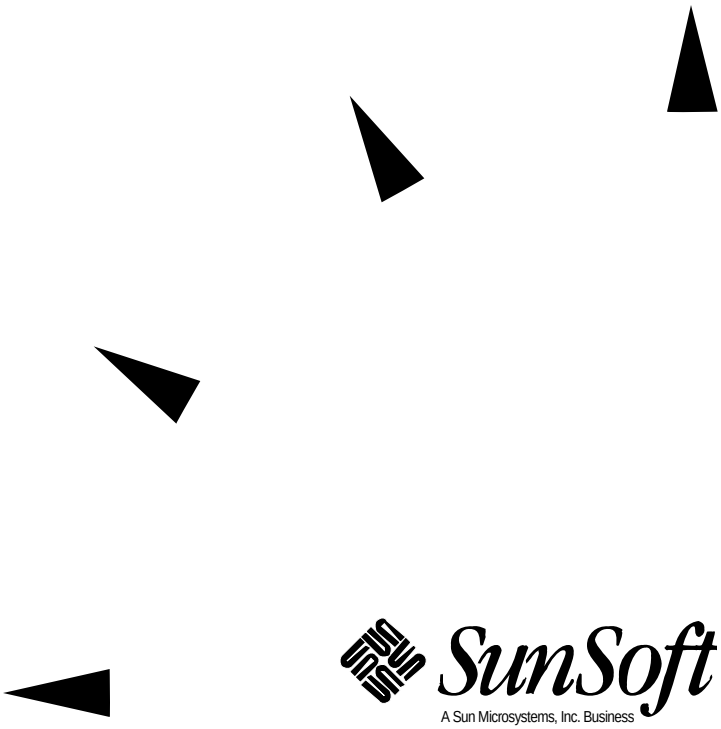
Peripherals Administration



2550 Garcia Avenue
Mountain View, CA 94043
U.S.A.

SunSoft
A Sun Microsystems, Inc. Business

Please
Recycle

Adobe PostScript

Contents

Preface

Peripherals Administration explains how to administer peripheral devices such as modems, disk drives, compact disk read-only memory (CD-ROM) devices, and alphanumeric terminals on an existing Solaris™ 2.x network.

Note – Unless otherwise stated, all the procedures in this book require root or superuser privileges.

Don't Try to Use This Book Just Yet

To minimize the number of pages you must read to get your job done, each chapter has these two or three major sections:

- The *About* section
- The *Instructions* section
- The *Reference* section (where applicable)

The *About* section of a chapter explains the background and concepts you need to do the tasks described in that chapter. Read as much of this section as you need to familiarize yourself with the procedure you need to perform.

If you do not require any background or explanation, skip the *About* section. Use the table on the first page of each chapter to find out where to turn for step-by-step instructions.

The *Instructions* section contains step-by-step directions for each essential administration task. The directions are presented in a “no-frills” manner, without explanations and theory. Turn directly to this section when you already understand the background information.

The *Reference* section contains charts and tables you may need to consult when performing tasks.

Using Cross References in AnswerBook

If you are reading this manual from within the Solaris AnswerBook® document viewer, you can double-click with your mouse on a cross-reference to quickly access the referenced information. (Cross-references appear as text enclosed by boxes.)

Double-clicking on the following boxed words takes you directly to the last page of this Preface. Try it!

“Click Here.”

When you’re ready to return to your original position, click on the “Go Back” button in the document viewer.

Who Should Use This Book

Peripherals Administration is intended for administrators of systems and networks running the Solaris 2.x environment.

Other Books You May Need

Peripheral Documentation

Whenever you install a new peripheral, you will need this book plus any documentation accompanying the peripheral. The peripheral documentation will tell you how to connect the device and how to set switches if necessary to work with your system.

Disk Drives

Though this book covers tasks related to administering hard disks, system disks are special. You generally partition, format, and load software on a system disk when you first install the system.

Refer to these installation guides for more information about these system disk-related procedures

- *SPARC: Installing Solaris Software*
- *x86: Installing Solaris Software*

Other Helpful Books

Other books you may find useful when administering devices and drivers include:

- *File System Administration*
- *Writing Device Drivers*
- *TCP/IP Network Administration Guide*

How This Book Is Organized

This book contains the following information:

Chapter 1, "Terminals and Modems," describes the Serial Port Manager and the Service Access Facility (SAF), which are used to administer modems and terminals in the Solaris 2.x environment. The chapter also provides instructions for setting up terminals and providing dial-in, dial-out, and bidirectional modem support.

Chapter 2, "Disk Drives," explains formatting, labeling, and partitioning hard disk drives. This chapter also explains the various system configurations and gives hints on sizing disk partitions.

Chapter 3, "CD-ROM and Diskette Drives," provides instructions for attaching a compact disc reading device (CD-ROM) to your Solaris 2.x system and provides information on administering CD-ROMs and floppies.

Chapter 4, "Tape Drives," describes how to get a small computer systems interface (SCSI) tape device running on your Solaris 2.x system.

Chapter 5, "Device Drivers," describes how to add a device driver to support certain third-party devices.

Appendix A, "The Service Access Facility," provides extensive detail on the Service Access Facility, the `listen` and `ttymon` port monitors, and administering services.

Appendix B, "Connecting Devices to the Serial Port," provides advice on connecting cables and setting switches for several popular kinds of modems. It also provides general information and troubleshooting tips.

Appendix C, "format Utility," provides reference information on the `format` utility.

What Typographic Changes and Symbols Mean

Table P-1 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. `system% You have mail.`
`AaBbCc123`	What you type, contrasted with on-screen computer output	`system% su` `Password:`
AaBbCc123	Command-line placeholder: replace with a real name or value	To delete a file, type `rm` *filename*.
AaBbCc123	Book titles, new words or terms, or words to be emphasized	Read Chapter 6 in *User's Guide*. These are called *class* options. You *must* be root to do this.
Code samples are included in boxes and may display the following:		
%	C shell prompt	`system%`
#	Superuser prompt, C shell	system#
$	Bourne and Korn shell prompt	$
#	Superuser prompt, Bourne and Korn shells	#

Warnings

An international symbol appears in this book to warn you of any risk to yourself, your equipment or your data. Take care in following the described procedures whenever you see this symbol.

Warning – Be careful—you risk damaging your equipment or losing data if you do not follow the instructions.

Terminals and Modems 1

Terminals and modems provide both local and remote access to system and network resources. Maintaining this access with Serial Port Manager, an Administration Tool application, is an important, if infrequent, responsibility of a system administrator.

Use the following table to find specific information on setting up terminals and modems.

1

About Terminals, Modems, Ports, and Services

This section explains some of the concepts behind modem and terminal management in the Solaris 2.x environment.

About Terminals

A *terminal* consists of an input device, most often a keyboard, and a display device, usually a cathode ray tube (CRT) screen. Often, however, the word "terminal" is used as a shorthand for *alphanumeric terminal*—a serial port device capable of displaying letters, numbers, and other characters such as those produced by a typewriter. The DEC VT-100™ model, for instance, is a popular type of alphanumeric terminal and is often emulated by many other terminals.

Note – 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.

About Modems

A *modem* is a type of device known as data communication equipment. It translates, or *modulates*, digital information into an analog signal that can be transmitted via phone lines.

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 converse. 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, incorporates both dial-in and dial-out capabilities.

About Ports

A *port* is a channel through which a device communicates with the operating system. The most concrete way to think of a port is as a “receptacle” into which a terminal or modem cable may be plugged.

However, a port is not strictly a physical receptacle, but an entity having both 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.

About Serial Ports

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.

Serial Port Adapter Board

When many serial port devices must be connected to a single computer, it may 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.

About 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 configure 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 watches out 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.

There are two types of port monitors included with the Solaris 2.x environment:

- `listen` port monitor—controls access to network services, fielding remote print and file system requests. A common use of the `listen` port monitor is to listen for request from the LP printer service. For more information on the listen port monitor, see Appendix A, "The Service Access Facility." The `listen` port monitor is *not* used when you set up modems and alphanumeric terminals.
- `ttymon` port monitor—provides access to the login services needed by modems and alphanumeric terminals. Serial Port Manager automatically sets up a `ttymon` port monitor to process login requests from these devices.

Note – You may be familiar with an older port monitor called `getty`. The new `ttymon` is more powerful; a single `ttymon` can replace multiple occurrences of `getty`. Otherwise, these two programs serve the same function.

About Serial Port Manager and Service Access Facility

When you configure a serial port for use with a modem or terminal, you have a choice between two tools:

- *Serial Port Manager*—an Administration Tool application that provides a graphical user interface and the functionality of the Service Access Facility's `pmadm` command, a port monitor administration tool.
- *Service Access Facility*—a collection of background processes and administrative commands used from the command line to configure and administer port services and monitors.

The following table highlights some situations when you may choose to use one tool or the other.

Table 1-1 When to Use Serial Port Manager or Service Access Facility

Procedure	Suggested Tool	Comment
Set up terminals and modems	Serial Port Manager	Serial Port Manager quickly sets up typical port services for terminals and modems. Serial Port Manager provides most of the functionality of the `pmadm` command. Exceptions are noted in the following table entries.
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 Serial Port Manager.
Not hanging up a modem 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.

About Serial Port Manager

The Serial Port Manager configures the serial port software to work with terminals and modems by calling the `pmadm` command with the appropriate information. It features:

- Templates for common terminal and modem configurations
- Multiple port setup, modification, or deletion
- Quick visual status of each port

Serial Port Templates

Serial Port Manager provides templates for the most common terminal and modem configurations that you can then modify for a particular device.

- Terminal - Hardwired
- Modem - Dial-in Only
- Modem - Dial-out Only
- Modem - Bidirectional
- Initialize Only - No Connection

Serial Port Manager Security

The system being configured may be your local system or a remote system. Because the Serial Port Manager needs to modify system information, you must have special privileges for that system.

Table 1-2 User Privileges Needed to Run Administration Tool

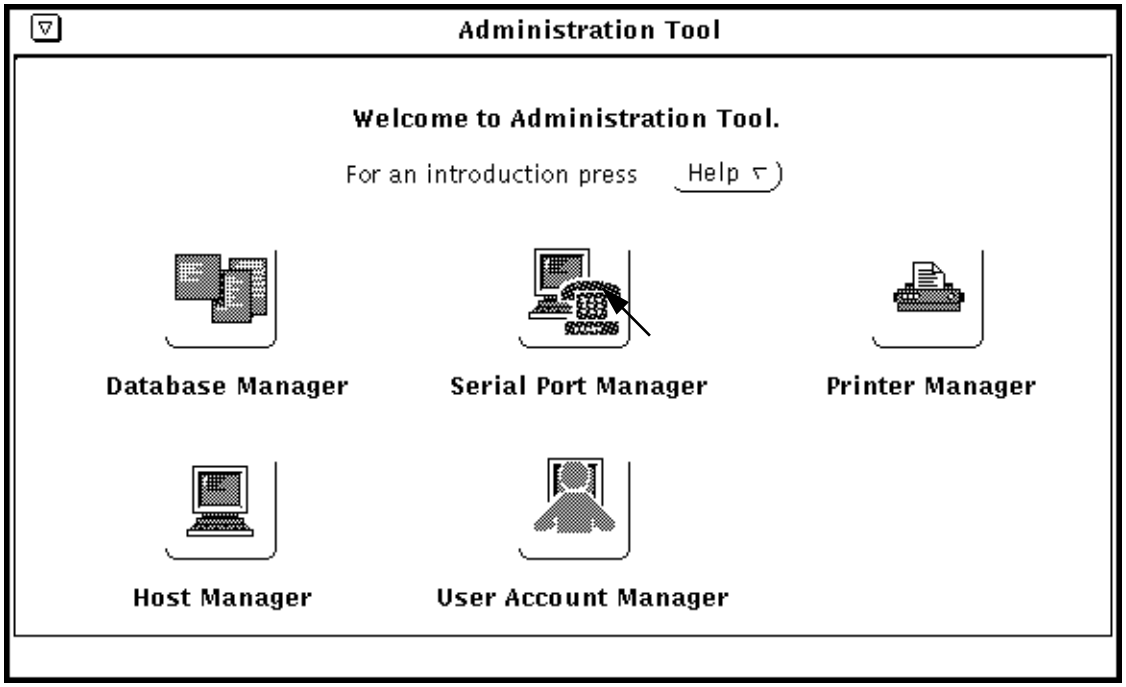
For a...	**You Must...**
Local system	• Run Administration Tool as root, or • Be a member of the UNIX sysadmin group (GID 14) for that system
Remote system	Be a member of the UNIX sysadmin group (GID 14) for that system

Starting Serial Port Manager

Start the Serial Port Manager from the Administration Tool window. If you don’t have Administration Tool running, start it by typing:

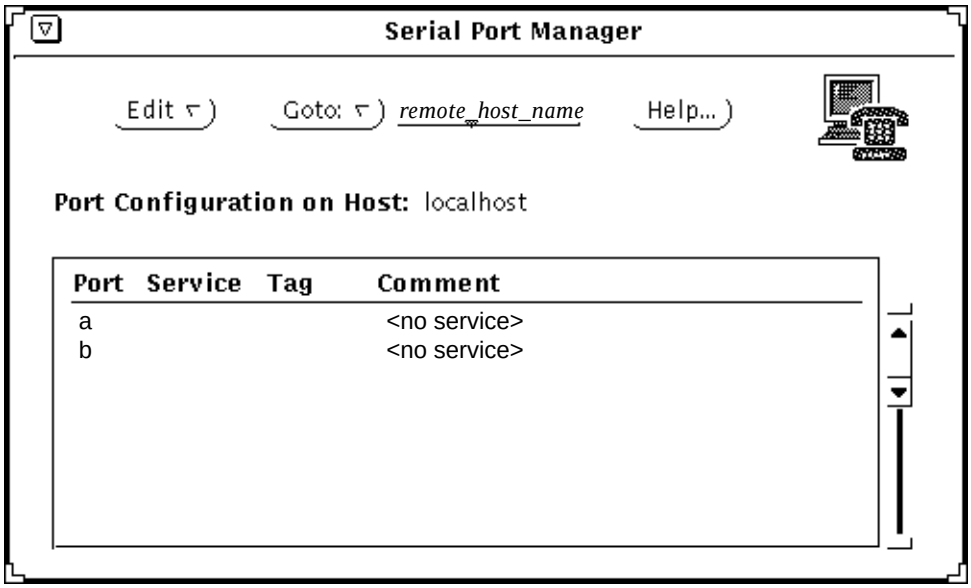
```
/usr/bin/admintool &
```

To start Serial Port Manager, click on the Serial Port Manager icon in the Administration Tool window.



What Serial Port Manager Looks Like

When Serial Port Manager starts, you will see Serial Port Manager main window listing the ports on the local system. You may specify a remote system and view the ports on it by entering the name of the remote system in the Goto: field.



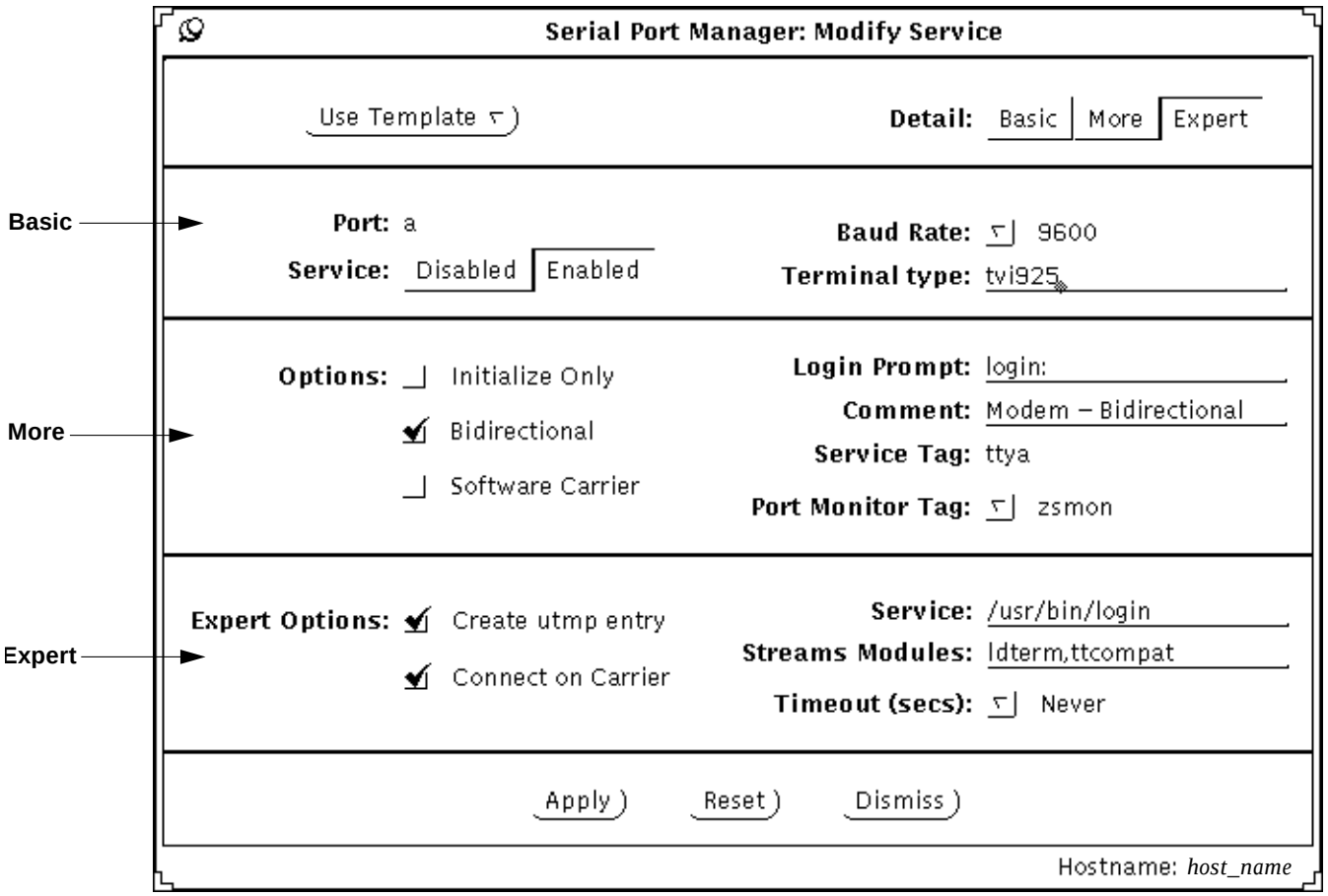
What You Can Do With Serial Port Manager

With the Serial Port Manager, you can

- Initialize a port without configuring the service
- Add a service
- Modify a service
- Disable a service
- Delete a service

These actions are taken by first selecting one or more ports and then choosing an item from the Edit menu. Detailed steps begin on page 11.

When configuring port information, you choose Modify Service... from the Edit menu to bring up the Modify Service 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 Service window will appear in the Basic detail mode. To view More or Expert details, select these from the Detail panel.

The descriptions of each item in the Modify Service window are listed in Table 1-3.

Table 1-3 Modify Service Window Items

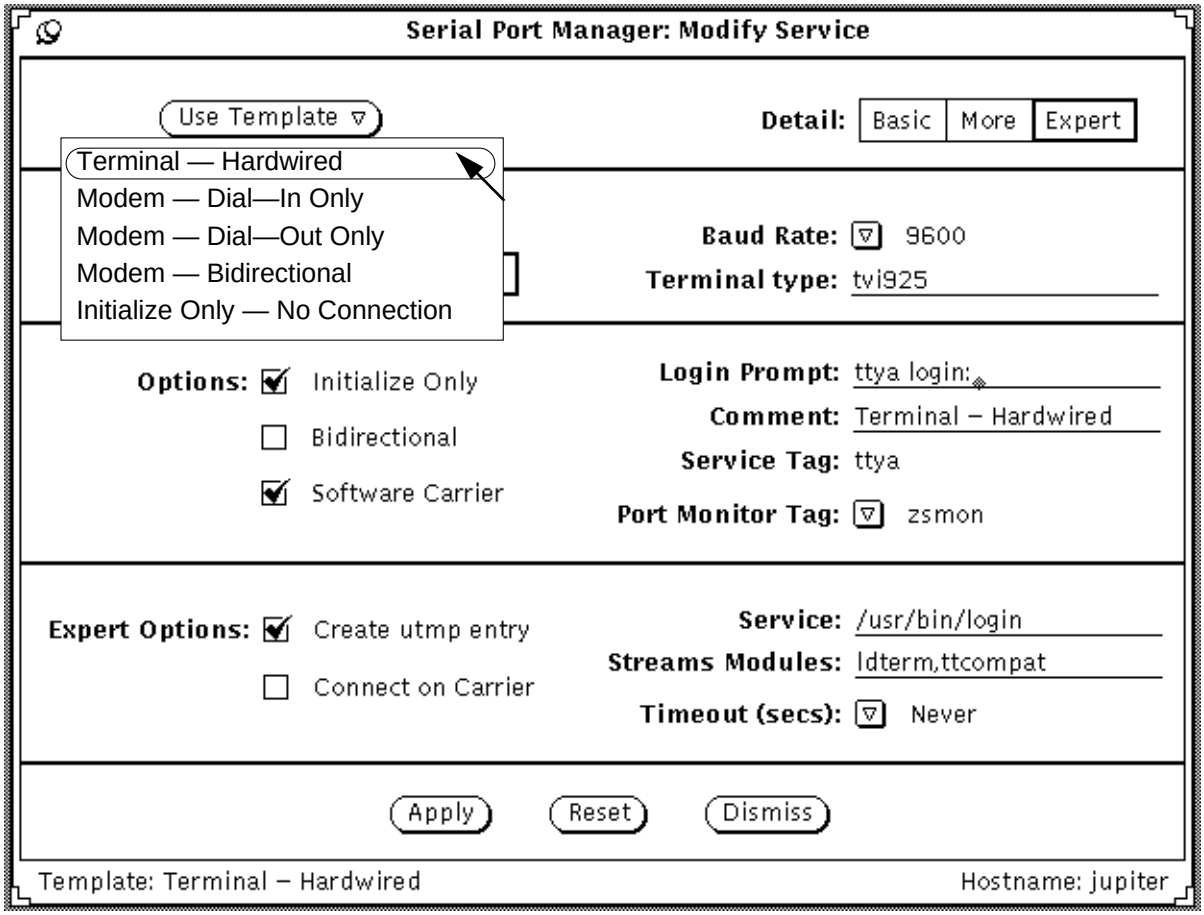
Detail	Item	Description
Basic	Port	Lists the port or ports you selected from Serial Port Manager's main window.
	Service	Specifies that the service for the specified port is turned on (enabled).
	Baud Rate	Specifies the line speed used to communicate with the terminal. The line speed represents an entry in `/etc/ttydefs`.
	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 environment variable `$TERM`.
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 `utmp` entry	Specifies that a `utmp` entry is created in the accounting files upon login. Note: This item must be checked 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.

Instructions for Setting Up Terminals, Modems, and Ports

▼ How to Add Terminals

To add terminals to a system, follow these steps.

1. **Select the port or ports that will be used with a terminal.**
2. **Choose Modify Service from the Edit menu.**
   The Modify Service window appears in the Basic Detail mode. For additional details, select either the More or Expert Detail modes.
3. **Choose Terminal-Hardwired from the Use Template menu.**



The Terminal-Hardwired template provides the following values for the selected port.

Table 1-4 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 `utmp` entry	yes
	Connect on Carrier	no
	Service	`/usr/bin/login`
	Streams Modules	`ldterm,ttcompat`
	Timeout (secs)	Never

4. Change values of template entries if desired.

5. Click on Apply to configure the port.

▼ Command Line Interface for Adding Terminals

The equivalent commands and steps for adding a terminal, as specified by the Terminal - Hardwired default values, are shown in Table 1-5.

Table 1-5 Command Line Interface for Adding Terminals

Step	Instruction	Command to Type
1	Determine the port monitor version number. In this example, the version number is 1.	# **ttyadm -V** 1
2	Type the sacadm command. Substitute the appropriate port monitor tag for *pmtag*. Also substitute the correct version number (as obtained in the previous step) for *vers*.	# **sacadm -a -p** *pmtag* **-t ttymon -c /usr/lib/saf/ttymon -v** *vers*
3	Use the pmadm command to add a login service. Substitute for the arguments as appropriate.	# **pmadm -a -p** *pmtag* **-s** *svctag* **-i root -fu -v** *vers* **-m "'ttyadm -S y -d** *dev_path* **-l** *ttylabel* **-s /usr/bin/login -m ldterm,ttcompat'"**
4	Attach all cords and cables and turn on the terminal. See the documentation accompanying the terminal for connection instructions.	

▼ How to Add Modems

To add a modem to a system, follow these steps.

1. **Select the port or ports that will be used with a modem.**
2. **Choose Modify Service from the Edit menu.**
   The Modify Service window appears in the Basic Detail mode. For additional details, select either the More or Expert Detail modes.
3. **Choose the modem configuration from the Use Template menu that meets or most closely matches your modem service.**
   The modem template choices are described here.

Table 1-6 Modem Templates

Modem Configuration	Description
Modem - Dial-In Only	Users may dial in to the modem but cannot dial out.
Modem - Dial-Out Only	Users may dial out from the modem but cannot dial in.
Modem - Bidirectional	Users may either dial in or out from the modem.

   See Table 1-7 on page 16 for the default values of each template. If a UUCP service will be used to dial in to your modem on a Solaris 2.x system, see "How to Configure a Modem for Use With UUCP" for the rest of the procedure.

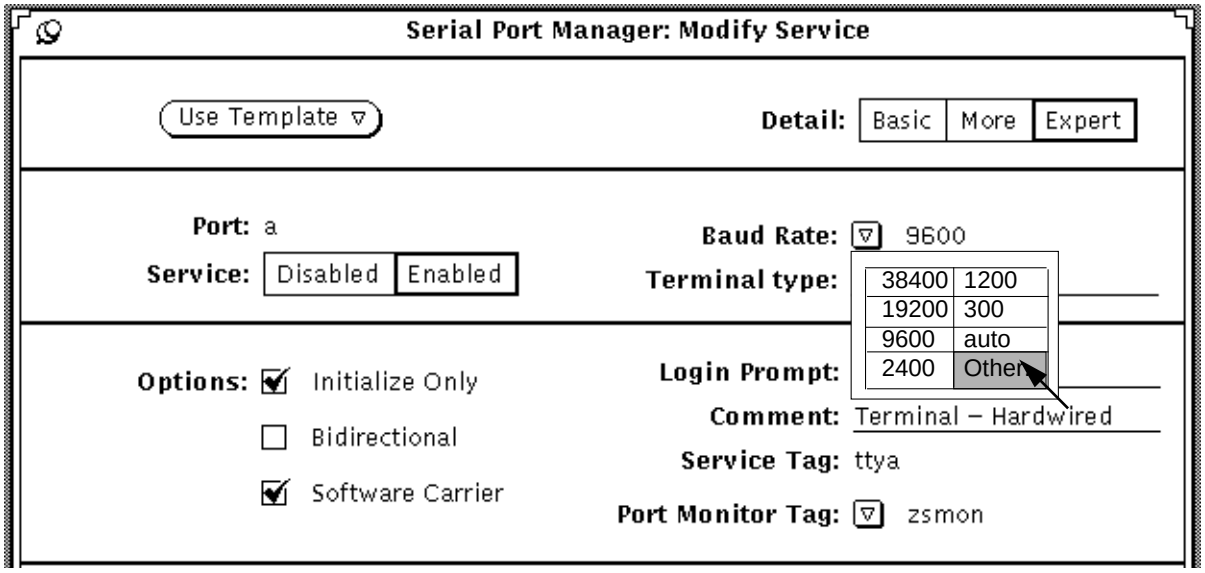
4. **Change values of template entries if desired.**
5. **Click on Apply to configure the port.**

▼ How to Configure a Modem for Use With UUCP

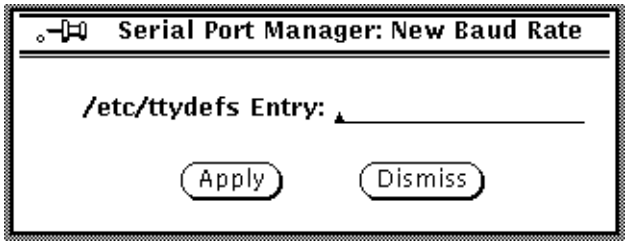
UUCP sends information to a service using seven bits and even parity. Solaris 2.x modem configurations use eight bits and no parity for internationalization purposes. To set up your modem service to work with UUCP, follow these instructions.

1. **Follow Step 1 through Step 3 in the "How to Add Modems" procedure.**

2. **Select the Other item from the Baud Rate menu.**



3. **Enter a baud rate value from the `/etc/ttydefs` file that provides seven bit, even parity service and click on Apply.**



In this example, the 9600E baud rate was selected. This provides a service with a 9600 baud rate, seven bits, and even parity.

4. **Change values of other template entries if desired.**

5. **Click on Apply to configure the port.**

Modem Template Default Values

The Modem templates provides the following values for the selected port.

Table 1-7 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 utmp 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

▼ Command Line Interface for a Dial-In Only Modem

The equivalent commands and steps for adding a dial-in only modem, as specified by the Modem - Dial-In Only default values, are shown here.

Table 1-8 Command Line Interface for Adding a Dial-In Only Modem

Step	Instruction	Command to Type
1	Make sure the modem is properly connected and that the modem switches are set to allow dial-in only service. See the documentation accompanying the modem.	
2	Use the `sacadm` command to display all the port monitors. The comment field indicates what port monitor is associated with each port. Examine the list of port monitors. If the port to which you want to attach the modem already has its own port monitor, skip to step 4.	# **`sacadm -l -t ttymon`**
3	Create a port monitor for the port to be used with the modem.	# **`sacadm -a -p`** *pmtag* **`-t ttymon -c /usr/lib/saf/ttymon -v 'ttyadm -V' -y "`***comment***"**
4	Determine if the port already has a service configured with the `pmadm` command. If it does not, skip to step 6.	# **`pmadm -l -s`** *svctag*
5	If the port already has a service configured, delete it.	# **`pmadm -r -p`** *pmtag* **`-s`** *svctag*
6	Create a new dial-in only port service. Use the port service tag name (listed in the `/dev/term` directory) for the *svctag*.	# **`pmadm -a -p`** *pmtag* **`-s`** *svctag* **`-i root -v 'ttyadm -V' -fu -m "'ttyadm -S n -d`** *dev_path* **`-s /usr/bin/login -l`** *ttylabel* **`-m ldterm,ttcompat'" -y "`***comment* **"**

▼ Command Line Interface for a Dial-Out Only Modem

The equivalent commands and steps for adding a dial-out only modem, as specified by the Modem - Dial-Out Only default values, are shown here.

Table 1-9 Command Line Interface for Adding Dial-Out Only Modems

Step	Instruction	Command to Type
1	Make sure the modem is properly connected and that the modem switches are set to allow dial-out only service. See the documentation accompanying the modem.	
2	Add the modem to the `/etc/uucp/Devices` file. The format to use is: `ACU cua/`*svctag* `-` *speed type* Use the port service tag name (listed in the `/dev/term` directory) for the *svctag*.	# **vi /etc/uucp/Devices**
3	Disable logins.	# **pmadm -d -p** *pmtag* **-s** *svctag*

▼ Command Line Interface for a Bidirectional Modem

The equivalent commands and steps for adding a bidirectional modem, as specified by the Modem - Bidirectional default values, are shown here.

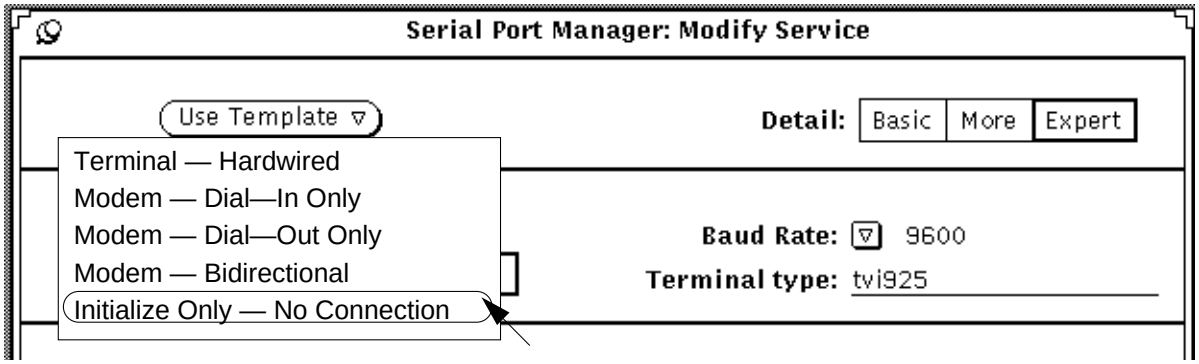
Table 1-10 Command Line Interface for Adding a Bidirectional Modem

Step	Instruction	Command to Type
1	Make sure the modem is properly connected and that the modem switches are set to allow bidirectional service. See the documentation accompanying the modem.	
2	Use the `sacadm` command to display all the port monitors. The comment field indicates what port monitor is associated with each port. Examine the list of port monitors. If the port to which you want to attach the modem already has its own port monitor, skip to step 4.	# **`sacadm -l -t ttymon`**
3	Create a port monitor for the port to be used with the modem.	# **`sacadm -a -p`** *pmtag* **`-t ttymon -c /usr/lib/saf/ttymon -v 'ttyadm -V' -y "`***comment***"**
4	Determine if the port already has a service configured with the `pmadm` command. If it does not, skip to step 6.	# **`pmadm -l -t ttymon`**
5	If the port already has a service configured, delete it.	# **`pmadm -r -p`** *pmtag* **`-s`** *svctag*
6	Create a new bidirectional port service. Use the port service tag name (listed in the `/dev/term` directory) for the *svctag*.	# **`pmadm -a -p`** *pmtag* **`-s`** *svctag* **`-i root -v 'ttyadm -V' -fu -m "'ttyadm -b -S n -d /dev/term/`***svctag* **`-s /usr/bin/login -l`** *ttylabel* **`-m ldterm,ttcompat'" -y "`***comment* **"**
7	Add the modem to the `/etc/uucp/Devices` file. The format to use is: `ACU cua/`*svctag* `-` *speed type* Use the port service tag name (listed in the `/dev/term` directory) for the *svctag*.	# **`vi /etc/uucp/Devices`**

▼ How to Initialize Ports Without Configuring

To initialize ports without configuring for a specific device, follow these steps.

1. **Select the port or ports that you want to initialize.**
2. **Choose Modify Service from the Edit menu.**
   The Modify Service window appears in the Basic Detail mode. For additional details, select either the More or Expert Detail modes.
3. **Choose Initialize Only - No Connection from the Use Template menu.**



The Initialize Only - No Connection template provides the following values for the selected port.

Table 1-11 Initialize Only - No Connection Default Values

Detail	Item	Default Value
Basic	Port	—
	Service	Enabled
	Baud Rate	9600
	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 utmp entry	yes
	Connect on Carrier	no
	Service	`/usr/bin/login`
	Streams Modules	`ldterm,ttcompat`
	Timeout (secs)	Never

4. Click on Apply to initialize the port.

▼ Command Line Interface for Only Initializing a Port

The equivalent command and step for initializing a port without allowing connections, as specified by the Initialize Only - No Connection values, are shown here.

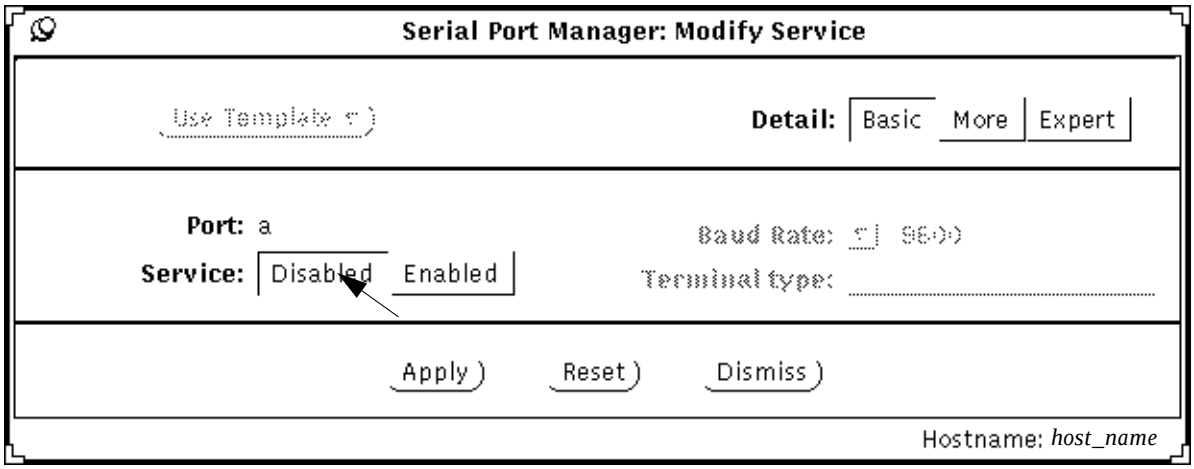
Table 1-12 Command Line Interface for Only Initializing a Port

Step	Instruction	Command to Type
1	Create and initialize a new port service but allow no connections.	# **pmadm -a -p** *pmtag* **-s** *svctag* **-i root -v ‘ttyadm -V‘ -fu -m "‘ttyadm -I -d** *dev_path* **-s /usr/bin/login -l** *ttylabel*‘" **-y** "*comment*"

▼ How to Disable Ports

To disable service on configured ports, follow these steps.

1. **Select the port or ports that you want to disable.**
2. **Choose Modify Service from the Edit menu.**
3. **Select Disable in the Modify Service window.**



4. **Click on Apply to disable the port.**

▼ Command Line Interface for Disabling Ports

The equivalent commands and steps for disabling a port, as specified by the Service: Disabled item in the Modify Service window, are shown here.

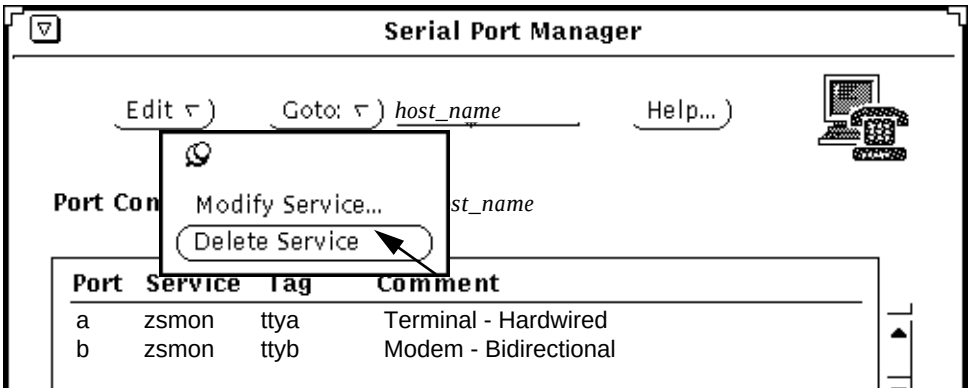
Table 1-13 Command Line Interface for Only Initializing a Port

Step	Instruction	Command to Type
1	Determine the port monitor tag and the port monitor service tag for the port you want to disable.	# **pmadm -l**
2	Disable the port monitor service.	# **pmadm -d -p** *pmtag* **-s** *svctag*

▼ How to Remove Port Services

To delete services on configured ports, follow these steps.

1. **Select the port or ports that has a service you want to delete.**
2. **Choose Delete Service from the Edit menu.**



You are asked if you really want to delete the service for the specified port or ports. You may cancel the delete operation or continue with it.

▼ Command Line Interface for Removing Port Services

The equivalent command for removing a port service, as specified by the Delete Service item in the Edit menu, is shown here.

Table 1-14 Command Line Interface for Removing Port Services

Step	Instruction	Command to Type
1	Delete the port service.	# **pmadm -r -p** *pmtag* **-s** *svctag*

Troubleshooting Tips for Terminals and Modems

See the section “Debugging Hints” in Appendix B, “Connecting Devices to the Serial Port” for more information about troubleshooting serial port devices.

If users are unable to log in over serial port lines after you’ve added a terminal or modem and set up the proper services, consider the following possible causes of failure.

1. Begin by checking 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.

2. 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*.

3. 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 `sacadm` 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.

4. 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

Disk Drives 2

This chapter provides conceptual and procedural information for setting up *secondary* hard disk drives connected to Solaris platforms. For information on administering *system* hard disk drives which contain the system software, see the *Administration Supplement for Solaris Platforms.*

Use the following table to find specific information on administering secondary disk drives.

2

About Preparing Hard Disks for Use

Whenever you add a new hard disk to a system or change the system's configuration, you need to prepare the disk to store and access data. Preparing the disk may require:

- Formatting
- Slice creation (sometimes referred to as partitioning)
- Labeling

About Hard Disks in General

Hard disks consist of several separate disks mounted on a common spindle. Data stored on each disk surface is written and read by disk heads.

The circular path that a disk head traces over a spinning disk is called a *track.* (See Figure 2-1.) Taken together, the set of tracks traced across all the individual disk surfaces for a single position of the heads is called a *cylinder.*

Each track is made up of a number of *sectors* laid end-to-end. A sector consists of a header, a trailer, and 512 bytes of data. The header and trailer contain error-checking information that helps ensure the accuracy of the data.

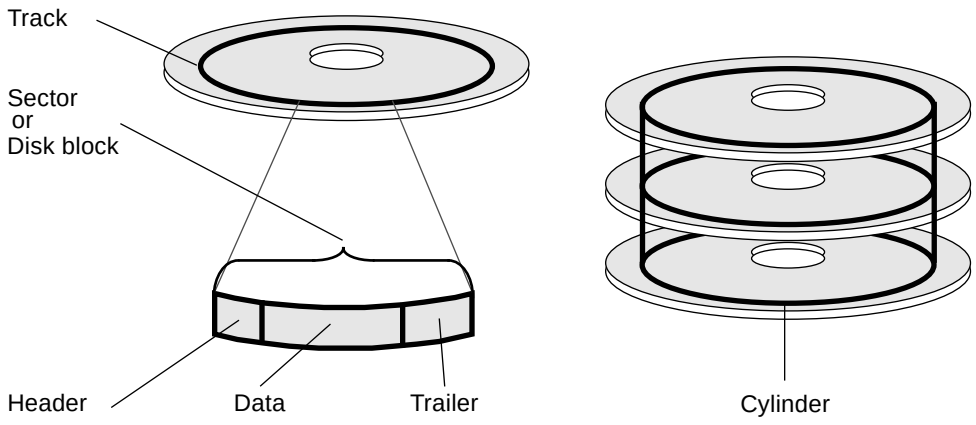


Figure 2-1 Arrangement of Data on Disk

Disk Controllers

Associated with every disk is a *controller*, which may be located on a separate circuit board, or may be *embedded*, or integrated, in the disk drive itself. The controller is an intelligent device responsible for organizing data on the disk.

Disks may contain areas where data cannot be written to and retrieved from reliably. These areas are called *defects*. The controller uses the error-checking information in each disk block's trailer to determine whether or not a defect is present in that block. When a block is found to be defective, the controller can be instructed to add it to a *defect list* and avoid using that block in the future.

Special Areas of the Disk

The beginning and ending portions of the disk are reserved. Either the first or second sector is where the *disk label* is stored. The disk label describes how information is arranged on the disk.

The last two cylinders are set aside for diagnostic use and for storing the disk defect list and a backup copy of the label.

Disk Drivers

The Solaris operating system does not directly communicate with disk controllers. Controllers for disk drives may require different data formats, protocols, and transmission rates. A *device driver* is a low-level program that allows the operating system to communicate with a specific piece of hardware such as a disk controller.

The Solaris environment provides a wide range of device drivers for various devices. These device drivers can be found in `/kernel/drv`.

If you have a disk drive that needs a device driver not listed in `/kernel/drv`, you will need to add the device driver to your system. See Chapter 5, "Device Drivers," for more information on how to install device drivers.

Disk Administration Tool

The Solaris tool for maintaining disks is the `format` utility. This name is something of a misnomer, however, because `format` allows you to:

- Analyze
- Format
- Partition (create slices)
- Repair

Sun cannot guarantee that its `format` utility will work properly with all third-party disk driver. If the disk driver is not compatible with the Solaris `format` utility, the disk drive vendor should supply you with a custom `format` program.

For reference information on the `format` utility, see Appendix C, “format Utility.”

About Disk Formatting

Before you can use a disk, it must be formatted. Formatting involves two separate processes:

- *Formatting* – Writing format information to the disk
- *Surface analysis* – Compiling an up-to-date list of disk defects

When you format a disk, header and trailer information (as well as other information) is superimposed on the disk.

During formatting, the controller scans the disk for defects—a process known as *surface analysis*.

Defects and formatting information reduce the total disk space available for data. This is why a new disk will usually hold only 90 to 95 percent of its capacity after formatting. This percentage varies according to disk geometry, and decreases as the disk ages and develops more defects.

When Should You Format a Disk?

Formatting is a destructive process—it overwrites data on the disk. For this reason, disks are usually formatted only once—by the manufacturer or reseller.

However, if you have reason to believe disk defects are responsible for recurring problems, an additional surface analysis may be warranted. The `format` utility provides surface analysis commands that do not corrupt data.

Is the Disk Properly Formatted?

The `format` utility will tell you if the disk is properly formatted. When you first run the program and select the appropriate disk, you'll see a message similar to the following:

```
selecting c0t0d0s0: <CDC Wren IV 94171-344>
[disk formatted]
```

About Disk Slices

Files stored on a disk are contained in file systems. Each file system on a disk is assigned to a *slice*—a group of cylinders on the disk that has been set aside for use by a file system. Each disk slice appears to the operating system (and to the system administrator) as though it were a separate disk drive.

Note – Slices are sometimes referred to as partitions. Strictly speaking, a *partition* is a collection of slices on certain Solaris platforms. This book attempts to use *slice* whenever possible. However, certain interfaces, such as the `format` utility, refer to slices as partitions.

The system administrator determines how large each disk slice should be. In so doing, it's important to bear in mind that each disk slice holds only one file system, and that no file system can span multiple slices.

When you set up a disk's slices, you choose not only the size of each slice, but also which slices to use. Your decisions about these matters depend on how you intend to use the system to which the disk is attached. See "Determining Which Slices to Use" for more information on system configurations and the slices each uses.

The Eight Solaris Slices

Solaris defines eight disk slices and assigns to each a conventional use. These slices are numbered 0 through 7. Your Solaris platform may contain more slices. See the *Administration Supplement for Solaris Platforms* for more information. Table 2-1 summarizes the contents of the eight Solaris slices.

Table 2-1 Solaris Disk Slices

Slice	File System	Purpose
0	`root`	Holds the files and directories that make up the operating system.
1	`swap`	Provides virtual memory, or *swap space*. Swap space is used when a new program you need to run is too large to fit in a computer's memory at the same time as other programs that are already running. When this happens, the operating system "swaps" different programs from the computer's memory to the disk—and vice versa—as needed.
2	—	Used by the operating system to reference the entire disk. It is defined automatically by Sun's `format` and the Solaris installation programs and should not be altered.
3	`/export`	Holds alternative versions of the operating system. These alternative versions are required by client machines whose architectures differ from that of the server. Clients with the same architecture type as the server obtain executables from slice 6.
4	`/export/swap`	Provides virtual memory space for the client rather than for the server.
5	`/opt`	Holds application software that is added to a system. If there is not enough room on the disk to put the `/opt` file system in slice 5, the file system is put in slice 0.
6	`/usr`	Holds operating system commands—also known as *executables*— designed to be run by users. This slice also holds documentation, system programs (`init` and `syslogd`, for example) and library routines.
7	`/home` or `/export/home`	Holds files that are created, arranged, and maintained by users.

Note – The Solaris installation program provides slice size recommendations based on the software you select for installation.

System vs. Secondary Disks

In a multiple disk arrangement, the disk containing the operating system software and swap space (that is, the disk holding slices 0 and 1) is called the *system disk*. Disks other than the system disk are called *secondary disks*.

Locating a system's slices on multiple disks allows you to modify file systems and slices on the secondary disks without having to shut down the system or reload operating system software.

Having more than one disk also increases input-output (I/O) volume. By distributing disk load across multiple disks, you can avoid I/O bottlenecks at the controller level.

Single vs. Multiple Disks

Although a single disk can hold all eight slices, it is also possible to use two or more disks to hold the slices required by a system.

Note – A slice cannot be split between two or more disks. However, multiple swap slices on separate disks are allowed.

For instance, a single disk might hold slices 0 and 1, while a separate disk is provided for slices 6 and 7.

Determining Which Slices to Use

There are five system configurations

- Servers
- Diskless clients
- Standalone systems
- Single systems
- Dataless clients

Each system configuration requires the use of different slices. Table 2-2 lists these requirements.

Table 2-2 System Configurations and Slice Requirements

Slice	Servers	Diskless Clients	Standalone	Single	Dataless
0	`root`	(on server)	`root`	`root`	`root`
1	`swap`	(on server)	`swap`	`swap`	`swap`
2	—	—	—	—	—
3	`/export`	—	—	—	—
4	`/export/swap`	—	—	—	—

Table 2-2 System Configurations and Slice Requirements

Slice	Servers	Diskless Clients	Standalone	Single	Dataless
5[1]	/opt	(on server)	/opt	/opt	/opt
6	/usr	(on server)	/usr	/usr	(on server)
7	/export/home	(on server)	/home	/home	(on server)

1. Slice 5 (for optional software) can be used in the configurations, though it is usually omitted

The Donor, or Free Hog, Slice

When you use the `format` utility to change the size of one or more disk slices, you designate a temporary slice that will expand and shrink to accommodate the resizing operations.

This slice donates, or "frees," space when you expand a slice, and receives, or "hogs," the discarded space when you shrink a slice. For this reason, the donor slice is sometimes called the *free hog*.

The donor slice exists only during installation or when you run the `format` utility. There is no permanent donor slice during day-to-day, normal operations.

Slice Tables and the `format` *Utility*

The `partition` submenu of the `format` utility allows you to manipulate a disk's slices by editing what is known as a *slice table*. A slice table lists the disk slices and contains entries describing each.

```
Part   Tag     Flag    Cylinders    Size       Blocks
0      root    ru      0             0         (0/0/0)
1      swap    wm      0             0         (0/0/0)
2      backup  -       0 - 1253     198.39MB    (1254/0/0)
3      -       -       0             0         (0/0/0)
4      -       -       0             0         (0/0/0)
5      -       -       0             0         (0/0/0)
6      /usr    rm      0 - 747      118.34MB    (748/0/0)
7      /home   wm    748 - 1253      80.05MB    (506/0/0)
```

The fields Tag and Flag describe the contents and access privileges associated with each slice. These fields are provided solely as a convenience—the operating system ignores them.

The range of numbers in the Cylinders field shows the first and last cylinder occupied by the slice. The Size field describes the slice size in megabytes or gigabytes, as appropriate.

The Blocks field provides still more size information arranged in a triplet of numbers separated by slashes. The numbers refer to the slice size in cylinders, heads, and sectors, respectively.

Note – To eliminate a slice, you simply set its size to 0 (see, for instance, slice 5 above).

About Disk Labels

A special area of every disk is set aside for storing information about the disk's controller, geometry, and slices. That information is called the disk's *label*.

To *label* a disk means to write slice information onto the disk. You label a disk only after changing its slices.

If you fail to label a disk after creating slices, the operating system has no way of "knowing" about the slices, so the disk will not work.

How to Examine a Label

You can examine some of the most important information stored on a disk's label by using the `prtvtoc` command:

```
# prtvtoc /dev/rdsk/c0t0d0s0
* c0t0d0s0 partition map
*
* Dimensions:
*     512 bytes/sector
*      36 sectors/track
*       9 tracks/cylinder
*     324 sectors/cylinder
*    1272 cylinders
*    1254 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
*                          First     Sector    Last
* Partition  Tag  Flags    Sector      Count    Sector  Mount Directory
       0      2    00          0     37260     37259   /
       1      3    01      37260     77760    115019
       2      5    01          0    406296    406295
       6      4    00     115020    283824    398843   /usr
       7      6    00     398844      7452    406295   /home
```

The label shows the number of cylinders (1272) and heads (which is the same as the number of tracks per cylinder, in this case 9), as well as how the disk's slices are arranged.

Autoconfiguration of SCSI Disk Drives

In Solaris 2.3 and subsequent releases, the `format` utility automatically configures SCSI disk drives even if that specific type of drive is not listed in the `/etc/format.dat` file. This feature enables you to format, slice, and label any disk drive compliant with SCSI-2 specification for mode sense pages. The SCSI disk drives must been turned on when a reconfiguration boot is performed so the `format` utility can issue SCSI commands to obtain the disk geometry and

capacity information. The partition table is then automatically generated by `format`. Here's the default slice rules that `format` uses to create the partition table.

Table 2-3 SCSI Disk Slice Rules

Disk size	Root	Swap
0 - 180 Mbytes	16 Mbytes	16 Mbytes
180 Mbytes - 280 Mbytes	16 Mbytes	32 Mbytes
280 Mbytes - 380 Mbytes	24 Mbytes	32 Mbytes
380 Mbytes - 600 Mbytes	32 Mbytes	32 Mbytes
600 Mbytes - 1.0 Gbytes	32 Mbytes	64 Mbytes
1.0 Gbytes - 2.0 Gbytes	64 Mbytes	128 Mbytes
2.0 Gbytes -	128 Mbytes	128 Mbytes

In all cases, slice 6 (`/usr`) gets the remainder of the space on the disk.

Here's an example of a `format`-generated partition table for a 1.3Gb SCSI disk drive.

```
Part      Tag    Flag     Cylinders        Size       Blocks
0         root    wm       0 -   96       64.41MB    (97/0/0)
1         swap    wu      97 -  289      128.16MB    (193/0/0)
2       backup    wu       0 - 1964        1.27GB    (1965/0/0)
6          usr    wm     290 - 1964        1.09GB    (1675/0/0)
```

2

Instructions for Setting Up Disks

This section includes step-by-step instructions for performing tasks related to disks. For some tasks, you will find an example of the screen input and output after the instructions.

Note – You must be superuser to perform the following procedures.

Adding a System Disk

See the Solaris installation guides—*SPARC: Installing Solaris Software* and *x86: Installing Solaris Software*— for the specific tasks of adding a system disk. Adding a system disk includes:

- Loading Solaris from CD-ROM
- Selecting and loading the appropriate software clusters

Adding a Secondary Disk

To add a secondary disk, you may need to follow some or all of these procedures:

- "How to Perform a Reconfiguration Boot" on page 39
- "How to Format a Disk" on page 41
- "How to Create Slices and Label a Disk" on page 47
- "How to Create a File System for Each Slice" on page 50

Many disk drives are shipped already formatted. To determine if your disk is already formatted, see "How to Check If a Disk Is Formatted."

▼ How to Perform a Reconfiguration Boot

1. **Create a file called** `/reconfiguration` **that will be read when the system is booting.**

```
# touch /reconfigure
```

The `/reconfigure` file will cause the SunOS™ software to check for the presence of any newly installed peripheral devices when you power on or boot your system later.

2. **Change directories to / and shut down the system.**

```
# cd /
# /usr/sbin/shutdown -y -g30 -i0
.
.
.
ok
```

Note – The `0` in `i0` is a zero.

In the example above, the command sends a message to all users who are logged in stating they have 30 seconds (`-g30`) before the system begins to shut down. The `ok` or > prompt is displayed once the operating environment is shut down.

3. **Turn off power to the system after the** `ok` **or** > **prompt is displayed.**
Refer to the hardware installation guide that accompanies your system for the location of the power switch.

4. **Turn off power to all external peripheral devices.**
For location of power switches on any peripheral devices, refer to the hardware installation guides that accompany your peripheral devices.

5. **Install the peripheral device.**
Refer to the hardware installation guides that accompany the peripheral devices for information on how to install and connect those devices.

6. **Turn on the power to all external peripherals.**

7. **Turn on the power to the system.**
   The system will boot and you will be shown the login prompt.

▼ How to Check If a Disk Is Formatted

In most cases, disks are formatted by the manufacturer or reseller and do not need to be reformatted when you install the drive. To check a disk to see if it is already formatted, follow these steps.

1. **Enter the format utility by typing** `format` **at the root prompt and pressing Return.**

2. **Enter the number of the disk that you want to check from the list displayed on your screen.**
   If the disk you chose is formatted, you will see the following message:

   ```
   [disk formatted]
   ```

   Here's an example where the second external disk drive is checked.

```
# format
Searching for disks...done


AVAILABLE DISK SELECTIONS:
       0. c0t3d0 <SUN0424 cyl 1151 alt 2 hd 9 sec 80>
          /sbus@1,f8000000/esp@0,800000/sd@3,0
       1. c1t1d0 <SUN0207 cyl  1214 alt 2 hd 9 sec 36>
          /sbus@1,f8000000/esp@0,8000000/sd@1,0
       2. c1t2d0 <SUN0207 cyl  1214 alt 2 hd 9 sec 36>
          /sbus@1,f8000000/esp@0,8000000/sd@2,0
Specify disk (enter its number):2
 selecting c1t2d0
[disk formatted]
```

▼ How to Format a Disk

Note – Formatting a disk may not be necessary. Many disks are formatted by the manufacturer or reseller. To determine if your disk is formatted, see "How to Check If a Disk Is Formatted" on page 40.

Follow these steps to format a disk:

1. **Enter the format utility by typing** `format` **at the root prompt and pressing Return.**
2. **Enter the number of the disk that you want to format from the list displayed on your screen.**
   If the disks have already been labeled, the system then displays information similar to the following.

```
Searching for disks...done
AVAILABLE DISK SELECTIONS:
    0.  c0t3d0 <SUN0207 cyl  1214 alt 2 hd 9 sec 36>
        /sbus@1,f8000000/esp@0,8000000/sd@3,0
    1.  c1t1d0 <SUN0207 cyl  1214 alt 2 hd 9 sec 36>
        /sbus@1,f8000000/esp@0,8000000/sd@1,0
    2.  c1t2d0 <SUN0207 cyl  1214 alt 2 hd 9 sec 36>
        /sbus@1,f8000000/esp@0,8000000/sd@2,0
Specify disk (enter its number):1
```

In this example, disk drive 0 is the system disk and disk drives 1 and 2 are external drives that have been added to the system. Drive 1 with address `c1t1d0` is the first external hard disk that is connected to a SCSI card. Drive 2, with address `c1t2d0`, is the second external hard disk.

Caution – Do not select the system disk. Formatting your system disk deletes your operating system and any data that you may have on this disk.

If the Solaris software had been unable to automatically label drive 1 with address `c1t1d0`, the following is displayed:

```
Searching for disks...done

c1t1d0:  configured with capacity of 198 MB

AVAILABLE DISK SELECTIONS:
    0.  c0t3d0 <SUN0207 cyl  1214 alt 2 hd 9 sec 36>
        /sbus@1,f8000000/esp@0,8000000/sd@3,0
    1.  c1t1d0 <SUN0207 cyl  1214 alt 2 hd 9 sec 36>
        /sbus@1,f8000000/esp@0,8000000/sd@1,0
    2.  c1t2d0 <SUN0207 cyl  1214 alt 2 hd 9 sec 36>
        /sbus@1,f8000000/esp@0,8000000/sd@2,0
Specify disk (enter its number):1

Selecting c1t1d0
[disk formatted]
Disk not labeled. Label it now? y
```

The following prompts are only displayed when the operating system is unable to label a disk.

`c`*`n`*`t`*`n`*`d`*`n`*`:configured with capacity . . .`

`Disk not labeled. Label it now?`

If the format utility does not recognize the disk drive, you may need to create a `format.dat` entry. See “How to Add a Third-Party Disk” for instructions on how to add a `format.dat` entry. Another possible reason that the format utility does not recognize the disk drive is that the disk drive hardware is malfunctioning. Check the documentation accompanying the drive for troubleshooting procedures.

3. **Enter the defect menu by typing** `defect` **option at the** `format>` **prompt and pressing Return.**
   The Defect menu entries that are displayed depend on the type of disk drive that you are trying to format. The menu that your system displays may differ from the one below.

```
format> defect

DEFECT MENU:

   primary - extract manufacturer's defect list
   print   - display working list
   dump    - dump working list to file
   commit  - set current list = working list
   quit

format>
```

4. **Check the Defect menu to see if your disk drive has a SCSI interface.**
   If your disk drive has a SCSI interface, the commit option is not displayed. If this is the case, go to Step 7.

5. **Extract the manufacturer's defect list by typing** `primary` **at the** `defect>` **prompt and pressing Return.**

```
defect> primary

Extracting manufacturer's defect list ... Extraction complete.
Current Defect List updated, total of 20 defects.
```

6. **To use the manufacturer's defect list while formatting the disk drive, type** `commit` **and press Return. Confirm your choice by typing** `y`**.**

```
defect> commit
Ready to update Current Defect List, continue? y
Current Defect List updated, total of 20 defects.
Disk must be reformatted for changes to take effect.
```

7. Leave the Defect menu.

```
defect> quit
```

8. To begin formatting the disk, type `format` at the `format>` prompt. Confirm the command by typing `y`.

```
format> format
Ready to format. Formatting cannot be interrupted
and takes 10 minutes (estimated). Continue? y
Beginning format. The current time is Tue May  3 17:44:42 1994

Formatting ...
done

Verifying media ...
pass 0 - pattern = 0xc6dec6de
pass 1 - pattern = 0x6db6db6d
total of 0 defective blocks repaired.
format>
```

Note – Formatting may take anywhere from a few minutes to several hours, depending on the type and size of the disk.

▼ How to Check If a Disk Has Slices

This procedure assumes that you have selected the disk drive you want to examine when you entered the `format` utility.

Note – The `format` utility uses the term *partition* in place of *slice.*

---

1. Enter the partition menu by typing `partition` at the `format>` prompt.

```
format> partition
PARTITION MENU:
        0      - change ‘0’ partition
        1      - change ‘1’ partition
        2      - change ‘2’ partition
        3      - change ‘3’ partition
        4      - change ‘4’ partition
        5      - change ‘5’ partition
        6      - change ‘6’ partition
        7      - change ‘7’ partition
        select - select a predefined table
        modify - modify a predefined partition table
        name   - name the current table
        print  - display the current table
        label  - write partition map and label to the disk
        quit
partition>
```

2. At the `partition>` **prompt, type** `print` **to display the slice table for the current disk drive.**

```
partition> print
Current partition table (original):
Part      Tag    Flag     Cylinders        Size       Blocks
  0       root    wm       0 -  136       48.16MB    (137/0/0)
  1       swap    wu     137 -  228       32.34MB    (92/0/0)
  2     backup    wm       0 - 1150      404.65MB    (1151/0/0)
  3 unassigned    wm       0               0         (0/0/0)
  4          -    wm     229 -  285       20.04MB    (57/0/0)
  5          -    wm     286 -  414       45.35MB    (129/0/0)
  6        usr    wm     415 -  926      180.00MB    (512/0/0)
  7       home    wm     927 - 1150       78.75MB    (224/0/0)
```

In this example, the drive has slices. Specific slice tags and sizes have been assigned. If you see that no slice sizes are assigned, the disk drive probably does not have slices.

▼ How to Create Slices and Label a Disk

Creating slices on a disk may not be necessary. Many manufacturers or resellers create slices on disks before shipping the disks. To determine if your disk has slices, see “How to Check If a Disk Has Slices.”

Note – The `format` utility uses the term *partition* in place of *slice.*

---

1. Enter the partition menu by typing `partition` at the `format>` prompt.

```
format> partition
PARTITION MENU:
        0      - change ‘0’ partition
        1      - change ‘1’ partition
        2      - change ‘2’ partition
        3      - change ‘3’ partition
        4      - change ‘4’ partition
        5      - change ‘5’ partition
        6      - change ‘6’ partition
        7      - change ‘7’ partition
        select - select a predefined table
        modify - modify a predefined partition table
        name   - name the current table
        print  - display the current table
        label  - write partition map and label to the disk
        quit
partition>
```

2. At the `partition>` prompt, type `modify`.

```
partition> modify
```

3. Type `0` for `Current partition table`, and type `y` to confirm your choice.

```
Select partitioning base:
 0. Current partition table (original sd3)
 1. All Free Hog
Choose base (enter number) [0]? 0

Part    Tag        Flag    Cylinders     Size         Blocks
 0      unassig    flag    0 - 505       80.05MB      (506/0/0)
        ned
 1      -          -       506 - 568     9.97MB       (63/0/0)
 2      -          -       0 - 1253      198.39MB     (1254/0/0)
 3      -          -       0             0            (0/0/0)
 4      -          -       0             0            (0/0/0)
 5      -          -       0             0            (0/0/0)
 6      -          -       569 - 937     58.38MB      (369/0/0)
 7      -          -       938 - 1253    49.99MB      (316/0/0)

Do you wish to continue creating a new partition
table based on above table[yes]? y
```

4. Type the number of the slice you choose as the donor (free hog) slice.

```
Free Hog partition[6]? 6
```

5. For each slice, type the desired size in megabytes.
Type the letters `mb` (for megabytes) after the size, and press Return after each entry.

```
Enter size of partition '0' [163944b, 506c, 80.05mb]: 0
Enter size of partition '1' [20412b, 63c, 9.97mb]: 0
Enter size of partition '3' [0b, 0c, 0.00mb]: 0
Enter size of partition '4' [0b, 0c, 0.00mb]: 0
Enter size of partition '5' [0b, 0c, 0.00mb]: 0
Enter size of partition '7' [102384b, 316c, 49.99mb]: 80mb
```

Caution – When editing a slice table, be careful to avoid having slices overlap. This will cause problems!

6. **Type** `y` **to confirm that the altered slice table is correct.**

```
Part    Tag        Flag    Cylinders     Size         Blocks
 0      -          -       0             0            (0/0/0)
 1      -          -       0             0            (0/0/0)
 2      -          -       0 - 1253      198.39MB     (1254/0/0)
 3      -          -       0             0            (0/0/0)
 4      -          -       0             0            (0/0/0)
 5      -          -       0             0            (0/0/0)
 6      -          -       0 - 747       118.34MB     (748/0/0)
 7      -          -       748 - 1253    80.05MB      (506/0/0)

Okay to make this the current partition table[yes]? y
```

7. **(Optional) Type a name for the new slice table.**
   If you do not want to name the slice table, just press Return. If you use a name that includes embedded spaces, enclose the name in quotation marks.

```
Enter table name (remember quotes): new_improved
```

8. **At the** `partition>` **prompt, type** `label`**. At the confirmation prompt, type** `y`**.**

```
partition> label
ok to label disk? y
```

9. **Type** `q` **twice to exit from the** `format` **utility.**

▼ How to Create a File System for Each Slice

1. **Type the command** `prtvtoc /dev/rdsk/`*device*`s2`
   Substitute the device identifier of the disk you're adding for *device*. (Don't type any spaces between *device* and `s2`.)

```
# prtvtoc /dev/rdsk/c0t2d0s2

[...]

*                          First    Sector   Last     Mount
*Partition   Tag   Flags   Sector   Count    Sector   Directory
2            5     01      0        406296   406295
6            4     00      0        327564   327563
7            0     00      327564   78732    406295
```

2. **Note what slices are available on the disk.**

3. **Prepare a file system for each slice. Type the command:** `newfs /dev/rdsk/`*deviceslice*, **and then type** `y` **to confirm.**
   Substitute the proper device and slice identifiers for *device* and *slice*. (Don't type any spaces between *device* and *slice*.)

```
# newfs /dev/rdsk/c0t2d0s7
newfs: construct a new file system /dev/rdsk/c0t2d0s7 (y/n)? y
/dev/rdsk/c0t2d0s7:   A sectors in B cylinders of C tracks...
     83.9MB in 32 cyl groups (16 c/g, 2.65MB/g, 1216 i/g)
super-block backups (for fsck -b #) at:
 32, 5264, 10496, 15728, 20960, 26192, 31424...
[...]
#
```

4. **Repeat the** `newfs` **command (as above) for each disk slice.**

5. **Mount the file systems.**
   The disk is ready for you to mount the file systems. For information about mounting file systems, see *File System Administration*.

▼ How to Add a Third-Party Disk

The Solaris environment supports many third-party disks. However, you may need to supply either a device driver, a `format.dat` entry, or both of these.

If the third-party disk was designed to work with standard SunOS operating system-compatible device drivers, creating an appropriate `format.dat` entry should be enough to allow the disk to be recognized by the `format` utility. In other cases, you'll need to load a third-party device driver to support the disk.

This section discusses what to do if some of this software support is missing. Typically, this occurs when you invoke the `format` utility and find that the disk type is not recognized.

Supply the missing software as described in this section, and then refer to the appropriate configuration procedure for system disks (page 38) or secondary disks (page 38).

▼ Adding a Device Driver

If a nonstandard device driver is required, it should be provided by the third-party vendor. Following the instructions in Chapter 5, "Device Drivers," for adding a device driver.

Note – Sun cannot guarantee that its `format` utility will work properly with all third-party disk driver. If the disk driver is not compatible with the Solaris `format` utility, the disk drive vendor should supply you with a custom `format` program.

▼ Creating a `format.dat` Entry

Unrecognized disks cannot be formatted without precise information about the disk's geometry and operating parameters. This information is supplied in the `/etc/format.dat` file.

Note – SCSI-2 drives do not require a `format.dat` entry. Starting in Solaris 2.3, the `format` utility automatically configures the SCSI-2 drives if the drives are powered on during a reconfiguration boot.

If your disk was not recognized, use a text editor to create an entry in `format.dat` for the disk. You'll need to gather all the pertinent technical specifications about the disk and its controller before you start. This information should have been provided with the disk. If not, contact the disk manufacturer or your supplier. See Appendix C, "format Utility," for more information on the `/etc/format.dat` file.

The type of information generally required is shown in Table 2-4. Following the table is a sample `format.dat` entry.

Note – The values shown in Table 2-4 are examples; they are not necessarily the correct values to describe your disk's geometry and operating parameters.

Table 2-4 Information in the `/etc/format.dat` File

Information Needed	How Specified	Notes
Controller type	`ctlr = XY450`	Controllers supported by SunOS operating system utilities: • Xylogics® XY450 or XY451 SMD • Xylogics 7053 (SMD) • Emulex® MD21: SCSI with ESDI devices • SCSI: True SCSI (CCS or SCSI-2) • ISP-80: IPI controller
Alternate cylinders	`acyl = 2`	
Alternate sectors per track	`asect = 2`	
Alternate tracks	`atrks = 2`	
Formatting time per cylinder	`fmt_time = 4`	
Number of logical cylinders	`ncyl = 840`	Required
Number of logical heads	`nhead = 20`	Required
Number of logical sectors per track	`nsect = 46`	Required
Number of physical cylinders	`pcyl = 842`	Required

Table 2-4 Information in the `/etc/format.dat` File *(Continued)*

Information Needed	How Specified	Notes
Number of physical heads	`phead = 20`	
Number of physical sectors per track	`psect = 48`	`psect` = `nsect` + `asect`
Rotational speed	`rpm = 3961`	Required
Number of bytes per track	`bpt = 28160`	Required—SMD disks only
Number of bytes per sector	`bps = 595`	Required—SMD disks only
Drive type	`drive_type = 1`	Required—XY450 SMD disks only
Read retries	`read_retries = 1`	SCSI and MD-21 disks only
Write retries	`write_retries = 1`	SCSI and MD-21 disks only
Cylinder skew	`cyl_skew = 2`	SCSI and MD-21 disks only
Track skew	`trk_skew = 2`	SCSI and MD-21 disks only
Tracks per zone	`trks_zone = 15`	SCSI and MD-21 disks only
Cache parameter	`cache = 0x11`	SCSI and MD-21 disks only
Prefetch parameter	`prefetch = 2`	SCSI and MD-21 disks only
Maximum prefetch	`max_prefetch = 5`	SCSI and MD-21 disks only
Minimum prefetch	`min_prefetch = 2`	SCSI and MD-21 disks only

Example of a `format.dat` *Entry*

The following is a sample entry from a `format.dat` file:

```
disk_type = "Fujitsu-M2351 Eagle" \
 : ctlr = XY450 : fmt_time = 4 \
 : ncyl = 840 : acyl = 2 : pcyl = 842 : nhead = 20 : nsect = 46 \
 : rpm = 3961 : bpt = 28160 : bps = 595 : drive_type = 0
```

▼ How to Remove a Disk Drive

You would remove a disk drive from a system to:

- Replace a disk that is defective or inadequate
- Move the drive to another system that needs the extra disk space

Replacing a Defective Disk

To replace a defective disk, do the following:

1. **Back up any data you want to save. See** *File System Administration* **for more information on backing up files.**
2. **Unmount any file systems that may be mounted on the defective disk with** `unshare(1M)`**.**
3. **Physically disconnect the defective drive and connect the replacement.**
4. **Turn to the appropriate instructions in this document for adding a system disk (page 38) or a secondary disk (page 38).**
5. **Restore any data you previously backed up. See** *File System Administration* **for more information on backing up files.**

CD-ROM and Diskette Drives 3

CD-ROMs provide large data storage capacity and have quickly become the medium of choice for software distribution.

Diskettes provide small amounts of data storage and are often used to transfer files to a non-networked personal computer or to store individual user files.

Software called *Volume Management* (added in Solaris 2.2) changes the way you administer and obtain access to the data on CD-ROMs and diskettes.

Use the following table to find specific information on administering CD-ROM and diskette drives.

3

How Solaris Manages CD-ROM and Diskette Drives

A layer of software called Volume Management manages CD-ROM and diskette drives. This software automates the interaction between you and your CD-ROMs and diskettes by automatically mounting CD-ROMs and diskettes that contain file systems.

Table 3-1 lists the changes in locations of CD-ROMs and diskettes from previous releases.

Table 3-1 CD-ROM and Diskette Locations

Media	Pre-Solaris 2.2 Location	Solaris 2.2 and After Location
CD-ROM	`/dev/dsk/c0t6d0s0`	`/cdrom/`*cdrom_name*
	`/dev/rdsk/c0t6d0s0`	`/vol/dev/aliases/cdrom0`
Diskette	`/dev/diskette`	`/floppy/`*floppy_name*
	`/dev/rdiskette`	`/vol/dev/aliases/floppy0`

If you attempt to use these previous locations while using Solaris 2.2 system software and later releases, you will see the messages listed in Table 3-2.

Table 3-2 Error Messages From Using Previous CD-ROM and Diskette Locations

If You Attempt to Use Previous Locations...	You Will See These New Error Messages...
`# mkdir /cdrom`	`mkdir: Failed to make directory "/cdrom"; File exists`
`# mount -F ufs -o ro /dev/dsk/c0t6d0s0 /cdrom`	`mount: /dev/dsk/c0t6d0s0 is already mounted, /cdrom is busy, or allowable number of mount points exceeded`
`% tar cvf /dev/diskette proposal status`	`tar: cannot open /dev/diskette.`
`% eject /dev/diskette`	`/dev/rdiskette is busy (try /vol name?)`

Note – If you are using the Wabi™ or SunPC™ products, you must use that product's interface for accessing CD-ROMs and diskettes.

Using Diskettes

Before you can access a diskette, Volume Management needs to be informed of the diskette's presence. Use one of the following to do this:

- Type `volcheck` and press Return.
- Select Check For Floppy from the File Menu of File Manager (a DeskSet application).

See the `volcheck(1)` man page for more information on `volcheck`. You do not need to use this command if you are going to use the commands `fdformat` or `eject`. For more information on formatting diskettes, see the *Administration Supplement for Solaris Platforms.*

Automatically Sharing CD-ROMs and Diskette File Systems

You can configure Volume Management to automatically share mounted CD-ROM and diskette file systems with other network users. You can specify that any mounted CD-ROM or diskette file system be shared or limit the sharing to a specifically named file system.

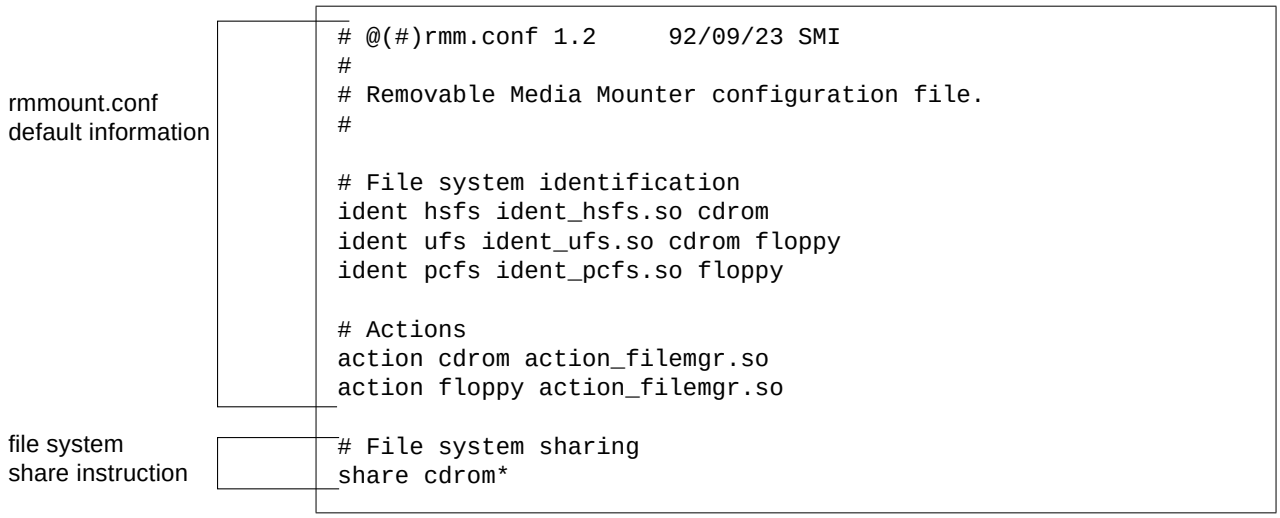
To share any of the mounted CD-ROM file systems on your system, add the following lines to the `/etc/rmmount.conf` file:

```
# File system sharing
share cdrom*
```

To share a specific CD-ROM file system, indicate the name of the file system:

```
# File system sharing
share Solaris_2.3*
```

Here is the `/etc/rmmount.conf` with the general share option included:



You can specify any of the `share` command line options. See `share(1M)`and `rmmount.conf(4)` for more information.

Instructions for Adding a CD-ROM

This section includes step-by-step instructions for adding a CD-ROM drive to a system.

Note – You must be superuser to perform the following procedures.

▼ How to Add a CD-ROM Drive

1. **Create a file called** `/reconfigure` **that will be read when the system is booting.**

```
# touch /reconfigure
```

The `/reconfigure` file will cause the SunOS software to check for the presence of any newly installed peripheral devices when you power on or boot your system later.

2. **Shut down the system.**

```
# /usr/sbin/shutdown -y -g30 -i0
.
.
.
.
ok
```

Note – The 0 in `i0` is a zero.

In the example above, the command sends a message to all users who are logged in stating they have 30 seconds (`-g30`) before the system begins to shut down. The `ok` or > prompt is displayed once the operating environment is shut down.

3. **Turn off power to the system after the** `ok` **or** > **prompt is displayed.** Refer to the hardware installation guide that accompanies your system for the location of the power switch.

4. **Turn off power to all external peripheral devices.**
   For location of power switches on any peripheral devices, refer to the hardware installation guides that accompany your peripheral devices.
5. **Install the peripheral device.**
   Refer to the hardware installation guides that accompany the peripheral devices for information on how to install and connect those devices.
6. **Turn on the power to all external peripherals.**
7. **Turn on the power to the system.**
   The system will boot and you will be shown the login prompt.

Instructions for Using CD-ROM and Diskette Media

Formatting Diskettes Before Use

See the *Administration Supplement for Solaris Platforms* for information on how to format your diskettes. The procedure varies depending on what type of diskette you use.

▼ How to Find Out What Media Has Been Inserted

Volume Management keeps track of CD-ROM and diskette file systems during a system session (rebooting will clear the in-memory database).

♦ **To view the media that has been inserted during a system session, list `/vol/dsk`.**

```
% ls /vol/dsk
solaris_2_1/   unnamed_cdrom#1
unnamed_cdrom  unnamed_floppy
```

Troubleshooting Tips for CD-ROMs and Diskette Devices

Disabling Volume Management

If you need to disable Volume Management to diagnose system or device problems, use the following procedure:

1. **Rename the `/etc/rc2.d/S92volmgt` file to stop the automatic startup of Volume Management at run time.**

   ```
   # mv /etc/rc2.d/S92volmgt /etc/rc2.d/S92volmgt.old
   ```

2. **Stop Volume Management.**

   ```
   # /etc/init.d/volmgt stop
   ```

Public Domain Programs

Many public domain CD-ROM software programs do not take advantage of Volume Management. To run these programs, you will need to disable Volume Management as described above.

Volume Management Does Not Recognize Media

If you have inserted a CD-ROM or diskette and Volume Management does not recognize the media and mount it (if appropriate), eject and reinsert the media.

If you have a diskette, be sure to issue the `volcheck` command or choose the Check For Floppy item from the File Manager's File menu after you insert the diskette. These commands instruct Volume Management to check the diskette drive for media.

Problems with Manually Removing Diskettes

Do not remove diskettes manually before issuing the `eject(1)` command. If you manually remove a diskette without ejecting, any of the following problems can occur:

- The kernel could panic. If this happens, the system should reboot itself.
- Volume Management might hang or end up in an unrecoverable state.

In either case, you need to follow this procedure.

1. **Stop Volume Management by typing** `/etc/init.d/volmgt stop`**.**
2. **Verify that Volume Management stopped by typing** `ps -e | grep vold`**.**
   If Volume Management is not stopped, you need to reboot your system.
3. **Once Volume Management has stopped, type** `/etc/init.d/volmgt start`**.**

Reference Material for CD-ROM and Diskette Devices

CD-ROM and diskette file systems are automatically mounted in default locations by Volume Management when the media is inserted. For security reasons, these file system are mounted `setuid`. See `mount(1M)` for a description of this and other mount options. See Table 3-3 for mount locations.

Table 3-3 CD-ROM and Diskette File System Mount Points

Media Type	Mount Location	State of Media
Diskette	`/floppy/floppy`*n*	Symbolic link to mounted diskette in a local diskette drive. Examples of locations include `/floppy/floppy0` and`/floppy/floppy1`.
	`/floppy/`*floppy_name*	Mounted named diskette
	`/floppy/unnamed_floppy`	Mounted unnamed diskette
CD-ROM	`/cdrom/cdrom`*n*	Symbolic link to mounted CD-ROM in a local CD-ROM drive. Examples of locations include `/cdrom/cdrom0` and`/cdrom/cdrom1`.
	`/cdrom/`*CD-ROM_name*	Mounted named CD-ROM
	`/cdrom/`*CD-ROM_name*`/`*partition*	Mounted named CD-ROM with partitioned file system
	`/cdrom/unnamed_cdrom`	Mounted unnamed CD-ROM

If there is no file system on the media, Volume Management provides block and character devices in the `/vol` file system. See Table 3-4 for the location of diskette and CD-ROM media in the `/vol` file system.

Table 3-4 CD-ROM and Diskette Device Locations in `/vol`—No File System Present

Media Type	Device Location	State of Media
Diskette	`/vol/dev/diskette`*n*`/unnamed_floppy`	Formatted unnamed diskette—block device access
	`/vol/dev/rdiskette`*n*`/unnamed_floppy`	Formatted unnamed diskette—raw device access
	`/vol/dev/diskette`*n*`/unlabeled`	Unlabeled diskette—block device access
	`/vol/dev/rdiskette`*n*`/unlabeled`	Unlabeled diskette—raw device access
CD-ROM	`/vol/dev/dsk/c0t`*n*`/unnamed_cdrom`	CD-ROM—block device access
	`/vol/dev/rdsk/c0t`*n*`/unnamed_cdrom`	CD-ROM—raw device access

Volume Management Man Pages

The following man pages describe the components and commands provided by Volume Management.

Table 3-5 Volume Management Man Pages

Man Page	Description
`volcancel(1)`	cancels user's request for removable media that is not currently in drive
`volcheck(1)`	checks for media in a drive. Default checks all diskette media.
`volmissing(1)`	notifies user that volume requested is not in the CD-ROM or diskette drive
`rmmount(1M)`	removable media mounter that automatically mounts a file system on a CD-ROM and diskette
`vold(1M)`	Volume Management daemon to manage CD-ROM and diskette devices
`rmmount.conf(4)`	removable media mounter configuration file
`vold.conf(4)`	Volume Management configuration file
`volfs(7)`	Volume Management file system

3

Tape Drives 4

Backing up data is perhaps a system administrator's most important responsibility. This chapter does not discuss backups (see *File System Administration*), but it does explain how to set up and maintain the tape devices often used to backup data.

4

Instructions for Adding a Tape Drive

This section includes step-by-step instructions for adding a tape drive.

Note – You must be superuser to perform the following procedure.

▼ How to Add a SCSI Tape Drive

1. **Create a file called** `/reconfigure` **that will be read when the system is booting.**

```
# touch /reconfigure
```

The `/reconfigure` file will cause the SunOS software to check for the presence of any newly installed peripheral devices when you power on or boot your system later.

2. **Shut down the system.**

```
# /usr/sbin/shutdown -y -g30 -i0
.
.
.
.
ok
```

Note – The 0 in `i0` is a zero.

In the example above, the command sends a message to all users who are logged in stating they have 30 seconds (`-g30`) before the system begins to shut down. The `ok` or > prompt is displayed once the operating environment is shut down.

3. **Turn off power to the system after the** `ok` **or** > **prompt is displayed.**
Refer to the hardware installation guide that accompanies your system for the location of the power switch.

4. **Turn off power to all external peripheral devices.**
   For location of power switches on any peripheral devices, refer to the hardware installation guides that accompany your peripheral devices.

5. **Install the peripheral device.**
   Refer to the hardware installation guides that accompany the peripheral devices for information on how to install and connect those devices.

   Set the SCSI unit number as specified in the manufacturer's documentation. You will often find a small switch located at the back of the device for this purpose. Check the documentation accompanying the tape drive for more information.

6. **Turn on the power to all external peripherals.**

7. **Turn on the power to the system.**
   The system will boot and you will be shown the login prompt.

Reference Material for Tape Drives

The following tape drives, which are available from Sun Microsystems, Inc., are supported by Solaris 2.x.

Table 4-1

Tape Format	Model
1/4"	Archive Viper™ QIC-150 streaming tape drive
	Emulex® MT-02 tape controller
1/2"	HP®-88780 tape drive
4mm	Archive Python DAT tape subsystem
8mm	Exabyte® ESB-8200/8500 cartridge tape

Drives not purchased from Sun Microsystems, Inc. are not guaranteed to work unmodified with Solaris 2.x.

If you have a tape drive that is not listed above or has not been purchased from Sun Microsystems, Inc., see the `st(7)` man page for more information on how to add a third-party tape drive.

4

Device Drivers 5

Device drivers do not usually enter into the day-to-day duties of a system administrator. You need be concerned about drivers only when adding an unsupported device to the system.

Use the following table to find specific information on administering devices drivers.

About Device Drivers

A computer typically uses a wide range of peripheral and mass-storage devices. Your machine, for example, probably has a SCSI disk drive, a keyboard and a mouse, and some kind of magnetic backup medium. Other commonly used devices include CD-ROM drives, printers and plotters, light pens, touch-sensitive screens, digitizers, and tablet-and-stylus pairs.

The Solaris software does not directly communicate with all these devices. Each type of device requires different data formats, protocols, and transmission rates.

5

A device driver is a low-level program that allows the operating system to communicate with a specific piece of hardware. The driver serves as the operating system's "interpreter" for that piece of hardware.

Device Drivers in the Solaris 2.x Environment

Drivers needed to support a wide range of standard devices are included in the Solaris 2.x environment. These drivers can be found under the directory `/kernel/drv`.

However, when you install a unsupported device, you'll need to add a new driver to allow that device to work properly. The Solaris 2.x environment makes this job much easier—just add the software package containing the device driver.

What You Need From an Unsupported Device Manufacturer

If you've purchased an unsupported device, the manufacturer should provide the software needed for the device to be properly installed, maintained, and administered.

At a minimum, this software includes a device driver and its associated configuration (`.conf`) file. In addition, the device may be incompatible with the utilities provided in the Solaris 2.x product, and may require custom maintenance and administrative utilities.

Contact your device manufacturer for more information.

Instructions for Adding and Removing Device Drivers

This section includes step-by-step instructions for performing tasks related to device drivers.

Note – You must be superuser to perform the following procedures.

▼ How to Add a Device Driver

1. **Place the tape, diskette, or CD-ROM into the drive.**
2. **Use the** `pkgadd` **command. Type the following:**

```
# pkgadd -d device packagename
```

Replace *device* and *packagename* as appropriate.

3. **Create a file called** `/reconfigure` **that will be read when the system is booting.**

```
# touch /reconfigure
```

The `/reconfigure` file will cause the SunOS software to check for the presence of any newly installed device drivers when you power on or boot your system later.

4. **Shut down the system.**

```
# /usr/sbin/shutdown -y -g30 -i0
.
.
.
.
ok
```

Note – The `0` in `i0` is a zero.

In the example above, the command sends a message to all users who are logged in stating they have 30 seconds (`-g30`) before the system begins to shut down. The `ok` or `>` prompt is displayed once the operating environment is shut down.

5. **Turn off power to the system after the `ok` or `>` prompt is displayed.**
   Refer to the hardware installation guide that accompanies your system for the location of the power switch.

6. **Turn on the power to the system.**
   The system will boot and you will be shown the login prompt.

▼ How to Remove a Device Driver

It is not necessary to remove a driver when its associated device is removed from a system. However, if you want to clean up your file systems and conserve space, you can easily remove a driver.

♦ **Type the following:**

```
# rem_drv drivername
```

Caution – A single driver can serve multiple devices. If you want to remove a driver, first make sure that there are no devices on the system that require that driver!

The Service Access Facility A

This appendix explains in detail what a system or network administrator needs to know about the Service Access Facility (SAF) of the Solaris 2.x environment.

If you want to see examples of specific SAF commands, skip the first section, “Overview of the Service Access Facility,” and use the following table to find examples of the instructions you need.

A

Overview of the Service Access Facility

The SAF is the tool used for administering terminals, modems, and other network devices. In particular, SAF enables you to:

- Add and administer `ttymon` and `listen` port monitors (using the `sacadm` command)
- Add and administer `ttymon` port monitor services (using the `pmadm` and `ttyadm` commands)
- Add and administer `listen` port monitor services (using the `pmadm` and `nlsadmin` commands)
- Administer and troubleshoot TTY devices
- Administer and troubleshoot incoming network requests for printing service
- Administer 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.

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 may differ from network to network, SAC and the administrative programs `sacadm` and `pmadm` are network independent.

Table A-1 illustrates 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 A-1 SAF Functions and Associated Programs

Function	Program	Description
Overall Administration	`sacadm`	Command for adding and removing port monitors
Service Access Controller	`sac`	SAF's master program
Port Monitors	`ttymon` `listen`	Monitors serial port login requests 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

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 may 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 (technically speaking, 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 port monitor and 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 Figure A-1, 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`). Figure A-1 shows only a single `ttymon` port monitor.

After `ttymon` has been started, it monitors the serial port lines for service requests.

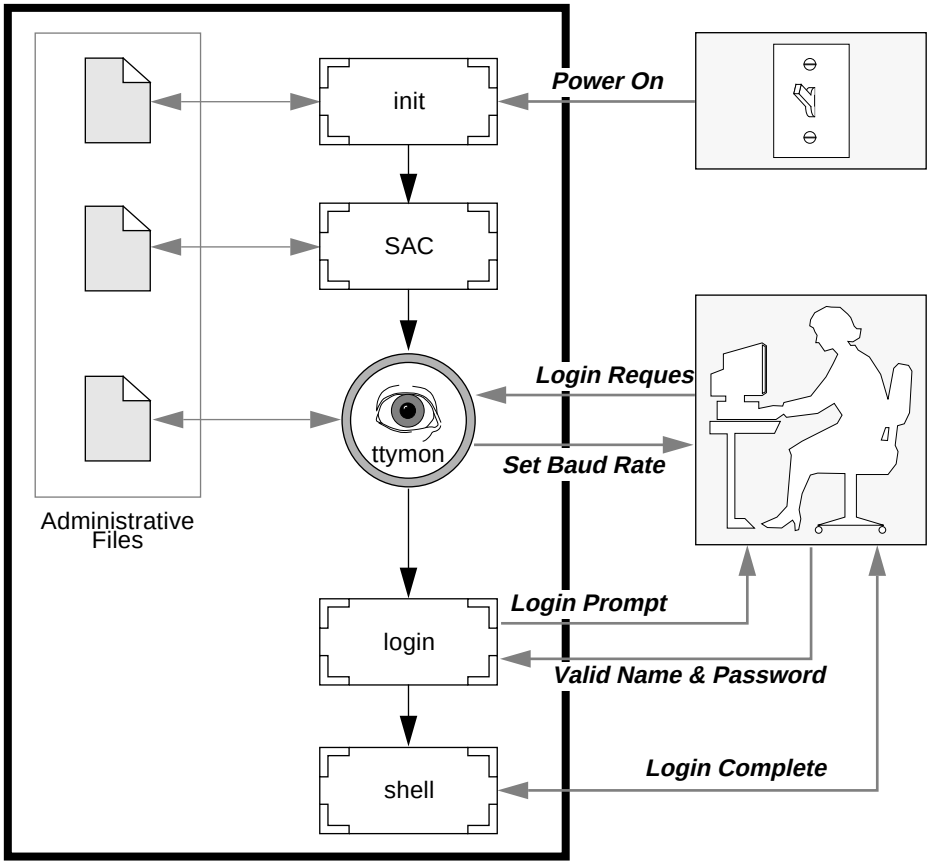


Figure A-1 How `ttymon` Helps Process a Login Request

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/utmp` entry if required, establishes the service environment, and invokes the service associated with the port.

After the service terminates, `ttymon` cleans up the `/etc/utmp` 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 2.x environment—`ttymon` and `listen`.

A

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 2.x users the same services that `getty` did under previous versions of Solaris 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.

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 the `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.

For more information about writing your own port monitor, see the *Network Interfaces Programmer's Guide*.

The network listener process may be used with any connection-oriented transport provider that conforms to the Transport Layer Interface (TLI) specification. In the Solaris 2.x environment, `listen` port monitors provide additional network services not provided by `inetd`, such as print service.

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`. See `nlsadmin(1M)` for more details.

Each network has 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 a detailed description.

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"
```

Where:

`-a`	is 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
```

Where:

`-l`	is the *list* port monitor status flag
`-p`	specifies the *pmtag* `mbmon` as the port monitor tag

▼ How to Stop a `ttymon` Port Monitor

♦ **To kill a `ttymon` port monitor, type:**

```
# sacadm -k -p mbmon
```

Where:

`-k`	is 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
```

Where:

`-s`	is 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
```

Where:

`-d`	is 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
```

Where:

`-e`	is 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
```

Where:

`-r`	is 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`.

Where:

`-a`	is 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. In that embedded command:

`-b`	is 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
```

Where:

`-l`	is the flag for a list of 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 a Port Monitor

♦ **To list all services of a port monitor, type:**

```
# pmadm -l -p mbmon
```

Where:

`-l`	is the flag for a list of service information
`-p`	specifies the *pmtag* `mbmon` as the port monitor tag

Listing All Services of All Port Monitors

♦ **To list all services of all port monitors, type:**

```
# pmadm -l
```

Where:

`-l`	is the flag for a list of service information

▼ How to Enable a Port Monitor Service

♦ **To enable a disabled port monitor service, type:**

```
# pmadm -e -p mbmon -s a
```

Where:

`-e`	is 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
```

Where:

`-d`	is the *disable* flag
`-p`	specifies the *pmtag* `mbmon` as the port monitor tag
`-s`	specifies the *svctag* `a` as the port monitor *service* tag

Administering `listen` *Port Monitors*

Use the `sacadm` command to add, list, kill, start, enable, disable, or remove a `listen` port monitor.

Note – You must be superuser to perform the following procedures.

▼ How to Add a `listen` Port Monitor

♦ **To add a `listen` port monitor, type:**

```
# sacadm -a -p tcp -t listen -c /usr/lib/saf/listen -v ‘nlsadmin -V‘ -y "le0 ethernet"
```

Where:

`-a`	is the *add* port monitor flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag
`-t`	specifies the port monitor type as `listen`
`-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 `listen` Port Monitor Status

♦ **To list the status of a `listen` port monitor, type:**

```
# sacadm -l -p tcp
```

Where:

`-l`	is the *list* port monitor status flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag

▼ How to Stop a `listen` Port Monitor

♦ **To kill a `listen` port monitor, type:**

```
# sacadm -k -p tcp
```

Where:

`-k`	is the *kill* port monitor flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag

▼ How to Start a `listen` Port Monitor

♦ **To start a `listen` port monitor, type:**

```
# sacadm -s -p tcp
```

Where:.

`-s`	is the *start* port monitor flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag

▼ How to Enable a `listen` Port Monitor

♦ **To enable a `listen` port monitor, type:**

```
# sacadm -e -p tcp
```

Where:

`-e`	is the *enable* port monitor flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag

▼ How to Disable a `listen` Port Monitor

♦ **To disable a `listen` port monitor, type:**

```
# sacadm -d -p tcp
```

Where:

`-e`	is the *disable* port monitor flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag

▼ How to Remove a `listen` Port Monitor

♦ **To remove a `listen` port monitor, type:**

```
# sacadm -r -p tcp
```

Where:

`-r`	is the *remove* port monitor flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag

Administering `listen` *Port Monitor Services*

Use the `pmadm` command to add, enable, disable, and list the services associated with a `listen` port monitor.

Note – You must be superuser to perform the following procedures.

▼ How to Add a `listen` Port Monitor Service

♦ **To add a `listen` port monitor service, type:**

```
# pmadm -a -p tcp -s lp -i root -v ‘nlsadmin -V‘ -m "‘nlsadmin -o
/var/spool/lp/fifos/listenS5‘"
```

In this example, a listen service is added for print requests from remote SunOS 5.x machines. This service does not listen for requests from SunOS 4.x machines.

Note – In this example, the input wraps to the next line. Do not put a Return or line feed after the `-o`.

Where:

`-a`	is the *add* port monitor service flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag
`-s`	specifies the *svctag* `lp` 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 `listen`-specific configuration data formatted by `nlsadmin`

The above `pmadm` command contains an embedded `nlsadmin` command. In that embedded command `-o` specifies the full path name of a first-in first-out (FIFO) or named STREAM through which a standing server will receive the connection.

▼ How to List `listen` Port Monitor Services

♦ **To list the services associated with a `listen` port monitor, type:**

```
# pmadm -l -p tcp
```

Where:

`-l`	is the *list* port monitor service flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag

▼ How to Enable a `listen` Port Monitor Service

♦ **To enable a `listen` port monitor service, type:**

```
# pmadm -e -p tcp -s lp
```

Where:

`-e`	is the *enable* flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag
`-s`	specifies the *svctag* `lp` as the port monitor *service* tag

▼ How to Disable a `listen` Port Monitor Service

♦ **To disable a `listen` port monitor service, type:**

```
# pmadm -d -p tcp -s lp
```

Where:

`-d`	is the *disable* flag
`-p`	specifies the *pmtag* `tcp` as the port monitor tag
`-s`	specifies the *svctag* `lp` as the port monitor *service* tag

Troubleshooting the Network Listener: `listen` *Port Monitor*

Here are some tips for remedying `listen` port monitor difficulties.

1. Begin with the network.

   The network listener is suspect when users report that they cannot print to a network printer. Begin by issuing the `/usr/sbin/ping` command from the print server to the client and back to determine if the network is up.

2. Check the `listen` port monitor.

   Use `sacadm` to check that the listener is starting. Use `pmadm` to check that the print service is configured correctly and that the service is enabled.

3. Check the print server's configuration.

   See *User Accounts, Printers, and Mail Administration* for more information.

Reference Material for Service Access Facility Administration

Files Associated With SAF

SAF uses configuration files which can be modified by using the `sacadm` and `pmadm` commands. It should not be necessary for you to edit them manually.

Table A-2 SAF-Associated Files

File Name	Description
`/etc/saf/_sysconfig`	Per-system configuration script
`/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

Service States

The `sacadm` command controls the states of services. The possible states are shown below.

Table A-3 `sacadm` Service States

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
```

A

Port Monitor States

The `sacadm` command controls the states of `ttymon` and `listen` port monitors. The possible states are shown below.

Table A-4 Port Monitor States

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*.
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 may be enabled or disabled depending on the state of the port monitor that controls them.

Table A-5 Port States

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, `ttymon` will send out the "disabled" message when it receives a connection request.
Network (`listen`) Port States	
Enabled	The `listen` port monitor scans the network for service requests and invokes services in response to those requests.
Disabled	If the `listen` port monitor is killed or disabled, the ports it controls are automatically disabled. If you specify, `listen` will send out the "disabled" message when it receives a connection request.

A

Connecting Devices to the Serial Port B

This appendix provides general information about connecting equipment to serial ports and explains some of the terminology related to that information. For specific instructions on adding terminals or modems, see Chapter 1, “Terminals and Modems.”

General Information About Serial Port Devices

A serial port sends a byte of information, bit-by-bit, over a single line. Some manufacturers’ serial ports have RS-232-C cabling conventions, but use RS-423 communications signal conventions. You can attach modems, terminals, printers, plotters, or other peripheral serial devices that accept the RS-423 signaling to the serial port connectors on your machine.

Ports are generally wired as data terminal equipment (DTE). This means that the data transmit signal from the port is on pin 2 and the receive data from the peripheral is on pin 3.

Modems

Connect modems directly to the serial port. Modems are wired as data communications equipment (DCE) devices. A DCE device receives serial data from a DTE device on pin 2 and sends it down the telephone line. A DCE device receives telephone data and sends it out on pin 3 to a DTE device.

B

Other Peripherals

To connect a system directly to a terminal, printer, computer, or other DTE device, use a standard *null modem* cable that crosses lines 2 and 3, thereby enabling the proper signal connection at the other end.

Some serial interfaces require special additional wiring. For example, the MTI board requires pin 5 to be asserted. This requirement can be satisfied if the peripheral supplies an appropriate signal; otherwise, pin 4 and 5 can be connected at the MTI connector end of the cable. Check the documentation that came with the peripheral to see if any special wiring is needed.

Make sure the serial device is supported. Note that the Solaris 2.x environment contains drivers to support many devices. However, if you purchase a nonstandard device from a non-Sun manufacturer, you may need to obtain and add an appropriate serial driver from the manufacturer (see Chapter 5, “Device Drivers”).

Connecting Terminals

Make sure you have an available serial port on your system. The number of serial ports varies from one system to another. Most systems provide at least two asynchronous serial ports, often controlled by the serial communications controller on the central processing unit board.

If all the existing serial ports are in use, and if you are not planning to disconnect any current peripheral devices to make a port available, you will need to add a new board to increase the number of serial ports. See the appropriate board installation guide for instructions.

When you do have a port available, connect the terminal to the system with a null modem cable. A basic null modem cable swaps lines 2 and 3 so that the proper transmit and receive signals are communicated between two DTE devices. Line 7 goes straight through, connecting pin 7 of the devices at each end of the null modem cable.

Connecting Modems

The Solaris 2.x environment supports many popular modems. Models that have been found to work especially well include: the Hayes® Smartmodem 1200™ and Telebit™ TrailBlazer™; and, the USRobotics® Courier 2400™. In this section, you'll find general advice for setting switches on each of these three models.

Note – See the documentation accompanying the modem for specific procedures on how to set the switches. If you have questions about the proper modem settings, contact the modem manufacturer.

Connecting Cables

Connect the modem to an open serial port on the system with an RS-232-C cable that has pins 2 through 8 and pin 20 wired straight through. You may also use a full 25-pin cable to connect the modem to the system. Make sure all the connections are secure.

Setting Modem Switches

The switch setting examples listed below work for use with `tip` and `uucp`. Always read the manufacturer's manual before attempting to adjust equipment.

Note – After changing the switch settings on any model, you must turn off power to the modem, wait a few seconds, and then turn on power again.

Looking at the front of a properly installed modem, you should see TR light up when the modem is not in use. If auto-answer is enabled (switch #5), AA should also be lit. The lights should be lit or blinking when the modem is in use.

Example 1 – Hayes Smartmodem 1200

The proper switch settings follow. There is only one switch panel on the Hayes Smartmodem 1200 modem; down is On and up is Off.

- Switch 1—up for hardware data terminal ready
- Switch 2—down for numeric result codes
- Switch 3—down to send result codes
- Switch 4—down to not echo commands
- Switch 5—up to answer incoming calls
- Switch 6—up for hardware carrier detect
- Switch 7—up for connection to RJ11 modular jack
- Switch 8—down to enable command recognition

Example 2 – USRobotics Courier 2400

The proper switch settings follow. There are two switch panels on the USRobotics Courier 2400 modem: a 10-switch panel and a single "Quad" switch panel. On the USRobotics modem, down is On and up is Off.

- Switch 1—up for hardware data terminal ready
- Switch 2— down for numeric result codes
- Switch 3—down to send result codes
- Switch 4—down to not echo commands
- Switch 5—up to answer incoming calls
- Switch 6—up for hardware carrier detect
- Switch 7—up for connection to RJ11 modular jack
- Switch 8—down to enable command recognition
- Switch 9—down to not disconnect with +++
- Switch 10—up for feature not used
- "Quad" switch—up
- Pins 2 and 3 wired straight through

Example 3 – Telebit TrailBlazer

The TrailBlazer modem must have internal registers set to work properly with cu and uucp. The first time the system is connected to the modem, you must enter the following setup sequence to use the TrailBlazer:

```
AT &F Q6 S51=254 S52=2 S53=1 S54=3 S58=0 S111=30 &W
```

Note that the above sequence may be slightly different from the one shown in the TrailBlazer documentation. You should read the TrailBlazer documentation fully to make sure the sequence is right for your system.

Settings for Other Modems

Hayes-compatible modems that use the Hayes AT command set *may* work with cu and uucp software. Configure the modem to:

- Use hardware data terminal ready (DTR). That is, when the system drops DTR (for example, when someone logs off) the modem should hang up.
- Use hardware carrier detect (CD). That is, the modem only raises the CD line when there is an active carrier signal on the phone connection. When carrier drops, either due to the other end of the connection being terminated or the phone connection being broken, the system will be notified and act appropriately. The CD signal is also used for coordinating dial-in *and* dial-out use on a single serial port and modem.
- Respond with numeric result codes.
- Send result codes.
- Not echo commands.

Debugging Hints

If you encounter problems with your new hardware, check the cabling and connections first.

Verify that the new device accepts the RS-423 communications protocol, and consult the manufacturer's manual to make sure the device has been installed properly.

Are the system and the new device set up for compatible communication? Many systems default to the following signal parameters:

- Seven data bits
- Even parity
- 9600 baud rate
- Flow control enabled (XON/XOFF)

Check which control signals the other equipment expects.

Next, check the condition of the cable. Sometimes a cable becomes damaged, resulting in signals being flipped randomly. A line may receive when it should transmit, or the reverse.

To check for cable problems it is helpful to have a *breakout box*. A breakout box plugs into the RS-232-C cable, providing a patch panel that allows you to connect any pin to any other pin(s); it will often contain light-emitting diodes, which display the presence or absence of a signal on each pin. These tools can be purchased through most computer hardware suppliers.

Even the most routine installation can sometimes hit unexpected snags. If something does not appear to be working properly, try the following:

- Check that all devices (modems, terminals, and computer) are properly turned on.
- If you cannot access a port, and find a process running on it when you type the `ps` command, then make sure you have pin 8 connected in your cable. If that does not work, check that your device driver is configured properly to set the correct flag for the line to Off.
- Sometimes even when both the hardware and software are correct, the device driver gets into a state where it will not let the alternate port be opened. You must do a `kill -1 1` to notify `init` and reset the flags on the device driver.
- If you get a `can't synchronize with hayes` error message when using a Hayes-compatible modem, check internal and external modem switch settings, and check the cable connection. Turn the modem off and then on again if necessary.
- If you get a `can't synchronize with ventel` error message when using a Hayes-compatible modem, look in the `/etc/remote` file and make sure you have changed `at=ventel` to `at=hayes`.

- The message `all ports busy` may mean that the port is actually busy running a dial-in user or the data carrier detect (DCD) signal is being asserted by the modem. You can do a `ps` to see what is running.
  You should check that you have properly set up the carrier detect signal from the modem as well.
- Check the `/var/spool` and `/var/spool/locks` directories. If no process is currently using the serial port, there may be a leftover lock file. It will have a name like `LCK.cua0`. If you change any modem switch, unplug the modem and turn its power off. Then, after waiting several seconds, turn on the modem's power again.

B

`format` *Utility*

C

This appendix contains these sections on the `format` utility:

About `format`

If you do not know much about the `format` utility, read the following information to gain a conceptual view of the `format` utility and what it is used for before proceeding to the instruction or reference sections.

`format` *Definition*

The `format` utility is a system administration tool used to prepare hard disk drives for use on your Solaris system. `format` cannot be used on diskette drives, CD-ROM drives, or tape drives.

If you need more information on disk drive terminology, see Chapter 2, “Disk Drives.” This chapter introduces many disk drive concepts.

C

Features and Benefits

`format` provides the following features and associated benefits.

Table C-1 `format` Features and Benefits

Feature	Benefit
Searches your system for all attached disk drives	Reports • target location • disk geometry • whether the disk is formatted • if the disk has mounted partitions
Retrieves disk labels	Used in repair operations
Repairs defective sectors	Allows disk drives to be repaired by knowledgeable administrators instead of sending the drive back to the manufacturer
Formats and analyzes a disk	Creates sectors on the disk and verifies each sector
Partitions a disk	Divides a disk so individual file systems can be created on separate slices
Labels a disk	Writes disk name and configuration information to the disk for future retrieval (usually for repair operations)

When to Use `format`

Disk drives are formatted, partitioned, and labeled by the Solaris installation program as part of installing Solaris. You may need to use `format` when:

- Adding a disk drive to an existing system
- Repairing a disk drive

The need for formatting and partitioning a disk drive has dropped as more and more manufacturers ship their disk drives already formatted and partitioned. You may not need to use `format` when adding a disk drive to an existing system.

Also, some customer sites prefer to replace rather than repair defective drives. If your site has a repair contract with the disk drive manufacturer, you may not need to use `format` to repair disk drives.

Requirements or Restrictions for Using `format`

You must be root to use the `format` utility. If you are not root, you will see the following error message when you try to use `format`.

```
% format
Searching for disk...done
No permission (or no disk found)!
```

Prerequisite Information Needed to Use `format`

Recommendations for Preserving Information

- Back up all files on the disk drive before doing anything else.
- Save all your defect lists in files by using the `format dump` command. The file name should include the drive type, model number, and serial number.
- Save the paper copies of the manufacturer's defect list shipped with your drive.

Informing `format` *of a Third-party Drive*

If you are adding a disk drive that is not supplied by Sun Microsystems, Inc., you may need to provide disk information in the `format.dat` file. To determine if you need to add a `format.dat` entry:

1. **Check your disk drive manufacturer literature to see if the disk drive interface is SCSI-2 (small computer system interface, version 2).**
   If it is, you do not need to supply a `format.dat` entry. The Solaris software will automatically retrieve information about the drive.
2. **If you do not have a SCSI-2 disk drive, check the `format.dat` file for an entry matching your drive.**
   If you cannot find an entry that matches, follow the instructions described in "How to Add a Third-Party Disk" on page 51.

C

Instructions for Using `format`

Many of the instructions for using `format` are covered in these books:

Peripherals Administration (this book)

- "Adding a Secondary Disk" on page 38
- "How to Check If a Disk Is Formatted" on page 40
- "How to Format a Disk" on page 41
- "How to Check If a Disk Has Slices" on page 45
- "How to Create Slices and Label a Disk" on page 47
- "How to Add a Third-Party Disk" on page 51

Administration Supplement for Solaris Platforms

- Identifying Disk Devices on Your System
- How to Add a System Disk

The two procedures covered in this appendix are:

- How to Identify a Defective Sector
- How to Repair a Defective Sector

Repairing a Defective Sector

If a disk on your system has a defective sector, you can repair it using the instructions in the following procedures. You may become aware of defective sectors when you:

- Run surface analysis on a disk
  See "analyze Menu" on page 115 for more information on the analysis functionality of `format`.
- Get a number of error messages from the disk driver concerning a particular portion of the disk while your system is running.
  The defective area reported while your system is running may not be accurate. Since the system does disk operations many sectors at a time, it is often hard to pinpoint exactly which sector caused a given error. Use "How to Identify a Defective Sector" on page 109 to find the exact sector(s).

Note – The following procedures in this section require you to be superuser.

▼ How to Identify a Defective Sector

1. **Unmount the file system in the slice that contains the defective sector. See** `mount(1M)` **for more information.**
2. **Enter the format utility by typing** `format`**.**
3. **Select the affected disk.**

```
# format
searching for disk...done

AVAILABLE DISK SELECTIONS:
    0.  c0t2d0 <SUN0424 cyl  1151 alt 2 hd 9 sec 80> opt
        /sbus@1,f8000000/esp@0,8000000/sd@2,0
    1.  c0t3d0 <SUN0424 cyl  1151 alt 2 hd 9 sec 80>
        /sbus@1,f8000000/esp@0,8000000/sd@3,0
Specify disk (enter its number):0
selecting c0t2d0: opt
[disk formatted]
Warning: Current Disk has mounted partitions.
```

4. **Enter the analyze menu by typing** `analyze`**.**
5. **Set up the analysis parameters for the search step. Use the parameters shown here**:

```
analyze> setup
Analyze entire disk [no]? n
Enter starting block number [0, 0/0/0]: 12330
Enter ending block number [584159, 1216/9/47]: 12360
Loop continuously [no]? y
Repair defective blocks [yes]? n
Stop after first error [no]? n
Use random bit patterns [no]? n
Enter number of blocks per transfer [31, 0/0/31]: 1
Verify media after formatting [yes]? y
Enable extended messages [no]? n
Restore defect list [yes]? y
Create defect label [yes]? y
```

6. Use the `read` **command to find the defect.**

```
analyze> read
Ready to analyze (won't harm SunOS). This takes a long time,
but is interruptible with Control-C. Continue? y
        pass 0
   25/7/24
        pass 1
Block 12354  (18/4/18), Corrected media error (hard data ecc)
   25/7/24
        pass 3
Block 12354  (18/4/18), Corrected media error (hard data ecc)
   25/7/24
^C
Total of 0 defective blocks repaired.
```

▼ How to Repair a Defective Sector

1. Enter the repair command from the format menu.

```
# format
searching for disk...done
.
.

FORMAT MENU:
.
.
          repair    - repair a defective sector
.
.
          quit
format> repair
```

2. **Enter the defective block number.**
   If you are unsure of the format used to identify the defective sector, see "Block Numbers" on page 125 for more information.

```
format> repair
   Enter absolute block number of defect: 18/4/18
   Ready to repair defect, continue? y
   Repairing block 12354 (18/4/18)...ok.
format>
```

Background for `format`

This section provides more detailed information on `format` and contains these discussions.

C

Format Menu and Command Descriptions

The main menu for the *format* utility looks like this.

```
FORMAT MENU:
          disk      - select a disk
          type      - select (define) a disk type
          partition - select (define) a partition table
          current   - describe the current disk
          format    - format and analyze the disk
          repair    - repair a defective sector
          label     - write label to the disk
          analyze   - surface analysis
          defect    - defect list management
          backup    - search for backup labels
          verify    - read and display labels
          save      - save new disk/partition definitions
          inquiry   - show vendor, product and revision
          volname   - set 8-character volume name
          quit
```

Here are the descriptions of each submenu and command.

Table C-2 `format` Main Menu Item Descriptions

Item	Command or Menu?	Allows You To...
`disk`	command	Choose the disk that will be used in subsequent operations (known as the current disk). All of the system's drives are listed.
`type`	command	Identify the manufacturer and model of the current disk. All Solaris-supported drives are listed. Choose the Auto configure option for all SCSI-2 disk drives.
`partition`	menu	Create and modify slices. See "partition Menu" for more information.
`current`	command	Display the following information about the current disk: • device name • manufacturer and model • number of cylinders, alternate cylinders, heads and sectors • bus address
`format`	command	Format the current disk. Uses the information found in the `format.dat` file or prompts you for information if there is no `format.dat` entry.

Table C-2 `format` Main Menu Item Descriptions (Continued)

Item	Command or Menu?	Allows You To...
`fdisk`	menu	Run the fdisk program. See the `fdisk(1M)` man page and the *Administration Supplement for Solaris Platforms* for more information.
`repair`	command	Repair a specific blocks on the disk.
`label`	command	Write a new label to the current disk
`analyze`	menu	Run read, write, compare tests. See "analyze Menu" for more information.
`defect`	menu	Retrieve and print defect lists. See "defect Menu" for more information.
`backup`	command	Search for backup labels.
`verify`	command	Print the following information about the disk: • device name • manufacturer and model • number of cylinders, alternate cylinders, heads and sectors • bus address • partition table
`save`	command	Save new disk and partition information.
`inquiry`	command	Print the vendor, product name, and revision level of the current drive
`volname`	command	Label the disk with a new eight character volume name.
`quit`	command	Exit the `format` menu.

`partition` *Menu*

The partition menu looks like this.

```
format> partition


PARTITION MENU:
          0       - change ‘0' partition
          1       - change ‘1' partition
          2       - change ‘2' partition
          3       - change ‘3' partition
          4       - change ‘4' partition
          5       - change ‘5' partition
          6       - change ‘6' partition
          7       - change ‘7' partition
          select  - select a predefined table
          modify  - modify a predefined partition table
          name    - name the current table
          print   - display the current table
          label   - write partition map and label to the disk
          quit
partition>
```

Here are the descriptions for each command.

Table C-3 `partition` Menu Item Descriptions

The Command...	**Allows You To...**
`change ‘x’ partition`	Specify new slice • identification tag • permission flags • starting cylinder • size
`select`	Choose a predefined slice table.
`modify`	Change the disk slices of the current disk.
`name`	Specify a name for the current slice table.

Table C-3 `partition` Menu Item Descriptions (Continued)

The Command...	Allows You To...
`print`	View the current slice table
label	Write the slice map and label to the current disk.
quit	Exit the partition menu.

`analyze` *Menu*

The analyze menu looks like this.

```
format> analyze

ANALYZE MENU:
          read      - read only test    (doesn't harm SunOS)
          refresh   - read then write   (doesn't harm data)
          test      - pattern testing   (doesn't harm data)
          write     - write then read         (corrupts data)
          compare   - write, read, compare (corrupts data)
          purge     - write, read, write    (corrupts data)
          print     - display data buffer
          setup     - set analysis parameters
          config    - show analysis parameters
          quit
analyze>
```

Here are the descriptions for each command.

Table C-4 `analyze` Menu Item Descriptions

The Command...	Allows You To...
read	Read each sector on this disk. Repairs defective blocks as a default.
refresh	Read then write data on the disk without harming the data. Repairs defective blocks as a default.
test	Write a pattern (`0xc6dec6de`) to the disk without harming the data. Repairs defective blocks as a default.

Table C-4 `analyze` Menu Item Descriptions (Continued)

The Command...	Allows You To...
write	Write a pattern (`0xc6dec6de`) to the disk then read the data on the disk back. Destroys existing data on the disk. Repairs defective blocks as a default.
compare	Write a pattern (`0xc6dec6de`) to the disk, read the data back, and compare it to the data in the write buffer. Destroys existing data on the disk. Repairs defective blocks as a default.
purge	Write a pattern to the disk then read the data back. The default behavior is that four passes are run. The patterns written to the disk are: pass 0 = `0xaaaaaaaa` Destroys existing data on the disk. Repairs defective blocks as a default.
print	View the data in the read / write buffer.

setup	Specify the following analysis parameters
	`Analyze entire disk? **yes**`
	`Starting block number:` *depends on drive*
	`Ending block number:` *depends on drive*
	`Loop continuously? **no**`
	`Number of passes: **4**`
	`Repair defective blocks? **yes**`
	`Stop after first error? **no**`
	`Use random bit patterns? **no**`
	`Number of blocks per transfer: **126 (0/*n*/*nn*)**`
	`Verify media after formatting? **yes**`
	`Enable extended messages? **no**`
	`Restore defect list? **yes**`
	`Restore disk label? **yes**`
	Defaults are shown in bold.

config	View the current analysis parameters.
quit	Exit the analyze menu.

`defect` *Menu*

The defect menu looks like this.

```
format> defect

DEFECT MENU:
          primary   - extract manufacturer's defect list
          grown     - extract manufacturer's and repaired defects lists
          both       - extract both primary and grown defects lists
          print      - display working list
          dump       - dump working list to file
          quit
defect>
```

Here are the descriptions for each command.

Table C-5 `defect` Menu Item Descriptions

The Command...	Allows You To...
primary	Read the manufacturer's defect list from the disk drive and update the in-memory defect list.
grown	Read the grown defect list (defects that have been detected during analysis) and update the in-memory defect list.
both	Read both the manufacturer's and grown defect list and update the in-memory defect list.
print	View the in-memory defect list.
dump	Save the in-memory defect list to a file.
quit	Exit the defect menu.

Files Use by `format`—`format.dat`

The `format` data, `/etc/format.dat`, contains:

- default disk types
- default slice tables

C

The `format`.dat file shipped with the Solaris operating system supports many standard disks. If your disk drive is not listed in the `format.dat` file, you can choose to add an entry for it or allow `format` to prompt you for the information it needs while it is performing operations.

Adding an entry to the `format.dat` file can save time if the disk drive will be used throughout your site. To use the `format.dat` file on other systems, copy the file to each system that will use the specific disk drive you added to the `format.dat` file.

You should modify the data file for your system if you have one of the following:

- A disk that has a unit number or controller number not found in the `GENERIC` configuration file
- A disk that is not supported by the Solaris operating system
- A disk with a slice table that is different from the Solaris operating system default configuration

---

Note – Do not alter default entries. If you want to alter the default entries, copy the entry, give it a different name, and make the modification to avoid confusion.

Structure

The `format` data file (`format.dat`) contains specific disk drive information used by the `format` utility. Three items are defined in the `format.dat` file:

- search paths
- disk types
- slice tables

`format.dat` *Syntax*

The following syntax rules apply to the data file:

- The pound sign (#) is the comment character. Any text on a line after a pound sign is not interpreted by `format`.

- Each definition in the `format.dat` file appears on a single logical line. If the definition is more than one line long, all but the last line of the definition must end with a backslash (\).
- A definition consists of a series of assignments that have an identifier on the left side and one or more values on the right side. The assignment operator is the equal sign (=). The assignments within a definition must be separated by a colon (:).
- White space is ignored by `format`. If you want an assigned value to contain white space, enclose the entire value in double quotes ("). This will cause the white space within the quotes to be preserved as part of the assignment value.
- Some assignments can have multiple values on the right hand side. Separate values by a comma (,).

`format.dat` *Keywords*

The data file contains disk definitions that are read in by `format` when it starts up. Each definition starts with one of the following keywords: `search_path`, `disk_type`, and `partition`.

Table 5-1

Keyword	Use
`search_path`	Tells `format` which disks it should search for when it starts up. The list in the default data file contains all the disks in the `GENERIC` configuration file. If your system has disks that are not in the `GENERIC` configuration file, add them to the `search_path` definition in your data file. The data file can contain only one `search_path` definition. However, this single definition lets you specify all the disks you have in your system.
`disk_type`	Defines the controller and disk model. Each `disk_type` definition contains information concerning the physical geometry of the disk. The default data file contains definitions for the controllers and disks that the Solaris operating system supports. You need to add a new `disk_type` only if you have an unsupported disk. You can add as many `disk_type` definitions to the data file as you want.
`partition`	Defines a slice table for a specific disk type. The slice table contains the slice information, plus a name that lets you refer to it in `format`. The default data file contains default slice definitions for several kinds of disk drives. Add a slice definition if you recreated slices on any of the disks on your system. Add as many slice definitions to the data file as you need.

Search Path

`search_path` tells `format` which disks it should search for when it starts up. The list in the default data file contains all the disks in the `GENERIC` configuration file. If your system has disks that are not in the `GENERIC` configuration file, add them to the `search_path` definition in your data file. The data file can contain only one `search_path` definition. However, this single definition lets you specify all the disks you have in your system. The disk names are as they appear in the boot messages. Here is an example of the `search_path` definition:

```
# This is the search path for format. It contains all the disks that
# will be searched for if no disk list is given on the command line.
#
        search_path = xy0, xy1, xy2, xy3, xd0, xd1, xd2, xd3, xd4, xd5, xd6, \
                xd7, xd8, xd9, xd10, xd11, xd12, xd13, xd14, xd15, sd0, sd1, sd2, sd3
```

Disk Type

`disk_type` defines the controller and disk model. Each `disk_type` definition contains the physical geometry of the disk. The default data file contains definitions for the controllers and disks that the Solaris operating system supports. You need to add a new `disk_type` only if you have an unsupported disk. You can add as many `disk_type` definitions to the data file as you want.

The keyword itself is assigned the name of the disk type. This name appears in the disk's label, and is used to identify the disk type whenever `format` is run. Enclose the name in double quotes to preserve any white space in the name. The following identifiers must also be assigned values in all `disk_type` definitions:

Table C-6 Required `disk_type` Identifiers

Identifier	Description
`ctlr`	Valid controller type the disk types for the disk type. Currently, the supported values for this assignment are XY450 for Xylogics 450/451 controllers, XD7053 for Xylogics® 7053 controllers, MD21 for Emulex® MD21 controllers (and embedded SCSI disks), and ACB4000 for Adaptec® ACB4000 controllers.
`ncyl`	The number of data cylinders in the disk type. This determines how many logical cylinders of the disk the system will be allowed to access.
`acyl`	The number of alternate cylinders in the disk type. These cylinders are used by `format` to store information such as the defect list for the drive. You should always leave at least two cylinders for alternates.
`pcyl`	The number of physical cylinders in the disk type. This number is used to calculate the boundaries of the disk media. This number is usually equal to `ncyl` plus `acyl`, but there are some circumstances under which it is not. For instance, the Emulex MD21 controller requires four cylinders for internal controller use, so they must be left off the other assignments. Also, to make disks field-replaceable with other disks, some disks are artificially limited to be the same size as another type.
`nhead`	The number of heads in the disk type. This number is used to calculate the boundaries of the disk media.
`nsect`	The number of data sectors per track in the disk type. This number is used to calculate the boundaries of the disk media. Note that this is only the data sectors, any spares are not reflected in the assignment.
`rpm`	The rotations per minute of the disk type. This information is put in the label and later used by the file system to calculate the optimal placement of file data.
`bpt`	The physical number of bytes per track for the disk type. This number is used to calculate the boundaries for defects that are in bytes from index format.

Other assignments may be necessary depending on the controller. For XY450/451 controllers, the following assignments are also required:

Table C-7 `disk_type` Identifiers for XY450/451 Controllers

Identifier	Description
`bps`	The total number of bytes per sector, including the header and gaps, in the disk type. This number is necessary to locate defects within a track. See the disk manual for information on how to calculate this number.
`drive_type`	The drive type of the disk type. The drive type is a number between 0 and 3 that the 450/451 controller uses to identify the disk's geometry. See the controller manual for more information.
`fmt_time`	A number indicating how long it takes to format a given drive. See the controller manual for more information.

For XD7053 controllers, the following assignments are also required:

Table C-8 `disk_type` Identifiers for XD7053 Controllers

Identifier	Description
`bps`	The total number of bytes per sector, including the header and gaps, in the disk type. This number is necessary to locate defects within a track. See the disk manual for information on how to calculate this number.

For MD21 and SCSI controllers, the following assignments are also required:

Table C-9 `disk_type` Identifiers for MD21 and SCSI Controllers

Identifier	Description
`fmt_time`	A number indicating how long it takes to format a given drive. See the controller manual for more information.
`cache`	A number that controls the operation of the onboard cache while `format` is operating. See the controller manual for more information.
`trks_zone`	A number that specified how many tracks you have per defect zone, to be used in alternate sector mapping. See the controller manual for more information.
`asect`	The number assigned to this parameter specifies how many sectors are available for alternate mapping within a given defect zone. See the controller manual for more information.

Below are some examples of `disk_type` definitions:

```
disk_type = "Fujitsu-M2361 Eagle" \
    : ctlr = XY450 : fmt_time = 4 \
    : ncyl = 840 : acyl = 2 : pcyl = 842 : nhead = 20 : nsect = 67 \
    : rpm = 3600 : bpt = 40960 : bps = 600 : drive_type = 3
disk_type = "CDC EMD 9720" \\
    : ctlr = XD7053 \\
    : ncyl = 1147 : acyl = 2 : pcyl = 1217 : nhead = 10 : nsect = 48 \
    : rpm = 3600 : bpt = 30240 : bps = 613
disk_type = "Micropolis 1355" \
    : ctlr = MD21 \
    : ncyl = 1018 : acyl = 2 : pcyl = 1024 : nhead = 8 : nsect = 34 \
    : rpm = 3600 : bpt = 20832
disk_type = "Fujitsu M2243AS" \
    : ctlr = ACB4000 \
    : ncyl = 752 : acyl = 2 : pcyl = 754 : nhead = 11 : nsect = 17 \
    : rpm = 3600 : bpt = 10416 : skew = 2 : precomp = 754
```

Slice Tables

A partition definition keyword is assigned the name of the slice table. Enclose the name in double quotes to preserve any white space in the name. The following identifiers must also be assigned values in all slice definitions:

Table C-10 Required Identifiers for slice Tables

Identifier	Description
`disk`	The name of the `disk_type` that this slice table is defined for. This name must appear exactly as it does in the `disk_type` definition.
`ctlr`	The controller type disks this slice table can be attached to. Currently, the supported values for this assignment are XY450 for Xylogics 450/451 controllers, XD7053 for Xylogics 7053 controllers, MD21 for Emulex MD21 controllers, ACB4000 for Adaptec ACB4000 controllers, and SCSI for SCSI controllers. The controller type specified here must also be defined for the `disk_type` chosen above.

The other assignments in a slice definition describe the actual slice information. The identifiers are the numbers 0 through 7. These assignments are optional. Any slice not explicitly assigned is set to 0 length. The value of each of these

assignments is a pair of numbers separated by a comma. The first number is the starting cylinder for the slice, and the second is the number of sectors in the slice. Below are some examples of slice definitions:

```
partition = "Fujitsu-M2351 Eagle" \
    : disk = "Fujitsu-M2351 Eagle" : ctlr = XY450 \
    : a = 0, 16560 : b = 18, 34040 : c = 0, 772800 : g = 55, 722200
partition = "Micropolis 1355" \
    : disk = "Micropolis 1355" : ctlr = MD21 \
    : a = 0, 16048 : b = 59, 33456 : c = 0, 276896 : g = 182, 227392
partition = "Toshiba MK 156F" \
    : disk = "Toshiba MK 156F" : ctlr = MD21 \
    : a = 0, 15980 : b = 47, 33660 : c = 0, 277100 : g = 146, 227460
```

`format.dat` *Location*

`format` learns of the location of your data file by the following methods.

1. If a path name is given with the `-x` command line option, that file is always used as the data file.
2. If the `-x` option is not specified, then `format` looks in the current directory for a file named `format.dat`. If the file exists, it is used as the data file.
3. If neither of these methods yields a data file, `format` uses `/etc/format.dat` as the data file. This file is shipped with the Solaris operating system and should always be present.

Associated Man Pages

The man page associated with the format utility is `format(1M)` which describes the basic `format` utility capabilities and provides descriptions of all command line variables.

Rules for Input to `format` *Commands*

When using format, you need to provide information in various formats. This section describes the rules for the formats.

Numbers

Several places in `format` require an integer as input. You must either specify the data or select one from a list of choices. In either case, the `help` facility causes `format` to print the upper and lower limits of the integer expected. Simply enter the number desired. The number is assumed to be in decimal unless a base is explicitly specified as part of the number (for example, 0x for hexadecimal).

The following are examples of integer input:

```
Enter number of passes [2]: 34
Enter number of passes [34] 0xf
```

Block Numbers

Whenever you are required to specify a disk block number, there are two ways to input the information:

- block number as an integer
- block number in the cylinder / head / sector format

You can specify the information as an integer representing the logical block number. You can specify the integer in any base, but the default is decimal. The maximum operator (a dollar sign, `$`) can also be used here to let `format` select the appropriate value. Logical block format is used by the SunOS disk drivers in error messages.

The other way to specify a block number is the cylinder / head / sector format. In this format, you must specify explicitly the three logical components of the block number, the cylinder, head, and sector values. These values are still logical, but they allow you to define regions of the disk related to the layout of the media.

If any of the cylinder / head / sector numbers are not specified, the appropriate value is assumed to be zero. You can also use the maximum operator in place of any of the numbers and let `format` select the appropriate value. Below are some examples of cylinder, head, and sector entries:

```
Enter defective block number: 34/2/3
Enter defective block number: 23/1/
Enter defective block number: 457//
Enter defective block number: 12345
Enter defective block number: 0xabcd
Enter defective block number: 334/$/2
Enter defective block number: 892//$
```

`format` always prints block numbers, in both of the above formats. Also, the `help` facility shows you the upper and lower bounds of the block number expected, in both formats.

Command Names

Command names are needed as input whenever `format` is displaying a menu prompt. You can abbreviate the command names, as long as what is entered is sufficient to uniquely identify the command desired.

Other Names

There are certain times in `format` when you must name something. In these cases, you are free to specify any string desired for the name. If the name has white space in it, the entire name must be enclosed in double quotes ("). Otherwise, only the first word of the name is used.

Index

M

N

P

R

S

T

V

W