing System Crash Dump Information

Keep the following key points in mind when you are working with system crash information:

- You must be superuser to access and manage system crash information.
- Do not disable the option of saving system crash dumps.
- System crash dump files provide an invaluable way to determine what is causing the system to crash.
- Do not remove important system crash information until it has been sent to your customer service representative

▼ How to Display the Current Crash Dump Configuration

1. **Become superuser.**

2. **Display the current crash dump configuration by using the** `dumpadm` **command without any options.**

   ```
   # dumpadm
         Dump content: kernel pages
          Dump device: /dev/dsk/c0t3d0s1 (swap)
   Savecore directory: /var/pluto
     Savecore enabled: yes
   ```

   The above example output means:

   - The dump content is kernel memory pages.
   - Kernel memory will be dumped on a swap device, `/dev/dsk/c0t3d0s1`. You can identify all your swap areas with the `swap -l` command.
   - System crash dump files will be written in the `/var/crash/venus` directory.
   - Saving crash dump files is enabled.

▼ How to Modify a Crash Dump Configuration

1. **Become superuser.**

2. **Identify the current crash dump configuration by using the** dumpadm **command.**

```
# dumpadm
      Dump content: kernel pages
       Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/pluto
  Savecore enabled: yes
```

This the default dump configuration for a system running the Solaris 9 release.

3. **Modify the crash dump configuration by using the** dumpadm **command.**

```
# dumpadm -c content -d dump-device -m nnnk | nnnm | nnn% -n -s savecore-dir
```

-c *content*	Specifies the type of data to dump. Use kernel to dump of all kernel memory or all to dump all of memory. The default dump content is kernel memory.
-d *dump-device*	Specifies the device that stores dump data temporarily as the system crashes. The primary swap device is the default dump device.
-m *nnnk* | *nnnm* | *nnn%*	Specifies the minimum free disk space for saving crash dump files by creating a minfree file in the current savecore directory. This parameter can be specified in kilobytes (nnnk), megabytes (nnnm) or file system size percentage (nnn%). The savecore command consults this file prior to writing the crash dump files. If writing the crash dump files, based on their size, would decrease the amount of free space below the minfree threshold, the dump files are not written and an error message is logged. See "How to Recover From a Full Crash Dump Directory (Optional)" on page 394 for recovering from this scenario.
-n	Specifies that savecore should not be run when the system reboots. This dump configuration is not recommended. If system crash information is written to the swap device, and savecore is not enabled, the crash dump information is overwritten when the system begins to swap.
-s	Specifies an alternate directory for storing crash dump files. The default directory is /var/crash/hostname where hostname is the output of the uname -n command.

Example—Modifying a Crash Dump Configuration

In this example, all of memory is dumped to the dedicated dump device, /dev/dsk/c0t1d0s1, and the minimum free space that must be available after the crash dump files are saved is 10% of the file system space.

```
# dumpadm
      Dump content: kernel pages
```

```
      Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/pluto
  Savecore enabled: yes
 # dumpadm -c all -d /dev/dsk/c0t1d0s1 -m 10%
     Dump content: all pages
      Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
  Savecore enabled: yes
```

▼ How to Examine a Crash Dump

1. **Become superuser.**

2. **Examine a crash dump by using the** mdb **utility.**

   # **/usr/bin/mdb** [**-k**] *crashdump-file*

-k	Specifies kernel debugging mode by assuming the file is an operating system crash dump file.
crashdump-file	Specifies the operating system crash dump file..

3. **Display crash status information.**

   # **/usr/bin/mdb** *file-name*
   > **::status**
      .
      .
      .
   > **::system**
      .
      .
      .

Example—Examining a Crash Dump

The following example shows sample output from the mdb utility, which includes system information and identifies the tunables that are set in this system's /etc/system file.

```
# /usr/bin/mdb -k unix.0
Loading modules: [ unix krtld genunix ip nfs ipc ptm ]
> ::status
debugging crash dump /dev/mem (64-bit) from ozlo
operating system: 5.9 Generic (sun4u)
> ::system
set ufs_ninode=0x9c40 [0t40000]
set ncsize=0x4e20 [0t20000]
```

```
set pt_cnt=0x400 [0t1024]
```

▼ How to Recover From a Full Crash Dump Directory (Optional)

In this scenario, the system crashes but there is no room in the savecore directory, and you want to save some critical system crash dump information.

1. **Log in as superuser after the system reboots.**

2. **Clear out the savecore directory, usually** /var/crash/*hostname*, **by removing existing crash dump files that have already been sent to your service provider. Or, run the** savecore **command and specify an alternate directory that has sufficient disk space. (See the next step.)**

3. **Manually run the** savecore **command and if necessary, specify an alternate savecore directory.**

   ```
   # savecore [ directory ]
   ```

▼ How to Disable or Enable Saving Crash Dumps (Optional)

1. **Become superuser.**

2. **Disable or enable the saving of crash dumps on your system by using the** dumpadm **command.**

Example—Disabling the Saving of Crash Dumps

This example illustrates how to disable the saving of crash dumps on your system.

```
# dumpadm -n
      Dump content: all pages
       Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
  Savecore enabled: no
```

Example—Enabling the Saving of Crash Dumps

This example illustrates how to enable the saving of crash dump on your system.

```
# dumpadm -y
      Dump content: all pages
       Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
  Savecore enabled: yes
```

Troubleshooting Miscellaneous Software Problems (Tasks)

This chapter describes miscellaneous software problems that may occur occasionally and are relatively easy to fix. Troubleshooting miscellaneous software problems includes solving problems that aren't related to a specific software application or topic, such as unsuccessful reboots and full file systems. Resolving these problems are described in the following sections.

This is a list of information in this chapter.

- "What to Do If Rebooting Fails" on page 397
- "What to Do if a System Hangs" on page 401
- "What to Do if a File System Fills Up" on page 402
- "What to Do if File ACLs Are Lost After Copy or Restore" on page 403
- "Troubleshooting Backup Problems" on page 403

What to Do If Rebooting Fails

If the system does not reboot completely, or if it reboots and then crashes again, there may be a software or hardware problem that is preventing the system from booting successfully.

Problem — A System Won't Boot Because ...	How to Fix the Problem
The system can't find `/platform/`uname`-m`/kernel/unix`.	You may need to change the `boot-device` setting in the PROM on a SPARC system. See "SPARC: Booting a System (Tasks)" in *System Administration Guide: Basic Administration* for information on changing the default boot device.

Problem — A System Won't Boot Because ...	How to Fix the Problem
There is no default boot device on an IA system. The message displayed is: `Not a UFS filesystem.`	Boot the system using the Configuration Assistant/Boot diskette and select the disk from which to boot.
There's an invalid entry in the `/etc/passwd` file.	See "Shutting Down a System (Tasks)" in *System Administration Guide: Basic Administration* for information on recovering from an invalid `passwd` file.
There's a hardware problem with a disk or another device.	Check the hardware connections: ■ Make sure the equipment is plugged in. ■ Make sure all the switches are set properly. ■ Look at all the connectors and cables, including the Ethernet cables. ■ If all this fails, turn off the power to the system, wait 10 to 20 seconds, and then turn on the power again.

If none of the above suggestions solve the problem, contact your local service provider.

SPARC: Troubleshooting 64–bit Solaris Boot Problems

After the 64–bit Solaris release is installed on an UltraSPARC system, the 64–bit kernel will be booted automatically unless any of the following conditions are true:

■ A FLASH PROM upgrade may be required on an UltraSPARC system before it can successfully boot the 64–bit kernel. Refer to your hardware manufacturer's documentation to determine whether your UltraSPARC system requires a firmware upgrade.

■ The Open Boot PROM `boot-file` parameter is set to `kernel/unix`. If booting the 64–bit kernel fails and this parameter is set, unset it, and reboot the system.

■ On some UltraSPARC systems, the 64-bit Solaris kernel is not booted by default, even when the system is completely installed with all the 64-bit Solaris components and the correct firmware is installed. Without booting the 64-bit Solaris kernel, 64-bit applications are unable to run.

To find out more about this issue, and how to enable booting the 64-bit Solaris kernel by default, see `boot`(1M).

You can always discover which Solaris kernel the system is currently running by using the `isainfo -kv` command.

```
$ isainfo -kv
64-bit sparcv9 kernel modules
```

This output means the system is running the 64–bit Solaris kernel.

You cannot boot the 64-bit Solaris operating environment on a 32-bit Solaris system.

What to Do if You Forgot Root Password

If you forget the root password and you cannot log into the system, you will have to:

- Stop the system by using the keyboard abort sequence.
- Boot from a boot or install server or from a local CD-ROM.
- Mount the root (/) file system.
- Remove the root password from the /etc/shadow file.
- Reboot the system.
- Log in and set root's password.

This procedure is fully described in "SPARC: Booting a System" in *System Administration Guide: Basic Administration* and "IA: Booting a System" in *System Administration Guide: Basic Administration*.

The examples below describe how to recovery from a forgotten root password on both SPARC and Intel systems.

SPARC: Example—What to Do if You Forgot Root Password

The following example shows how to recover when you forget the root password by booting from the network. This example assumes that the boot server is already available. Be sure to apply a new root password after the system has rebooted.

```
(Use keyboard abort sequence--Press Stop A keys to stop the system)
ok boot net -s
# mount /dev/dsk/c0t3d0s0 /a
# cd /a/etc
# TERM=vt100
# export TERM
# vi shadow
(Remove root's encrypted password string)
# cd /
# umount /a
# init 6
```

IA: Example—What to Do if You Forgot Root Password

The following example shows how to recover when you forget root's password by booting from the network. This example assumes that the boot server is already available. Be sure to apply a new root password after the system has rebooted.

```
(Use keyboard abort sequence to stop the system
Type any key to continue

SunOS Secondary Boot version 3.00


Solaris Intel Platform Edition Booting System

Running Configuration Assistant...
Autobooting from Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a

If the system hardware has changed, or to boot from a different
device, interrupt the autoboot process by pressing ESC.

Press ESCape to interrupt autoboot in 5 seconds.
     .
     .
     .
Boot Solaris

Select one of the identified devices to boot the Solaris kernel and
choose Continue.

To perform optional features, such as modifying the autoboot and property
settings, choose Boot Tasks.

An asterisk (*) indicates the current default boot device.

> To make a selection use the arrow keys, and press Enter to mark it [X].

[X]  NET : DEC 21142/21143 Fast Ethernet
on Board PCI at Dev 3
[ ]  DISK: (*) Target 0, QUANTUM  FIREBALL1280A
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ]  DISK: Target 1:ST5660A
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ]  DISK: Target 0:Maxtor 9 0680D4
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ]  CD  : Target 1:TOSHIBA  CD-ROM XM-5602B  1546
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1

F2_Continue   F3_Back   F4_Boot Tasks   F6_Help
     .
     .
     .
             <<< Current Boot Parameters >>>
```

```
Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a
Boot args: kernel/unix -r

Select the type of installation you want to perform:

1 Solaris Interactive
2 Custom JumpStart
3 Solaris Web Start

Enter the number of your choice followed by <ENTER> the key.

If you enter anything else, or if you wait for 30 seconds,
an interactive installation will be started.

Select type of installation:  b -s
     .
     .
     .
# mount /dev/dsk/c0t0d0s0 /a
     .
     .
     .
# cd /a/etc
# vi shadow
(Remove root's encrypted password string)
# cd /
# umount /a
# init 6
```

What to Do if a System Hangs

A system can freeze or hang rather than crash completely if some software process is stuck. Follow these steps to recover from a hung system.

1. Determine whether the system is running a window environment and follow the suggestions listed below. If these suggestions don't solve the problem, go to step 2.

   ■ Make sure the pointer is in the window where you are typing the commands

   ■ Press Control-q in case the user accidentally pressed Control-s, which freezes the screen. Control-s freezes only the window, not the entire screen. If a window is frozen, try using another window.

   ■ If possible, log in remotely from another system on the network. Use the pgrep command to look for the hung process. If it looks like the window system is hung, identify the process and kill it.

2. Press Control-\ to force a "quit" in the running program and (probably) write out a core file.

3. Press Control-c to interrupt the program that might be running.

4. Log in remotely and attempt to identify and kill the process that is hanging the system.

5. Log in remotely, become superuser and reboot the system.

6. If the system still does not respond, force a crash dump and reboot. See Chapter 28 for information on forcing a crash dump and booting.

7. If the system still does not respond, turn the power off, wait a minute or so, then turn the power back on.

8. If you cannot get the system to respond at all, contact your local service provider for help.

What to Do if a File System Fills Up

When the root (/) file system or any other file system fills up, you will see the following message in the console window:

```
.... file system full
```

There are several reasons why a file system fills up. The following sections describe several scenarios for recovering from a full file system. See Chapter 16 for information on routinely cleaning out old and unused files to prevent full file systems.

File System Fills Up Because a Large File or Directory Was Created

Reason Error Occurred	How to Fix the Problem
Someone accidentally copied a file or directory to the wrong location. This also happens when an application crashes and writes a large core file into the file system.	Log in as superuser and use the `ls -tl` command in the specific file system to identify which large file is newly created and remove it. See "How to Find and Delete core Files" on page 244 to remove core files.

A `TMPFS` File System is Full Because the System Ran Out of Memory

Reason Error Occurred	How to Fix the Problem
This can occur if `TMPFS` is trying to write more than it is allowed or some current processes are using a lot of memory.	See `tmpfs`(7FS) for information on recovering from `tmpfs`-related error messages.

What to Do if File ACLs Are Lost After Copy or Restore

Reason Error Occurred	How to Fix the Problem
If files or directories with ACLs are copied or restored into the `/tmp` directory, the ACL attributes are lost. The `/tmp` directory is usually mounted as a temporary file system, which doesn't support UFS file system attributes such as ACLs.	Copy or restore files into the `/var/tmp` directory instead.

Troubleshooting Backup Problems

This section describes some basic troubleshooting techniques to use when backing up and restoring data.

The root (/) File System Fills Up After You Back Up a File System

You back up a file system, and the root (/) file system fills up. Nothing is written to the media, and the `ufsdump` command prompts you to insert the second volume of media.

Reason Error Occurred	How to Fix the Problem
If you used an invalid destination device name with the -f option, the ufsdump command wrote to a file in the /dev directory of the root (/) file system, filling it up. For example, if you typed /dev/rmt/st0 instead of /dev/rmt/0, the backup file /dev/rmt/st0 was created on the disk rather than being sent to the tape drive.	Use the ls -tl command in the /dev directory to identify which file is newly created and abnormally large, and remove it.

Make Sure the Backup and Restore Commands Match

You can only use ufsrestore to restore files backed up with ufsdump. If you back up with tar, restore with tar. If you use the ufsrestore command to restore a tape that was written with another command, an error message tells you that the tape is not in ufsdump format.

Check to Make Sure You Have the Right Current Directory

It is easy to restore files to the wrong location. Because the ufsdump command always copies files with full path names relative to the root of the file system, you should usually change to the root directory of the file system before running ufsrestore. If you change to a lower-level directory, after you restore the files you will see a complete file tree created under that directory.

Interactive Commands

When you use the interactive command, a ufsrestore> prompt is displayed, as shown in this example:

```
# ufsrestore ivf /dev/rmt/0
Verify volume and initialize maps
Media block size is 126
Dump   date: Thu Aug 30 10:13:46 2001
Dumped from: the epoch
Level 0 dump of /export/home on starbug:/dev/dsk/c0t0d0s7
Label: none
Extract directories from tape
Initialize symbol table.
ufsrestore >
```

At the `ufsrestore>` prompt, you can use the commands listed on "UFS Backup and Restore Commands (Reference)" in *System Administration Guide: Basic Administration* to find files, create a list of files to be restored, and restore them.

Troubleshooting File Access Problems (Tasks)

This is a list of troubleshooting topics in this chapter.

- "Solving Problems With Search Paths (`Command not found`)" on page 407
- "Solving File Access Problems" on page 410
- "Recognizing Problems With Network Access" on page 411

Users frequently experience problems—and call on a system administrator for help—because they cannot access a program, a file, or a directory that they could previously use. Whenever you encounter such a problem, investigate one of three areas:

- The user's search path may have been changed, or the directories in the search path may not be in the proper order.
- The file or directory may not have the proper permissions or ownership.
- The configuration of a system accessed over the network may have changed.

This chapter briefly describes how to recognize problems in each of these three areas and suggests possible solutions.

Solving Problems With Search Paths (`Command not found`)

A message of `Command not found` indicates one of the following:

- The command is not available on the system.
- The command directory is not in the search path.

To fix a search path problem, you need to know the pathname of the directory where the command is stored.

If the wrong version of the command is found, a directory that has a command of the same name is in the search path. In this case, the proper directory may be later in the search path or may not be present at all.

You can display your current search path by using the echo $PATH command.

```
$ echo $PATH
/home/kryten/bin:/sbin:/usr/sbin:/usr/bin:/usr/dt:/usr/dist/exe
```

Use the which command to determine whether you are running the wrong version of the command.

```
$ which maker
/usr/doctools/frame5.1/bin/maker
```

Note – The which command looks in the .cshrc file for path information. The which command may give misleading results if you execute it from the Bourne or Korn shell and you have a .cshrc file than contains aliases for the which command. To ensure accurate results, use the which command in a C shell, or, in the Korn shell, use the whence command.

▼ How to Diagnose and Correct Search Path Problems

1. **Display the current search path to verify that the directory for the command is not in your path or that it isn't misspelled.**

   ```
   $ echo $PATH
   ```

2. **Check the following:**

   - Is the search path correct?
   - Is the search path listed before other search paths where another version of the command is found?
   - Is the command in one of the search paths?

   If the path needs correction, go to step 3. Otherwise, go to step 4.

3. **Add the path to the appropriate file, as shown in this table.**

Shell	File	Syntax	Notes
Bourne and Korn	`$HOME/.profile`	`$` `PATH=`*$HOME/bin:/sbin:/usr/local* */bin ...* `$ export PATH`	A colon separates path names.

C	`$HOME/.cshrc`
	
	 or
	
	 `$HOME/.login`
		hostname% `set path=`(*~bin* */sbin /usr/local/bin ...)*	A blank space separates path names.

4. **Activate the new path as follows:**

Shell	File Where Path Is Located	Activate The Path With ...
Bourne and Korn	`.profile`	`$ . ./.profile`
C	`.cshrc`	*hostname%* `source .cshrc`
	`.login`	*hostname%* `source .login`

5. **Verify the path using the command shown below.**

   `$ ` **`which`** *command*

Example—Diagnosing and Correcting Search Path Problems

This example shows that the `mytool` executable is not in any of the directories in the search path using the `which` command.

```
venus% mytool
mytool: Command not found
venus% which mytool
no mytool in /sbin /usr/sbin /usr/bin /etc /home/ignatz/bin  .
venus% echo $PATH
/sbin /usr/sbin /usr/bin /etc /home/ignatz/bin
venus% vi ~.cshrc
(Add appropriate command directory to the search path)
venus% source .cshrc
venus% mytool
```

If you cannot find a command, look at the man page for its directory path. For example, if you cannot find the `lpsched` command (the `lp` printer daemon), `lpsched`(1M) tells you the path is `/usr/lib/lp/lpsched`.

Solving File Access Problems

When users cannot access files or directories that they previously could access, the permissions or ownership of the files or directories probably has changed.

Changing File and Group Ownerships

Frequently, file and directory ownerships change because someone edited the files as superuser. When you create home directories for new users, be sure to make the user the owner of the dot (`.`) file in the home directory. When users do not own "`.`" they cannot create files in their own home directory.

Access problems can also arise when the group ownership changes or when a group of which a user is a member is deleted from the `/etc/group` database.

See Table 30–1 for information about how to change the permissions or ownership of a file that you are having problems accessing.

TABLE 30–1 Solving File Access Problems

If You Need to Change the ...	Use the ...	For More Details, See ...
Permission on a file	`chmod`(1) command	See "Securing Files (Tasks)" in *System Administration Guide: Security Services* for more information
Ownership of a file	`chown`(1) command	See "Securing Files (Tasks)" in *System Administration Guide: Security Services* for more information.
Group ownership of a file	`chgrp`(1) command	See "Securing Files (Tasks)" in *System Administration Guide: Security Services* for more information

Recognizing Problems With Network Access

If users have problems using the `rcp` remote copy command to copy files over the network, the directories and files on the remote system may have restricted access by setting permissions. Another possible source of trouble is that the remote system and the local system are not configured to allow access.

See "Strategies for NFS Troubleshooting" in *System Administration Guide: Resource Management and Network Services* for information about problems with network access and problems with accessing systems through AutoFS.

Troubleshooting Printing Problems (Tasks)

This chapter explains how to troubleshoot printing problems that may occur when you set up or maintain printing services.

This is a list of step-by-step instructions in this chapter.

- "How to Troubleshoot No Printer Output" on page 419
- "How to Troubleshoot Incorrect Output" on page 425
- "How to Unhang the LP Print Service" on page 429
- "How to Troubleshoot an Idle (Hung) Printer" on page 430
- "How to Resolve Conflicting Printer Status Messages" on page 433

See Chapter 2 for information about printing and the LP print service.

Tips on Troubleshooting Printing Problems

Sometimes after setting up a printer, you find that nothing prints. Or, you may get a little farther in the process: something prints, but it is not what you expect—the output is incorrect or illegible. Then, when you get past these problems, other problems may occur, such as:

- LP commands hanging
- Printers becoming idle
- Users getting conflicting messages

> **Note –** Although many of the suggestions in this chapter are relevant to parallel printers, they are geared toward the more common serial printers.

Troubleshooting No Output (Nothing Prints)

When nothing prints, there are three general areas to check:

- The printer hardware
- The network
- The LP print service

If you get a banner page, but nothing else, this is a special case of incorrect output. See "Troubleshooting Incorrect Output" on page 415.

Check the Hardware

The hardware is the first area to check. As obvious as it sounds, you should make sure that the printer is plugged in and turned on. In addition, you should refer to the manufacturer's documentation for information about hardware settings. Some computers use hardware switches that change the characteristics of a printer port.

The printer hardware includes the printer, the cable that connects it to the computer, and the ports into which the cable plugs at each end. As a general approach, you should work your way from the printer to the computer. Check the printer. Check where the cable connects to the printer. Check the cable. Check where the cable connects to the computer.

Check the Network

Problems are more common with remote print requests—those going from a print client to a print server. You should make sure that network access between the print server and print clients is enabled.

If the network is running the Network Information Service Plus (NIS+), see the *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)* for instructions to enable access between systems. If the network is not running the Network Information Service (NIS) or NIS+, before you set up print servers and print clients, include the Internet address and system name for each client system in the /etc/hosts file on the print server. Also, the Internet address and system name for the print server must be included in the /etc/hosts file of each print client system.

Check the LP Print Service

For printing to work, the LP scheduler must be running on both the print server and print client. If it is not running, you need to start it using the `/usr/lib/lp/lpsched` command. If you have trouble starting the scheduler, see "How to Restart the Print Scheduler" on page 79.

In addition to the scheduler running, a printer must be enabled and accepting requests before it will produce any output. If the LP print service is not accepting requests for a printer, the submitted print requests are rejected. Usually, in that instance, the user receives a warning message after submitting a print request. If the LP print service is not enabled for a printer, print requests remain queued on the system until the printer is enabled.

In general, you should analyze a printing problem as follows:

- Follow the path of the print request step-by-step.
- Examine the status of the LP print service at each step.

  - Is the configuration correct?
  - Is the printer accepting requests?
  - Is the printer enabled to process requests?

- If the request is hanging on transmission, set up `lpr.debug` in `syslog.conf` to display the flow.
- If the request is hanging locally, examine the `lpsched` log (`/var/lp/logs/lpsched`).
- If the request is hanging locally, have notification of the printer device errors (faults) mailed to you, and re-enable the printer.

The procedures found in "Troubleshooting Printing Problems" on page 419 use this strategy to help you troubleshoot various problems with the LP print service.

Troubleshooting Incorrect Output

If the printer and the print service software are not configured correctly, the printer may print, but it may provide output that is not what you expect.

Check the Printer Type and File Content Type

If you used the wrong printer type when you set up the printer with the LP print service, inappropriate printer control characters can be sent to the printer. The results are unpredictable: nothing may print, the output may be illegible, or the output may be printed in the wrong character set or font.

If you specified an incorrect file content type, the banner page may print, but that is all. The file content types specified for a printer indicate the types of files the printer can print directly, without filtering. When a user sends a file to the printer, the file is sent directly to the printer without any attempt to filter it. The problem occurs if the printer cannot handle the file content type.

When setting up print clients, you increase the chance for a mistake because the file content types must be correct on both the print server and the print client. If you set up the print client as recommended with any as the file content type, files are sent directly to the print server and the print server determines the need for filtering. Therefore, the file content types have to be specified correctly only on the server.

You can specify a file content on the print client to off-load filtering from the server to the client, but the content type must be supported on the print server.

Check the `stty` Settings

Many formatting problems can result when the default `stty` (standard terminal) settings do not match the settings required by the printer. The following sections describe what happens when some of the settings are incorrect.

Wrong Baud Settings

When the baud setting of the computer does not match the baud setting of the printer, usually you get some output, but it does not look like the file you submitted for printing. Random characters are displayed, with an unusual mixture of special characters and undesirable spacing. The default for the LP print service is 9600 baud.

Note – If a printer is connected by a parallel port, the baud setting is irrelevant.

Wrong Parity Setting

Some printers use a parity bit to ensure that data received for printing has not been garbled during transmission. The parity bit setting for the computer and the printer must match. If they do not match, some characters either will not be printed at all, or will be replaced by other characters. In this case, the output looks approximately correct; the word spacing is all right and many letters are in their correct place. The LP print service does not set the parity bit by default.

Wrong Tab Settings

If the file contains tabs, but the printer expects no tabs, the printed output may contain the complete contents of the file, but the text may be jammed against the right margin. Also, if the tab settings for the printer are incorrect, the text may not have a left margin, it may run together, it may be concentrated to a portion of the page, or it may be incorrectly double-spaced. The default is for tabs to be set every eight spaces.

Wrong Return Setting

If the output is double-spaced, but it should be single-spaced, either the tab settings for the printer are incorrect or the printer is adding a line feed after each return. The LP print service adds a return before each line feed, so the combination causes two line feeds.

If the print zigzags down the page, the `stty` option `onlcr` that sends a return before every line feed is not set. The `stty=onlcr` option is set by default, but you may have cleared it while trying to solve other printing problems.

Troubleshooting Hung LP Commands

If you type any of the LP commands (such as `lpsystem`, `lpadmin`, or `lpstat`) and nothing happens (no error message, status information, or prompt is displayed), chances are something is wrong with the LP scheduler. Such a problem can usually be resolved by stopping and restarting the LP scheduler. See "How to Stop the Print Scheduler" on page 78 for instructions.

Troubleshooting Idle (Hung) Printers

You may find a printer that is idle, even though it has print requests queued to it. A printer may seem idle when it should not be for one of the following reasons:

- The current print request is being filtered.
- The printer has a fault.
- Networking problems may be interrupting the printing process.

Check the Print Filters

Slow print filters run in the background to avoid tying up the printer. A print request that requires filtering will not print until it has been filtered.

Check Printer Faults

When the LP print service detects a fault, printing resumes automatically, but not immediately. The LP print service waits about five minutes before trying again, and continues trying until a request is printed successfully. You can force a retry immediately by enabling the printer.

Check Network Problems

When printing files over a network, you may encounter the following types of problems:

- Requests sent to print servers may back up in the client system (local) queue.
- Requests sent to print servers may back up in the print server (remote) queue.

Print Requests Backed Up in the Local Queue

Print requests submitted to a print server may back up in the client system queue for the following reasons:

- The print server is down.
- The printer is disabled on the print server.
- The network between the print client and print server is down.
- Underlying network software was not set up properly.

While you are tracking the source of the problem, you should stop new requests from being added to the queue. See "How to Accept or Reject Print Requests for a Printer" on page 94 for more information.

Print Requests Backed Up in the Remote Queue

If print requests back up in the print server queue, the printer has probably been disabled. When a printer is accepting requests, but not processing them, the requests are queued to print. Unless there is a further problem, once the printer is enabled, the print requests in the queue should print.

Troubleshooting Conflicting Status Messages

A user may enter a print request and be notified that the client system has accepted it, then receive mail from the print server that the print request has been rejected. These conflicting messages may occur for the following reasons:

- The print client may be accepting requests, while the print server is rejecting requests.
- The definition of the printer on the print client might not match the definition of that printer on the print server. More specifically, the definitions of the print job components, like filters, character sets, print wheels, or forms are not the same on the client and server systems.

You should check that identical definitions of these job components are registered on both the print clients and print servers so that local users can access printers on the print servers.

Troubleshooting Printing Problems

This section contains step-by-step instructions that explain:

- How to troubleshoot no output
- How to troubleshoot incorrect output
- How to unhang the LP commands
- How to troubleshoot an idle (hung) printer
- How to resolve conflicting status messages

▼ How to Troubleshoot No Printer Output

This task includes the following troubleshooting procedures to try when you submit a print request to a printer and nothing prints:

- Check the hardware ("How to Check the Printer Hardware" on page 420).
- Check the network ("How To Check the Printer Network Connections" on page 421).
- Check the LP print service basic functions ("How to Check the Basic Functions of the LP Print Service" on page 421).
- Check printing from a SunOS 5.9 print client to a SunOS 5.9 print server ("How to Check Printing From a SunOS 5.9 Print Client to a SunOS 5.9 Print Server" on page 424).

Try the first three procedures in the order in which they are listed, before going to the specific print client/server case that applies. However, if the banner page prints, but nothing else does, turn to the instructions under "How to Troubleshoot Incorrect Output" on page 425.

▼ How to Check the Printer Hardware

1. **Check that the printer is plugged in and turned on.**

2. **Check that the cable is connected to the port on the printer and to the port on the system or server.**

3. **Make sure that the cable is the correct cable and that it is not defective.**

   Refer to the manufacturer's documentation. If the printer is connected to a serial port, verify that the cable supports hardware flow control; a NULL modem adapter supports this. The table below shows the pin configuration for NULL modem cables.

   TABLE 31–1 Pin Configuration for NULL Modem Cables

	Host	Printer
Mini-Din-8	25-Pin D-sub	25-Pin D-sub
-	1 (FG)	1(FG)
3(TD)	2(TD)	3(RD)
5(RD)	3(RD)	2(TD)
6(RTS)	4(RTS)	5(CTS)
2(CTS)	5(CTS)	4(RTS)
4(SG)	7(SG)	7(SG)
7(DCD)	6(DSR), 8(DCD)	20(DTR)
1(DTR)	20(DTR)	6(DSR), 8(DCD)

4. **Check that any hardware switches for the ports are set properly.**

   See the printer documentation for the correct settings.

5. **Check that the printer is operational.**

   Use the printer's self-test feature, if the printer has one. Check the printer documentation for information about printer self-testing.

6. **Check that the baud settings for the computer and the printer are correct.**

   If the baud settings are not the same for both the computer and the printer, sometimes nothing will print, but more often you get incorrect output. For instructions, see "How to Troubleshoot Incorrect Output" on page 425.

▼ How To Check the Printer Network Connections

1. **Check that the network link between the print server and the print client is set up correctly.**

   print_client# **ping** *print_server*
   print_server is alive
   print_server# **ping** *print_client*
   print_client not available

   If the message says the system is alive, you know you can reach the system, so the network is all right. The message also tells you that either a name service or the local `/etc/hosts` file has translated the host (system) name you entered into an IP address; otherwise, you would need to enter the IP address.

   If you get a `not available` message, try to answer the following questions: How is NIS or NIS+ set up at your site? Do you need to take additional steps so that print servers and print clients can communicate with one another? If your site is not running NIS or NIS+, have you entered the IP address for the print server in each print client's `/etc/hosts` file, and entered all print client IP addresses in the `/etc/hosts` file of the print server?

2. **(On a SunOS 5.0–5.1 print server only) Check that the** `listen` **port monitor is configured correctly.**

3. **(On a SunOS 5.0–5.1 print server only) Check that the network listen services are registered with the port monitor on the print server.**

▼ How to Check the Basic Functions of the LP Print Service

This procedure uses the printer `luna` as an example of checking basic LP print service functions.

1. **On both the print server and print client, make sure that the LP print service is running.**

   a. **Check whether the LP scheduler is running.**

      # **lpstat -r**
      scheduler is running

   b. **If the scheduler is not running, become superuser or** `lp`**, and start the scheduler.**

      # **/usr/lib/lp/lpsched**

      If you have trouble starting the scheduler, see "How to Unhang the LP Print Service" on page 429.

2. **On both the print server and print client, make sure that the printer is accepting requests.**

   a. **Check that the printer is accepting requests.**

   ```
   # lpstat -a
   mars accepting requests since Jul 12 14:23 2001
   luna not accepting requests since Jul 12 14:23 2001
   unknown reason
   ```

   This command verifies that the LP system is accepting requests for each printer configured for the system.

   b. **If the printer is not accepting requests, become superuser or** lp, **and allow the printer to accept print requests.**

   ```
   # accept luna
   ```

   The specified printer now accepts requests.

3. **On both the print server and print client, make sure that the printer is enabled to print submitted print requests.**

   a. **Check that the printer is enabled.**

   ```
   # lpstat -p luna
   printer luna disabled since Jul 12 14:25 2001.
   available.
   unknown reason
   ```

   This command displays information about printer status. You can omit the printer name to obtain information about all printers set up for the system. The following example shows a printer that is disabled.

   b. **If the printer is disabled, become superuser or** lp, **and enable the printer.**

   ```
   # enable luna
   printer "luna" now enabled.
   ```

   The specified printer is enabled to process print requests.

4. **On the print server, make sure that the printer is connected to the correct serial port.**

   a. **Check that the printer is connected to the correct serial port.**

   ```
   # lpstat -t
   scheduler is running
   system default destination: luna
   device for luna: /dev/term/a
   ```

   The message device for *printer-name* shows the port address. Is the cable connected to the port to which the LP print service says is connected? If the port is correct, skip to step 5.

   b. **Become superuser or** lp.

c. **Change the file ownership of the device file that represents the port.**

   # **chown lp** *device-filename*

   This command assigns the special user `lp` as the owner of the device file. In this command, *device-filename* is the name of the device file.

d. **Change the permissions on the printer port device file.**

   # **chmod 600** *device-filename*

   This command allows only superuser or `lp` to access the printer port device file.

5. **On both the print server and print client, make sure that the printer is configured properly.**

   a. **Check that the printer is configured properly.**

   ```
   # lpstat -p luna -l
   printer luna is idle. enabled since Jul 12 14:24 2001. available
           Content types: postscript
           Printer types: PS
   ```

   The above example shows a PostScript printer that is configured properly, and that is available to process print requests. If the printer type and file content type are correct, skip to step 6.

   b. **If the printer type or file content type is incorrect, try setting the print type to** unknown **and the content type to** any **on the print client.**

   # **lpadmin -p** *printer-name* **-T** *printer-type* **-I** *file-content-type*

6. **On the print server, make sure that the printer is not faulted.**

   a. **Check that the printer is not waiting because of a printer fault.**

   # **lpadmin -p** *printer-name* **-F continue**

   This command instructs the LP print service to continue if it is waiting because of a fault.

   b. **Force an immediate retry by re-enabling the printer.**

   # **enable** *printer-name*

   c. **(Optional) Instruct the LP print service to enable quick notification of printer faults.**

   # **lpadmin -p** *printer-name* **-A 'write root'**

   This command instructs the LP print service to set a default policy of writing root—sending the printer fault message to the terminal on which root is logged in—if the printer fails. This may help you get quick notification of faults as you try to fix the problem.

7. **Make sure that the printer is not set up incorrectly as a login terminal.**

Note – It is easy to mistakenly set up a printer as a login terminal, so be sure to check this possibility even if you think it does not apply.

---

a. **Look for the printer port entry in the** `ps -ef` **command output.**

```
# ps -ef
    root   169   167  0   Apr 04 ?         0:08 /usr/lib/saf/listen tcp
    root   939     1  0 19:30:47 ?         0:02 /usr/lib/lpsched
    root   859   858  0 19:18:54 term/a    0:01 /bin/sh -c \ /etc/lp
/interfaces/luna
luna-294 rocket!smith "passwd\n##
#
```

In the output from this command, look for the printer port entry. In the above example, port `/dev/term/a` is set up incorrectly as a login terminal. You can tell by the `"passwd\n##` information at the end of the line. If the port is set correctly, skip the last steps in this procedure.

b. **Cancel the print request(s).**

```
# cancel request-id
```

In this command, *request-id* is the request ID number for a print request to be canceled.

c. **Set the printer port to be a nonlogin device.**

```
# lpadmin -p printer-name -h
```

d. **Check the** `ps -ef` **command output to verify that the printer port is no longer a login device.**

If you do not find the source of the printing problem in the basic LP print service functions, continue to one of the following procedures for the specific client/server case that applies.

▼ How to Check Printing From a SunOS 5.9 Print Client to a SunOS 5.9 Print Server

1. **Check the basic functions of the LP print service on the print server, if you have not done so already.**

For instructions on checking basic functions, see "How to Check the Basic Functions of the LP Print Service" on page 421. Make sure that the printer works locally before trying to figure out why nothing prints when a request is made from a print client.

2. **Check the basic functions of the LP print service on the print client, if you have not done so already.**

   For instructions on checking basic functions, see "How to Check the Basic Functions of the LP Print Service" on page 421. On the print client, the LP scheduler has to be running, and the printer has to be enabled and accepting requests before any request from the client will print.

   Note – For most of the following steps, you must be logged in as root or `lp`.

   ---

3. **Make sure that the print server is accessible.**

   a. **On the print client, send an "are you there?" request to the print server.**

      print_client# **ping** *print_server*

      If you receive the message *print_server* `not available`, you may have a network problem.

4. **On SunOS 5.1 print client only, make sure that the print server is identified as type `s5` by viewing the Modify Printer window in Admintool.**

5. **Verify that the print server is operating properly.**

   ```
   # lpstat -t luna
   scheduler is running
   system default destination: luna
   device for luna: /dev/term/a
   luna accepting requests since Jul 12 14:26 2001
   printer luna now printing luna-314. enabled since Jul 12 14:26 2001.
   available.
   luna-129            root                488   Jul 12 14:32
   #
   ```

   The above example shows a print server up and running.

6. **If the print server is not operating properly, go back to step 1.**

▼ How to Troubleshoot Incorrect Output

1. **Log in as superuser or `lp`.**

2. **Make sure that the printer type is correct.**

   An incorrect printer type may cause incorrect output. For example, if you specify printer type `PS` and the pages print in reverse order, try printer type `PSR`. (These type names must be in uppercase.) Also, an incorrect printer type may cause missing text, illegible text, or text with the wrong font. To determine the printer type, examine the entries in the `terminfo` database. For information on the structure of the `terminfo` database, see "Printer Type" on page 46.

a. **On the print server, display the printer's characteristics.**

```
$ lpstat -p luna -l
printer luna is idle. enabled since Thu Jul 12 15:02:32 MDT 2001. available.
        Form mounted:
        Content types: postscript
        Printer types: PS
        Description:
        Connection: direct
        Interface: /usr/lib/lp/model/standard
        After fault: continue
        Users allowed:
                (all)
        Forms allowed:
                (none)
        Banner not required
        Character sets:

        Default pitch:
        Default page size: 80 wide 66 long
        Default port settings:
$
```

b. **Consult the printer manufacturer's documentation to determine the printer model.**

c. **If the printer type is not correct, change it with Admintool's Modify Printer option, or use the following** lpadmin **command.**

   # **lpstat -p** *printer-name* **-T** *printer-type*

   On the print client, the printer type should be unknown. On the print server, the printer type must match a terminfo entry that is defined to support the model of printer you have. If there is no terminfo entry for the type of printer you have, see "How to Add a terminfo Entry for an Unsupported Printer" on page 138.

3. **If the banner page prints, but there is no output for the body of the document, check the file content types.**

   File content types specified for a printer indicate the types of files the printer can print directly without filtering. An incorrect file content type causes filtering to be bypassed when it may be needed.

   a. **Note the information on file content type that was supplied in the previous step by the** lpstat **command.**

      On the print client, the file content type should be any, unless you have good reason to specify one or more explicit content types. If a content is specified on the client, filtering is done on the print client, rather than the print server. In addition, content types on the client must match the content types specified on the print server, which in turn must reflect the capabilities of the printer.

b. **Consult your printer manufacturer's documentation to determine which types of files the printer can print directly.**

The names you use to refer to these types of files do not have to match the names used by the manufacturer. However, the names you use must agree with the names used by the filters known to the LP print service.

c. **If the file content type is not correct, change it with Admintool's Modify Printer option, or the following** `lpadmin` **command.**

lpadmin -p *printer-name* **-I** *file-content-type(s)*

Run this command on either the print client, or print server, or both, as needed. Try `-I any` on the print client, and `-I ""` on the print server. The latter specifies a null file content type list, which means an attempt should be made to filter all files, because the printer can directly print only files that exactly match its printer type.

This combination is a good first choice when files are not printing. If it works, you may want to try specifying explicit content types on the print server to reduce unnecessary filtering. For a local PostScript printer, you should use `postscript`, or `postscript,simple`— if the printer supports these types. Be aware that `PS` and `PSR` are not file content types; they are printer types.

If you omit `-I`, the file content list defaults to `simple`. If you use the `-I` option and want to specify file content types in addition to `simple`, `simple` must be included in the list.

When specifying multiple file content types, separate the names with commas. Or you can separate names with spaces and enclose the list in quotation marks. If you specify `any` as the file content type, no filtering will be done and only file types that can be printed directly by the printer should be sent to it.

4. **Check that the print request does not bypass filtering needed to download fonts.**

If a user submits a print request to a PostScript printer with the command `lp -T PS`, no filtering is done. Try submitting the request with the command `lp -T postscript` to force filtering, which may result in the downloading of non-resident fonts needed by the document.

5. **Make sure that the** `stty` **settings for the printer port are correct.**

a. **Read the printer documentation to determine the correct** `stty` **settings for the printer port.**

Note – If a printer is connected to a parallel or USB port, the baud setting is irrelevant.

---

b. **Examine the current settings by using the** `stty` **command.**

```
# stty -a < /dev/term/a
speed 9600 baud;
rows = 0; columns = 0; ypixels = 0; xpixels = 0;
```

```
eucw 1:0:0:0, scrw 1:0:0:0
intr = ^c; quit = ^|; erase = ^?; kill = ^u;
eof = ^d; eol = <undef>; eol2 = <undef>; swtch = <undef>;
start = ^q; stop = ^s; susp = ^z; dsusp = ^y;
rprnt = ^r; flush = ^o; werase = ^w; lnext = ^v;
parenb -parodd cs7 -cstopb -hupcl cread -clocal -loblk -parext
-ignbrk brkint -ignpar -parmrk -inpck istrip -inlcr -igncr icrnl -iuclc
ixon -ixany -ixoff imaxbel
isig icanon -xcase echo echoe echok -echonl -noflsh
-tostop echoctl -echoprt echoke -defecho -flusho -pendin iexten
opost -olcuc onlcr -ocrnl -onocr -onlret -ofill -ofdel tab3
#
```

This command shows the current stty settings for the printer port.

The table below shows the default stty options used by the LP print service's standard printer interface program.

TABLE 31–2 Default stty Settings Used by the Standard Interface Program

Option	Meaning
-9600	Set baud rate to 9600
-cs8	Set 8-bit bytes
-cstopb	Send one stop bit per byte
-parity	Do not generate parity
-ixon	Enable XON/XOFF (also known as START/STOP or DC1/DC3)
-opost	Do "output post-processing" using all the settings that follow in this table
-olcuc	Do not map lowercase to uppercase
-onlcr	Change line feed to carriage return/line feed
-ocrnl	Do not change carriage returns into line feeds
-onocr	Output carriage returns even at column 0
-n10	No delay after line feeds
-cr0	No delay after carriage returns
-tab0	No delay after tabs
-bs0	No delay after backspaces
-vt0	No delay after vertical tabs
-ff0	No delay after form feeds

c. **Change the** `stty` **settings.**

   # **lpadmin -p printer-name -o "stty=** *options* **"**

   Use the table below to choose `stty` options to correct various problems affecting print output.

   TABLE 31–3 `stty` Options to Correct Print Output Problems

`stty` **Values**	**Result**	**Possible Problem From Incorrect Setting**
`110`, `300`, `600`, `1200`, `1800`, `2400`, `4800`, `9600`, `19200`, `38400`	Sets baud rate to the specified value (enter only one baud rate)	Random characters and special characters may be printed and spacing may be inconsistent
`oddp`	Sets odd parity	Missing or incorrect characters appear randomly
`evenp`	Sets even parity	
`-parity`	Sets no parity	
`-tabs`	Sets no tabs	Text is jammed against right margin
`tabs`	Sets tabs every eight spaces	Text has no left margin, is run together, or is jammed together
`-onlcr`	Sets no carriage return at the beginning of line(s)	Incorrect double spacing
`onlcr`	Sets carriage return at beginning of line(s)	The print zigzags down the page

   You can change more than one option setting by enclosing the list of options in single quotation marks and separating each option with spaces. For example, suppose the printer requires you to enable odd parity and set a 7-bit character size. You would type a command similar to that shown in the following example:

   # **lpadmin -p neptune -o "stty='parenb parodd cs7'"**

   The `stty` option `parenb` enables parity checking/generation, `parodd` sets odd parity generation, and `cs7` sets the character size to 7 bits.

6. **Verify that the document prints correctly.**

   # **lp -d** *printer-name* *filename*

▼ How to Unhang the LP Print Service

1. **Log in as superuser or lp.**

2. **Stop the LP print service.**

   # **lpshut**

   If this command hangs, press Control-c and proceed to the next step. If this command succeeds, skip to step 4.

3. **Identify the LP process IDs.**

   ```
   # ps -el | grep lp
      134 term/a   0:01 lpsched
   #
   ```

   Use the process ID numbers (PIDs) from the first column in place of the *pid* variables in the next step.

4. **Stop the LP processes by using the** kill -15 **command.**

   # **kill -15 103 134**

   This should stop the LP print service processes. If the processes do not stop, as a last resort go to step 5.

5. **As a last resort, terminate the processes abruptly.**

   # **kill -9 103 134**

   All the lp processes are terminated.

6. **Remove the** SCHEDLOCK **file so you can restart the LP print service.**

   # **rm /usr/spool/lp/SCHEDLOCK**

7. **Restart the LP print service.**

   # **/usr/lib/lp/lpsched**

   The LP print service should restart. If you are having trouble restarting the scheduler, see "How to Restart the Print Scheduler" on page 79.

▼ How to Troubleshoot an Idle (Hung) Printer

This task includes a number of procedures to use when a printer appears idle but it should not be. It makes sense to try the procedures in order, but the order is not mandatory.

▼ How to Check That the Printer is Ready to Print

1. **Display printer status information.**

   # **lpstat -p** *printer-name*

The information displayed shows you whether the printer is idle or active, enabled or disabled, or available or not accepting print requests. If everything looks all right, continue with other procedures in this section. If you cannot run the `lpstat` command, see "How to Unhang the LP Print Service" on page 429.

2. **If the printer is not available (not accepting requests), allow the printer to accept requests.**

   # **accept** *printer-name*

   The printer begins to accept requests into its print queue.

3. **If the printer is disabled, re-enable it.**

   # **enable** *printer-name*

   This command re-enables the printer so that it will act on the requests in its queue.

▼ How to Check for Print Filtering

Check for print filtering by using the `lpstat -o` command.

```
$ lpstat -o luna
luna-10          fred          1261   Mar 12 17:34 being filtered
luna-11          iggy          1261   Mar 12 17:36 on terra
luna-12          jack          1261   Mar 12 17:39 on terra
$
```

See if the first waiting request is being filtered. If the output looks like the above example, the file is being filtered; the printer is not hung, it just is taking a while to process the request.

▼ How to Resume Printing After a Printer Fault

1. **Look for a message about a printer fault and try to correct the fault if there is one.**

   Depending on how printer fault alerts have been specified, messages may be sent to root by email or written to a terminal on which root is logged in.

2. **Re-enable the printer.**

   # **enable** *printer-name*

   If a request was blocked by a printer fault, this command will force a retry. If this command does not work, continue with other procedures in this section.

▼ How to Send Print Requests to a Remote Printer When They Back Up in the Local Queue

1. On the print client, stop further queuing of print requests to the print server.

```
# reject printer-name
```

2. On the print client, send an "are you there?" request to the print server.

```
print_client# ping print_server
print_server is alive
```

If you receive the message *print_server* `not available`, you may have a network problem.

3. After you fix the above problem, allow new print requests to be queued.

```
# accept printer-name
```

4. If necessary, re-enable the printer.

```
# enable printer-name
```


▼ How to Free Print Requests From a Print Client That Back Up in the Print Server Queue

1. On the print server, stop further queuing of print requests from any print client to the print server.

```
# reject printer-name
```

2. Display the `lpsched` **log file.**

```
# more /var/lp/logs/lpsched
```

The information displayed may help you pinpoint what is preventing the print requests from the print client to the print server from being printed.

3. After you fix the problem, allow new print requests to be queued.

```
# accept printer-name
```

4. If necessary, re-enable the printer on the print server.

```
# enable printer-name
```

▼ How to Resolve Conflicting Printer Status Messages

1. **On the print server, verify the printer is enabled and is accepting requests.**

   # **lpstat -p** *printer-name*

   Users will see conflicting status messages when the print client is accepting requests, but the print server is rejecting requests.

2. **On the print server, check that the definition of the printer on the print client matches the definition of the printer on the print server.**

   # **lpstat -p -l** *printer-name*

   Look at the definitions of the print job components, like print filters, character sets, print wheels, and forms, to be sure they are the same on both the client and server systems so that local users can access printers on print server systems.

Resolving UFS File System Inconsistencies (Tasks)

This is a list of the information in this chapter.

See "Checking File System Integrity (Tasks)" in *System Administration Guide: Basic Administration* for information about the `fsck` program and how to use it to check file system integrity.

`fsck` Error Messages

Normally, `fsck` is run non-interactively to *preen* the file systems after an abrupt system halt in which the latest file system changes were not written to disk. Preening automatically fixes any basic file system inconsistencies and does not try to repair more serious errors. While preening a file system, `fsck` fixes the inconsistencies it expects from such an abrupt halt. For more serious conditions, the command reports the error and terminates.

When you run `fsck` interactively, `fsck` reports each inconsistency found and fixes innocuous errors. However, for more serious errors, the command reports the inconsistency and prompts you to choose a response. When you run `fsck` using the

-y or -n options, your response is predefined as yes or no to the default response suggested by fsck for each error condition.

Some corrective actions will result in some loss of data. The amount and severity of data loss may be determined from the fsck diagnostic output.

fsck is a multipass file system check program. Each pass invokes a different phase of the fsck program with different sets of messages. After initialization, fsck performs successive passes over each file system, checking blocks and sizes, path names, connectivity, reference counts, and the map of free blocks (possibly rebuilding it). It also performs some cleanup.

The phases (passes) performed by the UFS version of fsck are:

- Initialization
- Phase 1 – Check blocks and sizes
- Phase 2 – Check path names
- Phase 3 – Check connectivity
- Phase 4 – Check reference counts
- Phase 5 – Check cylinder groups

The next sections describe the error conditions that may be detected in each phase, the messages and prompts that result, and possible responses you can make.

Messages that may appear in more than one phase are described in "General fsck Error Messages " on page 437. Otherwise, messages are organized alphabetically by the phases in which they occur.

Many of the messages include the abbreviations shown in the table below:

TABLE 32–1 Error Message Abbreviations

Abbreviation	Meaning
BLK	Block number
DUP	Duplicate block number
DIR	Directory name
CG	Cylinder group
MTIME	Time file was last modified
UNREF	Unreferenced

Many of the messages also include variable fields, such as inode numbers, which are represented in this book by an italicized term, such as *inode-number*. For example, this screen message:

```
INCORRECT BLOCK COUNT I=2529
```

is shown as:

```
INCORRECT BLOCK COUNT I=inode-number
```

General `fsck` Error Messages

The error messages in this section may be displayed in any phase after initialization. Although they offer the option to continue, it is generally best to regard them as fatal. They reflect a serious system failure and should be handled immediately. When confronted with such a message, terminate the program by entering n(o). If you cannot determine what caused the problem, contact your local service provider or another qualified person.

```
CANNOT SEEK: BLK block-number (CONTINUE)
```

Cause

A request to move to a specified block number, *block-number*, in the file system failed. This message indicates a serious problem, probably a hardware failure.

If you want to continue the file system check, `fsck` will retry the move and display a list of sector numbers that could not be moved. If the block was part of the virtual memory buffer cache, `fsck` will terminate with a fatal I/O error message.

Action

If the disk is experiencing hardware problems, the problem will persist. Run `fsck` again to recheck the file system.

If the recheck fails, contact your local service provider or another qualified person.

```
CANNOT READ: BLK block-number (CONTINUE)
```

Cause

A request to read a specified block number, *block-number*, in the file system failed. The message indicates a serious problem, probably a hardware failure.

If you want to continue the file system check, `fsck` will retry the read and display a list of sector numbers that could not be read. If the block was part of the virtual memory buffer cache, `fsck` will terminate with a fatal I/O error message. If `fsck` tries to write back one of the blocks on which the read failed, it will display the following message:

```
WRITING ZERO'ED BLOCK sector-numbers TO DISK
```

Action

If the disk is experiencing hardware problems, the problem will persist. Run `fsck` again to recheck the file system. If the recheck fails, contact your local service provider or another qualified person.

```
CANNOT WRITE: BLK block-number (CONTINUE)
```

Cause
    A request to write a specified block number, *block-number*, in the file system failed.

    If you continue the file system check, fsck will retry the write and display a list of
    sector numbers that could not be written. If the block was part of the virtual
    memory buffer cache, fsck will terminate with a fatal I/O error message.

Action
    The disk may be write-protected. Check the write-protect lock on the drive. If the
    disk has hardware problems, the problem will persist. Run fsck again to recheck
    the file system. If the write-protect is not the problem or the recheck fails, contact
    your local service provider or another qualified person.

Initialization Phase `fsck` Messages

In the initialization phase, command-line syntax is checked. Before the file system
check can be performed, fsck sets up tables and opens files.

The messages in this section relate to error conditions resulting from command-line
options, memory requests, the opening of files, the status of files, file system size
checks, and the creation of the scratch file. All such initialization errors terminate fsck
when it is preening the file system.

```
Can't roll the log for device-name.

DISCARDING THE LOG MAY DISCARD PENDING TRANSACTIONS.
DISCARD THE LOG AND CONTINUE?
```

Cause
    fsck was unable to flush the transaction log of a logging UFS file system prior to
    checking the file system for errors.

Action
    Answering yes means the file system operations that were in the log, but had not
    been applied to the file system, are lost. In this case, fsck runs the same checks it
    always runs and asks the following question in phase 5:

```
FREE BLK COUNT(S) WRONG IN SUPERBLK (SALVAGE)
```

    Answering yes at this point reclaims the blocks that were used for the log. The next
    time the file system is mounted with logging enabled, the log will be recreated.

    Answering no preserves the log and exits, but the file system isn't mountable.

```
bad inode number inode-number to ginode
```

Cause
    An internal error occurred because of a nonexistent inode *inode-number*. fsck exits.

Action

Contact your local service provider or another qualified person.

```
cannot alloc size-of-block map bytes for blockmap
cannot alloc size-of-free map bytes for freemap
cannot alloc size-of-state map bytes for statemap
cannot alloc size-of-lncntp bytes for lncntp
```

Cause

Request for memory for its internal tables failed. `fsck` terminates. This message indicates a serious system failure that should be handled immediately. This condition may occur if other processes are using a very large amount of system resources.

Action

Killing other processes may solve the problem. If not, contact your local service provider or another qualified person.

```
Can't open checklist file: filename
```

Cause

The file system checklist file *filename* (usually `/etc/vfstab`) cannot be opened for reading. `fsck` terminates.

Action

Check if the file exists and if its access modes permit read access.

```
Can't open filename
```

Cause

`fsck` cannot open file system *filename*. When running interactively, `fsck` ignores this file system and continues checking the next file system given.

Action

Check to see if read and write access to the raw device file for the file system is permitted.

```
Can't stat root
```

Cause

`fsck` request for statistics about the root directory failed. `fsck` terminates.

Action

This message indicates a serious system failure. Contact your local service provider or another qualified person.

```
Can't stat filename
Can't make sense out of name filename
```

Cause

`fsck` request for statistics about the file system *filename* failed. When running interactively, `fsck` ignores this file system and continues checking the next file system given.

Action

    Check if the file system exists and check its access modes.

filename: (NO WRITE)

Cause

    Either the `-n` option was specified or `fsck` could not open the file system *filename* for writing. When `fsck` is running in no-write mode, all diagnostic messages are displayed, but `fsck` does not attempt to fix anything.

Action

    If `-n` was not specified, check the type of the file specified. It may be the name of a regular file.

IMPOSSIBLE MINFREE=*percent* IN SUPERBLOCK (SET TO DEFAULT)

Cause

    The superblock minimum space percentage is greater than 99 percent or less than 0 percent.

Action

    To set the `minfree` parameter to the default 10 percent, type `y` at the default prompt. To ignore the error condition, type `n` at the default prompt.

filename: BAD SUPER BLOCK: *message*
USE AN ALTERNATE SUPER-BLOCK TO SUPPLY NEEDED INFORMATION;
e.g., fsck[-f ufs] -o b=# [special ...]
where # is the alternate superblock. See fsck_ufs(1M)

Cause

    The superblock has been corrupted.

Action

    One of the following messages may be displayed:

```
CPG OUT OF RANGE
FRAGS PER BLOCK OR FRAGSIZE WRONG
INODES PER GROUP OUT OF RANGE
INOPB NONSENSICAL RELATIVE TO BSIZE
MAGIC NUMBER WRONG
NCG OUT OF RANGE
NCYL IS INCONSISTENT WITH NCG*CPG
NUMBER OF DATA BLOCKS OUT OF RANGE
NUMBER OF DIRECTORIES OUT OF RANGE
ROTATIONAL POSITION TABLE SIZE OUT OF RANGE
SIZE OF CYLINDER GROUP SUMMARY AREA WRONG
SIZE TOO LARGE
BAD VALUES IN SUPERBLOCK
```

    Try to rerun `fsck` with an alternative superblock. Specifying block 32 is a good first choice. You can locate an alternative copy of the superblock by running the `newfs -N` command on the slice. Be sure to specify the `-N` option; otherwise, `newfs` overwrites the existing file system.

```
UNDEFINED OPTIMIZATION IN SUPERBLOCK (SET TO DEFAULT)
```

Cause

The superblock optimization parameter is neither OPT_TIME nor OPT_SPACE.

Action

To minimize the time to perform operations on the file system, type y at the SET TO DEFAULT prompt. To ignore this error condition, type n.

Phase 1: Check Blocks and Sizes Messages

This phase checks the inode list. It reports error conditions encountered while:

- Checking inode types
- Setting up the zero-link-count table
- Examining inode block numbers for bad or duplicate blocks
- Checking inode size
- Checking inode format

All errors in this phase except INCORRECT BLOCK COUNT, PARTIALLY TRUNCATED INODE, PARTIALLY ALLOCATED INODE, and UNKNOWN FILE TYPE terminate fsck when it is preening a file system.

These messages (in alphabetical order) may occur in phase 1:

block-number BAD I=*inode-number*

Cause

Inode *inode-number* contains a block number *block-number* with a number lower than the number of the first data block in the file system or greater than the number of the last block in the file system. This error condition may generate the EXCESSIVE BAD BLKS error message in phase 1 if inode *inode-number* has too many block numbers outside the file system range. This error condition generates the BAD/DUP error message in phases 2 and 4.

Action

N/A

BAD MODE: MAKE IT A FILE?

Cause

The status of a given inode is set to all 1s, indicating file system damage. This message does not indicate physical disk damage, unless it is displayed repeatedly after fsck -y has been run.

Action

Type y to reinitialize the inode to a reasonable value.

BAD STATE *state-number* TO BLKERR

Cause

An internal error has scrambled the fsck state map so that it shows the impossible value *state-number*. fsck exits immediately.

Action

Contact your local service provider or another qualified person.

block-number DUP I=*inode-number*

Cause

Inode *inode-number* contains a block number *block-number*, which is already claimed by the same or another inode. This error condition may generate the EXCESSIVE DUP BLKS error message in phase 1 if inode *inode-number* has too many block numbers claimed by the same or another inode. This error condition invokes phase 1B and generates the BAD/DUP error messages in phases 2 and 4.

Action

N/A

DUP TABLE OVERFLOW (CONTINUE)

Cause

There is no more room in an internal table in fsck containing duplicate block numbers. If the -o p option is specified, the program terminates.

Action

To continue the program, type y at the CONTINUE prompt. When this error occurs, a complete check of the file system is not possible. If another duplicate block is found, this error condition repeats. Increase the amount of virtual memory available (by killing some processes, increasing swap space) and run fsck again to recheck the file system. To terminate the program, type n.

EXCESSIVE BAD BLOCKS I=inode-number (CONTINUE)

Cause

Too many (usually more than 10) blocks have a number lower than the number of the first data block in the file system or greater than the number of the last block in the file system associated with inode *inode-number*. If the -o p (preen) option is specified, the program terminates.

Action

To continue the program, type y at the CONTINUE prompt. When this error occurs, a complete check of the file system is not possible. You should run fsck again to recheck the file system. To terminate the program, type n.

EXCESSIVE DUP BLKS I=inode-number (CONTINUE)

Cause

Too many (usually more than 10) blocks are claimed by the same or another inode or by a free-list. If the -o p option is specified, the program terminates.

Action

To continue the program, type y at the CONTINUE prompt. When this error occurs, a complete check of the file system is not possible. You should run fsck again to recheck the file system. To terminate the program, type n.

INCORRECT BLOCK COUNT I=inode-number (*number-of-BAD-DUP-or-missing-blocks* should be
number-of-blocks-in-filesystem) (CORRECT)

Cause

The block count for inode *inode-number* is *number-of-BAD-DUP-or-missing-blocks*, but should be *number-of-blocks-in-filesystem*. When preening, fsck corrects the count.

Action

To replace the block count of inode *inode-number* by *number-of-blocks-in-filesystem*, type y at the CORRECT prompt. To terminate the program, type n.

LINK COUNT TABLE OVERFLOW (CONTINUE)

Cause

There is no more room in an internal table for fsck containing allocated inodes with a link count of zero. If the -o p (preen) option is specified, the program exits and fsck has to be completed manually.

Action

To continue the program, type y at the CONTINUE prompt. If another allocated inode with a zero-link count is found, this error condition repeats. When this error occurs, a complete check of the file system is not possible. You should run fsck again to recheck the file system. Increase the virtual memory available by killing some processes or increasing swap space, then run fsck again. To terminate the program, type n.

PARTIALLY ALLOCATED INODE I=*inode-number* (CLEAR)

Cause

Inode *inode-number* is neither allocated nor unallocated. If the -o p (preen) option is specified, the inode is cleared.

Action

To deallocate the inode *inode-number* by zeroing out its contents, type y. This may generate the UNALLOCATED error condition in phase 2 for each directory entry pointing to this inode. To ignore the error condition, type n. A no response is appropriate only if you intend to take other measures to fix the problem.

PARTIALLY TRUNCATED INODE I=*inode-number* (SALVAGE)

Cause

fsck has found inode *inode-number* whose size is shorter than the number of blocks allocated to it. This condition occurs only if the system crashes while truncating a file. When preening the file system, fsck completes the truncation to the specified size.

Action

   To complete the truncation to the size specified in the inode, type y at the SALVAGE prompt. To ignore this error condition, type n.

```
UNKNOWN FILE TYPE I=inode-number (CLEAR)
```

Cause

   The mode word of the inode *inode-number* shows that the inode is not a pipe, special character inode, special block inode, regular inode, symbolic link, FIFO file, or directory inode. If the -o p option is specified, the inode is cleared.

Action

   To deallocate the inode *inode-number* by zeroing its contents, which results in the UNALLOCATED error condition in phase 2 for each directory entry pointing to this inode, type y at the CLEAR prompt. To ignore this error condition, type n.

Phase 1B: Rescan for More DUPS Messages

When a duplicate block is found in the file system, this message is displayed:

block-number DUP I=*inode-number*

Cause

   Inode *inode-number* contains a block number *block-number* that is already claimed by the same or another inode. This error condition generates the BAD/DUP error message in phase 2. Inodes that have overlapping blocks may be determined by examining this error condition and the DUP error condition in phase 1.

Action

   When a duplicate block is found, the file system is rescanned to find the inode that previously claimed that block.

Phase 2: Check Path Names Messages

This phase removes directory entries pointing to bad inodes found in phases 1 and 1B. It reports error conditions resulting from:

- Incorrect root inode mode and status
- Directory inode pointers out of range
- Directory entries pointing to bad inodes
- Directory integrity checks

When the file system is being preened (-o p option), all errors in this phase terminate fsck, except those related to directories not being a multiple of the block size, duplicate and bad blocks, inodes out of range, and extraneous hard links.

These messages (in alphabetical order) may occur in phase 2:

```
BAD INODE state-number TO DESCEND
```

Cause

An `fsck` internal error has passed an invalid state *state-number* to the routine that descends the file system directory structure. `fsck` exits.

Action

If this error message is displayed, contact your local service provider or another qualified person.

```
BAD INODE NUMBER FOR '.' I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

Cause

A directory *inode-number* has been found whose inode number for "." does not equal *inode-number*.

Action

To change the inode number for "." to be equal to *inode-number*, type y at the FIX prompt To leave the inode numbers for "." unchanged, type n.

```
BAD INODE NUMBER FOR '..' I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

Cause

A directory *inode-number* has been found whose inode number for ".." does not equal the parent of *inode-number*.

Action

To change the inode number for ".." to be equal to the parent of *inode-number*, type y at the FIX prompt. (Note that ".." in the root inode points to itself.)To leave the inode number for ".." unchanged, type n.

```
BAD RETURN STATE state-number FROM DESCEND
```

Cause

An `fsck` internal error has returned an impossible state *state-number* from the routine that descends the file system directory structure. `fsck` exits.

Action

If this message is displayed, contact your local service provider or another qualified person.

```
BAD STATE state-number FOR ROOT INODE
```

Cause

An internal error has assigned an impossible state *state-number* to the root inode. `fsck` exits.

Action

If this error message is displayed, contact your local service provider or another qualified person.

```
BAD STATE state-number FOR INODE=inode-number
```

Cause

An internal error has assigned an impossible state *state-number* to inode *inode-number*. `fsck` exits.

Action

If this error message is displayed, contact your local service provider or another qualified person.

`DIRECTORY TOO SHORT I=`*inode-number* `OWNER=`*UID* `MODE=`*file-mode*
`SIZE=`*file-size* `MTIME=`*modification-time* `DIR=`*filename* `(FIX)`

Cause

A directory *filename* has been found whose size *file-size* is less than the minimum directory size. The owner *UID*, mode *file-mode*, size *file-size*, modify time *modification-time*, and directory name *filename* are displayed.

Action

To increase the size of the directory to the minimum directory size, type `y` at the `FIX` prompt. To ignore this directory, type `n`.

`DIRECTORY` *filename*`: LENGTH` *file-size* `NOT MULTIPLE OF` *block-number* `(ADJUST)`

Cause

A directory *filename* has been found with size *file-size* that is not a multiple of the directory block size *block-number*.

Action

To round up the length to the appropriate block size, type `y`. When preening the file system (`-o p` option), `fsck` only displays a warning and adjusts the directory. To ignore this condition, type `n`.

`DIRECTORY CORRUPTED I=`*inode-number* `OWNER=`*UID* `MODE=`*file-mode*
`SIZE=`*file-size* `MTIME=`*modification-time* `DIR=`*filename* `(SALVAGE)`

Cause

A directory with an inconsistent internal state has been found.

Action

To throw away all entries up to the next directory boundary (usually a 512-byte boundary), type `y` at the `SALVAGE` prompt. This drastic action can throw away up to 42 entries. Take this action only after other recovery efforts have failed. To skip to the next directory boundary and resume reading, but not modify the directory, type `n`.

`DUP/BAD I=`*inode-number* `OWNER=O MODE=M SIZE=`*file-size*
`MTIME=`*modification-time* `TYPE=`*filename* `(REMOVE)`

Cause

Phase 1 or phase 1B found duplicate blocks or bad blocks associated with directory or file entry *filename*, inode *inode-number*. The owner *UID*, mode *file-mode*, size *file-size*, modification time *modification-time*, and directory or file name *filename* are displayed. If the `-p` (preen) option is specified, the duplicate/bad blocks are removed.

Action

To remove the directory or file entry *filename*, type y at the REMOVE prompt. To ignore this error condition, type n.

DUPS/BAD IN ROOT INODE (REALLOCATE)

Cause

Phase 1 or phase 1B has found duplicate blocks or bad blocks in the root inode (usually inode number 2) of the file system.

Action

To clear the existing contents of the root inode and reallocate it, type y at the REALLOCATE prompt. The files and directories usually found in the root inode will be recovered in phase 3 and put into the lost+found directory. If the attempt to allocate the root fails, fsck will exit with: CANNOT ALLOCATE ROOT INODE. Type n to get the CONTINUE prompt. Type: y to respond to the CONTINUE prompt, and ignore the DUPS/BAD error condition in the root inode and continue running the file system check. If the root inode is not correct, this may generate many other error messages. Type n to terminate the program.

EXTRA '.' ENTRY I=*inode-number* OWNER=*UID* MODE=*file-mode*
SIZE=*file-size* MTIME=*modification-time* DIR=*filename* (FIX)

Cause

A directory *inode-number* has been found that has more than one entry for ".".

Action

To remove the extra entry for "." type y at the FIX prompt. To leave the directory unchanged, type n.

EXTRA '..' ENTRY I=*inode-number* OWNER=*UID* MODE=*file-mode*
SIZE=*file-size* MTIME=*modification-time* DIR=*filename*(FIX)

Cause

A directory *inode-number* has been found that has more than one entry for ".." (the parent directory).

Action

To remove the extra entry for '..' (the parent directory), type y at the FIX prompt. To leave the directory unchanged, type n.

hard-link-number IS AN EXTRANEOUS HARD LINK TO A DIRECTORY *filename* (REMOVE)

Cause

fsck has found an extraneous hard link *hard-link-number* to a directory *filename*. When preening (-o p option), fsck ignores the extraneous hard links.

Action

To delete the extraneous entry *hard-link-number* type y at the REMOVE prompt. To ignore the error condition, type n.

inode-number OUT OF RANGE I=*inode-number* NAME=*filename* (REMOVE)

Cause

A directory entry *filename* has an inode number *inode-number* that is greater than the end of the inode list. If the -p (preen) option is specified, the inode will be removed automatically.

Action

To delete the directory entry *filename* type y at the REMOVE prompt. To ignore the error condition, type n.

MISSING '.' I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
MTIME=*modification-time* DIR=*filename* (FIX)

Cause

A directory *inode-number* has been found whose first entry (the entry for ".") is unallocated.

Action

To build an entry for "." with inode number equal to *inode-number*, type y at the FIX prompt. To leave the directory unchanged, type n.

MISSING '.' I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
MTIME=*modification-time* DIR=*filename* CANNOT FIX, FIRST ENTRY IN
DIRECTORY CONTAINS *filename*

Cause

A directory *inode-number* has been found whose first entry is *filename*. fsck cannot resolve this problem.

Action

If this error message is displayed, contact your local service provider or another qualified person.

MISSING '.' I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
MTIME=*modification-time* DIR=*filename* CANNOT FIX, INSUFFICIENT
SPACE TO ADD '.'

Cause

A directory *inode-number* has been found whose first entry is not ".". fsck cannot resolve the problem.

Action

If this error message is displayed, contact your local service provider or another qualified person.

MISSING '..' I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
MTIME=*modification-time* DIR=*filename* (FIX)

Cause

A directory *inode-number* has been found whose second entry is unallocated.

Action

To build an entry for ".." with inode number equal to the parent of *inode-number*, type y at the FIX prompt. (Note that ".." in the root inode points to itself.) To leave the directory unchanged, type n.

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, SECOND ENTRY IN
DIRECTORY CONTAINS filename
```

Cause

A directory *inode-number* has been found whose second entry is *filename*. `fsck` cannot resolve this problem.

Action

If this error message is displayed, contact your local service provider or another qualified person.

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, INSUFFICIENT SPACE
TO ADD '..'
```

Cause

A directory *inode-number* has been found whose second entry is not "`..`" (the parent directory). `fsck` cannot resolve this problem.

Action

If this error message is displayed, contact your local service provider or another qualified person.

```
NAME TOO LONG filename
```

Cause

An excessively long path name has been found, which usually indicates loops in the file system name space. This error can occur if a privileged user has made circular links to directories.

Action

Remove the circular links.

```
ROOT INODE UNALLOCATED (ALLOCATE)
```

Cause

The root inode (usually inode number 2) has no allocate-mode bits.

Action

To allocate inode 2 as the root inode, type `y` at the `ALLOCATE` prompt. The files and directories usually found in the root inode will be recovered in phase 3 and put into the `lost+found` directory. If the attempt to allocate the root inode fails, `fsck` displays this message and exits: `CANNOT ALLOCATE ROOT INODE`. To terminate the program, type `n`.

```
ROOT INODE NOT DIRECTORY (REALLOCATE)
```

Cause

The root inode (usually inode number 2) of the file system is not a directory inode.

Action

To clear the existing contents of the root inode and reallocate it, type `y` at the `REALLOCATE` prompt. The files and directories usually found in the root inode will

be recovered in phase 3 and put into the `lost+found` directory. If the attempt to allocate the root inode fails, `fsck` displays this message and exits :`CANNOT ALLOCATE ROOT INODE`. To have `fsck` prompt with FIX, type n.

UNALLOCATED I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
MTIME=*modification-time* type=*filename*(REMOVE)

Cause

A directory or file entry *filename* points to an unallocated inode *inode-number*. The owner *UID*, mode *file-mode*, size *file-size*, modify time *modification-time*, and file name *filename* are displayed.

Action

To delete the directory entry *filename*, type y at the REMOVE prompt. To ignore the error condition, type n.

ZERO LENGTH DIRECTORY I=*inode-number* OWNER=*UID* MODE=*file-mode*
SIZE=*file-size* MTIME=*modification-time* DIR=*filename* (REMOVE)

Cause

A directory entry *filename* has a size *file-size* that is zero. The owner UID, mode *file-mode*, size *file-size*, modify time *modification-time*, and directory name *filename* are displayed.

Action

To remove the directory entry *filename*, type y at the REMOVE prompt. This results in the BAD/DUP error message in phase 4. To ignore the error condition, type n.

Phase 3: Check Connectivity Messages

This phase checks the directories examined in phase 2 and reports error conditions resulting from:

- Unreferenced directories
- Missing or full `lost+found` directories

These messages (in alphabetical order) may occur in phase 3:

BAD INODE *state-number* TO DESCEND

Cause

An internal error has caused an impossible state *state-number* to be passed to the routine that descends the file system directory structure. `fsck` exits.

Action

If this occurs, contact your local service provider or another qualified person.

DIR I=*inode-number1* CONNECTED. PARENT WAS I=*inode-number2*

Cause

This is an advisory message indicating a directory inode *inode-number1* was successfully connected to the `lost+found` directory. The parent inode *inode-number2* of the directory inode *inode-number1* is replaced by the inode number of the `lost+found` directory.

Action

N/A

`DIRECTORY` *filename* `LENGTH` *file-size* `NOT MULTIPLE OF` *block-number* `(ADJUST)`

Cause

A directory *filename* has been found with size *file-size* that is not a multiple of the directory block size B. (This condition can recur in phase 3 if it is not adjusted in phase 2.)

Action

To round up the length to the appropriate block size, type `y` at the `ADJUST` prompt. When preening, `fsck` displays a warning and adjusts the directory. To ignore this error condition, type `n`.

`lost+found IS NOT A DIRECTORY (REALLOCATE)`

Cause

The entry for `lost+found` is not a directory.

Action

To allocate a directory inode and change the `lost+found` directory to reference it, type `y` at the `REALLOCATE` prompt. The previous inode reference by the `lost+found` directory is not cleared and it will either be reclaimed as an unreferenced inode or have its link count adjusted later in this phase. Inability to create a `lost+found` directory displays the message: `SORRY. CANNOT CREATE lost+found DIRECTORY` and aborts the attempt to link up the lost inode, which generates the `UNREF` error message in phase 4. To abort the attempt to link up the lost inode, which generates the `UNREF` error message in phase 4, type `n`.

`NO lost+found DIRECTORY (CREATE)`

Cause

There is no `lost+found` directory in the root directory of the file system. When preening, `fsck` tries to create a `lost+found` directory.

Action

To create a `lost+found` directory in the root of the file system, type `y` at the `CREATE` prompt. This may lead to the message `NO SPACE LEFT IN / (EXPAND)`. If the `lost+found` directory cannot be created, `fsck` displays the message: `SORRY. CANNOT CREATE lost+found DIRECTORY` and aborts the attempt to link up the lost inode. This in turn generates the `UNREF` error message later in phase 4. To abort the attempt to link up the lost inode, type `n`.

`NO SPACE LEFT IN /lost+found (EXPAND)`

Cause

Another entry cannot be added to the `lost+found` directory in the root directory of the file system because no space is available. When preening, `fsck` expands the `lost+found` directory.

Action

To expand the `lost+found` directory to make room for the new entry, type y at the EXPAND prompt. If the attempted expansion fails, `fsck` displays: SORRY. NO SPACE IN `lost+found` DIRECTORY and aborts the request to link a file to the `lost+found` directory. This error generates the UNREF error message later in phase 4. Delete any unnecessary entries in the `lost+found` directory. This error terminates `fsck` when preening is in effect. To abort the attempt to link up the lost inode, type n.

`UNREF DIR I=`*inode-number* `OWNER=`*UID* `MODE=`*file-mode* `SIZE=`*file-size*
`MTIME=`*modification-time* `(RECONNECT)`

Cause

The directory inode *inode-number* was not connected to a directory entry when the file system was traversed. The owner *UID*, mode *file-mode*, size *file-size*, and modification time *modification-time* of directory inode *inode-number* are displayed. When preening, `fsck` reconnects the non-empty directory inode if the directory size is non-zero. Otherwise, `fsck` clears the directory inode.

Action

To reconnect the directory inode *inode-number* into the `lost+found` directory, type y at the RECONNECT prompt. If the directory is successfully reconnected, a CONNECTED message is displayed. Otherwise, one of the `lost+found` error messages is displayed. To ignore this error condition, type n. This error causes the UNREF error condition in phase 4.

Phase 4: Check Reference Counts Messages

This phase checks the link count information obtained in phases 2 and 3. It reports error conditions resulting from:

- Unreferenced files
- A missing or full `lost+found` directory
- Incorrect link counts for files, directories, symbolic links, or special files
- Unreferenced files, symbolic links, and directories
- Bad or duplicate blocks in files and directories
- Incorrect total free-inode counts

All errors in this phase (except running out of space in the `lost+found` directory) are correctable when the file system is being preened.

These messages (in alphabetical order) may occur in phase 4:

BAD/DUP *type* I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
MTIME=*modification-time* (CLEAR)

Cause

   Phase 1 or phase 1B found duplicate blocks or bad blocks associated with file or
   directory inode *inode-number*. The owner *UID*, mode *file-mode*, size *file-size*, and
   modification time *modification-time* of inode *inode-number* are displayed.

Action

   To deallocate inode *inode-number* by zeroing its contents, type y at the CLEAR
   prompt. To ignore this error condition, type n.

(CLEAR)

Cause

   The inode mentioned in the UNREF error message immediately preceding cannot be
   reconnected. This message does not display if the file system is being preened
   because lack of space to reconnect files terminates fsck.

Action

   To deallocate the inode by zeroing out its contents, type y at the CLEAR prompt. To
   ignore the preceding error condition, type n.

LINK COUNT *type* I=*inode-number* OWNER=*UID* MODE=*file-mode*
SIZE=*file-size*
MTIME=*modification-time* COUNT *link-count* SHOULD BE
corrected-link-count (ADJUST)

Cause

   The link count for directory or file inode *inode-number* is *link-count* but should be
   corrected-link-count. The owner *UID*, mode *file-mode*, size *file-size*, and modification
   time *modification-time* of inode *inode-number* are displayed. If the -o p option is
   specified, the link count is adjusted unless the number of references is increasing.
   This condition does not occur unless there is a hardware failure. When the number
   of references is increasing during preening, fsck displays this message and exits:
   LINK COUNT INCREASING

Action

   To replace the link count of directory or file inode *inode-number* with
   corrected-link-count, type y at the ADJUST prompt. To ignore this error condition,
   type n.

lost+found IS NOT A DIRECTORY (REALLOCATE)

Cause

   The entry for lost+found is not a directory.

Action

   To allocate a directory inode and change the lost+found directory to reference it,
   type y at the REALLOCATE prompt. The previous inode reference by the
   lost+found directory is not cleared. It will either be reclaimed as an unreferenced

inode or have its link count adjusted later in this phase. Inability to create a
`lost+found` directory displays this message: `SORRY. CANNOT CREATE`
`lost+found DIRECTORY` and aborts the attempt to link up the lost inode. This
error generates the `UNREF` error message later in phase 4. To abort the attempt to
link up the lost inode, type `n`.

`NO lost+found DIRECTORY (CREATE)`

Cause

There is no `lost+found` directory in the root directory of the file system. When
preening, `fsck` tries to create a `lost+found` directory.

Action

To create a `lost+found` directory in the root of the file system, type `y` at the
`CREATE` prompt. If the `lost+found` directory cannot be created, `fsck` displays the
message: `SORRY. CANNOT CREATE lost+found DIRECTORY` and aborts the
attempt to link up the lost inode. This error in turn generates the `UNREF` error
message later in phase 4. To abort the attempt to link up the lost inode, type `n`.

`NO SPACE LEFT IN / lost+found (EXPAND)`

Cause

There is no space to add another entry to the `lost+found` directory in the root
directory of the file system. When preening, `fsck` expands the `lost+found`
directory.

Action

To expand the `lost+found` directory to make room for the new entry, type `y` at the
`EXPAND` prompt. If the attempted expansion fails, `fsck` displays the message:
`SORRY. NO SPACE IN lost+found DIRECTORY` and aborts the request to link a
file to the `lost+found` directory. This error generates the `UNREF` error message
later in phase 4. Delete any unnecessary entries in the `lost+found` directory. This
error terminates `fsck` when preening (`-o p` option) is in effect. To abort the attempt
to link up the lost inode, type `n`.

`UNREF FILE I=`*inode-number* `OWNER=`*UID* `MODE=`*file-mode* `SIZE=`*file-size*
`MTIME=`*modification-time* `(RECONNECT)`

Cause

File inode *inode-number* was not connected to a directory entry when the file system
was traversed. The owner *UID*, mode *file-mode*, size *file-size*, and modification time
modification-time of inode *inode-number* are displayed. When `fsck` is preening, the
file is cleared if either its size or its link count is zero; otherwise, it is reconnected.

Action

To reconnect inode *inode-number* to the file system in the `lost+found` directory,
type `y`. This error may generate the `lost+found` error message in phase 4 if there
are problems connecting inode *inode-number* to the `lost+found` directory. To ignore
this error condition, type `n`. This error always invokes the `CLEAR` error condition in
phase 4.

UNREF *type* `I=`*inode-number* `OWNER=`*UID* `MODE=`*file-mode* `SIZE=`*file-size*
`MTIME=`*modification-time* `(CLEAR)`

Cause

Inode *inode-number* (whose *type* is directory or file) was not connected to a directory entry when the file system was traversed. The owner *UID*, mode *file-mode*, size *file-size*, and modification time *modification-time* of inode *inode-number* are displayed. When `fsck` is preening, the file is cleared if either its size or its link count is zero; otherwise, it is reconnected.

Action

To deallocate inode *inode-number* by zeroing its contents, type `y` at the `CLEAR` prompt. To ignore this error condition, type `n`.

`ZERO LENGTH DIRECTORY I=`*inode-number* `OWNER=`*UID* `MODE=`*file-mode*
`SIZE=`*file-size* `MTIME=`*modification-time*`(CLEAR)`

Cause

A directory entry *filename* has a size *file-size* that is zero. The owner *UID*, mode *file-mode*, size *file-size*, modification time *modification-time*, and directory name *filename* are displayed.

Action

To deallocate the directory inode *inode-number* by zeroing out its contents, type `y`. To ignore the error condition, type `n`.

Phase 5: Check Cylinder Groups Messages

This phase checks the free-block and used-inode maps. It reports error conditions resulting from:

- Allocated inodes missing from used-inode maps
- Free blocks missing from free-block maps
- Free inodes in the used-inode maps
- Incorrect total free-block count
- Incorrect total used inode count

These messages (in alphabetical order) may occur in phase 5:

`BLK(S) MISSING IN BIT MAPS (SALVAGE)`

Cause

A cylinder group block map is missing some free blocks. During preening, `fsck` reconstructs the maps.

Action

To reconstruct the free-block map, type `y` at the `SALVAGE` prompt. To ignore this error condition, type `n`.

`CG` *character-for-command-option*`: BAD MAGIC NUMBER`

Cause

  The magic number of cylinder group *character-for-command-option* is wrong. This
  error usually indicates that the cylinder group maps have been destroyed. When
  running interactively, the cylinder group is marked as needing reconstruction. `fsck`
  terminates if the file system is being preened.

Action

  If this occurs, contact your local service provider or another qualified person.

```
FREE BLK COUNT(S) WRONG IN SUPERBLK (SALVAGE)
```

Cause

  The actual count of free blocks does not match the count of free blocks in the
  superblock of the file system. If the `-o p` option was specified, the free-block count
  in the superblock is fixed automatically.

Action

  To reconstruct the superblock free-block information, type `y` at the `SALVAGE`
  prompt. To ignore this error condition, type `n`.

```
SUMMARY INFORMATION BAD (SALVAGE)
```

Cause

  The summary information is incorrect. When preening, `fsck` recomputes the
  summary information.

Action

  To reconstruct the summary information, type `y` at the `SALVAGE` prompt. To ignore
  this error condition, type `n`.

Cleanup Phase Messages

Once a file system has been checked, a few cleanup functions are performed. The
cleanup phase displays the following status messages.

number-of `files`, *number-of-files*
`used`, *number-of-files* `free` (*number-of* `frags`, *number-of* `blocks`,
percent `fragmentation`)

This message indicates that the file system checked contains *number-of* files using
number-of fragment-sized blocks, and that there are *number-of* fragment-sized blocks
free in the file system. The numbers in parentheses break the free count down into
number-of free fragments, *number-of* free full-sized blocks, and the *percent*
fragmentation.

```
***** FILE SYSTEM WAS MODIFIED *****
```

This message indicates that the file system was modified by `fsck`. If this file system is
mounted or is the current root (/) file system, reboot. If the file system is mounted,

you may need to unmount it and run `fsck` again; otherwise, the work done by `fsck` may be undone by the in-core copies of tables.

filename `FILE SYSTEM STATE SET TO OKAY`

This message indicates that file system *filename* was marked as stable. Use the `fsck -m` command to determine if the file system needs checking.

filename `FILE SYSTEM STATE NOT SET TO OKAY`

This message indicates that file system *filename* was *not* marked as stable. Use the `fsck -m` command to determine if the file system needs checking.

Troubleshooting Software Administration Problems (Tasks)

This chapter describes problems you may encounter when installing or removing software packages. There are two sections: Specific Software Administration Errors, which describes package installation and administration errors you might encounter, and General Software Administration Problems, which describes behavioral problems that might not result in a particular error message.

This is a list of information in this chapter.

- "Specific Software Administration Errors " on page 460
- "General Software Administration Problems" on page 461

See "Software Administration (Overview)" in *System Administration Guide: Basic Administration* for information about managing software packages.

Troubleshooting Software Package Symbolic Link Problems

In previous Solaris releases, there was no way to specify a symbolic link target in the `pkgmap` file when creating a software package. This meant a package or patch-related symbolic link was always followed to the source of the symbolic link rather than to the target of the symbolic link when a package was added with the `pkgadd` command. This created problems when upgrading a package or a patch package that needed to change a symbolic link target destination to something else.

In this Solaris release, the default behavior is that if a package needs to change the target of a symbolic link to something else, the target of the symbolic link and not the source of the symbolic link is inspected by the `pkgadd` command.

Unfortunately, this means that some packages may or may not conform to the new `pkgadd` behavior.

The `PKG_NONABI_SYMLINKS` environment variable might help you transition between the old and new `pkgadd` symbolic link behaviors. If this environment variable is set to true, `pkgadd` follows the source of the symbolic link.

Setting this variable enables a non-conforming package to revert to the old behavior if set by the administrator before adding a package with the `pkgadd` command

The new `pkgadd` symbolic link behavior might cause an existing package to fail when added with the `pkgadd` command. You might see the following error message in this situation:

```
unable to create symbolic link to <path>
```

If a package doesn't install due to this problem, do the following:

1. If this is a Sun-supplied package, call the Resolution Center and report the non-conforming package name.

2. Set the `PKG_NONABI_SYMLINKS` environment variable and try adding the package with the `pkgadd` command again:

   ```
   # PKG_NONABI_SYMLINKS=true
   # export PKG_NONABI_SYMLINKS
   # pkgadd pkg-name
   ```

Specific Software Administration Errors

```
WARNING: filename <not present on Read Only file system>
```

Reason Error Occurred	How to Fix the Problem
This error message indicates that not all of a package's files could be installed. This usually occurs when you are using `pkgadd` to install a package on a client. In this case, `pkgadd` attempts to install a package on a file system that is mounted from a server, but `pkgadd` doesn't have permission to do so.	If you see this warning message during a package installation, you must also install the package on the server. See "Software Administration (Overview)" in *System Administration Guide: Basic Administration* for details.

General Software Administration Problems

Reason Error Occurred	How to Fix the Problem
There is a known problem with adding or removing some packages developed prior to the Solaris 2.5 release and compatible versions. Sometimes, when adding or removing these packages, the installation fails during user interaction or you are prompted for user interaction and your responses are ignored.	Set the following environment variable and try to add the package again. `NONABI_SCRIPTS=TRUE`

Index

adding *(continued)*
    printer, new
        with Solaris Print Manager, 57
    `terminfo` entry, 135
address space map, 323
adjusting printer port characteristics, 133
administering
    character sets, 33, 105, 113
    fonts, 33, 127, 130
    forms, 33, 118, 127
    print filters, 33, 114, 118
    printers, 73, 103
Admintool
    terminals and modems, 182
Admintool: Serial Ports
    Modify window field descriptions, 186
    prerequisites for running, 190
    starting, 190
alert message priority (for `syslogd`), 375
alerts
    for mounting character sets, 110
    for mounting font cartridges, 108
    for mounting forms, 119, 123
    for mounting print wheels, 108
    for printer faults, 86
alias for selectable character set, 106, 112, 113
alignment pattern
    defining, 156
    printing, 122
    protection of, 120
allow list
    for printer access to forms, 127
    for user access to forms, 126
    for user access to printers, 90
alphanumeric terminal, *See* terminals
application threads, 315, 316
ASCII file
    file content type of, 48
`at` command, 275, 276, 279
    `-l` option (list), 277, 278
    `-m` option (mail), 276, 277
    automatic scheduling of, 267
    controlling access to, 276, 279
        overview, 264
    denying access, 279
    error messages, 280
    overview, 264, 265, 275

`at` job files, 275, 278
    creating, 276, 277
    deleting, 278
    description, 265
    displaying, 277, 278
    displaying queue of, 277
    location of, 265
    submitting, 275
    verifying, 277
`at.deny` file, 276, 279
    description, 264
`atjobs` directory, 267
    description, 264
`atq` command, 277
automatic system activity data collection, 363, 365
automatic system activity reporting, 363, 364
automatic system task execution
    repetitive tasks, 273, 274
    single tasks, 275, 276, 279
automatically turning on quotas, 250
automating system task execution, 264
auxiliary (remote) console, 376

B
banner option, 83
banner pages
    making optional, 82
    nothing else prints, 426
    reasons to turn off, 82
    setting
        with `lpadmin` command, 81, 82
        with `-o nobanner` option, 83
        with Solaris Print Manager, 44
    troubleshooting incorrect output, 415
    turning off, 83
baud settings, 416
bidirectional modem service, 180, 204
billing users, 285, 288, 289, 300
booting
    displaying messages generated during, 373
    running `sadc` command when, 363

C

displaying *(continued)*
  system information
    commands for, 225, 230
  time, 225, 229
  virtual memory with `vmstat`, 338
`dmesg` command, 373
DNLC improvements, 311
`dodisk` script, 283
  caution, 284
  `crontab` entry that runs, 288
  files created by, 284, 285, 296, 305
  overview, 283, 284, 285
download filter, 128, 129
downloaded PostScript fonts, 130
downloading
  fonts, 128, 427
  host-resident fonts, 129
`dtmp` file, 305
`du` command, 239, 241
  displaying size of directories, 239
`dumpadm`
  displaying crash dump configuration, 391
  managing system crash information, 389
  modifying crash dump configuration, 391

E
editing
  `crontab` files, 268, 269
`edquota` command
  changing quotas for individual users, 259
  disabling quotas for individual users, 260, 261
  -p option (prototype), 253
  -t option (time limit), 259
  overview, 250, 251, 258
  setting up user quotas, 252, 253
`enable` command, 94
enabling
  a global core file path with `coreadm`, 385
  a per-process core file path with `coreadm`, 385
  an auxiliary console with `consadm` command, 378
  auxiliary console across system reboots, 379
  printers, 96

enabling *(continued)*
  saving crash dumps, 394
environment variables
  `LPDEST`, 81
  `PRINTER`, 81
error messages
  `at` command, 280
  crash messages, 373
  crash related, 372
  `crontab` command, 274
  customizing logging of, 374
  log file for, 370, 373
  priorities for, 375
  `runacct` script, 291
  sources of, 374
  specifying storage location for, 373, 374
`/etc/acct/holidays` file, 287, 288
`/etc/cron.d/at.deny` file, 276, 279
`/etc/cron.d/cron.allow` file, 272, 273, 274
`/etc/cron.d/cron.deny` file, 272, 273
`/etc/init.d/acct` file, 287
`/etc/init.d/perf` file, 363, 365
`/etc/inittab` file, 202
`/etc/lp` directory, 161
`/etc/lp/classes/printer-class` file, 85
`/etc/lp/default` file, 81
`/etc/lp/fd` directory, 114, 170
`/etc/lp/filter.table` file
  filter deleted from, 116, 114
  filter added in, 116
`/etc/lp/forms` directory, 120
`/etc/lp/forms/form-name` file, 121
`/etc/lp/forms/form-name/alert.sh` file, 123
`/etc/lp/forms/form-name/allow` file, 126
`/etc/lp/forms/form-name/deny` file, 126
`/etc/lp/forms/form-name/describe` file, 120
`/etc/lp/printers` directory, 161
  of print client, 75
  of print server, 75
`/etc/lp/printers/printer-name/alert.sh` file, 87
`/etc/lp/printers/printer-name/comment` file, 80