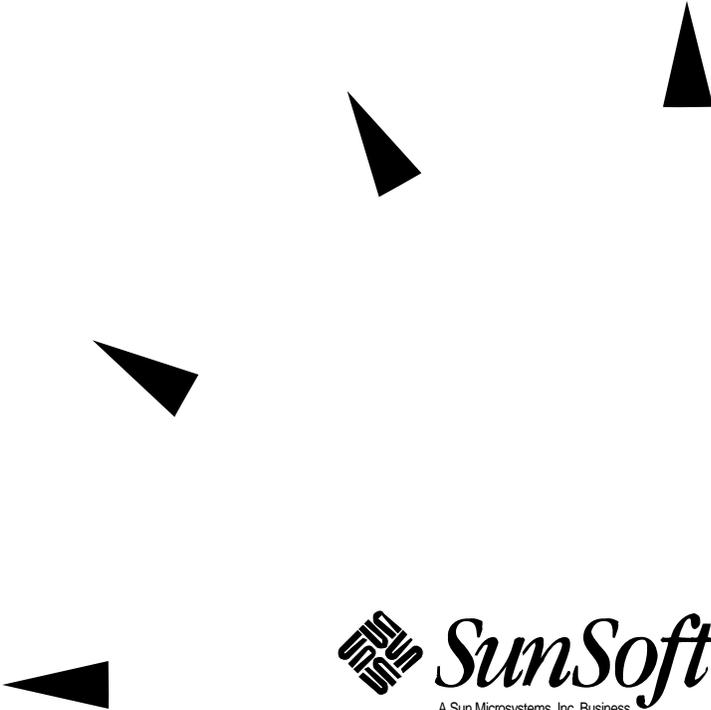


Solaris *PowerPC Edition*: Installing Solaris Software

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



Copyright 1996 Sun Microsystems, Inc., 2550 Garcia Avenue, Mountain View, California 94043-1100 U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

Portions of this product may be derived from the UNIX[®] system, licensed from Novell, Inc., and from the Berkeley 4.3 BSD system, licensed from the University of California. UNIX is a registered trademark in the United States and other countries and is exclusively licensed by X/Open Company Ltd. Third-party software, including font technology in this product, is protected by copyright and licensed from Sun's suppliers.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 and FAR 52.227-19.

Sun, Sun Microsystems, the Sun logo, Solaris, Solstice, SunOS, OpenWindows, ONC, NFS, DeskSet are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the United States and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc. ELC, IPC, IPX, SLC, SPARCstation, SPARCserver, SPARCclassic, SPARCcenter, SPARCengine, Voyager, Ultra, and UltraServer are licensed exclusively to Sun Microsystems, Inc. The PowerPC name is a trademark of International Business Machines Corporation.

The OPEN LOOK[®] and Sun[™] Graphical User Interfaces were developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

X Window System is a trademark of X Consortium, Inc.

THIS PUBLICATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.



Contents

About This Book	ix
1. About Installing Solaris.	1
Definition: Installing Solaris Software.	1
2. Preparing to Install Solaris	5
Why You Should Not Ignore This Chapter.	5
Steps to Prepare to Install Solaris.	6
Task Map: Interactive Installations	13
Task Map: Custom JumpStart Installation	14
3. Preparing to Install Solaris Over a Network	15
About Installing Solaris Over a Network	15
Servers Required for Network Installation.	16
Requirements for Hands-Off Network Installation.	18
Commands You Should Know About.	19
Creating an Install Server	20
▼ How to Set Up a New System to be an Install Server ..	20

▼ How to Create an Install Server	22
▼ How to Create a Boot Server on a Subnet	25
Setting Up Servers for Network Installation	28
▼ How to Set Up Servers for Network Installation of a Standalone, Server, or Dataless Client	29
Using <code>add_install_client</code> to Set Up Servers for Network Installation	35
▼ How to Use <code>add_install_client</code> to Set Up Servers for Network Installation	36
Preconfiguring the Default Locale	38
▼ How to Preconfigure Default Locale Using NIS	38
▼ How to Preconfigure Default Locale Using NIS+	41
4. Preparing Custom JumpStart Installations	45
Definition: Custom JumpStart Installation	45
Reasons to Choose a Custom JumpStart Installation	46
Tasks to Prepare for Custom JumpStart Installations	47
What Happens During a Custom JumpStart Installation	48
Creating a JumpStart Directory on a Diskette	51
▼ How to Create a JumpStart Directory on a Diskette for x86 and PowerPC Systems	51
Creating a JumpStart Directory on a Server	57
▼ How to Create a JumpStart Directory on a Server	57
Enabling All Systems to Access the JumpStart Directory	59
▼ How to Enable All Systems to Access the JumpStart Directory	59
Creating a Profile	61

What Is a Profile	61
Requirements for Profiles	61
▼ How to Create a Profile	62
Profile Examples	63
Profile Keyword and Profile Value Descriptions	66
How the Size of Swap Is Determined	76
Creating the <code>rules</code> File	76
What Is the <code>rules</code> File	76
When Does a System Match a Rule	77
▼ How to Create the <code>rules</code> File	77
Rule Examples	80
Important Information About the <code>rules</code> File	81
Rule Keyword and Rule Value Descriptions	82
How the Installation Program Sets the Value of <code>rootdisk</code>	86
Using <code>check</code> to Validate the <code>rules</code> File	87
▼ How to Use <code>check</code> to Validate the <code>rules</code> File	88
5. Using Optional Custom JumpStart Features	91
Overview	91
Creating Begin Scripts	92
What Is a Begin Script	92
Important Information About Begin Scripts	92
Ideas for Begin Scripts	92
Creating Derived Profiles With Begin Scripts	92
Creating Finish Scripts	94

What Is a Finish Script	94
Important Information About Finish Scripts	94
Ideas for Finish Scripts	94
Adding Files With Finish Scripts	95
Customizing the Root Environment	96
Setting the System's Root Password With Finish Scripts ..	96
Using <code>pfinstall</code> to Test Profiles	98
Why Use <code>pfinstall</code>	98
Ways to Use <code>pfinstall</code>	98
▼ How to Use <code>pfinstall</code> to Test a Profile	99
<code>pfinstall</code> Examples	100
Creating a Disk Configuration File for a PowerPC System ...	101
▼ How to Create a Disk Configuration File for a PowerPC System	101
▼ How to Create a Multiple Disk Configuration File for a PowerPC System	104
Using a Site-Specific Installation Program	108
6. Booting and Installing Solaris: Interactive	109
7. Booting and Installing Solaris: Custom JumpStart	115
8. Where to Go After Installing Solaris	121
A. Worksheets for the Solaris Installation Program	125
B. System Identification Label	133
C. Platform Names and Groups	139
D. Sample Custom JumpStart Installation	141
E. Troubleshooting	149

Specific Installation Errors	150
F. Time Zones.....	155

About This Book

Who Can Use This Book

This book is for anyone installing the Solaris™ operating environment on networked or non-networked systems. Site policy and/or level of expertise will determine who can perform the tasks required to install Solaris software.

Don't Read the Entire Book!

Because this book covers different ways of installing the Solaris software to accommodate a variety of site needs, you do not need to read the entire book.

Read chapters 1 and 2 — they'll help you decide which method of installing is best for your site, and point you to a task map that tells you exactly what sections of the book to use.

How This Book Is Organized

This book is organized by tasks in the categories of before, during, and after installing Solaris software.

Note – This book does not include instructions for setting up system hardware or other peripherals. Setting up hardware and peripherals is described in your hardware guides.

———— *Before Installing Solaris* ————

- 1 About Installing Solaris
- 2 Preparing to Install Solaris
- 3 Preparing to Install Solaris Over a Network
- 4 Preparing for Custom JumpStart™ Installations
- 5 Using Optional Custom JumpStart Features

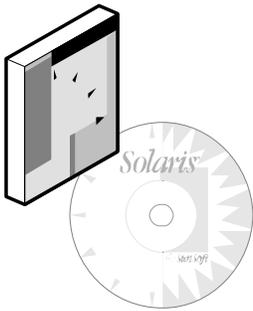
———— *Installing Solaris* ————

- 6 Installing Solaris: Interactive
- 7 Installing Solaris: Custom JumpStart

———— *After Installing Solaris* ————

- 8 Where To Go After Installing Solaris

Related Information



You may need to refer to the following books or online information when installing Solaris software:

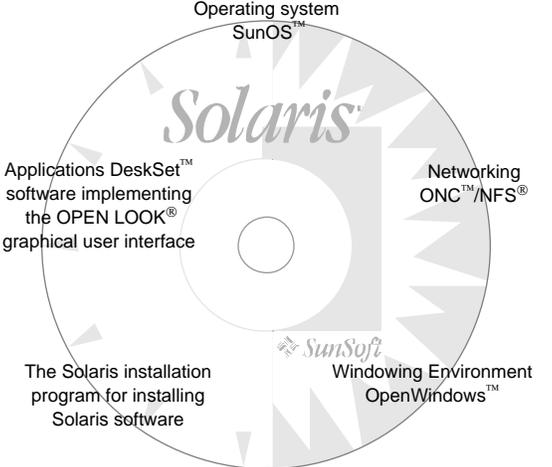
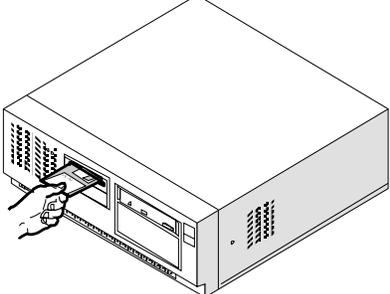
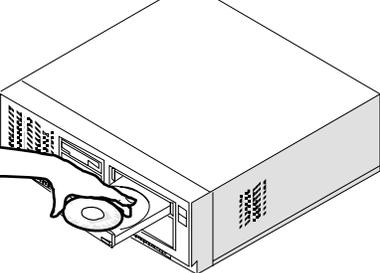
- *Solaris PowerPC Edition: Hardware Compatibility List*
Contains vendor-specific information on compatible hardware.
- *Installation Notes*
Describes any late-breaking news about installing Solaris software including known problems.
- *Solstice AdminSuite 2.2 Administration Guide*
Describes applications such as Solstice™ Host Manager, which you can use if you're setting up network installations.
- *System Administration Guide, Volume I*
Describes how to back up system files.

About Installing Solaris



Definition: Installing Solaris Software

As shown on the following pages, the process of installing Solaris software means copying it from the Solaris CD to a system's local disk.

Stage	Diagram	What Happens
<p>1</p>		<p>You choose a Solaris CD for your hardware platform. The Solaris CD contains the SunOS operating system and other software.</p>
<p>2</p>		<p>You insert the Solaris boot diskette into the system's a: diskette drive.</p>
<p>3</p>		<p>You load the Solaris CD into the CD-ROM drive.</p>

Stage	Diagram	What Happens
4	<pre>Automatically booting in 5 seconds. Hit any key to abort. ok boot net</pre>	After you power on the system, you choose to boot from the system's local CD-ROM drive, or from another system on the network. (Sample screen shown.)
5	<pre>What type of terminal are you using? 1 ANSI Standard CRT 2 DEC VT52 3 DEC VT100 4 Heathkit 19 5 Lear Siegler ADM31 6 PC Console 7 Sun Command Tool 8 Sun Workstation 9 Televideo 900 10 Televideo 925 11 Wyse Model 50 12 X Terminal Emulator (xterm) 13 Other Enter the number of your choice followed by the Enter key.</pre>	You choose a terminal type. (Sample screen shown.)
6	<pre>The Solaris Installation Program ----- You are now interacting with the Solaris installation program. The program is divided into a series of short sections. At the end of each section, you will see a summary of the choices you've made, and be given the opportunity to make changes. ----- Continue Exit Help</pre>	The Solaris installation program copies the Solaris software from the CD to the system's local disk. This is done <i>interactively</i> using a graphical (shown) or character interface, or <i>automatically</i> without user intervention.

Preparing to Install Solaris



This chapter guides you step-by-step through making decisions and completing the tasks required to prepare your system to install the Solaris software including:

<i>Determine if your system is networked.</i>	<i>page 6</i>
<i>Determine your system type.</i>	<i>page 6</i>
<i>Determine if you have required hardware.</i>	<i>page 8</i>
<i>Plan disk space and memory requirements.</i>	<i>page 9</i>
<i>Back up your system.</i>	<i>page 11</i>
<i>Choose a method for installing Solaris software.</i>	<i>page 11</i>
<i>Choose a task map and complete required tasks.</i>	<i>page 12</i>

Why You Should Not Ignore This Chapter

Successfully installing Solaris software requires a clear sense of what you're doing and why, or it can become difficult. This chapter provides all the information you need to determine the best way to install the Solaris software for your site. This chapter also directs you to specific chapters of this book you'll need.

Steps to Prepare to Install Solaris

Follow these steps before turning on your system.

1 Determine if your system is networked.

If your system is networked (connected to a network), an Ethernet connector or similar network adapter should be plugged into your system.

2 Determine your system type.

Before installing Solaris software, you must determine the *system type* which determines where the system gets important file systems. Using Figure 2-1 on page 7, choose a system type from the following lists:

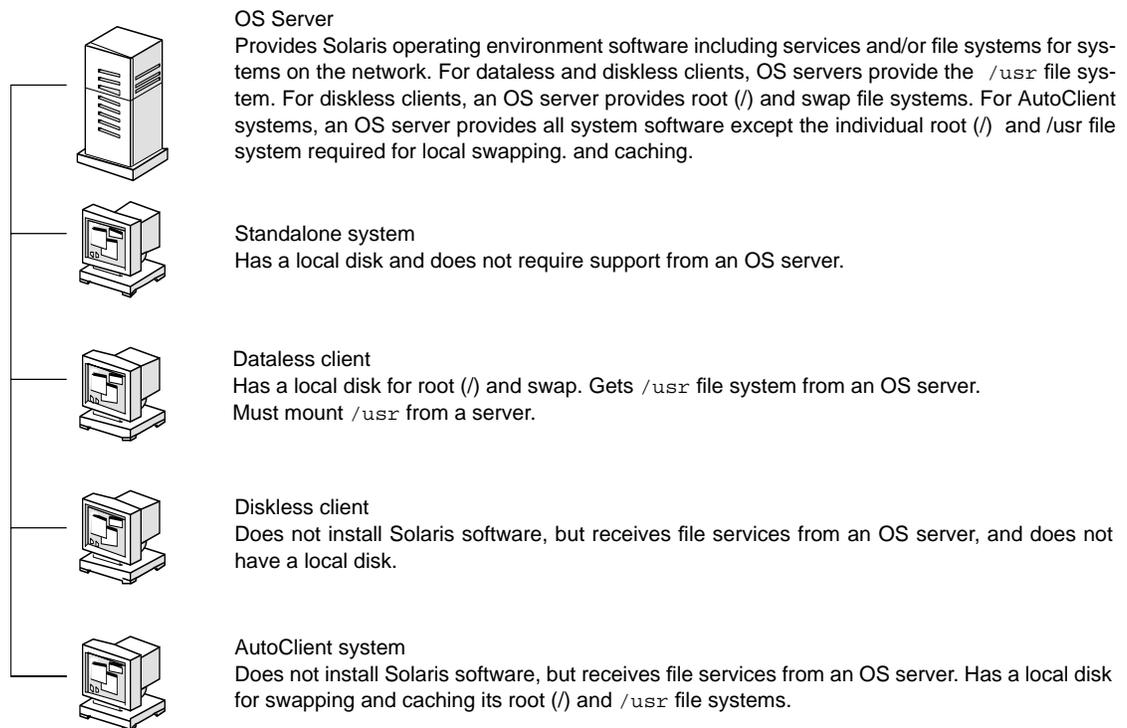
Networked systems:

- *OS server*
- *Standalone system*
- *Dataless client*
- *Diskless client*
- *AutoClient™ system*

Non-networked systems:

- *Standalone system*

Networked systems



Non-networked systems

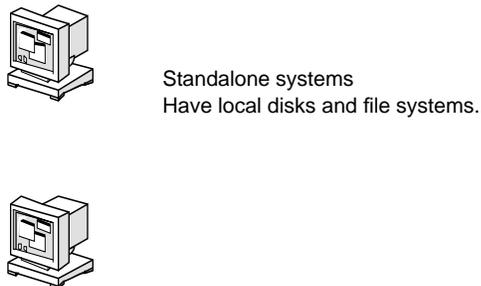


Figure 2-1 System Types

Note – A standalone system in the Solaris operating environment applies to *both* networked and non-networked systems. Whether networked or non-networked, a standalone system has all of its Solaris software on local hard disk, and does not require services from another system.

- 3 **If you are setting up diskless clients and/or AutoClient systems, and the OS server providing Solaris software is already installed with Solaris, do not go further in this book.**

Diskless clients and AutoClient systems do not install Solaris software; instead, they receive file services from an OS server. If the OS server providing Solaris software already has Solaris software installed, refer to the *Solstice AdminSuite 2.2 Administration Guide* for information on setting up diskless clients, or the *Solstice AutoClient 1.0 Administration Guide* for setting up AutoClients.

If you do not have an OS server set up with Solaris software, continue in this book.

- 4 **If you are setting up dataless clients, determine if you have required software.**

If you are setting up dataless clients to boot over the network from an OS server, you must have the Solstice™ Host Manager, which comes with the server software.

Note – SunSoft™ plans to remove support for the dataless client system type after Solaris 2.5. You can select this system type now, but in future releases you will need to select a different type.

- 5 **Determine if you have required hardware.**

For a detailed description of hardware requirements, see the *Solaris PowerPC Edition: Hardware Compatibility List*.

Table 2-1 Hardware Requirements

Hardware Platform	Minimum Memory	Disk Interfaces	Buses	Device for Installing Solaris
PowerPC™ system ¹	16 Mbytes	SCSI or IDE hard disk	<ul style="list-style-type: none">• PCI• ISA	You must have a diskette drive, and one of the following devices for installing Solaris software: <ul style="list-style-type: none">• Local CD-ROM drive• Remote CD-ROM drive available over the network• Remote hard disk available over the network

1. See the *Solaris PowerPC Edition: Hardware Compatibility List* for a list of supported systems.

6 Plan disk space and memory requirements.

There are many variables involved in determining disk space and memory requirements. Determining physical memory is easy—the minimum is 16 Mbytes, but more is better. While a system requires 32 Mbytes of virtual memory (physical and swap file/slices), determining requirements beyond this is not easy; it depends on the type of applications that are running. Determining disk space is also not so easy; it depends on the following:

- System type selected (for example, OS server, diskless client, AutoClient system, standalone system, dataless client)
- Language selected (for example, Chinese, Japanese, Korean)
- Software group selected (see following table)

If You Want to Install This Software Group ...	Which Installs ...	Then You'll Need Approximately This Much Disk Space For Solaris Domestic CD ...
Core System Support	The minimum software required to boot and run Solaris software.	73 Mbytes
End User System Support	The core group plus the recommended software for an end user including OpenWindows and the DeskSet software.	173 Mbytes
Developer System Support	The end user software plus software for developing software including libraries, include files, man pages, and programming tools. Compilers and debuggers are not included.	284 Mbytes
Entire Distribution	The entire Solaris release (everything on the CD). Compilers and debuggers are not included.	315 Mbytes

- Software packages that are selected or deselected in the software group (for example, you select Chinese, but deselect the 200-Mbyte font package)
- Disks selected (for example, ten 104-Mbyte disks will waste more space trying to make things fit than a single 1-GByte disk)
- Solaris file system overhead or local file systems (for example, mail, printer spooling, users' personal file systems, swap space)
- Size of other applications that are running (for example, AnswerBooks, PowerPC compilers, FrameMaker[®], IslandDraw[®])

Depending on which installation method you choose, you can perform a “dry run” to determine if you have enough disk space before actually installing Solaris software.

- **Interactive** – This method of installation lets you interactively select disks, locales, software, and lay out file systems; you can determine your requirements from a summary at the end of the session, and exit the program prior to installing Solaris software.
- **Custom JumpStart** – This method of installation lets you run the `pfinstall` command to test specific installation profiles without actually installing the Solaris software on a system.

7 Back up your system.

If your system has any files on it that you want to save, make sure you perform a backup. The safest way to back up files is to do a level 0 dump. If you do not have a backup procedure in place, see *System Administration Guide, Volume I* for instructions.

8 Choose a method for installing Solaris software.

There are two methods for installing Solaris software:

- ① **Interactive**—easy, hands-on method of installing Solaris software. The Solaris installation program guides you step-by-step through identifying your system and installing Solaris software. You're in control all the way!
- ② **Custom JumpStart** (formerly called auto-install)—for the advanced user with experience in Bourne shell scripting. By creating profiles and rules files, you can set up systems to automatically install Solaris software in different ways on different systems. This method requires up-front work before systems are turned on, but it's the most cost-effective way to install Solaris software for large, enterprise sites. You can even set up a hands-off installation where the user just boots the system and nothing more needs to be done!

Methods	Booting Phase	Identifying Your System	Installing Solaris
① Interactive	<div style="border: 1px solid black; padding: 5px; width: fit-content;">You boot system</div>	<div style="border: 1px solid black; padding: 5px; width: fit-content;">You interact with the Solaris installation program to identify your system</div>	<div style="border: 1px solid black; padding: 5px; width: fit-content;">You interact with the Solaris installation program to install Solaris software</div>
② Custom JumpStart	<div style="border: 1px solid black; padding: 5px; width: fit-content;">You boot system</div>	<div style="border: 1px solid black; padding: 5px; width: fit-content;">You interact with the Solaris installation program to identify your system</div>	<div style="border: 1px solid black; padding: 5px; width: fit-content; background-color: #f0f0f0;">Solaris software is automatically installed using a profile that you create in advance</div>

9 Choose a task map and complete required tasks.

Choose the task map from the following pages that matches the method you've chosen for installing Solaris software: interactive or custom JumpStart. The task maps guide you through all the tasks you need to complete before turning on, booting, and installing Solaris software on a system.

Task Map: Interactive Installations

Activity	Description	For Instructions, Go To	
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin-bottom: 10px;">Gather information</div> <div style="border: 1px solid black; padding: 5px; width: fit-content;">Install Solaris Software</div>	Optional. Use worksheets to gather information that may need to be supplied during the Solaris installation program.	Appendix A, "Worksheets for the Solaris Installation Program"	page 125
	From another system on the network Set up systems (standalone, OS server, or dataless client) for network installations.	Chapter 3, "Preparing to Install Solaris Over a Network"	page 15
	From local CD-ROM Boot and install Solaris software.	Chapter 7, "Booting and Installing Solaris: Custom JumpStart"	page 115

Figure 2-2 Task Map for Interactive Installations

Task Map: Custom JumpStart Installation

Activity	Description	For Instructions, Go To	
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin-bottom: 10px;"> Set up system for custom JumpStart </div> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> Install Solaris Software </div>	Perform the following tasks: <ul style="list-style-type: none"> • Create a JumpStart directory • Enable clients to access the JumpStart directory • Create profiles • Create a <code>rules</code> file • Use <code>check</code> to validate the <code>rules</code> file 	Chapter 4, "Preparing Custom JumpStart Installations"	page 45
	From another system on the network Set up systems (standalone, OS server, or dataless client) for network installations.	Chapter 3, "Preparing to Install Solaris Over a Network"	page 15
	From local CD-ROM Boot and install Solaris software.	Chapter 7, "Booting and Installing Solaris: Custom JumpStart"	page 115
		Chapter 7, "Booting and Installing Solaris: Custom JumpStart"	page 115

Figure 2-3 Task Map for Custom JumpStart Installations

Preparing to Install Solaris Over a Network



<i>How to Set Up a New System to be an Install Server</i>	<i>page 20</i>
<i>How to Create an Install Server</i>	<i>page 22</i>
<i>How to Create a Boot Server on a Subnet</i>	<i>page 25</i>
<i>How to Set Up Servers for Network Installation of a Standalone, Server, or Dataless Client</i>	<i>page 29</i>
<i>How to Preconfigure Default Locale Using NIS</i>	<i>page 38</i>
<i>How to Preconfigure Default Locale Using NIS+</i>	<i>page 41</i>

About Installing Solaris Over a Network

A typical way to install Solaris software is to use the installation program to copy the Solaris CD shipped with your system. However, it is uncommon at most sites for every system to have its own local CD-ROM drive. When a system does not have a local CD-ROM drive, you can perform a *network installation*. Network installation means that you install software over the network—from a system with a CD-ROM drive to a system without a CD-ROM drive.

Note – Instructions in this chapter are valid for either an x86, PowerPC, or SPARC server being used for network installations. An x86 server can provide the Solaris CD image for SPARC and PowerPC systems, a SPARC server can provide the Solaris CD image for x86 and PowerPC systems, and a PowerPC server can provide the Solaris CD image for x86 and SPARC systems.

Servers Required for Network Installation

As shown in Figure 3-1, systems that install Solaris software over the network require:

- *Install server* – A networked system with the CD-ROM drive that provides installation services for other systems.
- *Name server* – A system that manages a distributed network database (such as NIS or NIS+) containing information about users and other systems on the network.

Note – The install server and name server may be the same or separate systems.

- *Boot server* – A system that boots the system to be installed over the network. A boot server and install server are typically the same system. However, if the system to be installed is on a *different* subnet than the install server, a boot server is required on that subnet.

Dataless clients also require:

- *OS server* – A system that provides Solaris operating environment software including services and/or file systems. For dataless clients, OS servers provide the `/usr` file system.

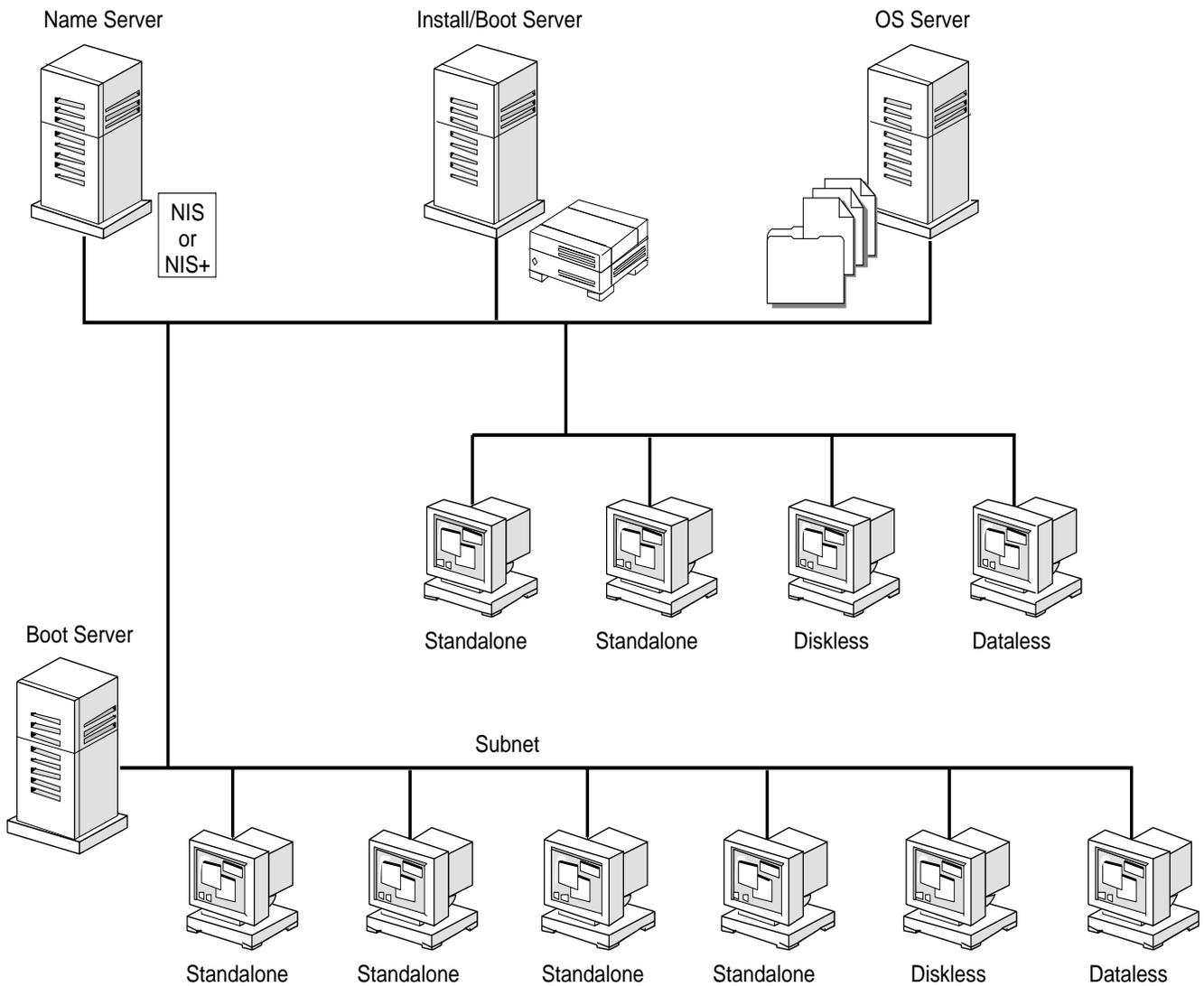


Figure 3-1 Network Installation Servers

Requirements for Hands-Off Network Installation

To set up your site to install Solaris software on systems over the network with no user intervention, you must:

- Use the custom JumpStart installation method. (See Chapter 4, “Preparing Custom JumpStart Installations.”)
- Make sure all systems are properly configured in the name service. (Procedures in this chapter include information on how to add systems to the name service.)
- Preconfigure network information, such as the date, time, geographic region, site subnet mask, and language. By using the Solstice Host Manager to set up a server for network installations, you automatically preconfigure network information. This eliminates many prompts that are otherwise necessary to identify the system during an installation.

Commands You Should Know About

Table 3-1 shows programs available for use when setting up for network installation.

Table 3-1 Network Installation Commands

Program	Description
<code>setup_install_server</code>	A script that copies all or part of the Solaris CD onto a server's local disk. This enables you to perform network installations from the install server's disk instead of its CD-ROM drive. (Installing from the install server's disk is faster than installing from the server's CD-ROM drive.) See the <code>setup_install_server(1m)</code> man page for more information.
Host Manager	A graphical user interface that is available from within the Solstice AdminSuite (<code>solstice</code>). You can use Host Manager to specify naming services, system's IP and Ethernet addresses, and other information to be used for installations across a network.
<code>mount</code>	A command that shows mounted file systems, including the Solaris CD file system. See the <code>mount(1M)</code> man page for more information.
<code>uname -m</code>	A command for determining a system's platform name (for example, <code>prep</code>). This information is sometimes required during installation. See the <code>uname(1)</code> man page for more information.
<code>kdmconfig</code>	A command to run on the install server to preconfigure the keyboard, display, and other peripherals for a system being installed over a network. See the <code>kdmconfig(1M)</code> man page for more information.

Creating an Install Server

If you are installing systems over the network, you must have an install server—a system with a CD-ROM drive or with Solaris software copied to its local disk. This system will provide the installation services for systems on the network that do not have a local CD-ROM drive.

You can create an install server to install the following system types:

- OS server
- Standalone system
- Dataless client

This section describes how to:

- Set up a new system to be an install server.
- Create an install server by mounting the Solaris CD or by copying the Solaris CD to the install server's local disk.

Note – If you intend to do frequent installations over the network, you should copy the Solaris CD image from the Solaris CD to the install server's local disk. Network installations from the install server's local disk are faster than from its CD-ROM drive. Copying Solaris CD image to the install server's disk also frees the CD-ROM drive for other uses.

- Create a separate boot server (required *only* if systems are not on same subnet as the install server).

▼ How to Set Up a New System to be an Install Server

Overview – Setting up a new system to be an install server involves:

- Installing the system hardware, including a CD-ROM drive
- Connecting the system to a network
- Installing Solaris software

Follow this procedure to create an install server

- 1. Install the system hardware, including a CD-ROM drive.**
Follow the hardware documentation for detailed information.

2. Connect the system to the network.

Follow the hardware documentation for detailed information.

3. Install Solaris software.

During an interactive installation, you are prompted to select a name service. Follow the guidelines in the next table:

Is a Name Service Running at Your Site ...	Then ...
Yes	Select NIS or NIS+, whichever your site uses, when prompted for a name service. After the Solaris software is installed, go to How to Create an Install Server on page 22.
No	Select None when prompted for a name service. After the Solaris software is installed, go to Step 4.

4. If you want this system to be a name server, set up a name service after you have installed the Solaris software. For detailed information, see the *NIS+ and DNS Setup and Configuration Guide*.

Task Complete

The system is now set up so that you can make it an install server. Next, you must make a Solaris CD image available to other systems. To continue, go to the How to Create an Install Server on page 22.

▼ **How to Create an Install Server**

Overview – Creating an install server involves:

- Choosing a system with a CD-ROM drive to be the install server
- Mounting the Solaris CD
- Using the `setup_install_server` command to copy the Solaris CD to the install server’s local disk (optional, but recommended)

Follow this procedure to create an install server.

1. On the system that is going to be the install server, log in and become root.

This system must have a CD-ROM drive.

2. Insert the Solaris CD into the CD-ROM drive.

3. Mount the Solaris CD (if needed) and change the directory to the mounted CD:

If the Install Server Is ...	Then ...
Running Volume Management	<code>cd /cdrom/cdrom0/s0</code> or <code>cd /cdrom/cdrom0/s2</code>
Not running Volume Management	1) <code>mount -F hsfs -o ro /dev/dsk/c0t6d0s0 /cdrom</code> or <code>mount -F hsfs -o ro /dev/dsk/c0t6d0p0 /cdrom</code> 2) <code>cd /cdrom</code>

Note: Volume Management is running if the `/vol` directory on the system contains files. Systems running Solaris 2.0 or 2.1 do not have Volume Management.

4. Determine your next step based on whether or not you want to copy the Solaris CD to the install server’s local disk:

If You ...	Then ...
Want to copy the Solaris CD	Go to Step 5.
Do not want to copy the Solaris CD	Go to Step 7.

5. Use the `setup_install_server` command to copy the contents of the Solaris CD to the install server's local disk.

```
# ./setup_install_server install_dir_path
```

In this command,

<i>install_dir_path</i>	Specifies the directory where the Solaris CD image will be copied. You can substitute any directory path.
-------------------------	---

For example, the following command copies the Solaris CD image from the Solaris CD to the `/export/install` directory on the local disk:

```
./setup_install_server /export/install
```

Note - The `setup_install_server` command will indicate if you do not have enough disk space to copy the Solaris CD image from the Solaris CD. Use the `df -k1` command to determine available disk space.

6. Type `cd install_dir_path` and press Return.

7. Determine your next step based on whether or not the install server is on the same subnet as the system to be installed.

If Install Server Is ...	Then ...
On same subnet as the system to be installed	Go to Task Complete on page 24.
Not on the same subnet as the system to be installed	Follow the procedure How to Create a Boot Server on a Subnet on page 25. You must complete this procedure when the install server is <i>not</i> on the same subnet as the system to be installed. After completing that procedure, go to Task Complete on page 24.



The install server is now created. Next, you must add information to the install server's configuration files so it recognizes the systems to be installed. To continue, go to How to Set Up Servers for Network Installation of a Standalone, Server, or Dataless Client on page 29.

▼ How to Create a Boot Server on a Subnet

You can install Solaris software over the network from any install server on the network. However, a system that will use an install server on another subnet *requires* a separate boot server on its own subnet.

Overview – Creating a boot server involves:

- Choosing a system with a CD-ROM drive to be the boot server for the subnet
- Mounting the Solaris CD
- Using the `setup_install_server -b` command to copy required platform information from the Solaris CD to the local disk

Follow this procedure to set up a boot server on a subnet.

1. On the system that will be the boot server for the subnet, log in and become root.

This system must have a CD-ROM drive or an NFS mount of a Solaris CD image. The system must also be in the NIS or NIS+ name service. (If your site doesn't use the NIS or NIS+ name service, you must distribute information about this system by following your site's policies.)

2. Determine your next step based on whether the boot server uses a local CD-ROM drive or an NFS mount of a Solaris CD image.

If the Boot Server Uses ...	Then ...
Local CD-ROM drive	1) Insert the Solaris CD into the CD-ROM drive. 2) Go to Step 3.
NFS mount of a Solaris CD image	1) <code>mount -F nfs -o ro server_name:path /mnt</code> where <code>server_name:path</code> is the host name and absolute path to the Solaris CD image. 2) <code>cd /mnt</code> 3) Go to Step 4.

3. Mount the Solaris CD (if needed) and change the directory to the mounted CD:

If the Boot Server Is ...	Then ...
Running Volume Management	<code>cd /cdrom/cdrom0/s0</code> or <code>cd /cdrom/cdrom0/s2</code>
Not running Volume Management	1) <code>mount -F hsfs -o ro /dev/dsk/c0t6d0s0 /cdrom</code> or <code>mount -F hsfs -o ro /dev/dsk/c0t6d0p0 /cdrom</code> 2) <code>cd /cdrom</code>

Note: Volume Management is running if the `/vol` directory on the system contains files. Systems running Solaris 2.0 or 2.1 do not have Volume Management.

4. Use the `setup_install_server` command to set up the boot server for the subnet.

The `setup_install_server` command copies all supported platform information to the local disk.

```
# ./setup_install_server -b boot_dir_path
```

In this command,

<code>-b</code>	Specifies that the system will be set up as a boot server.
<code><i>boot_dir_path</i></code>	Specifies the directory where the platform information will be copied. You can substitute any directory path.

For example, the following command copies platform information necessary for booting supported platforms over the network:

```
./setup_install_server -b /export/install
```

Note – The `setup_install_server` command will indicate if you do not have enough disk space to copy the platform dependent information. Use the `df -kl` command to determine available disk space.

Task
Complete

The boot server is now set up to boot supported systems on a subnet. To continue, go to [How to Set Up Servers for Network Installation of a Standalone, Server, or Dataless Client](#) on page 29.

Setting Up Servers for Network Installation

This section describes how to set up appropriate servers necessary to install a system over a network. The next table shows servers required for each system type you want to install.

If System Type You Are Installing Is ¹ ...	Then You Need This Server Support ...
Standalone system	Install server
OS server	Install server
Dataless client	Install server and OS server

1. Systems also require a boot server if they are on a different subnet than the install server.

Note – When you install a server, you must allocate disk space required for the clients that server will support. Then, *after* the server is installed, you must use the Solstice Host Manager and add the platform support required by those clients.

▼ How to Set Up Servers for Network Installation of a Standalone, Server, or Dataless Client

Once you have an install server set up, you are ready to use it to install Solaris software on other systems on the network. Before you can actually do that, however, you need to provide some basic system information about the systems that you are going to install. You do this by using the Solstice Host Manager.

Note – SunSoft plans to remove support for the dataless client system type after Solaris 2.5. You can set up this system type now, but in future releases you will need to change it to a different type (standalone, OS server, diskless client, or AutoClient system.)

The procedure to add system information about standalone systems, servers, and dataless clients to the install server can all be accomplished using the Solstice Host Manager. You simply fill out a form and apply the information you provide, and the Solstice Host Manager updates the appropriate files and name server maps or tables.

Overview – This procedure involves:

- Using the Solstice Host Manager to update the name service, to add information about the standalone, server, or dataless client to the install server's configuration files, and, if necessary, to set up a file server

Follow this procedure to set up the install server to install a standalone system, a server, or a dataless client on the network.

1. On the install server, log in as a user in the administration group (group 14).

Caution – If your system is part of the NIS+ name service, you must run the Solstice AdminSuite™ while logged in as a user in the NIS+ admin group. Otherwise, you will not have permission to update configuration information on the name server.

2. Start the Solstice AdminSuite with the following command.

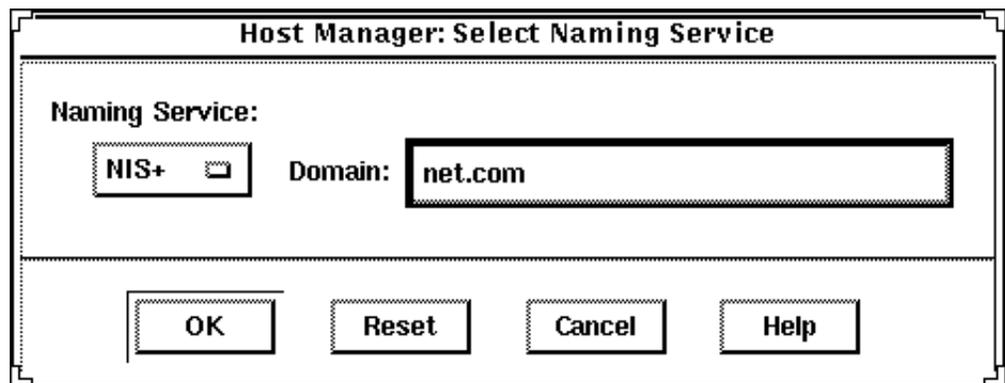
```
$ /usr/bin/solstice &
```

3. After the Solstice AdminSuite main window appears, click on the Host Manager icon.

Note – If your site uses the Domain Name Service (DNS), you will have to modify the `/etc/nsswitch.conf` file and create the `/etc/resolv.conf` file manually. For detailed information, see the *NIS+ and FNS Administration Guide*.

4. On the Host Manager: Select Naming Service screen, select a naming service and click on the Apply button. See the sample Naming Service screen below.

If the Name Service Is ...	Then Select ...
NIS+	NIS+. Host Manager will update the NIS+ tables.
NIS	NIS. Host Manager will update the NIS maps.
None	None. Host Manager will store the information you provide in the <code>/etc</code> files. This will provide enough information for systems to boot, and to install Solaris software over the network.



5. On the Host Manager main window, choose Add... from the Edit menu.

6. Determine your next step based on what kind of system you want to install.

System to Be Installed Is ...	Then ...
Standalone system	Go to Step 7.
OS server	Go to Step 10.
Dataless client	Go to Step 13.

7. On the Host Manager: Add Host screen, complete all fields and click on the OK or Apply button.

Type - Select standalone.

Install Server - If enabling remote install for a network installation, specify the install server and set the path to the location of the Solaris CD image.

Boot Server - If the install client is on a different subnet than the install server, specify the boot server that resides on the install client's subnet.

Profile Server - If using custom JumpStart installations, specify the system with the custom Jumpstart profiles on it.

Remote Install - Select if setting up a network installation. The install server should have been set up already.

8. On the Host Manager main window, choose Save Changes from the File menu.

9. Go to Task Complete on page 35.

Step 10 through Step 13 are only required if you are adding an OS server or dataless client.

10. On the Host Manager: Add Host screen, complete all fields and click on the OK or Apply button.

Type - Select OS server.

Install Server - If enabling remote install for a network installation, specify the install server and set the path to the location of the Solaris CD image.

Boot Server - If the install client is on a different subnet than the install server, specify the boot server that resides on the install client's subnet.

Profile Server - If using custom JumpStart installations, specify the system with the custom Jumpstart profiles on it.

Remote Install - Select if setting up a network installation. The install server should have been set up already.

Solstice Host Manager: Add

Host Name:

IP Address:

Ethernet Address:

System Type: **OS server**

Timezone Region: **United States**

Timezone: **Mountain**

Remote Install: **Enable Remote Install**

Install Server: **pluto** **Set Path...**

OS Release:

Boot Server: **none**

Profile Server: **none**

OK **Apply** **Reset** **Cancel** **Help**

11. On the Host Manager main window, choose **Save Changes** from the **File** menu.

12. Go to **Task Complete** on page 35.
Step 13 is required only for adding dataless clients for network installation.

13. On the Host Manager: Add Host screen, complete all fields and click on the OK or Apply button.

Solstice Host Manager: Add

Host Name:

IP Address:

Ethernet Address:

System Type:

Timezone Region:

Timezone:

File Server: File Server - Specify the system that you want to provide /usr file systems for the dataless client.

OS Release: OS Release - Select the OS release you want installed on the dataless client. The proper OS support must reside on the install server. (You set up this support by using Host Manager after the server has been installed. This is known as "adding services.")

Remote Install: Enable Remote Install

Install Server:

Boot Server:

Profile Server:

OS Release - Select the OS release you want installed on the dataless client. The proper OS support must reside on the install server. (You set up this support by using Host Manager after the server has been installed. This is known as "adding services.")

The example shows the OS release for a SPARC image. For an x86 image it would be: i386 i86pc Solaris 2.x; for a PowerPC image it would be: ppc prep Solaris 2.x.)

14. On the Host Manager main window, choose Save Changes from the File menu.

Task
Complete

The standalone system, OS server, or dataless client is now added for network installation. You are now ready to boot and install over the network. To find the correct booting and installing procedure in this book, see the appropriate chapter for the installation you want to perform:

- Chapter 6, “Booting and Installing Solaris: Interactive”
- Chapter 7, “Booting and Installing Solaris: Custom JumpStart”

Using `add_install_client` to Set Up Servers for Network Installation

Install servers running versions of Solaris released prior to Solaris 2.3.2 software may not have access to the Solstice AdminSuite. You can install the Solstice products on Solaris 2.3 and 2.4 systems and follow the instructions in How to Set Up Servers for Network Installation of a Standalone, Server, or Dataless Client on page 29. Alternatively, you can use the `add_install_client` command to set up the install server for network installations, as described in the next section.

▼ **How to Use `add_install_client` to Set Up Servers for Network Installation**

Use the `add_install_client` command to add information about the system to be installed to the boot server configuration files.

```
# ./add_install_client [-c server:jumpstart_dir_path] -s install_server:install_dir_path host_name platform_group
```

In this command,

- `-c` Specifies a JumpStart directory for custom JumpStart installations. This option and its arguments are required only for custom JumpStart installations.
- `server:jumpstart_dir_path` `server` is the host name of the server on which the JumpStart directory is located. `jumpstart_dir_path` is the absolute path of the JumpStart directory.
- `-s` Specifies the install server.
- `install_server:install_dir_path` `install_server` is the host name of the install server. `install_dir_path` is the absolute path name of the mounted Solaris CD or the directory that has the copy of the Solaris CD image.
- `host_name` Is the host name of the standalone system or the server where Solaris software will be installed over the network. (This is not the host name of the install server). The host must be in the name service for this command to work.
- `platform_group` Is the platform group of the system to be installed. (For a detailed list of platform groups, See Appendix C, “Platform Names and Groups.”)

For example, the following command copies boot information from Solaris CD image on an install server’s local disk in `/export/install`. `add_install_client` sets up the `/rplboot` directory on the local system,

which will be the boot server. The system that will be installed is named `watson`, and it is a PowerPC system. The platform group for a PowerPC system is `prep`.

```
./add_install_client -s install_server1:/export/install watson prep
```

Task
Complete

The standalone system, server, or dataless client is now added for network installation. You are now ready to boot and install over the network. To find the booting and installing procedure in this book, see the appropriate chapter for the installation you want to perform:

- Chapter 6, “Booting and Installing Solaris: Interactive”
- Chapter 7, “Booting and Installing Solaris: Custom JumpStart”

Preconfiguring the Default Locale

When installing a localized version of Solaris software, you are prompted for the locale (the language) you want to use for the duration of the installation process. The locale you choose for installing Solaris software is also the default locale the installed version of Solaris will provide to the system's user.

You can set up a default locale of your choice. You do this by modifying the naming service (NIS or NIS+). After you modify the name service, as described in this section, the operating system will use this default locale for users. Also, the installation software will use the default locale for all future installations.

Choose from the following two procedures, depending on whether your site uses the NIS or NIS+ name service.

▼ How to Preconfigure Default Locale Using NIS

Overview – The procedure to preconfigure the default locale at sites using the NIS name service involves:

- Using the `vi` or text editor to edit files in the `/var/yp/Makefile` file
- Creating a locale file in the `/etc` directory
- Updating the NIS maps

Follow these instructions to set up a default locale for the system being installed over the network.

1. On the name server, log in and become root and edit the `/var/yp/Makefile` file.
Add the following text after the other `*.time` entries.

```
locale.time: $(DIR)/locale
    -@if [ -f $(DIR)/locale ]; then \
        sed -e "/^#/d" -e s/#.*$$// $(DIR)/locale \
        | awk '{for (i = 2; i<=NF; i++) print $$i, $$0}' \
        | $(MAKEDBM) - $(YPDBDIR)/$(DOM)/locale.byname; \
        touch locale.time; \
        echo "updated locale"; \
        if [ ! $(NOPUSH) ]; then \
            $(YPPUSH) locale.byname; \
            echo "pushed locale"; \
        else \
            : ; \
        fi \
    else \
        echo "couldn't find $(DIR)/locale"; \
    fi
```

2. Edit the `/var/yp/Makefile` file.
 - a. Add `locale` to the line starting with the word `all`.
 - b. Add `locale: locale.time` on a new line.

```
all: passwd group hosts ethers networks rpc services protocols netgroup bootparams aliases \
timezone locale
locale: locale.time
```

3. Create the file `/etc/locale` and make one entry for each domain.

```
domain_name locale
```

The entry specifies the default locale for the domain. For example, the following line specifies French to be the default locale for the `worknet.com` domain:

```
worknet.com fr
```

You can also use a host name instead of the domain name to specify the default locale for a particular host. For example, the following line specifies Korean to be the default locale for system named `sherlock`:

```
sherlock ko
```

See the table on page 42 for a list of valid locale values.

Note – Not all locales are available on all Solaris CDs. The locale you select will be used for installation if it is present on the Solaris CD.

4. Make the maps.

```
# cd /var/yp; make
```

Task
Complete

Systems in the name service are now set up to use the default locale. The default language you have specified will be used during the installation and will also be the language provided to system users.

▼ How to Preconfigure Default Locale Using NIS+

Overview – The procedure to preconfigure the default locale at sites using the NIS+ name service involves:

- Using the `nistbladm` command to create a locale table and add information to it
- Updating the NIS+ tables

Note – The installation software is not translated to the Asian locales, so this procedure will not work for any of the Asian locales.

Follow these instructions to set up a default locale for a system being installed over the network. (This procedure assumes the NIS+ domain is set up. Setting up the NIS+ domain is documented in the *NIS+ and FNS Administration Guide*.)

1. **Log in to a name server as root or as a user in the NIS+ admin group.**
2. **Use the following `nistbladm` command to create the locale table.**

```
# nistbladm -D access=og=rmcd,nw=r -c locale_tbl
name=SI,nogw= locale=,nogw= comment=,nogw=
locale.org_dir.'nisdefaults -d'
```

3. Add an entry to the `locale.org_dir` table by typing the following `nistbladm` command.

```
# nistbladm -a name=domain_name locale=locale comment=comment
locale.org_dir.'nisdefaults -d'
```

In this command,

domain_name Is either the domain name or a specific host name for which you want to preconfigure a default locale.

locale Is the locale you want to use to install the system and the locale you want to come up on users' systems. The following table shows valid values for *locale*.

<u>Language</u>	<u>Valid <i>locale</i> Values</u>
Chinese	zh
English (Solaris default)	C
French	fr
German	de
Italian	it
Japanese	ja
Korean	ko
Latin American	es
Swedish	sv
Taiwanese	zh_TW

comment Is the comment field. Use double quotation marks to begin and end comments that are longer than one word.

Note – Not all locales are available on all Solaris CDs. The locale you select will be used for installation if it is present on the Solaris CD.

Task
Complete

Systems in the name service are now set up to use the default locale. The default language you have specified will be used during the installation and will also be the language provided to system users.

Preparing Custom JumpStart Installations



<i>How to Create a JumpStart Directory on a Diskette for x86 and PowerPC Systems</i>	<i>page 51</i>
<i>How to Create a JumpStart Directory on a Server</i>	<i>page 57</i>
<i>How to Enable All Systems to Access the JumpStart Directory</i>	<i>page 59</i>
<i>How to Create a Profile</i>	<i>page 62</i>
<i>How to Create the rules File</i>	<i>page 77</i>
<i>How to Use check to Validate the rules File</i>	<i>page 88</i>

Definition: Custom JumpStart Installation

A custom JumpStart installation is a type of installation in which the Solaris software is automatically installed on a system based on a user-defined profile. You can create customized profiles for different types of users.

Note – Appendix D, “Sample Custom JumpStart Installation” provides an example of how a fictitious site is prepared for custom JumpStart installations.

Note – Instructions in this chapter are valid for either an x86, PowerPC, or SPARC server that is being used for custom JumpStart installations. An x86 server can provide custom JumpStart files for SPARC and PowerPC systems, a

SPARC system can provide custom JumpStart files for x86 and PowerPC systems, and a PowerPC server can provide custom JumpStart files for SPARC and x86 systems.

Reasons to Choose a Custom JumpStart Installation

You should choose custom JumpStart installations when you have to install Solaris software on:

- Many systems.
- Particular groups of systems.

For example, the following scenario would be ideal for performing custom JumpStart installations:

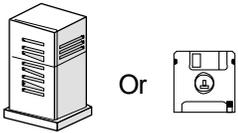
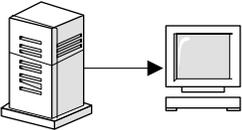
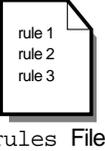
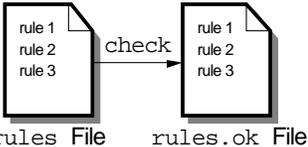
- You need to install the Solaris software on 100 new systems.
- The engineering group owns 70 out of the 100 new systems, and its systems must be installed as standalone systems with the developer software group.
- The marketing group owns 30 out of the 100 new systems, and its systems must be installed as standalone clients with the end user software group.

These installations would be time-consuming and tedious if you chose to perform an interactive installation on each system.

Tasks to Prepare for Custom JumpStart Installations

Table 4-1 shows the tasks that are required to prepare for custom JumpStart installations.

Table 4-1 Tasks to Prepare for Custom JumpStart Installations

Task		Description
Creating a JumpStart directory on a diskette or on a server		You must create a JumpStart directory to hold the custom JumpStart files. If you are going to use a diskette for custom JumpStart installations, see “Creating a JumpStart Directory on a Diskette” on page 51. If you are going to use a server for custom JumpStart installations, see “Creating a JumpStart Directory on a Server” on page 57.
Enabling all clients to access the JumpStart directory		When you use a server to provide the JumpStart directory, you can enable all clients to access the JumpStart directory. See “Enabling All Systems to Access the JumpStart Directory” on page 59 for detailed information.
Creating profiles		A profile is a text file used as a template by the custom JumpStart installation software. It defines how to install the Solaris software on a system (for example, initial or upgrade installation option, system type, disk partitioning, software group), and it is named in the <code>rules</code> file. See “Creating a Profile” on page 61 for detailed information.
Creating a <code>rules</code> file		The <code>rules</code> file is a text file used to create the <code>rules.ok</code> file. The <code>rules</code> file is a look-up table consisting of one or more rules that define matches between system attributes and profiles. See “Creating the rules File” on page 76 for detailed information.
Using <code>check</code> to validate the <code>rules</code> file		The <code>rules.ok</code> file is a generated version of the <code>rules</code> file, and it is required by the custom JumpStart installation software to match a system to a profile. You <i>must</i> use the <code>check</code> script to create the <code>rules.ok</code> file. See “Using <code>check</code> to Validate the rules File” on page 87 for detailed information.

What Happens During a Custom JumpStart Installation

Figure 4-1 describes what happens after you boot a system to perform a custom JumpStart installation.

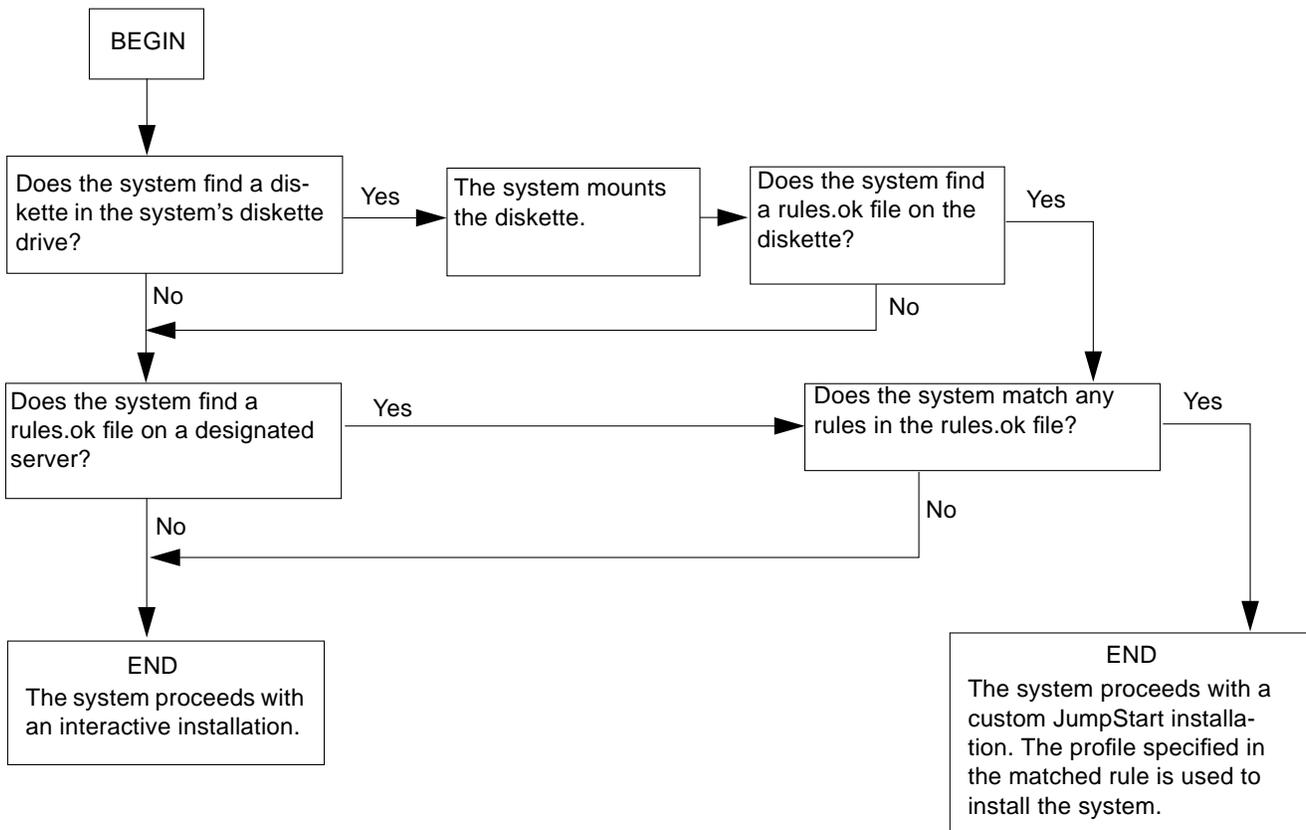


Figure 4-1 What Happens During a Custom JumpStart Installation

Figure 4-2 is an example of how a custom JumpStart installation works on a standalone, non-networked system using the system's diskette drive.

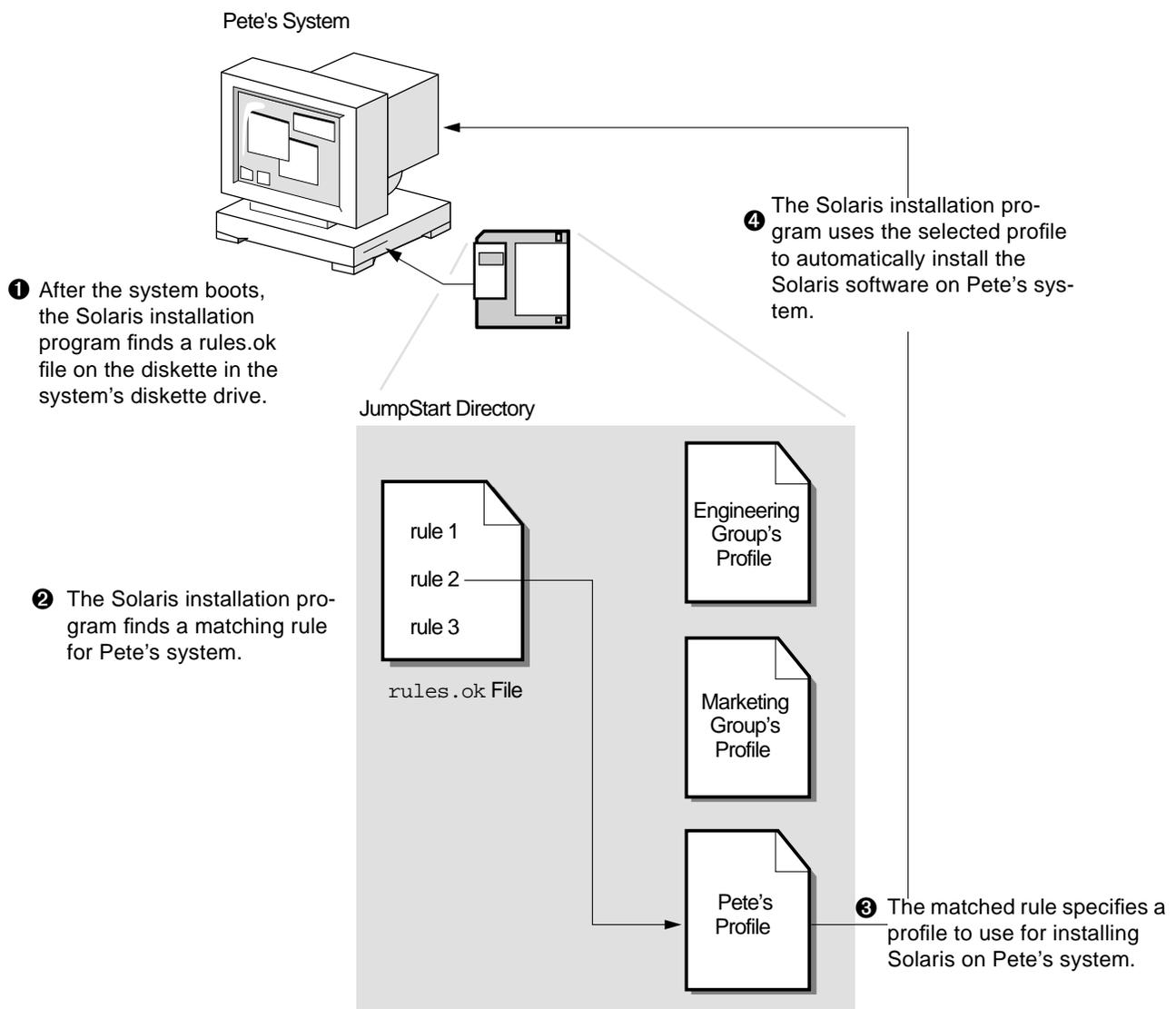


Figure 4-2 How a Custom JumpStart Installation Works: Non-Networked Example

Figure 4-3 is an example of how a custom JumpStart installation works for multiple systems on a network where different profiles are accessed from a single server.

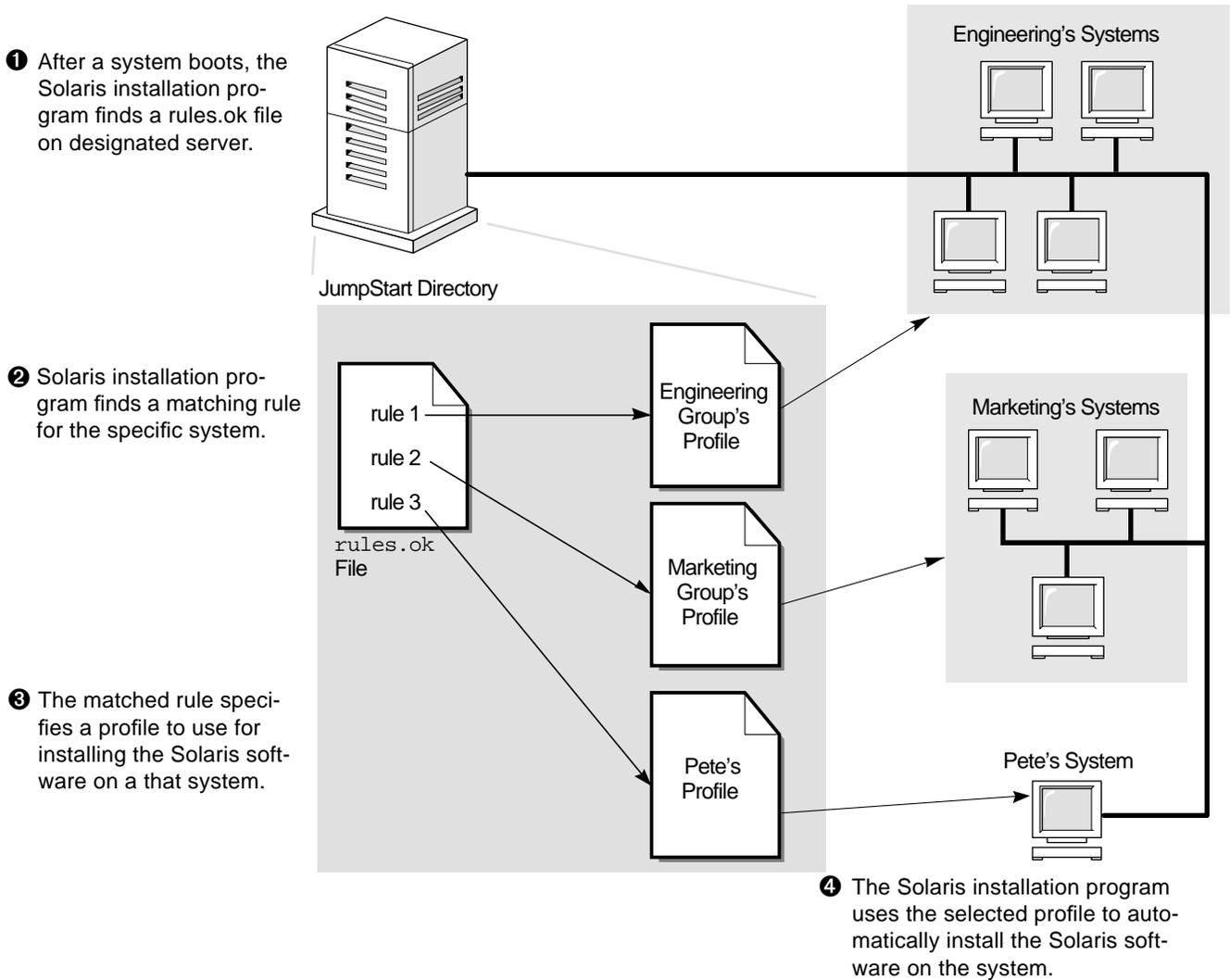


Figure 4-3 How a Custom JumpStart Installation Works: Networked Example

Creating a JumpStart Directory on a Diskette

You should use a diskette for a custom JumpStart installation if the system:

- Has a diskette drive
- Has a local CD-ROM drive
- Is *not* connected to a network

When you use a diskette for custom JumpStart installations, the JumpStart directory must be the root directory on the diskette that contains all the essential custom JumpStart installation files (for example, the `rules` file, `rules.ok` file, and profiles). The JumpStart directory should be owned by root and have permissions equal to 755.

The following section describes how to create a JumpStart directory on a diskette for x86 and PowerPC systems. (A SPARC system must be used to create a JumpStart directory on a diskette for SPARC systems.)

▼ How to Create a JumpStart Directory on a Diskette for x86 and PowerPC Systems

Overview – The procedure to create a JumpStart directory on a diskette for x86 and PowerPC systems involves:

- Making a copy of the Solaris boot diskette (the copied Solaris boot diskette has a PCFS file system)
- Copying sample custom JumpStart installation files into the diskette's root directory

Follow this procedure to create a JumpStart directory on a diskette for x86 systems.

1. **Log in as root on an x86, PowerPC, or SPARC system that has a diskette drive and a CD-ROM drive.**
2. **Insert the Solaris boot diskette into the diskette drive.**

3. Choose the appropriate steps, depending on whether or not the system is running Volume Management:

If the System Is ...	Then ...
Running Volume Management	Go to Step 4. Note: Volume Management is running if the <code>/vol</code> directory on the system contains files. Systems running Solaris 2.0 or 2.1 do not have Volume Management.
Not running Volume Management	Go to Step 12.

4. Make sure Volume Management knows about the diskette:

```
# volcheck
```

5. Copy the Solaris boot diskette image to the system's hard disk:

```
# dd if=/vol/dev/aliases/floppy0 of=boot_image
```

In this command,

boot_image Is the file name where the Solaris boot diskette image is copied. You can specify an absolute path name.

For example, the following command would copy a Solaris boot diskette to the `boot_save` file.

```
dd if=/vol/dev/aliases/floppy0 of=boot_save
```

6. Eject the Solaris boot diskette:

Note – The following command is not required for x86 and PowerPC systems; you must manually eject the diskette on an x86 and PowerPC system.

```
# eject floppy
```

7. Insert a blank diskette into the diskette drive.**8. Make sure Volume Management knows about the diskette:**

```
# volcheck
```

9. Format the diskette:

Caution – This step will overwrite any data on the diskette.

```
# fdformat -d -U
```

10. Copy the Solaris boot diskette image from the system's hard disk to the formatted diskette:

```
# dd if=boot_image of=/vol/dev/aliases/floppy0
```

The *boot_image* variable should be the same as in Step 5.

11. Go to Step 17.

Step 12 through Step 16 are used if the system is not running Volume Management.

12. Copy the Solaris boot diskette image to the system's hard disk:

```
# dd if=/dev/rdiskette of=boot_image
```

In this command,

boot_image Is the file name where the Solaris boot diskette image is copied. You can specify an absolute path name.

For example, the following command would copy a Solaris boot diskette to the *boot_save* file.

```
dd if=/dev/rdiskette of=boot_save
```

13. Eject the Solaris boot diskette:

Note - The following command is not required for x86 and PowerPC systems; you must manually eject the diskette on an x86 and PowerPC system.

```
# eject fd
```

14. Insert a blank diskette into the diskette drive.

15. Format the diskette:



Caution - This step will overwrite any data on the diskette.

```
# fdformat -d
```

16. Copy the Solaris boot diskette image from the system's hard disk to the formatted diskette:

```
# dd if=boot_image of=/dev/rdiskette
```

The *boot_image* variable should be the same as in Step 12.

17. Mount the diskette:

If the System Is ...	Then ...
Running Volume Management	<ol style="list-style-type: none"> 1) Eject the copied Solaris boot diskette. 2) Insert the copied Solaris boot diskette back into the diskette drive. 3) Make sure Volume Management knows about the diskette: volcheck
Not running Volume Management	<pre>mount -F pcfs /dev/diskette <i>jump_dir_path</i></pre> <p>Note: <i>jump_dir_path</i> is the absolute directory path where the diskette is mounted.</p>

18. Determine your next step based on where the Solaris CD is located.

If You Want to Use The ...	Then ...
Solaris CD in the local CD-ROM drive	<ol style="list-style-type: none"> 1) Insert the Solaris CD into the CD-ROM drive. 2) Go to Step 19.
Solaris CD image on local disk	<ol style="list-style-type: none"> 1) Change the directory to the Solaris CD image on the local disk. For example: cd /export/install 2) Go to Step 20.

19. Mount the Solaris CD (if needed) and change the directory to the mounted CD:

If the System Is ...	Then ...
Running Volume Management	cd /cdrom/cdrom0/s0 or cd /cdrom/cdrom0/s2
Not running Volume Management	1) mount -F hsfs -o ro /dev/dsk/c0t6d0s0 /cdrom or mount -F hsfs -o ro /dev/dsk/c0t6d0p0 /cdrom 2) cd /cdrom

20. Copy the JumpStart installation files from the auto_install_sample directory on the Solaris CD into the JumpStart directory (root directory) of the diskette:

If the System Is ...	Then ...
Running Volume Management	cp -r auto_install_sample/* /floppy/floppy0/.
Not running Volume Management	cp -r auto_install_sample/* <i>jumpstart_dir_path</i> Note: <i>jump_dir_path</i> is the absolute directory path where the diskette is mounted.



Caution – File names on PCFS file systems can be only 11 characters long (an 8-character file name and a 3-character extension). When copying JumpStart installation files to a diskette for x86 and PowerPC systems, be aware that the file transfer may truncate file names.

Note – The custom JumpStart installation files must be in the root directory of the diskette.



You have completed creating a JumpStart directory on the diskette. To continue, see “How to Create a Profile” on page 62.

Creating a JumpStart Directory on a Server

If you want to perform custom JumpStart installations by using a server on the network, you must create a JumpStart directory on the server. When you use a server for custom JumpStart installations, the JumpStart directory is a directory on the server that contains all the essential custom JumpStart files (for example, the `rules` file, `rules.ok` file, and profiles). The JumpStart directory should be owned by root and have permissions equal to 755.

▼ How to Create a JumpStart Directory on a Server

Overview – The procedure to create a JumpStart directory on a server involves:

- Creating a directory on the server
- Editing the `/etc/dfs/dfstab` file
- Copying sample custom JumpStart installation files into the directory on the server

Follow this procedure to create a JumpStart directory on a server.

1. **Log in as root on the server where you want the JumpStart directory to reside.**
2. **Create the JumpStart directory anywhere on the server:**

```
# mkdir jumpstart_dir_path
```

In this command,

`jumpstart_dir_path` Is the absolute path of the JumpStart directory.

For example, the following command would create the directory called `jumpstart` in the root file system:

```
mkdir /jumpstart
```

3. **Edit the `/etc/dfs/dfstab` file. Add the following entry:**

```
share -F nfs -o ro,anon=0 jumpstart_dir_path
```

For example, the following entry would be correct for the example shown in Step 2:

```
share -F nfs -o ro,anon=0 /jumpstart
```

4. Type `unshareall` and press Return.

5. Type `shareall` and press Return.

6. Determine your next step based on where the Solaris CD is located.

If You Want to Use The ...	Then ...
Solaris CD in the local CD-ROM drive	1) Insert the Solaris CD into the CD-ROM drive. 2) Go to Step 7.
Solaris CD image on the local disk	1) Change the directory to the Solaris image on the local disk. For example: <code>cd /export/install</code> 2) Go to Step 8.

7. Mount the Solaris CD (if needed) and change the directory to the mounted CD:

If the System Is ...	Then ...
Running Volume Management	<code>cd /cdrom/cdrom0/s0</code> or <code>cd /cdrom/cdrom0/s2</code>
Not running Volume Management	Note: Volume Management is running if the <code>/vol</code> directory on the system contains files. Systems running Solaris 2.0 or 2.1 do not have Volume Management. 1) <code>mount -F hsfs -o ro /dev/dsk/c0t6d0s0 /cdrom</code> or <code>mount -F hsfs -o ro /dev/dsk/c0t6d0p0 /cdrom</code> 2) <code>cd /cdrom</code>

8. Copy the contents of the `auto_install_sample` directory from the Solaris CD-ROM into the JumpStart directory:

```
# cp -r auto_install_sample/* jumpstart_dir_path
```

For example, the following command would copy the `auto_install_sample` directory into the JumpStart directory created in Step 2:

```
cp -r auto_install_sample/* /jumpstart
```

Task
Complete

You have completed creating a JumpStart directory on the server. To continue, see “How to Create a Profile” on page 62.

Enabling All Systems to Access the JumpStart Directory

When you create a JumpStart directory on a server, you must make sure systems can access it during a custom JumpStart installation. There are two ways to do this:

- Using the `-c` option of the `add_install_client` command every time you add a system for network installation.

or

- Enabling all systems to access the JumpStart directory.

To save you time when adding systems for network installations, use the following procedure to enable all systems to access the JumpStart directory from a server.

Note – The following procedure is not necessary if you are using a diskette for the JumpStart directory.

▼ How to Enable All Systems to Access the JumpStart Directory

Overview – The procedure to enable all systems to access the JumpStart directory from a server involves:

- Editing the `/etc/bootparams` file
- Updating the name service (NIS or NIS+) with the information you’ve added to the `/etc/bootparams` file

Follow this procedure to enable all systems to access the JumpStart directory from a server.

1. On the NIS or NIS+ master server, log in as root.
2. Edit the `/etc/bootparams` file or create the `/etc/bootparams` file if it does not exist. Add the following entry:

```
* install_config=server:jumpstart_dir_path
```

In this entry,

- `*` Is a wildcard character specifying all systems.
- `server` Is the host name of the server where the JumpStart directory is located.
- `jumpstart_dir_path` Is the absolute path of the JumpStart directory.

For example, the following entry would enable all systems to access the `/jumpstart` directory on the server named `sherlock`:

```
* install_config=sherlock:/jumpstart
```

Caution – Using this procedure may produce the following error message when booting an install client:
WARNING: getfile: RPC failed: error 5: (RPC Timed out).
See page 152 for more details on this error message.

3. Update the NIS or NIS+ tables (if necessary) with the information you added to the `/etc/bootparams` files.

If Your Site Uses ...	Then ...
NIS	<code>/var/yp/make</code>
NIS+	<code>/usr/lib/nis/nispopulate -F -p /etc bootparams</code>
No name service	Go to Task Complete on page 61.

Task
Complete

All systems can now access the JumpStart directory. You no longer need to use the `-c` option of the `add_install_client` command when adding systems for network installations.

Creating a Profile

What Is a Profile

A profile is a text file used as a template by the custom JumpStart installation software. It defines how to install the Solaris software on a system (for example, initial or upgrade installation option, system type, disk partitioning, software group), and it is named in the `rules` file.

A profile consists of one or more profile keywords and their values. Each profile keyword is a command that controls one aspect of how the Solaris installation program will install the Solaris software on a system. For example, the profile keyword and value

```
system_type server
```

tells the Solaris installation program to install the system as a server.

Note – If you created the JumpStart directory by using the procedures on page 51 or page 57, example profiles should already be in the JumpStart directory.

Requirements for Profiles

The following are requirements when creating a profile:

- The `install_type` profile keyword is required.
- Only one profile keyword can be on a line.

▼ How to Create a Profile

Overview – The procedure to create a profile involves:

- Editing a file
- Selecting profile keywords and profile values to define how to install the Solaris software on a system

Follow this procedure to create as many profiles as you need for your site.

1. Open a file (the profile) and give it a descriptive name.

You can create a new file or edit one of the sample profiles in the JumpStart directory you created.

The name of a profile should reflect how it will install the Solaris software on a system (for example, `basic_install`, `eng_profile`, or `user_profile`).

2. Add profile keywords and profile values to the profile.

Be aware of these things as you edit the profile:

- Profile Examples on page 63 provides some examples of profiles.
- Table 4-2 on page 66 provides the list of valid profile keywords and values.
- You can have as many lines in the profile as necessary to define how to install the Solaris software on a system.
- You can add a comment after the pound sign (#) anywhere on a line. If a line begins with a #, the entire line is a comment line. If a # is specified in the middle of a line, everything after the # is considered a comment. Blank lines are also allowed in a profile.
- The profile keywords and their values *are* case sensitive.
- Profiles should be owned by root and have permissions equal to 644.

Note – See “Using `pinstall` to Test Profiles” on page 98 for detailed information about testing profiles.

Task
Complete

This completes the procedure to create a profile. To continue setting up for a custom JumpStart installation, see How to Create the rules File on page 77.

Profile Examples

The following profile examples describe how you can use different profile keywords and profile values to control how the Solaris software is installed on a system. See Table 4-2 on page 66 for the list of profile keywords and profile values.

#	profile keywords	profile values
	# -----	-----
①	install_type	initial_install
②	system_type	standalone
③	partitioning	default
	fileSYS	any 60 swap # specify size of /swap
	fileSYS	s_ref:/usr/share/man - /usr/share/man ro
	fileSYS	s_ref:/usr/openwin/share/man - /usr/openwin/share/man ro,quota
④	cluster	SUNWCprog
⑤	package	SUNWman delete
	package	SUNWolman delete
	package	SUNWxwman delete
	package	SUNWoldem add
	package	SUNWxdem add
	package	SUNWoldim add
	package	SUNWxdim add

- ① This profile keyword is required in every profile.
- ② This profile keyword defines that the system will be installed as a standalone system.
- ③ The file system slices are determined by the software to be installed (default value); however, the size of swap is set to 60 Mbytes and it is installed on any disk (any value). The standard and OpenWindows man pages are mounted from the file server, s_ref, on the network.
- ④ The developer software group (SUNWCprog) is installed on the system.
- ⑤ Because the man pages are being mounted remotely, those packages are selected *not* to be installed on the system; however, the packages containing the OpenLook and X Windows demo programs and images are selected to be installed on the system.

```

# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone

❶ partitioning         default
filesys                c0t0d0s0 auto /
filesys                c0t3d0s1 32 swap
❷ cluster              SUNWCall

```

- ❶ The file system slices are determined by the software to be installed (default value). However, the size of root is based on the selected software (auto value) and it is installed on c0t0d0s0, and the size of swap is set to 32 Mbytes and it is installed on c0t3d0s1.
- ❷ The entire distribution software group (SUNWCall) is installed on the system.

```

# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone

❶ fdisk                c0t0d0 0x04 delete
❷ fdisk                c0t0d0 solaris maxfree
❸ cluster              SUNWCall
❹ cluster              SUNWCacc delete

```

- ❶ All fdisk partitions of type DOSOS16 (04 hexadecimal) are deleted from the c0t0d0 disk.
- ❷ A Solaris fdisk partition is created on the largest contiguous free space on the c0t0d0 disk.
- ❸ The entire distribution software group (SUNWCall) is installed on the system.
- ❹ The system accounting utilities (SUNWCacc) are selected *not* to be installed on the system.

	# profile keywords	profile values
	# -----	-----
❶	install_type	upgrade
❷	package	SUNWbcp delete
❸	package	SUNWolman add
	package	SUNWxwman add
	cluster	SUNWCumux add
❹	locale	de

- ❶ This profile upgrades a system (SPARC only).
- ❷ The binary compatibility package (SUNWbcp) is selected to be deleted from the system or prevented from being installed.
- ❸ This code ensures that the OpenLook and X Windows man pages and the universal multiplexor software are selected to be installed if they are not installed on the system. (All packages already on the system are automatically upgraded.)
- ❹ The German localization packages are selected to be installed on the system.

Profile Keyword and Profile Value Descriptions

Table 4-2 shows the profile keywords and profile values that you can use in a profile.

Table 4-2 Profile Keyword and Profile Value Descriptions (1 of 10)

Profile Keyword	Profile Values and Description
<code>client_arch</code>	<p><i>karch_value</i></p> <p><code>client_arch</code> defines that the server will support a different platform group than it uses. If you do not specify <code>client_arch</code>, any diskless client must have the same platform group as the server. You must specify <code>client_arch</code> once for each platform group.</p> <p>Valid values for <i>karch_value</i> are <code>sun4d</code>, <code>sun4c</code>, <code>sun4m</code>, <code>sun4u</code>, <code>i86pc</code>, or <code>prep</code>. (See Appendix C, “Platform Names and Groups” for a detailed list of the platform names of various systems.)</p> <p>Restriction: <code>client_arch</code> can be used only when <code>system_type</code> is specified as <code>server</code>.</p>
<code>client_root</code>	<p><i>root_size</i></p> <p><code>client_root</code> defines the amount of root space (<i>root_size</i> in Mbytes) to allocate for each client. If you do not specify <code>client_root</code> in a server’s profile, the installation software will automatically allocate 15 Mbytes of root space per client. The size of the client root area is used in combination with the <code>num_clients</code> keyword to determine how much space to reserve for the <code>/export/root</code> file system.</p> <p>Restriction: <code>client_root</code> can be used only when <code>system_type</code> is specified as <code>server</code>.</p>
<code>client_swap</code>	<p><i>swap_size</i></p> <p><code>client_swap</code> defines the amount of swap space (<i>swap_size</i> in Mbytes) to allocate for each diskless client. If you do not specify <code>client_swap</code>, 24 Mbytes of swap space is allocated.</p> <p>Example: <code>client_swap 32</code></p> <p>The example defines that each diskless client will have a swap space of 32 Mbytes.</p> <p>Restriction: <code>client_swap</code> can be used only when <code>system_type</code> is specified as <code>server</code>.</p>

† Profile keywords that can be used for upgrading

Table 4-2 Profile Keyword and Profile Value Descriptions (2 of 10)

Profile Keyword	Profile Values and Description												
<code>cluster</code> (use for software groups)	<p><code>group_name</code></p> <p><code>cluster</code> designates what software group to add to the system. The cluster names for the software groups are:</p> <table border="0"> <thead> <tr> <th>Software Group</th> <th><code>group_name</code></th> </tr> </thead> <tbody> <tr> <td>Core</td> <td>SUNWCreq</td> </tr> <tr> <td>End user system support</td> <td>SUNWCuser</td> </tr> <tr> <td>Developer system support</td> <td>SUNWCprog</td> </tr> <tr> <td>Entire distribution</td> <td>SUNWCall</td> </tr> <tr> <td>Entire distribution plus OEM support</td> <td>SUNWCXall</td> </tr> </tbody> </table> <p>You can specify only one software group in a profile, and it must be specified before other <code>cluster</code> and <code>package</code> entries. If you do not specify a software group with <code>cluster</code>, the end user software group (SUNWCuser) is installed on the system by default.</p>	Software Group	<code>group_name</code>	Core	SUNWCreq	End user system support	SUNWCuser	Developer system support	SUNWCprog	Entire distribution	SUNWCall	Entire distribution plus OEM support	SUNWCXall
Software Group	<code>group_name</code>												
Core	SUNWCreq												
End user system support	SUNWCuser												
Developer system support	SUNWCprog												
Entire distribution	SUNWCall												
Entire distribution plus OEM support	SUNWCXall												
<code>cluster</code> [†] (use for clusters)	<p><code>cluster_name</code> [add delete]</p> <p><code>cluster</code> designates whether a cluster should be added or deleted from the software group that will be installed on the system. <code>add</code> or <code>delete</code> indicates whether the cluster should be added or deleted. If you do not specify <code>add</code> or <code>delete</code>, <code>add</code> is set by default.</p> <p><code>cluster_name</code> must be in the form <code>SUNWCname</code>. To view detailed information about clusters and their names, start Admintool on an installed system and select Software from the Browse menu.</p> <p>For Upgrade:</p> <ul style="list-style-type: none"> • All clusters already on the system are automatically upgraded. • If you specify <code>cluster_name</code> <code>add</code>, and <code>cluster_name</code> is not installed on the system, the cluster is installed. • If you specify <code>cluster_name</code> <code>delete</code>, and <code>cluster_name</code> is installed on the system, the package is deleted <i>before</i> the upgrade begins. 												

† Profile keywords that can be used for upgrading

Table 4-2 Profile Keyword and Profile Value Descriptions (3 of 10)

Profile Keyword	Profile Values and Description
dontuse	<p><i>disk_name</i></p> <p>dontuse designates a disk that the Solaris installation program should <i>not</i> use when partitioning default is specified. You can specify dontuse once for each disk, and <i>disk_name</i> must be specified in the form <i>cxydz</i> or <i>cydz</i>, for example, <i>c0t0d0</i>.</p> <p>By default, the Solaris installation program uses all the operational disks on the system.</p> <p>Restriction: You cannot specify the dontuse keyword and the usedisk keyword in the same profile.</p>
fdisk	<p><i>disk_name type size</i></p> <p>fdisk defines how the fdisk partitions are set up on an x86 system (only required for x86 systems), and you can specify fdisk more than once. This is what happens by default with fdisk partitions on x86 systems:</p> <ul style="list-style-type: none"> • All fdisk partitions on the disk are preserved unless you specifically delete them with the fdisk keyword (if <i>size</i> is <i>delete</i> or 0). Also, all existing fdisk partitions are deleted when <i>size</i> is <i>all</i>. • A Solaris fdisk partition that contains a root file system is always designated as the active partition on the disk (an x86 system boots from the active partition by default). • If no fdisk keyword is specified in a profile, the following fdisk keyword is specified during the installation: <pre>fdisk all solaris maxfree</pre> • fdisk entries are processed in the order they appear in the profile. <p><i>disk_name</i> - Choose where the fdisk partition will be created or deleted:</p> <ul style="list-style-type: none"> • <i>cxydz</i> or <i>cydz</i> - A specific disk, for example, <i>c0t3d0</i>. • <i>rootdisk</i> - The disk where the root file system is placed during an installation, which is determined by the Solaris installation program (described on page 86). • <i>all</i> - All the selected disks. <p><i>type</i> - Choose what type of fdisk partition will be created or deleted on the specified disk:</p> <ul style="list-style-type: none"> • <i>solaris</i> - A Solaris fdisk partition (SUNIXOS fdisk type). • <i>dosprimary</i> - An alias for primary DOS fdisk partitions (not for extended or data DOS fdisk partitions). When deleting fdisk partitions (<i>size</i> is <i>delete</i>), <i>dosprimary</i> is an alias for the DOSHUGE, DOSOS12, and DOSOS16 fdisk types (they are all deleted). When creating an fdisk partition, <i>dosprimary</i> is an alias for the DOSHUGE fdisk partition (a DOSHUGE fdisk partition is created).

† Profile keywords that can be used for upgrading

Table 4-2 Profile Keyword and Profile Value Descriptions (4 of 10)

Profile Keyword	Profile Values and Description																											
fdisk continued	<p><i>disk_name type size</i></p> <p><i>type</i> - Choose what type of fdisk partition will be created or deleted on the specified disk:</p> <ul style="list-style-type: none"> • <i>DDD</i> - A decimal fdisk partition. <i>DDD</i> is a decimal number (valid values are 1 through 255). Restriction: This value can be specified only if <i>size</i> is <code>delete</code>. • <i>0xHH</i> - A hexadecimal fdisk partition. <i>HH</i> is a hexadecimal number (valid values are 01 through FF). Restriction: This value can be specified only if <i>size</i> is <code>delete</code>. <p>The following table shows the decimal and hexadecimal numbers for some of the fdisk types:</p> <table border="1"> <thead> <tr> <th><u>fdisk type</u></th> <th><u>DDD</u></th> <th><u>HH</u></th> </tr> </thead> <tbody> <tr> <td>DOSOS12</td> <td>1</td> <td>01</td> </tr> <tr> <td>PCIXOS</td> <td>2</td> <td>02</td> </tr> <tr> <td>DOSOS16</td> <td>4</td> <td>04</td> </tr> <tr> <td>EXTDOS</td> <td>5</td> <td>05</td> </tr> <tr> <td>DOSHUGE</td> <td>6</td> <td>06</td> </tr> <tr> <td>DOSDATA</td> <td>86</td> <td>56</td> </tr> <tr> <td>OTHEROS</td> <td>98</td> <td>62</td> </tr> <tr> <td>UNIXOS</td> <td>99</td> <td>63</td> </tr> </tbody> </table> <p><i>size</i> - Choose one of the following:</p> <ul style="list-style-type: none"> • <i>DDD</i> - An fdisk partition of size <i>DDD</i> (in Mbytes) is created on the specified disk. <i>DDD</i> must be a decimal number and the Solaris installation program automatically rounds the number up to the nearest cylinder boundary. If 0 is specified, it is the same as specifying <code>delete</code>. • <code>all</code> - An fdisk partition is created on the entire disk (all existing fdisk partitions will be deleted). Restriction: This value can be specified only if <i>type</i> is <code>solaris</code>. • <code>maxfree</code> - An fdisk partition is created in the largest contiguous free space on the specified disk. If an fdisk partition of the specified <i>type</i> already exists on the disk, the existing fdisk partition is used (a new fdisk partition is <i>not</i> created on the disk). Note: There must be at least one unused fdisk partition on the disk and the disk must have free space, or an error will occur. Restriction: This value can be specified only if <i>type</i> is <code>solaris</code> or <code>dosprimary</code>. • <code>delete</code> - All fdisk partitions of the specified <i>type</i> are deleted on the specified disk. 	<u>fdisk type</u>	<u>DDD</u>	<u>HH</u>	DOSOS12	1	01	PCIXOS	2	02	DOSOS16	4	04	EXTDOS	5	05	DOSHUGE	6	06	DOSDATA	86	56	OTHEROS	98	62	UNIXOS	99	63
<u>fdisk type</u>	<u>DDD</u>	<u>HH</u>																										
DOSOS12	1	01																										
PCIXOS	2	02																										
DOSOS16	4	04																										
EXTDOS	5	05																										
DOSHUGE	6	06																										
DOSDATA	86	56																										
OTHEROS	98	62																										
UNIXOS	99	63																										

† Profile keywords that can be used for upgrading

Table 4-2 Profile Keyword and Profile Value Descriptions (5 of 10)

Profile Keyword	Profile Values and Description
filesys (use for mounting remote file systems)	<pre>server:path server_address mount_pt_name [mount_options]</pre> <p>This instance of <code>filesys</code> sets up the installed system to automatically mount remote file systems when it boots. You can specify <code>filesys</code> more than once.</p> <p>For Dataless Clients: Profiles for dataless clients (when <code>system_type dataless</code> is specified) must include a remote <code>filesys</code> entry for both the <code>/usr</code> file system. The following line is an example of the <code>filesys</code> entries that must be used in a profile to install the Solaris software on a dataless client:</p> <pre>filesys sherlock:/export/exec/Solaris_2.4_sparc.all/usr - /usr</pre> <p><i>server</i> - The name of the server where the remote file system resides (followed by a colon).</p> <p><i>path</i> - The remote file system's mount point name, for example, <code>/usr</code> or <code>/export/home</code>.</p> <p><i>server_address</i> - The IP address of the server specified in <i>server:path</i>. If you don't have a name service running on the network, this value can be used to populate the <code>/etc/hosts</code> file with the server's host name and IP address. If you don't want to specify the server's IP address (if you have a name service running on the network), you must specify a minus sign (-).</p> <p><i>mount_pt_name</i> - The name of the mount point that the remote file system will be mounted on.</p> <p><i>mount_options</i> - One or more mount options (-o option of the command) that are added to the <code>/etc/vfstab</code> entry for the specified <i>mount_pt_name</i>.</p> <p>Note: If you need to specify more than one mount option, the mount options must be separated by commas and no spaces. For example: <code>ro,quota</code></p>

† Profile keywords that can be used for upgrading

Table 4-2 Profile Keyword and Profile Value Descriptions (6 of 10)

Profile Keyword	Profile Values and Description
<code>filesys</code> (use for creating local file systems)	<p><code>slice size [file_system] [optional_parameters]</code></p> <p>This instance of <code>filesys</code> creates local file systems during the installation. You can specify <code>filesys</code> more than once.</p> <p><code>slice</code> - Choose one of the following:</p> <ul style="list-style-type: none"> <code>any</code> - The Solaris installation program places the file system on any disk. <p>Restriction: <code>any</code> cannot be specified when <code>size</code> is <code>existing</code>, <code>all</code>, <code>free</code>, <code>start:size</code>, or <code>ignore</code>.</p> <ul style="list-style-type: none"> <code>cwtxdysz</code> or <code>cxdysz</code> - The disk slice where the Solaris installation program places the file system, for example, <code>c0t0d0s0</code>. <code>rootdisk.sn</code> - The logical name of the disk where the installation program places the root file system. The <code>sn</code> suffix indicates a specific slice on the disk. <p><code>size</code> - Choose one of the following:</p> <ul style="list-style-type: none"> <code>num</code> - The size of the file system is set to <code>num</code> (in Mbytes). <code>existing</code> - The current size of the existing file system is used. <p>Note: When using this value, you can change the name of an existing slice by specifying <code>file_system</code> as a different <code>mount_pt_name</code>.</p> <ul style="list-style-type: none"> <code>auto</code> - The size the file system is automatically determined depending on the selected software. <code>all</code> - The specified <code>slice</code> uses the entire disk for the file system. When you specify this value, no other file systems can reside on the specified disk. <code>free</code> - The remaining unused space on the disk is used for the file system. <p>Restriction: If <code>free</code> is used as the value to <code>filesys</code>, it must be the last <code>filesys</code> entry in a profile.</p> <ul style="list-style-type: none"> <code>start:size</code> - The file system is explicitly partitioned: <code>start</code> is the cylinder where the slice begins; <code>size</code> is the number of cylinders for the slice.

† Profile keywords that can be used for upgrading

Table 4-2 Profile Keyword and Profile Value Descriptions (7 of 10)

Profile Keyword	Profile Values and Description
filesys (use for creating local file systems) continued	<p><i>slice size [file_system] [optional_parameters]</i></p> <p><i>file_system</i> - You can use this optional value when <i>slice</i> is specified as <i>any</i> or <i>cwtxdysz</i>. If <i>file_system</i> is not specified, <i>unnamed</i> is set by default, but then you can't specify the <i>optional_parameters</i> value. Choose one of the following:</p> <ul style="list-style-type: none"> • <i>mount_pt_name</i> - The file system's mount point name, for example, <i>/var</i>. • <i>swap</i> - The specified <i>slice</i> is used as <i>swap</i>. • <i>overlap</i> - The specified <i>slice</i> is defined as a representation of a disk region (VTOC value is <i>V_BACKUP</i>). By default, <i>slice 2</i> is an overlap slice that is a representation of the whole disk. Restriction: <i>overlap</i> can be specified only when <i>size</i> is <i>existing</i>, <i>all</i>, or <i>start.size</i>. • <i>unnamed</i> - The specified <i>slice</i> is defined as a raw slice, so <i>slice</i> will not have a mount point name. If <i>file_system</i> is not specified, <i>unnamed</i> is set by default. • <i>ignore</i> - The specified <i>slice</i> is not used or recognized by the Solaris installation program. This could be used to ignore a file system on a disk during an installation, so the Solaris installation program can create a new file system on the same disk with the same name. <p><i>optional_parameters</i> - Choose one of the following:</p> <ul style="list-style-type: none"> • <i>preserve</i> - The file system on the specified <i>slice</i> is preserved. Restriction: <i>preserve</i> can be specified only when <i>size</i> is <i>existing</i> and <i>slice</i> is <i>cwtxdysz</i>. • <i>mount_options</i> - One or more mount options (<i>-o</i> option of the <i>mount(1M)</i> command) that are added to the <i>/etc/vfstab</i> entry for the specified <i>mount_pt_name</i>. <p>Note: If you need to specify more than one mount option, the mount options must be separated by commas and no spaces. For example: <i>ro,quota</i></p>
install_type [†]	<p><i>initial_install upgrade</i></p> <p><i>install_type</i> defines whether to perform the initial installation option or upgrade option on the system.</p> <p>Restriction: <i>install_type</i> must be the first profile keyword in every profile.</p>

[†] Profile keywords that can be used for upgrading

Table 4-2 Profile Keyword and Profile Value Descriptions (8 of 10)

Profile Keyword	Profile Values and Description																				
locale [†]	<p><i>locale_name</i></p> <p>locale designates that the localization packages associated with the selected software should be installed (or added for upgrade) for the specified <i>locale_name</i>. The <i>locale_name</i> values are the same as the values used for the \$LANG environment variable. Solaris 2.5 supports the following localizations:</p> <table border="1"> <thead> <tr> <th>Language</th> <th><i>locale_name</i></th> </tr> </thead> <tbody> <tr> <td>Chinese</td> <td>zh</td> </tr> <tr> <td>French</td> <td>fr</td> </tr> <tr> <td>German</td> <td>de</td> </tr> <tr> <td>Italian</td> <td>it</td> </tr> <tr> <td>Japanese</td> <td>ja</td> </tr> <tr> <td>Korean</td> <td>ko</td> </tr> <tr> <td>Latin American</td> <td>es</td> </tr> <tr> <td>Swedish</td> <td>sv</td> </tr> <tr> <td>Taiwanese</td> <td>zh_TW</td> </tr> </tbody> </table> <p>The English localization packages are installed by default. You can specify locale once for each localization you need to support.</p>	Language	<i>locale_name</i>	Chinese	zh	French	fr	German	de	Italian	it	Japanese	ja	Korean	ko	Latin American	es	Swedish	sv	Taiwanese	zh_TW
Language	<i>locale_name</i>																				
Chinese	zh																				
French	fr																				
German	de																				
Italian	it																				
Japanese	ja																				
Korean	ko																				
Latin American	es																				
Swedish	sv																				
Taiwanese	zh_TW																				
num_clients	<p><i>client_num</i></p> <p>When a server is installed, space is allocated for each diskless client's root (/) and swap file systems. num_clients defines the number of diskless clients (<i>client_num</i>) that a server will support. If you do not specify num_clients, five diskless clients are allocated.</p> <p>Restriction: num_clients can be used only when system_type is specified as server.</p>																				

[†] Profile keywords that can be used for upgrading

Table 4-2 Profile Keyword and Profile Value Descriptions (9 of 10)

Profile Keyword	Profile Values and Description
package [†]	<p><i>package_name</i> [add delete]</p> <p>package designates whether a package should be added to or deleted from the software group that will be installed on the system. add or delete indicates whether the package should be added or deleted. If you do not specify add delete, add is set by default.</p> <p><i>package_name</i> must be in the form <i>SUNWname</i>. Use the <code>pkginfo -l</code> command or Admintool (select Software from the Browse menu) on an installed system to view detailed information about packages and their names.</p> <p>For Upgrade:</p> <ul style="list-style-type: none"> • All packages already on the system are automatically upgraded. • If you specify <i>package_name</i> add, and <i>package_name</i> is not installed on the system, the package is installed. • If you specify <i>package_name</i> delete, and <i>package_name</i> is installed on the system, the package is deleted <i>before</i> the upgrade begins. • If you specify <i>package_name</i> delete, and <i>package_name</i> is not installed on the system, the package is prevented from being installed if it is part of a cluster that is designated to be installed.

[†] Profile keywords that can be used for upgrading

Table 4-2 Profile Keyword and Profile Value Descriptions (10 of 10)

Profile Keyword	Profile Values and Description
partitioning	<p>default existing explicit</p> <p>partitioning defines how the disks are divided into slices for file systems during the installation. If you do not specify partitioning, default is set.</p> <p>default - The Solaris installation program selects the disks and creates the file systems on which to install the specified software, except for any file systems specified by the <code>filesys</code> keyword. <code>rootdisk</code> is selected first; additional disks are used if the specified software does not fit on <code>rootdisk</code>.</p> <p>existing - The Solaris installation program uses the existing file systems on the system's disks. All file systems except <code>/</code>, <code>/usr</code>, <code>/usr/openwin</code>, <code>/opt</code>, and <code>/var</code> are preserved. The installation program uses the last mount point field from the file system superblock to determine which file system mount point the slice represents.</p> <p>Restriction: When specifying the <code>filesys</code> profile keyword with <code>partitioning existing</code>, <code>size</code> must be <code>existing</code>.</p> <p>explicit - The Solaris installation program uses the disks and creates the file systems specified by the <code>filesys</code> keywords. If you specify only the root (<code>/</code>) file system with the <code>filesys</code> keyword, all the Solaris software will be installed in the root file system.</p> <p>Restriction: When you use the <code>explicit</code> profile value, you must use the <code>filesys</code> profile keyword to specify which disks to use and what file systems to create.</p>
system_type	<p>standalone dataless server</p> <p><code>system_type</code> defines the type of system being installed. If you do not specify <code>system_type</code> in a profile, <code>standalone</code> is set by default.</p>
usedisk	<p><code>disk_name</code></p> <p><code>usedisk</code> designates a disk that the Solaris installation program will use when <code>partitioning default</code> is specified. You can specify <code>usedisk</code> once for each disk, and <code>disk_name</code> must be specified in the form <code>cxydz</code> or <code>cydz</code>, for example, <code>c0t0d0</code>.</p> <p>If you specify the <code>usedisk</code> profile keyword in a profile, the Solaris installation program will only use the disks that you specify with the <code>usedisk</code> profile keyword.</p> <p>Restriction: You cannot specify the <code>usedisk</code> keyword and the <code>dontuse</code> keyword in the same profile.</p>

† Profile keywords that can be used for upgrading

How the Size of Swap Is Determined

If a profile does not explicitly specify the size of swap, the Solaris installation program determines the maximum size that swap can be, based on the system's physical memory. Table 4-3 shows how the maximum size of swap is determined during a custom JumpStart installation.

Table 4-3 How the Maximum Size of Swap Is Determined

Physical Memory (in Mbytes)	Maximum Size of Swap (in Mbytes)
16 - 64	32
64 - 128	64
128 - 512	128
512 >	256

The Solaris installation program will make the size of swap no more than 20% of the disk where it resides, unless there is free space left on the disk after laying out the other file systems. If free space exists, the Solaris installation program will allocate the free space to swap up to the maximum size shown in Table 4-3.

Note – Physical memory plus swap space must be a minimum of 32 Mbytes.

Creating the rules File

What Is the rules File

The `rules` file is a text file used to create the `rules.ok` file. The `rules` file is a look-up table consisting of one or more rules that define matches between system attributes and profiles. For example, the rule

```
karch sun4c - basic_prof -
```

matches a system with a `sun4c` platform name to the `basic_prof` profile, which the Solaris installation program would use to install the system.

Note – If you set up the JumpStart directory by using the procedures on page 51 or page 57, an example `rules` file should already be in the JumpStart directory; the example `rules` file contains documentation and some example rules. If you use the example `rules` file, make sure you comment out the example rules that you will not use.

When Does a System Match a Rule

During a custom JumpStart installation, the Solaris installation program attempts to match the rules in the `rules.ok` file in order: first rule through the last rule. A rule match occurs when the system being installed matches any of the rule values in the rule (as defined in Table 4-5 on page 82). As soon as a system matches a rule, the Solaris installation program stops reading the `rules.ok` file and begins to install the system as defined by the matched rule's profile.

▼ How to Create the `rules` File

Overview – The procedure to create a `rules` file involves:

- Editing a file
- Selecting rule keywords and rule values for each group of systems you want to install using custom JumpStart. Any systems that match the rule keyword and rule value will be installed as specified by the corresponding profile.

Follow this procedure to create a `rules` file.

1. Open a file (the `rules` file) and name it `rules`.

You can create a new file or edit the sample `rules` file provided in the JumpStart directory you created.

2. Add a rule in the `rules` file for each group of systems you want to install using custom JumpStart.

Be aware of these things as you add rules to the `rules` file:

- Rule Examples on page 80 provides some examples of rules.
- Table 4-5 on page 82 provides the list of valid rule keywords and values.
- The `rules` file must have at least one rule

- A rule must have at least a rule keyword, a rule value, and a corresponding profile.

A rule within the `rules` file must have the following syntax:

```
[!]rule_keyword rule_value [&& [!]rule_keyword rule_value]... begin profile finish
```

Table 4-4 describes the fields of a rule.

Table 4-4 Field Descriptions of a Rule

Field	Description
!	A symbol used before a rule keyword to indicate negation.
[]	A symbol used to indicate an optional expression or field.
...	A symbol used to indicate the preceding expression may be repeated.
<i>rule_keyword</i>	A predefined keyword that describes a general system attribute, such as host name (<code>hostname</code>) or memory size (<code>memsize</code>). It is used with the <code>rule</code> value to match a system with the same attribute to a profile. See Table 4-5 on page 82 for the list of <code>rule</code> keywords.
<i>rule_value</i>	A value that provides the specific system attribute for the corresponding <code>rule</code> keyword. See Table 4-5 on page 82 for the list of <code>rule</code> values.
&&	A symbol that must be used to join (logically AND) rule keyword and rule value pairs together in the same rule. During a custom JumpStart installation, a system must match every pair in the rule before the rule matches.

Table 4-4 Field Descriptions of a Rule (Continued)

Field	Description
<i>begin</i>	<p>A name of an optional Bourne shell script that can be executed before the installation begins. If no begin script exists, you <i>must</i> enter a minus sign (-) in this field. All begin scripts must reside in the JumpStart directory.</p> <p>See “Creating Begin Scripts” on page 92 for detailed information on how to create begin scripts.</p>
<i>profile</i>	<p>A name of a text file used as a template that defines how to install Solaris on a system. The information in a profile consists of profile keywords and their corresponding profile values. All profiles must reside in the JumpStart directory.</p> <p>Note - There are optional ways to use the profile field, which are described in “Using a Site-Specific Installation Program” on page 108 and “Creating Derived Profiles With Begin Scripts” on page 92.</p>
<i>finish</i>	<p>A name of an optional Bourne shell script that can be executed after the installation completes. If no finish script exists, you must enter a minus sign (-) in this field. All finish scripts must reside in the JumpStart directory.</p> <p>See “Creating Finish Scripts” on page 94 for detailed information on how to create finish scripts.</p>

Task
Complete

This completes the procedure to create a rules file. To validate the rules file, see How to Use check to Validate the rules File on page 88.

Rule Examples

The following illustration shows several example rules in a `rules` file. Each line has a rule keyword and a valid value for that keyword. The Solaris installation program scans the `rules` file from top to bottom. When the Solaris installation program matches a rule keyword and value with a known system, it installs the Solaris software specified by the profile listed in the profile field.

	# rule keywords and rule values	begin script	profile	finish script
	# -----	-----	-----	-----
❶	hostname eng-1	-	basic_prof	-
❷	network 192.43.34.0 && !model \ 'SUNW,Sun 4_50'	-	net_prof	-
❸	model SUNW,SPARCstation-LX	-	lx_prof	complete
❹	network 193.144.2.0 && karch i86pcsetup	-	x86_prof	done
❺	memsize 16-32 && arch ppc	-	prog_prof	-
❻	any -	-	generic_prof	-

- ❶ This rule matches if the system's host name is `eng-1`. The `basic_prof` profile is used to install the Solaris software on the system that matches this rule.
- ❷ The rule matches if the system is on subnet `192.43.34.0` and it is *not* a SPARCstation IPX™ (`SUNW,Sun 4_50`). The `net_prof` profile is used to install the Solaris software on systems that match this rule.
- ❸ The rule matches if the system is a SPARCstation LX. The `lx_prof` profile and the `complete` finish script are used to install the Solaris software on systems that match this rule. This rule also provides an example of rule wrap, which is defined on page 81.
- ❹ This rule matches if the system is on subnet `193.144.2.0` and the system is an x86. The `setup` begin script, the `x86_prof` profile, and the `done` finish script are used to install the Solaris software on systems that match this rule.
- ❺ This rule matches if the system has 16-32 Mbytes of memory and its processor type is `ppc` (PowerPC). The `prog_prof` profile is used to install the Solaris software on systems that match this rule.
- ❻ This rule matches any system that did not match the previous rules. The `generic_prof` profile is used to install the Solaris software on systems that match this rule. If used, `any` should always be in the last rule.

Important Information About the rules File

The following information is important to know about the `rules` file:

- **Name** - The `rules` file *must* have the file name, `rules`.
- **rules.ok file** - The `rules.ok` file is a generated version of the `rules` file, and it is required by the custom JumpStart installation software to match a system to a profile. You must run the `check` script to create the `rules.ok` file, and the `rules.ok` file should be owned by root and have permissions equal to 644.
- **Comments** - You can add a comment after the pound sign (#) anywhere on a line. If a line begins with a #, the entire line is a comment line. If a # is specified in the middle of a line, everything after the # is considered a comment. Blank lines are also allowed in the `rules` file.

Note - When creating the `rules.ok` file, the `check` script removes all the comment lines, comments at the end of a rule, and blank lines.

- **Rule wrap** - When a rule spans multiple lines, you can let a rule to wrap to a new line, or you can continue a rule on a new line by using a backslash (\) before the carriage return.
- **Rule fields** - The `rule_value`, `begin`, and `finish` fields must have a valid entry or a minus sign (-) to specify that there is no entry.

Rule Keyword and Rule Value Descriptions

Table 4-5 describes the rule keywords and rule values that you can use in the `rules` file.

Table 4-5 Rule Keyword and Rule Value Descriptions (1 of 4)

Rule Keyword	Rule Values	Description	
any	minus sign (-)	Match always succeeds.	
arch	<i>processor_type</i> <u>platform</u> SPARC x86 PowerPC	<i>processor_type</i> <u>processor_type</u> sparc i386 ppc	Matches a system's processor type. The <code>uname -p</code> command reports the system's processor type.
domainname	<i>domain_name</i>	Matches a system's domain name, which controls how a name service determines information. If you have a system already installed, the <code>domainname</code> command reports the system's domain name.	
disksize	<i>disk_name</i> <i>size_range</i> <i>disk_name</i> - A disk name in the form <code>c0t3d0</code> , such as <code>c0t3d0</code> , or the special word <code>rootdisk</code> . <code>rootdisk</code> should be used only when trying to match systems that contain the factory-installed JumpStart software. <code>rootdisk</code> is described on page 86. <i>size_range</i> - The size of the disk, which must be specified as a range of Mbytes (<code>xx-xx</code>).	Matches a system's disk (in Mbytes). Example: <code>disksize c0t3d0 250-300</code> The example tries to match a system with a <code>c0t3d0</code> disk that is between 250 and 300 Mbytes. Note: When calculating <i>size_range</i> , remember that a Mbyte equals 1,048,576 bytes. A disk may be advertised as a "207 Mbyte" disk, but it may have only 207 million bytes of disk space. The Solaris installation program will actually view the "207 Mbyte" disk as a 197 Mbyte disk because $207,000,000 / 1,048,576 = 197$. So, a "207 Mbyte" disk would not match a <i>size_range</i> equal to 200-210.	
hostaddress	<i>IP_address</i>	Matches a system's IP address.	

Table 4-5 Rule Keyword and Rule Value Descriptions (2 of 4)

Rule Keyword	Rule Values	Description
hostname	<i>host_name</i>	Matches a system's host name. If you have a system already installed, the <code>uname -n</code> command reports the system's host name.
installed	<i>slice version</i> <i>slice</i> - A disk slice name in the form <code>cwtxdysz</code> , such as <code>c0t3d0s5</code> , or the special words <code>any</code> or <code>rootdisk</code> . If <code>any</code> is used, any disk attached to the system attempts to match. <code>rootdisk</code> should be used only when trying to match systems that contain the factory-installed JumpStart software. <code>rootdisk</code> is described on page 86. <i>version</i> - A version name, such as <code>Solaris_2.5</code> , or the special words <code>any</code> or <code>upgrade</code> . If <code>any</code> is used, any Solaris or SunOS release is matched. If <code>upgrade</code> is used, any upgradable Solaris 2.1 or greater release is matched.	Matches a disk that has a root file system corresponding to a particular version of Solaris software.
karch	<i>platform_group</i> Valid values are <code>sun4d</code> , <code>sun4c</code> , <code>sun4m</code> , <code>sun4u</code> , <code>i86pc</code> , or <code>prep</code> . (See Appendix C, "Platform Names and Groups" for a detailed list of platform groups and names.)	Matches a system's platform name. If you have a system already installed, the <code>arch -k</code> command or the <code>uname -m</code> command reports the system's platform group.
memsize	<i>physical_mem</i> The value must be a range of Mbytes (<code>xx-xx</code>) or a single Mbyte value.	Matches a system's physical memory size (in Mbytes). Example: <code>memsize 16-32</code> The example tries to match a system with a physical memory size between 16 and 32 Mbytes. If you have a system already installed, the <code>prtconf</code> command (line 2) reports the system's physical memory size.

Table 4-5 Rule Keyword and Rule Value Descriptions (3 of 4)

Rule Keyword	Rule Values	Description																																																
model	<i>model_name</i>	Matches a system's model number, which is system-dependent and varies by the manufacturer. The list shown may not be complete.																																																
	<table border="0"> <tr> <td><u>System</u></td> <td><u>model_name</u></td> </tr> <tr> <td>Sun-4/110</td> <td>Sun 4_100 Series</td> </tr> <tr> <td>Sun-4/2xx</td> <td>Sun 4_200 Series</td> </tr> <tr> <td>SPARCstation 1 (4/60)</td> <td>Sun 4_60</td> </tr> <tr> <td>SPARCstation 1+ (4/65)</td> <td>Sun 4_65</td> </tr> <tr> <td>SPARCstation SLC™ (4/20)</td> <td>Sun 4_20</td> </tr> <tr> <td>SPARCstation IPC (4/40)</td> <td>SUNW,Sun 4_40</td> </tr> <tr> <td>SPARCstation ELC™ (4/25)</td> <td>SUNW,SUN 4_25</td> </tr> <tr> <td>SPARCstation IPX (4/50)</td> <td>SUNW,Sun 4_50</td> </tr> <tr> <td>SPARCstation 2 (4/75)</td> <td>SUNW,SUN 4_75</td> </tr> <tr> <td>Sun-4/3xx</td> <td>Sun SPARCsystem 300</td> </tr> <tr> <td>Sun-4/4xx</td> <td>Sun SPARCsystem 400</td> </tr> <tr> <td>SPARCserver™ 6xx</td> <td>SUNW,SPARCsystem-600</td> </tr> <tr> <td>SPARCstation 10</td> <td>SUNW,SPARCstation-10</td> </tr> <tr> <td>SPARCclassic™ (4/15)</td> <td>SUNW,SPARCclassic</td> </tr> <tr> <td>SPARCstation LX (4/30)</td> <td>SUNW,SPARCstation-LX</td> </tr> <tr> <td>SPARCcenter™ 1000</td> <td>SUNW,SPARCserver-1000</td> </tr> <tr> <td>SPARCcenter 2000</td> <td>SUNW,SPARCcenter-2000</td> </tr> <tr> <td>SPARCstation 10 SX</td> <td>SUNW,SPARCstation-10,SX</td> </tr> <tr> <td>SPARCstation 20</td> <td>SUNW,SPARCstation-20</td> </tr> <tr> <td>SPARCstation 5</td> <td>SUNW,SPARCstation-5</td> </tr> <tr> <td>SPARCstation Voyager</td> <td>SUNW,S240</td> </tr> <tr> <td>Sun Ultra™ 1</td> <td>SUNW,Ultra-1</td> </tr> <tr> <td>Sun UltraServer 1</td> <td>SUNW,Ultra-1</td> </tr> </table>	<u>System</u>	<u>model_name</u>	Sun-4/110	Sun 4_100 Series	Sun-4/2xx	Sun 4_200 Series	SPARCstation 1 (4/60)	Sun 4_60	SPARCstation 1+ (4/65)	Sun 4_65	SPARCstation SLC™ (4/20)	Sun 4_20	SPARCstation IPC (4/40)	SUNW,Sun 4_40	SPARCstation ELC™ (4/25)	SUNW,SUN 4_25	SPARCstation IPX (4/50)	SUNW,Sun 4_50	SPARCstation 2 (4/75)	SUNW,SUN 4_75	Sun-4/3xx	Sun SPARCsystem 300	Sun-4/4xx	Sun SPARCsystem 400	SPARCserver™ 6xx	SUNW,SPARCsystem-600	SPARCstation 10	SUNW,SPARCstation-10	SPARCclassic™ (4/15)	SUNW,SPARCclassic	SPARCstation LX (4/30)	SUNW,SPARCstation-LX	SPARCcenter™ 1000	SUNW,SPARCserver-1000	SPARCcenter 2000	SUNW,SPARCcenter-2000	SPARCstation 10 SX	SUNW,SPARCstation-10,SX	SPARCstation 20	SUNW,SPARCstation-20	SPARCstation 5	SUNW,SPARCstation-5	SPARCstation Voyager	SUNW,S240	Sun Ultra™ 1	SUNW,Ultra-1	Sun UltraServer 1	SUNW,Ultra-1	<p>If you have a system already installed, the <code>prtconf</code> command (line 5) reports the system's model number.</p> <p>Note: If the <i>model_name</i> contains spaces, the <i>model_name</i> must be inside a pair of single quotes ('). For example: 'SUNW,Sun 4_50'</p>
<u>System</u>	<u>model_name</u>																																																	
Sun-4/110	Sun 4_100 Series																																																	
Sun-4/2xx	Sun 4_200 Series																																																	
SPARCstation 1 (4/60)	Sun 4_60																																																	
SPARCstation 1+ (4/65)	Sun 4_65																																																	
SPARCstation SLC™ (4/20)	Sun 4_20																																																	
SPARCstation IPC (4/40)	SUNW,Sun 4_40																																																	
SPARCstation ELC™ (4/25)	SUNW,SUN 4_25																																																	
SPARCstation IPX (4/50)	SUNW,Sun 4_50																																																	
SPARCstation 2 (4/75)	SUNW,SUN 4_75																																																	
Sun-4/3xx	Sun SPARCsystem 300																																																	
Sun-4/4xx	Sun SPARCsystem 400																																																	
SPARCserver™ 6xx	SUNW,SPARCsystem-600																																																	
SPARCstation 10	SUNW,SPARCstation-10																																																	
SPARCclassic™ (4/15)	SUNW,SPARCclassic																																																	
SPARCstation LX (4/30)	SUNW,SPARCstation-LX																																																	
SPARCcenter™ 1000	SUNW,SPARCserver-1000																																																	
SPARCcenter 2000	SUNW,SPARCcenter-2000																																																	
SPARCstation 10 SX	SUNW,SPARCstation-10,SX																																																	
SPARCstation 20	SUNW,SPARCstation-20																																																	
SPARCstation 5	SUNW,SPARCstation-5																																																	
SPARCstation Voyager	SUNW,S240																																																	
Sun Ultra™ 1	SUNW,Ultra-1																																																	
Sun UltraServer 1	SUNW,Ultra-1																																																	
	x86	i86pc																																																
	ThinkPad Power Series 820	IBM,PPS_Model_6040 IBM,PPS_Model_6040_E																																																
	ThinkPad Power Series 850	IBM,PPS_Model_6042 IBM,PPS_Model_6042_E																																																
	Power Series 830 or 850	IBM,PPS_Model_6050_6070 IBM,PPS_Model_6050_6070_E																																																
	PowerStack Series E	Motorola,PowerStack_Series_E																																																
	PowerStack RISC PC DT	Motorola,PowerStack_Series_MT																																																
	PowerStack RISC PC MT	Motorola,PowerStack_Series_MT																																																

Table 4-5 Rule Keyword and Rule Value Descriptions (4 of 4)

Rule Keyword	Rule Values	Description
network	<i>network_num</i>	<p>Matches a system's network number, which the Solaris installation program determines by performing a logical AND between the system's IP address and the subnet mask.</p> <p>Example: <code>network 193.144.2.0</code></p> <p>The example would match a system with a 193.144.2.8 IP address (if the subnet mask were 255.255.255.0).</p>
osname	<i>Solaris_version</i>	<p>Matches a version of Solaris already installed on a system. <i>Solaris_version</i> is the version of Solaris environment installed on the system: for example, <i>Solaris_2.5</i>.</p>
totaldisk	<i>size_range</i> The value must be specified as a range of Mbytes (<i>xx-xx</i>).	<p>Matches the total disk space on a system (in Mbytes). The total disk space includes all the operational disks attached to a system.</p> <p>Example: <code>totaldisk 300-500</code></p> <p>The example tries to match a system with a total disk space between 300 and 500 Mbytes.</p> <p>Note: When calculating <i>size_range</i>, remember that a Mbyte equals 1048576 bytes. A disk may be advertised as a "207 Mbyte" disk, but it may have only 207 million bytes of disk space. The Solaris installation program will actually view the "207 Mbyte" disk as a 197 Mbyte disk because $207000000 / 1048576 = 197$. So, a "207 Mbyte" disk would not match a <i>size_range</i> equal to 200-210.</p>

How the Installation Program Sets the Value of `rootdisk`

`rootdisk` is the logical name of the disk where the root file system is placed during an installation. During a custom JumpStart installation, the Solaris installation program sets the value of `rootdisk` (that is, the actual disk it represents) depending on various situations; this is described in Table 4-6.

Table 4-6 How the Solaris Installation Program Sets the Value of `rootdisk`

Situation	What Happens
A system contains the factory-installed JumpStart software. (This applies to some SPARC systems only).	<code>rootdisk</code> is set to the disk that contains the factory-installed JumpStart software before the system tries to match any rules.
<code>rootdisk</code> has <i>not</i> been set and a system tries to match the following rule: <code>disksize rootdisk size_range</code> or <code>installed rootdisk version</code>	<code>rootdisk</code> is set to <code>c0t3d0</code> or the first available disk attached to the system. After <code>rootdisk</code> is set, the system tries to match the rule.
If <code>rootdisk</code> has been set and the system tries to match the following rule: <code>disksize rootdisk size_range</code> or <code>installed rootdisk version</code>	The system tries to match the rule.
A system tries to match the following rule: <code>installed disk version</code>	If <code>disk</code> is found on the system with a root file system that matches the specified <code>version</code> , the rule matches and <code>rootdisk</code> is set to <code>disk</code> .
A system tries to match the following rule: <code>installed any version</code>	If any disk is found on the system with a root file system that matches the specified <code>version</code> , the rule matches and <code>rootdisk</code> is set to the found disk. (If there is more than one disk on the system that can match, the system will match the first disk that is found.)

Table 4-6 How the Solaris Installation Program Sets the Value of `rootdisk` (Continued)

Situation	What Happens
<code>rootdisk</code> has not been set after a system matches a rule and the system is going to be upgraded (which is defined in the profile).	<code>rootdisk</code> is set to the first disk found with a root file system that matches an upgradable version of Solaris software. If no disk is found, the system proceeds with an interactive installation.
<code>rootdisk</code> has not been set after a system matches a rule.	<code>rootdisk</code> is set to <code>c0t3d0</code> or the first available disk attached to the system.

For the Solaris installation program to use the value of `rootdisk`, the following conditions must be true in the profile specified for the system:

- Default partitioning is used.
- No slice has been explicitly set for the root file system.

Using *check to Validate the rules File*

Before the `rules` file and profiles can be used, you must run the `check` script to validate that these files are set up correctly. The following table shows what the `check` script does.

Stage	Description
1	The <code>rules</code> file is checked for syntax. <code>check</code> makes sure that the rule keywords are legitimate, and the <i>begin</i> , <i>class</i> , and <i>finish</i> fields are specified for each rule (the <i>begin</i> and <i>finish</i> fields may be a minus sign [-] instead of a file name).
2	If no errors are found in the <code>rules</code> file, each profile specified in the <code>rules</code> is checked for syntax.
3	If no errors are found, <code>check</code> creates the <code>rules.ok</code> file from the <code>rules</code> file, removing all comments and blank lines, retaining all the rules, and adding the following comment line to the end: # version=2 checksum=num

▼ **How to Use `check` to Validate the `rules` File**

Overview – The procedure to use `check` to validate the `rules` file involves:

- Making sure the check script resides in the JumpStart directory
- Running the check script

Follow this procedure to use `check` to validate the `rules` file.

1. Make sure that the `check` script resides in the JumpStart directory.

Note – The `check` script is provided in the `auto_install_sample` directory on the Solaris CD.

2. Change the directory to the JumpStart directory:

```
$ cd jumpstart_dir_path
```

3. Run the `check` script to validate the `rules` file:

```
$ ./check [-p path] [-r file_name]
```

In this command,

- | | |
|---------------------------|---|
| <code>-p path</code> | Is the path to the Solaris 2.5 CD. You can use a Solaris CD image on a local disk or a mounted Solaris CD. This option ensures that you are using the most recent version of the <code>check</code> script. You should use this option if you are using <code>check</code> on a system that is running a previous version of Solaris. |
| <code>-r file_name</code> | Specifies a rules file other than the one named <code>rules</code> . Using this option, you can test the validity of a rule before integrating it into the <code>rules</code> file. |

As the check script runs, it reports that it is checking the validity of the rules file and the validity of each profile. If no errors are encountered, it reports: `The auto-install configuration is ok.`

Task
Complete

The rules files is now validated. To read about the optional features available for custom JumpStart installations, see Chapter 5, “Using Optional Custom JumpStart Features.” To perform a custom JumpStart installation on a system, see Chapter 7, “Booting and Installing Solaris: Custom JumpStart.”

Using Optional Custom JumpStart Features



<i>How to Use pinstall to Test a Profile</i>	<i>page 99</i>
<i>How to Create a Disk Configuration File for a PowerPC System</i>	<i>page 101</i>
<i>How to Create a Multiple Disk Configuration File for a PowerPC System</i>	<i>page 104</i>

Overview

This chapter describes the optional features available for custom JumpStart installations, and it is a supplement to Chapter 4, “Preparing Custom JumpStart Installations.” You can use the following optional features to enhance and test custom JumpStart installations:

- Begin scripts
- Finish scripts
- pinstall
- Site-specific installation program

Note – This chapter is valid for either an x86, PowerPC, or SPARC server that is being used for custom JumpStart installations. An x86 server can provide custom JumpStart files for SPARC and PowerPC systems, a SPARC system can provide custom JumpStart files for x86 and PowerPC systems, and a PowerPC server can provide custom JumpStart files for x86 and SPARC systems.

Creating Begin Scripts

What Is a Begin Script

A *begin script* is a user-defined Bourne shell script, specified within the `rules` file, that performs tasks before the Solaris software is installed on the system. Begin scripts can be used only with custom JumpStart installations.

Important Information About Begin Scripts

The following information is important to know about begin scripts:

- Be careful that you do not specify something in the script that would prevent the mounting of file systems onto `/a` during an initial or upgrade installation. If the Solaris installation program cannot mount the file systems onto `/a`, an error will occur and the installation will fail.
- Output from the begin script goes to `/var/sadm/begin.log`.
- Begin scripts should be owned by root and have permissions equal to 644.

Ideas for Begin Scripts

You could set up begin scripts to perform the following tasks:

- Creating derived profiles
- Backing up files before upgrade

Creating Derived Profiles With Begin Scripts

A *derived profile* is a profile that is dynamically created by a begin script during a custom JumpStart installation. Derived profiles are needed when you cannot set up the `rules` file to match specific systems to a profile (when you need more flexibility than the `rules` file can provide). For example, you may need to use derived profiles for identical system models that have different hardware components (for example, systems that have different frame buffers).

To set up a rule to use a derived profile, you must:

- Set the profile field to an equal sign (=) instead of a profile.

- Set the begin field to a begin script that will create a derived profile depending on which system is being installed.

When a system matches a rule with the profile field equal to an equal sign (=), the begin script creates the derived profile that is used to install the Solaris software on the system.

An example of a begin script that creates the same derived profile every time is shown below; however, you could add code to this example that would create a different derived profile depending on certain command's output.

```
#!/bin/sh
echo "install_type      initial_install"    > ${SI_PROFILE}
echo "system_type      standalone"        >> ${SI_PROFILE}
echo "partitioning     default"           >> ${SI_PROFILE}
echo "cluster          SUNWCprog"         >> ${SI_PROFILE}
echo "package          SUNWman    delete"  >> ${SI_PROFILE}
echo "package          SUNWolman  delete"  >> ${SI_PROFILE}
echo "package          SUNWxwman  delete"  >> ${SI_PROFILE}
```

As shown above, the begin script must use the `SI_PROFILE` environment variable for the name of the derived profile, which is set to `/tmp/install.input` by default.

Note – If a begin script is used to create a derived profile, make sure there are no errors in it. A derived profile is not verified by the check script, because it is not created until the execution of the begin script.

Creating Finish Scripts

What Is a Finish Script

A *finish script* is a user-defined Bourne shell script, specified within the `rules` file, that performs tasks after the Solaris software is installed on the system, but before the system reboots. Finish scripts can be used only with custom JumpStart installations.

Important Information About Finish Scripts

The following information is important to know about finish scripts:

- The Solaris installation program mounts the system's file systems onto `/a`. The file systems remain mounted on `/a` until the system reboots. Therefore, you can use the finish script to add, change, or remove files from the newly installed file system hierarchy by modifying the file systems respective to `/a`.
- Output from the finish script goes to `/var/sadm/finish.log`.
- Finish scripts should be owned by root and have permissions equal to 644.

Ideas for Finish Scripts

You could set up finish scripts to perform the following tasks:

- Installing patches
- Restoring backed up files
- Setting up print servers
- Adding entries to the automount map

The following finish scripts are provided as examples:

- Adding files
- Customizing the root environment
- Setting the system's root password

Adding Files With Finish Scripts

Through a finish script, you can add files from the JumpStart directory to the already installed system. This is possible because the JumpStart directory is mounted on the directory specified by the `SI_CONFIG_DIR` variable (which is set to `/tmp/install_config` by default).

Note – You can also replace files by copying files from the JumpStart directory to already existing files on the installed system.

The following procedure enables you to create a finish script to add files to a system after the Solaris software is installed on it:

1. Copy all the files you want added to the installed system into the JumpStart directory.
2. Insert the following line into the finish script for each file you want copied into the newly installed file system hierarchy.

```
cp ${SI_CONFIG_DIR}/file_name /a/path_name
```

For example, assume you have a special application, `site_prog`, developed for all users at your site. If you place a copy of `site_prog` into the JumpStart directory, the following finish script would copy the `site_prog` from the JumpStart directory into a system's `/usr/bin` directory during a custom JumpStart installation:

```
#!/bin/sh
cp ${SI_CONFIG_DIR}/site_prog /a/usr/bin
```

Customizing the Root Environment

Through a finish script, you can customize files already installed on the system. For example, the following finish script customizes the root environment by appending information to the `.cshrc` file in the root directory.

```
#!/bin/sh
#
# Customize root's environment
#
echo "***adding customizations in /.cshrc"
test -f a/.cshrc || {
cat >> a/.cshrc <<EOF
set history=100 savehist=200 filec ignoreeof prompt="\$user@\`uname -n`> "
alias cp cp -i
alias mv mv -i
alias rm rm -i
alias ls ls -FC
alias h history
alias c clear
unset autologout
EOF
}
```

Setting the System's Root Password With Finish Scripts

After Solaris software is installed on a system, the system reboots. Before the boot process is completed, the system prompts for the root password. This means that until someone enters a password, the system cannot finish booting.

The `auto_install_sample` directory provides a finish script called `set_root_pw` that sets the root password for you. This allows the initial reboot of the system to be completed without prompting for a root password.

The `set_root_pw` file is shown below.

```
#!/bin/sh
#
#      @(#)set_root_pw 1.4 93/12/23 SMI
#
# This is an example bourne shell script to be run after installation.
# It sets the system's root password to the entry defined in PASSWD.
# The encrypted password is obtained from an existing root password entry
# in /etc/shadow from an installed machine.

echo "setting password for root"

# set the root password
❶ PASSWD=dK05IBkSF42lw
#create a temporary input file
❷ cp /a/etc/shadow /a/etc/shadow.orig

mv /a/etc/shadow /a/etc/shadow.orig
nawk -F: '{
❸   if ( $1 == "root" )
       printf"%s:%s:%s:%s:%s:%s:%s:%s:%s\n", $1,passwd,$3,$4,$5,$6,$7,$8,$9
   else
       printf"%s:%s:%s:%s:%s:%s:%s:%s:%s\n", $1,$2,$3,$4,$5,$6,$7,$8,$9
   }' passwd="$PASSWD" /a/etc/shadow.orig > /a/etc/shadow
❹ #remove the temporary file
rm -f /a/etc/shadow.orig
❺ # set the flag so sysidroot won't prompt for the root password
sed -e 's/0# root/1# root/' ${SI_SYS_STATE} > /tmp/state.$$
mv /tmp/state.$$ ${SI_SYS_STATE}
```

There are several things you must do to set the root password in a finish script.

- ❶ Set the variable `PASSWD` to an encrypted root password obtained from an existing entry in a system's `/etc/shadow` file.
- ❷ Create a temporary input file of `/a/etc/shadow`.
- ❸ Change the root entry in the `/etc/shadow` file for the newly installed system using `$PASSWD` as the password field.
- ❹ Remove the temporary `/a/etc/shadow` file.

- ⑤ Change the entry from 0 to a 1 in the state file, so that the user will not be prompted for the root password. The state file is accessed using the variable `SI_SYS_STATE`, whose value currently is `/a/etc/.sysIDtool.state`. (To avoid problems with your scripts if this value changes, always reference this file using `$SI_SYS_STATE`.) The `sed` command shown here contains a tab character after the 0 and after the 1.

Note – If you set your root password by using a finish script, be sure to safeguard against those who will try to discover the root password from the encrypted password in the finish script.

Using `pfinstall` to Test Profiles

Why Use `pfinstall`

When `install_type initial_install` is defined in a profile, you can use the `pfinstall` command to test the profile without actually installing the Solaris software on a system. `pfinstall` shows the results of how a system would be installed according to the specified profile, before you actually perform a custom JumpStart installation.

Note – You cannot use `pfinstall` to test a profile using the upgrade option (when the `install_type upgrade` is defined in a profile).

Ways to Use `pfinstall`

`pfinstall` enables you to test a profile against:

- The system's disk configuration where `pfinstall` is being run.
- A disk configuration file that you can create with the `prtvtoc` command. A *disk configuration file* is a file that represents a structure of a disk (for example, bytes/sector, flags, slices). Disk configuration files enable you to use `pfinstall` from a single system to test profiles on different sized disks.

Note – You must run `pinstall` on the same platform type as the profiles you are testing. For example, you must run `pinstall` on an x86 system to test profiles for x86 systems (using the x86 system’s disk configuration or an x86 disk configuration file).

▼ How to Use `pinstall` to Test a Profile

Overview – The procedure to use `pinstall` to test a profile involves:

- Changing the directory to the JumpStart directory
- Using the `pinstall` command to test the profile

Follow this procedure to use `pinstall` to test a profile.

1. **To test the profile with a specific system memory size, set `SYS_MEMSIZE` to the specific memory size in Mbytes:**

```
$ SYS_MEMSIZE=memory_size
$ export SYS_MEMSIZE
```

2. **Change the directory to the JumpStart directory where the profile resides:**

```
$ cd jumpstart_dir_path
```

For example, the following command would change the directory to the `jumpstart` directory on the root file system.

```
cd /jumpstart
```

3. **Run the `pinstall -d` or `pinstall -D` command to test the profile:**



Caution – Without the `-d` or `-D` option, `pinstall` will install the Solaris software on the system by using the specified profile, and the data on the system will be overwritten.

```
$ /usr/sbin/install.d/pinstall -D | -d disk_config [-c path] profile
```

In this command,

<code>-D</code>	Tells <code>pfinstall</code> to use the current system's disk configuration to test the profile against. You must be root to execute <code>pfinstall</code> with the <code>-D</code> option.
<code>-d <i>disk_config</i></code>	Tells <code>pfinstall</code> to use a disk configuration file, <i>disk_config</i> , to test the profile against.
<code>-c <i>path</i></code>	Is the path to the Solaris CD. This is required if the Solaris CD is not mounted on <code>/cdrom</code> . (For example, use this option if you copied the Solaris CD image to disk or mounted the Solaris CD on a directory other than <code>/cdrom</code>).
<i>profile</i>	The name of the profile to test.

Note – You should run `pfinstall` on a system running the same version of Solaris software that will be installed by the profile. Otherwise, use `pfinstall` on the Solaris CD that will be installed by the profile, which is located in the `/export/exec/arch.Solaris_2.5/sbin/install.d` directory.

Run `pfinstall` from the directory where the *profile* and *disk_config* files reside (which should be the JumpStart directory). If the *profile* or *disk_config* file is not in the directory where `pfinstall` is run, you must specify the path.

4. Check to see if the results of `pfinstall` are as you expected. If not, change the profile and go to Step 3.

Task Complete

You have completed testing the profile. To perform a custom JumpStart installation on a system, see Chapter 7, “Booting and Installing Solaris: Custom JumpStart.”

`pfinstall` *Examples*

Below are some examples of using `pfinstall` to test the `basic_prof` profile against the `104_test` disk configuration file:

```
/usr/sbin/install.d/pfinstall -D basic_prof  
  
/usr/sbin/install.d/pfinstall -d 104_test basic_prof  
  
/usr/sbin/install.d/pfinstall -D -c /export/install basic_prof
```

Creating a Disk Configuration File for a PowerPC System

The following procedures enable you to use the `-d` option of the `pfinstall` command to test custom JumpStart installations for a PowerPC system.

▼ How to Create a Disk Configuration File for a PowerPC System

A disk configuration file represents a disk structure (for example, bytes/sectors, flags, slices), and it enables you to use `pfinstall` from a single system to test profiles against different sized disks. Disk configuration files for a PowerPC system must also contain information about a disk's `fdisk` partitions.

Overview – Creating a disk configuration file for a PowerPC system involves:

- Locating a PowerPC system with a disk that you want to test a profile against
- Saving the output of the `prtvtoc` command to a file
- Saving the output of the `fdisk` command to a file
- Concatenating the two files to create a disk configuration file

Follow this procedure to create a disk configuration file for a PowerPC system:

1. **Locate a PowerPC system with a disk that you want to test a profile against.**
2. **Determine the device name for the system's disk.**
3. **Redirect the output of the following `prtvtoc` command to a file:**

```
# prtvtoc /dev/rdisk/device_name > file1
```

where `/dev/rdisk/device_name` is the device name of the system's disk, and `file1` is the file that contains the output of the `prtvtoc` command. `device_name` must be in the form `cwtxdyp0` or `cxryp0`. Partition 0 must be specified in `device_name`.

4. Save the output of the following `fdisk` command to a file:

```
# fdisk -R -d -n /dev/rdisk/device_name 2>file2
```

Note – This version of the `fdisk` command may not be supported in the next release.

where `/dev/rdisk/device_name` is the device name of the system's disk. `file2` is the file that contains the output of the `fdisk` command. `device_name` must be in the form `cwtxdyp0` or `cxryp0`. Partition 0 must be specified in `device_name`.

5. Concatenate the two files to create a disk configuration file:

```
# cat file1 file2 > disk_config
```

Note – The output of the `prtvtoc` command must be first in a disk configuration file for a PowerPC system.

6. Copy the disk configuration file to the JumpStart directory:

```
# cp disk_config jumpstart_dir_path
```

Task Complete

You have created a disk configuration file for a PowerPC system. The following page provides an example of creating a disk configuration file. This example creates a disk configuration file, `500_test`, on a PowerPC system with a 500-Mbyte disk.

First, you would save the output of the `prtvtoc` command to a file:

```
# prtvtoc /dev/rdisk/c0t0d0p0 > output1
```

The output1 file is shown as follows:

```
* /dev/rdisk/c0t0d0p0 partition map
*
* Dimensions:
*   512 bytes/sector
*   85 sectors/track
*   9 tracks/cylinder
*   765 sectors/cylinder
*   2662 cylinders
*   2660 accessible cylinders
*
* Flags:
*   1: unmountable
*   10: read-only
*
*
* Partition  Tag  Flags      First   Sector   Last
* Partition  Tag  Flags      Sector  Count    Sector  Mount Directory
*   0         0    00         2295   655605   657899
*   1         3    01        657900 65790    723689
*   2         5    00          0    2034900 2034899
*   4         0    00        723690 654075   1377764
*   7         0    00       1377765 655605   2033369
*   8         1    01          0         765     764
*   9         9    01          765     1530    2294
```

Second, you would save the output of the `fdisk` command to a different file:

```
# fdisk -R -d -n /dev/rdisk/c0t0d0p0 2>output2
```

The output2 file is shown as follows:

```
fdisk physical geometry:
cylinders[2694] heads[9] sectors[85] sector size[512] blocks[2060910] mbytes[1006]
fdisk virtual (HBA) geometry:
cylinders[1023] heads[33] sectors[61] sector size[512] blocks[2059299] mbytes[1005]
fdisk table on entry:
SYSID  ACT  BHEAD  BSECT  BEGCYL   EHEAD  ESECT  ENDCYL   RELSECT  NUMSECT
65     0    1     2     0       0     35    1       1       2047
1     128  0    36    1       2     18    4       2048    6144
130   0    2    19    4       26    217   255     8192    2052718
100   0    0     0     0       0     0     0       100     100
```

Finally, you would concatenate the two files (output1 and output2) together to create the disk configuration file named 500_test.

```
# cat output1 output2 > 500_test
```

The 500_test file is shown as follows:

```
* /dev/rdisk/c0t0d0p0 partition map
*
* Dimensions:
*   512 bytes/sector
*   85 sectors/track
*   9 tracks/cylinder
*   765 sectors/cylinder
*   2662 cylinders
*   2660 accessible cylinders
*
* Flags:
*   1: unmountable
*   10: read-only
*
*
* Partition  Tag  Flags      First   Sector   Last
* Partition  Tag  Flags      Sector  Count    Sector  Mount Directory
*   0         0    00         2295   655605   657899
*   1         3    01        657900 65790    723689
*   2         5    00          0    2034900 2034899
*   4         0    00        723690 654075   1377764
*   7         0    00       1377765 655605   2033369
*   8         1    01          0        765     764
*   9         9    01         765     1530    2294
fdisk physical geometry:
cylinders[2694] heads[9] sectors[85] sector size[512] blocks[2060910] mbytes[1006]
fdisk virtual (HBA) geometry:
cylinders[1023] heads[33] sectors[61] sector size[512] blocks[2059299] mbytes[1005]
fdisk table on entry:
SYSID  ACT  BHEAD  BSECT  BEGCYL  EHEAD  ESECT  ENDCYL  RELSECT  NUMSECT
65     0    1     2     0     0     35    1     1     2047
1     128  0    36    1     2     18    4     2048    6144
130    0    2    19    4     26    217   255    8192    2052718
100    0    0     0     0     0     0     0     100     100
```

▼ How to Create a Multiple Disk Configuration File for a PowerPC System

If you need to test a profile on multiple disks, you can concatenate disk configuration files together to create multiple disk configuration scenarios.

Creating a multiple disk configuration file for a PowerPC system involves:

- Concatenating two or more disk configuration files to create a multiple disk configuration file
- Changing the target numbers of disks in the multiple disk configuration file

Note – Disk device target numbers must be unique on a system.

The following procedure creates a multiple disk configuration file. (The procedure uses the `500_test` file from the previous procedure.)

1. Concatenate a disk configuration file with itself and save it to a file.

The new file becomes the multiple disk configuration file. For example, the following command creates a multiple disk configuration file named `dual_500_test`:

```
$ cat 500_test 500_test > dual_500_test
```

2. Edit the disk configuration file so that each disk device name has a different target number.

The dual_500_test file is shown as follows:

```

❶ * /dev/rdisk/c0t0d0p0 partition map
*
* Dimensions:
*   512 bytes/sector
*   85 sectors/track
*   9 tracks/cylinder
*   765 sectors/cylinder
*   2683 cylinders
*   2681 accessible cylinders
*
* Flags:
*   1: unmountable
*   10: read-only
*
*
* Partition  Tag  Flags      First      Sector      Last
*           Tag  Flags      Sector      Count      Sector  Mount Directory
*   0         0   00          295       660195     662489
*   1         3   01      662490     68085     730574
*   2         5   00           0     2051730     2051729
*   4         0   00      730575     660195     1390769
*   7         0   00     1390770     660195     2050964
*   8         1   01           0         765         764
*   9         9   01       765       1530       2294
fdisk physical geometry:
cylinders[2694] heads[9] sectors[85] sector size[512] blocks[2060910] mbytes[1006]
fdisk virtual (HBA) geometry:
cylinders[1023] heads[33] sectors[61] sector size[512] blocks[2059299] mbytes[1005]
fdisk table on entry:
SYSID ACT BHEAD BSECT BEGCYL  EHEAD ESECT ENDCYL  RELSECT  NUMSECT
65    0  1    2    0    0    35    1    1    2047
1    128 0    36    1    2    18    4    2048    6144
130  0  2    19    4    26   217   255    8192   2052718
100  0  0    0    0    0    0    0    100    100

```

continued

```

❷ * /dev/rdisk/c0t1d0p0 partition map
*
* Dimensions:
*   512 bytes/sector
*   85 sectors/track
*   9 tracks/cylinder
*   765 sectors/cylinder
*   2683 cylinders
*   2681 accessible cylinders
*
* Flags:
* 1: unmountable
* 10: read-only
*
*
* Partition      Tag  Flags      First      Sector      Last
* Partition      Tag  Flags      Sector     Count       Sector  Mount Directory
*   0             0    00         295       660195     662489
*   1             3    01       662490     68085     730574
*   2             5    00           0     2051730     2051729
*   4             0    00       730575     660195     1390769
*   7             0    00     1390770     660195     2050964
*   8             1    01           0         765         764
*   9             9    01         765         1530        2294
fdisk physical geometry:
cylinders[2694] heads[9] sectors[85] sector size[512] blocks[2060910] mbytes[1006]
fdisk virtual (HBA) geometry:
cylinders[1023] heads[33] sectors[61] sector size[512] blocks[2059299] mbytes[1005]
fdisk table on entry:
SYSID ACT BHEAD BSECT BEGICYL  EHEAD ESECT ENDCYL  RELSECT  NUMSECT
65    0  1    2    0    0    35    1    1    2047
1    128 0    36    1    2    18    4    2048    6144
130  0  2    19    4    26   217   255    8192   2052718
100  0  0    0    0    0    0    0    100    100

```

Because the `dual_500_test` file was created by concatenating itself, the following editing was required:

- ❶ The first disk device name was left as is
- ❷ The second disk device name was changed from `/dev/rdisk/c0t0d0p0` to `/dev/rdisk/c0t1d0p0` so it has a unique target number.

Task
Complete

You have created a multiple disk configuration file for a PowerPC system.

Using a Site-Specific Installation Program

Through the use of begin and finish scripts, sites with special requirements can install the Solaris software by creating their own installation program. When a minus sign (-) is specified in the profile field, the begin and finish scripts control how the system is installed, instead of the profile and the Solaris installation program.

For example, if the following rule would match, the `x_install.beg` begin script and the `x_install.fin` finish script would install the system named `sherlock` (the Solaris installation program would not be used):

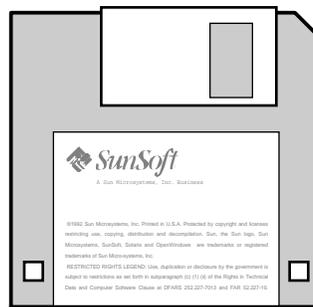
```
hostname sherlock x_install.beg - x_install.fin
```

Booting and Installing Solaris: Interactive



This chapter provides procedures to boot a system and perform an interactive installation using the Solaris installation program. The procedure in this chapter should be done on the system that is being installed.

- 1 Insert the Solaris boot diskette (shown below) into the system's a: diskette drive.

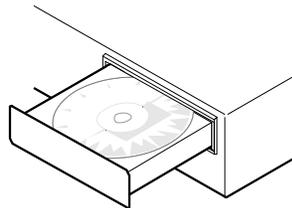


- 2 If you are using the system's local CD-ROM drive to install the Solaris software on the system, prepare the CD-ROM drive.

If your CD-ROM drive is not a tray version, see your hardware manual for instructions.

CD-ROM Drive - Tray Version

- 1 Press the eject button on the front panel of the CD-ROM drive (a tray will emerge if you have this type of caddy).



- 2 Insert the Solaris CD (logo up) into the tray.
 - 3 Push the tray back into CD-ROM drive.
-

- 3 **Make sure you've reviewed the task map for an interactive installation (shown in Figure 2-2 on page 13).**
- 4 **If you are installing a system connected through a tip line, ensure that your window display is at least 80 columns wide and 24 rows long.**

Otherwise, the character installation interface will display improperly. You can use the `stty` command to find out the current dimensions of your tip window.

- 5 **Turn on the system components.**

Follow the order listed in your hardware guide. The booting process starts automatically.

- 6 **Wait for booting to complete.**

Hardware and system components are checked during the booting phase. This lasts for several minutes. The following screen provides an example of what you should see.

During the booting phase, OpenWindows may be started for localization reasons, or for the graphical user interface; however, OpenWindows is not fully installed until the Solaris software is installed.



```
Boot device: /pci/pci8086,4840
Boot args:
...
Configuring the /devices directory
Configuring the /dev directory
...
Starting OpenWindows...
```

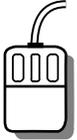
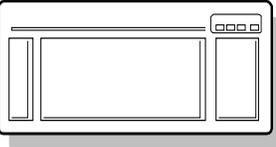
- 7 **Select a terminal type.**

For all PowerPC systems, select option 8 - SunWorkstation.

8 Install the Solaris software on the system by using the Solaris installation program displayed on the screen.

You are now in the hands of the Solaris installation program. This menu-driven, interactive program guides you step-by-step through installing Solaris software; it also has online help to answer your questions. As shown below, the program has two interfaces to accommodate different hardware.

If you want to record your responses, use the work sheet in Appendix A, “Worksheets for the Solaris Installation Program.” There is also a time zone map on page 156 to help you set your system’s time clock.

If Your System Has A ...	Then the Interface Is ...	And the Navigation Device Is ...
Graphics monitor	<p style="text-align: center;">Graphics-based</p> <div style="border: 1px solid black; padding: 10px;"> <p style="text-align: center;">The Solaris Installation Program</p> <p>You are now interacting with the Solaris installation program. The program is divided into a series of short sections. At the end of each section, you will see a summary of the choices you’ve made, and be given the opportunity to make changes.</p> <hr/> <p style="text-align: center;"> <input type="button" value="Continue"/> <input type="button" value="Exit"/> <input type="button" value="Help"/> </p> </div>	<p style="text-align: center;">Mouse</p> 
Non-graphics monitor, or an unsupported graphics card	<p style="text-align: center;">Character-based</p> <div style="border: 1px solid black; padding: 10px;"> <pre> --- The Solaris Installation Program You are now interacting with the Solaris installation program. The program is di- vided into a series of short sections. At the end of each section, you will see a summary of the choices you’ve made, and be given the opportunity to make changes. ----- F2_Continue F5_Help </pre> </div>	<p style="text-align: center;">Keyboard</p> 

- 9 Start the OpenWindows software to display the desktop:

```
$ /usr/openwin/bin/openwin
```

- 10 If you've allocated space for diskless clients, dataless clients, or AutoClient systems during the Solaris installation program, you must use the Solstice Host Manager to complete set up of these clients.

The Solaris installation program only allocates space for clients during an initial installation. The Solstice Host Manager completes client set up by providing their required directories. See the *System Administration Guide, Volume I*.

Task
Complete

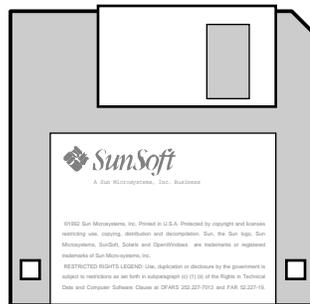
For post-installation information, see Chapter 8, "Where to Go After Installing Solaris."

Booting and Installing Solaris: Custom JumpStart



This chapter provides a procedure to boot a system and perform a custom JumpStart installation using profiles you've created to install Solaris software. The procedure in this chapter should be done on the system that is being installed.

- 1 Insert the Solaris boot diskette (shown below) into the system's a: diskette drive.



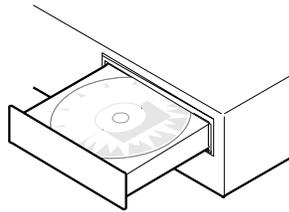
Note – If you are using a diskette to perform a custom JumpStart installation, you must insert a copy of the Solaris boot diskette that contains a JumpStart directory into the system's a: diskette drive.

- 2 If you are using the system's local CD-ROM drive to install the Solaris software on the system, prepare the CD-ROM drive.

If your CD-ROM drive is not a tray type, see your hardware manual for instructions.

CD-ROM Drive - Tray Version

- 1 Press the eject button on the front panel of the CD-ROM drive (a tray will emerge if you have this type of caddy).



- 2 Insert the Solaris CD (logo up) into the tray.
 - 3 Push the tray back into the CD-ROM drive.
-

3 Make sure you have reviewed the task map for a custom JumpStart installation (shown in Figure 2-3 on page 14).

4 Follow the instructions before you boot the system:

If The System Is ...	Then ...
Off	<ol style="list-style-type: none">1) Turn on the system components in the order recommended in your hardware guide.2) Go to Step 7.
On	<ol style="list-style-type: none">1) If the system is running Solaris, enter the following commands: \$ su root # halt2) Go to Step 7.

5 Boot the system using the appropriate boot command:

If You Are Booting From ...	Then Enter ...
A server on the network	<code>boot net</code>
The system's local CD-ROM drive	<code>boot cdrom</code>

6 Wait for booting to complete.

Hardware and system components are checked during the booting phase. This lasts for several minutes. The following screen provides an example of what you should see.

During the booting phase, OpenWindows may be started for localization reasons, or for the graphical user interface; however, OpenWindows is not fully installed until the Solaris software is installed.



```
ok boot net
Boot device: /pci/pci8086,4840
Boot args:
...
Configuring the /devices directory
Configuring the /dev directory
...
Starting OpenWindows...
```

7 If prompted, provide information about the system.

After the booting phase, the Solaris installation program may be displayed on the screen, prompting you to provide information about the system.

8 Wait as the Solaris installation program automatically installs the Solaris software on the system.

You're done for awhile; installing Solaris software can take between 15 minutes and 2 hours.

- 9 Start the OpenWindows software to display the desktop:

```
$ /usr/openwin/bin/openwin
```

- 10 If you installed Solaris on a server and allocated space for diskless clients, dataless clients, or AutoClient systems, and you selected the initial installation option, you must use the Solstice Host Manager to complete set up of these clients.

The Solaris installation program only allocates space for clients during an initial installation. The Solstice Host Manager completes client set up by providing their required directories. See the *System Administration Guide, Volume I*.

Task
Complete

For post-installation information, see Chapter 8, “Where to Go After Installing Solaris.”

Where to Go After Installing Solaris



Table 8-1 shows the Solaris documentation that you may need after installing the Solaris software on a system. For a complete description of all the Solaris 2.5 documentation, refer to the *Solaris 2.5 Introduction*.

Table 8-1 Where to Go After Installing Solaris Software

Information Needed	Manual Title
Adding and removing Solaris, third-party, or unbundled software	<i>Solstice AdminSuite 2.2 Administration Guide</i>
Setting up mail accounts	<i>Mail Administration Guide</i>
Installing software (for example AnswerBooks)	<i>System Administration Guide, Volume I</i> Note: The End User AnswerBook is the only AnswerBook installed with the Solaris software; and is only installed if the entire distribution software group is installed.
Setting up user accounts	<i>System Administration Guide, Volume I</i>
Halting a system	<i>System Administration Guide, Volume I</i>
Boot files	<i>System Administration Guide, Volume I</i>

Table 8-1 Where to Go After Installing Solaris Software (Continued)

Information Needed	Manual Title
Adding systems to a network	<i>System Administration Guide, Volume I</i>
Accessing remote files and systems	<i>System Administration Guide, Volume I</i>
Administering file systems	<i>System Administration Guide, Volume I</i>
Setting up system security	<i>System Administration Guide, Volume I</i>
CD-ROM and diskette drives	<i>System Administration Guide, Volume I</i>
Setting up printers	<i>System Administration Guide, Volume II</i>
Increasing your system's performance	<i>System Administration Guide, Volume II</i>
Managing disk use	<i>System Administration Guide, Volume II</i>
Examining and changing system information	<i>System Administration Guide, Volume II</i>
Using crontabs	<i>System Administration Guide, Volume II</i>
Adding and Maintaining Peripherals	<i>System Administration Guide, Volume II</i>
Accessing devices	<i>System Administration Guide, Volume II</i>
Setting up disks	<i>System Administration Guide, Volume II</i>
Terminals and modems, disk drives, tape drives, service access facility, connecting devices to serial port, format utility	<i>System Administration Guide, Volume II</i>

Table 8-1 Where to Go After Installing Solaris Software (Continued)

Information Needed	Manual Title
Using system administration tools	<i>Solstice AdminSuite 2.2 Administration Guide</i>

Worksheets for the Solaris Installation Program



What's in the Worksheets

The worksheets list information that may need to be supplied during the Solaris installation program. There are separate worksheets for the initial and upgrade options.

How to Use the Worksheets

The worksheets can be used before installing Solaris software, or to record your responses when you install Solaris software. For example, if you're a system administrator of a large site and need to delegate the task of installing Solaris software to less-experienced staff, you can fill out the worksheet in advance; installers will know exactly how you want them to install Solaris software on their system, thus reducing the errors that come from guessing.

What shaded areas mean
in the following table

Note – If you're using the following worksheet to gather information before installing Solaris software, you can ignore shaded areas if you are installing a system as a standalone, non-networked system.



Worksheet for the Initial Installation Option

The following worksheet lists important information you may need to supply when installing Solaris software on a system for the first time.

Information You May Need When Installing Solaris	Description/Example	Work Space
What type of terminal are you using?	Terminal type that you are installing Solaris software Example: 8) Sun Workstation	
Keyboard language	Language your keyboard supports. Example: UK English	
Graphics card	Type of graphics card in your system. Example: Diamond Viper	
Screen size	Size of the screen on your monitor from the following list: - 15 inch - 17 inch - 19 inch	
Color depth	Number of bits for the color plane that the graphics card supports. Example: 8, 24	
Display resolution	Screen resolution. Example: 640 X 480	

Information You May Need When Installing Solaris	Description/Example	Work Space
Monitor type	Type of monitor. IBM-VGA (31.5kHz)	
Pointing device	Type of mouse. Example: PS/2 style built in mouse	
Host name	Host name of the system. Example: sherlock	
Network connectivity	Is the system connected to a network?	
Primary network interface	Primary network interface for the system. Example: smc0	
IP address	Internet™ protocol address of the system you're installing. Example: 129.221.2.1	
Name service	Is the system a client of NIS+ or NIS? Part of another name service? Or is a name service not yet established? Example: Other, DCE	
Domain name	Domain in which the system resides. Example: chandy.West.Arp.COM	



Information You May Need When Installing Solaris	Description/Example	Work Space
Name server	Specify a name server for the system, or have software try to find one?	
Name server information	Name server's host name and IP address. Example: toucan, 125.221.2.1	
Subnet	Is the system part of a subnet?	
Netmask	Netmask of the subnet. Example: 255.255.0.0	
Time zone	Default time zone. Can be set by selecting geographic region, hours offset from GMT, or by pointing to a file in <code>/usr/share/zoneinfo</code> directory. Example: US/Mountain	
System type	System type: standalone system, OS server, or dataless client?	
Select platforms	All the platforms for clients that are <i>different</i> from the OS server's platform (see Appendix C, "Platform Names and Groups").	

Information You May Need When Installing Solaris	Description/Example	Work Space
Allocate client services	<p>Number of clients the OS server will support. Example: 5 (default)</p> <p>Number of megabytes of swap, root, or both to assign clients. Example: 24 Mbytes (default)</p>	
Dataless clients	<p>Note: SunSoft plans to remove support for the dataless client system type after Solaris 2.5. You can select this system type now, but in future releases you will need to select a different type.</p> <p>Host names and IP addresses of the server(s) where dataless clients will mount their <code>/usr</code> file systems. Example: morton, 112.221.2.1, /export/home</p>	
Languages	<p>Language(s) to select for displaying the user interface after installing Solaris software. Example: French, Spanish</p>	



Information You May Need When Installing Solaris	Description/Example	Work Space
Software	<p>Which software group to install?</p> <ul style="list-style-type: none">• Core• End user system support• Developer system support• Entire distribution <p>Should the software group be customized?</p> <p>Note: Removing or adding software often creates software dependencies; system administration knowledge is often required to fix dependencies.</p>	
Disk(s) to install Solaris software on	<p>Disk(s) for installing Solaris. Example: c0t0d0, c0t3d0, c0t4d1</p>	
Create Solaris partition	<p>If there is no Solaris partition on any of the disks selected, a Solaris partition must be created. Which partition, 1-4? What size should it be? Example: Solaris on partition 3, 198 Mbytes.</p>	
Auto-layout file systems?	<p>Should file systems be laid out on disks automatically or manually?</p> <p>Note: Manual layout requires advanced system administration skills.</p>	

Information You May Need When Installing Solaris	Description/Example	Work Space
File systems to auto-layout	Which file systems should be used for auto-layout? Example: /, /opt, /var	
Preserve existing data?	Should existing data on disk(s) be preserved?	
Disk slices to preserve data	Which slices should be preserved? Example: /opt on c0t2d2	
File system and disk layout	Accept the layout of file systems as created by auto-layout or manual layout, or customize? Note: Customizing requires advanced system administration skills.	
Mount remote file systems?	Mount remote file systems from a server?	



Information You May Need When Installing Solaris	Description/Example	Work Space
Mount remote file systems	Provide remote file system information: <ul style="list-style-type: none">• Local mount point• Server's host name• Server's IP address• File system path Example: Server: mitra Server IP address: 129.222.2.2 File system path: /export/home Local mount point: /export	
Reboot	Reboot after installing Solaris?	
Root password	Create root password. Example: cat6file	

System Identification Label



For your convenience, the following pages contain system identification labels (a sample, and one for photocopying). Attach them to systems at your site so users have access to information needed to install Solaris software and operate their system.



Sample - System Identification Label

As shown in the following sample, system identification labels can provide a quick reference to important system information.

Host name: roylat

System type

- Standalone system
- OS server
- Diskless client
- Dataless client
- AutoClient system

Boot device network

Keyboard language UK English

Graphics card Diamond Viper

Screen size 15"

Color Depth 8, 24

Display resolution 640 X 480

Monitor type IBM-VGA (31.5kHz)

Pointing device PS/2 style built in mouse

Networked? Yes

Primary network interface smc0

IP address 129.221.2.1

Name service NIS+

Domain name hawthorne.West.Arp.COM

System part of a subnet? Yes

Netmask 225.225.225.225

Name server and IP address hudson, 125.222.2.2

Ethernet address 8:0:20:b:40:e7



System Identification Label

Photocopying the following label and identifying system information for each system provides a quick reference for users.

Host name:

System type

- Standalone system
- OS server
- Diskless client
- Dataless client
- AutoClient system

Boot device _____

Keyboard language _____

Graphics card _____

Screen size _____

Color Depth _____

Display resolution _____

Monitor type _____

Pointing device _____

Networked? _____

Primary network interface _____

IP address _____

Name service _____

Domain name _____

System part of a subnet? _____

Netmask _____

Name server and IP address _____

Ethernet address _____

Platform Names and Groups



Table C-1 shows the platform names of various hardware platforms. You may need this information when preparing a system to install Solaris software.

Use `uname -i` to determine a system's platform name; use `uname -m` to determine a system's platform group.

Table C-1 Platform Names and Groups

System	Platform Name	Platform Group
x86	i86pc	i86pc
ThinkPad Power Series 820	IBM,PPS_Model_6040 IBM,PPS_Model_6040_E	prep
ThinkPad Power Series 850	IBM,PPS_Model_6042 IBM,PPS_Model_6042_E	prep
Power Series 830 or 850	IBM,PPS_Model_6050_6070 IBM,PPS_Model_6050_6070_E	prep
PowerStack Series E	Motorola,PowerStack_Series_E	prep
PowerStack RISC PC DT or MT	Motorola,PowerStack_MT	prep
SPARCstation 1	SUNW,Sun_4_60	sun4c
SPARCstation1+	SUNW,Sun_4_65	sun4c

Table C-1 Platform Names and Groups

System	Platform Name	Platform Group
SPARCstation SLC	SUNW,Sun_4_20	sun4c
SPARCstation ELC	SUNW,Sun_4_25	sun4c
SPARCstation IPC	SUNW,Sun_4_40	sun4c
SPARCstation IPX	SUNW,Sun_4_50	sun4c
SPARCstation 2	SUNW,Sun_4_75	sun4c
SPARCcenter 1000	SUNW,SPARCserver-1000	sun4d
SPARCcenter 2000	SUNW,SPARCcenter-2000	sun4d
SPARCstation 5	SUNW,SPARCstation-5	sun4m
SPARCstation 10	SUNW,SPARCstation-10	sun4m
SPARCstation 10SX	SUNW,SPARCstation-10,SX	sun4m
SPARCstation 20	SUNW,SPARCstation-20	sun4m
SPARCserver6xx	SUNW,SPARCsystem-600	sun4m
SPARCstation LX	SUNW,SPARCstation-LX	sun4m
SPARCstation LX+	SUNW,SPARCstation-LX+	sun4m
SPARCclassic	SUNW,SPARCclassic	sun4m
SPARCclassic X	SUNW,SPARCclassic-X	sun4m
SPARCEngine EC3	SUNW,SPARCEngine-EC-3	sun4m
SPARCstation Voyager	SUNW,S240	sun4m
Sun Ultra 1 Sun UltraServer 1	SUNW,Ultra-1	sun4u
Other SPARC systems	See your hardware vendor documentation for platform name information.	

Sample Custom JumpStart Installation



This example shows a set of steps a system administrator would take to do a custom JumpStart installation for a fictitious site.

Sample Site Setup

Figure D-1 shows the sample site setup for this example.

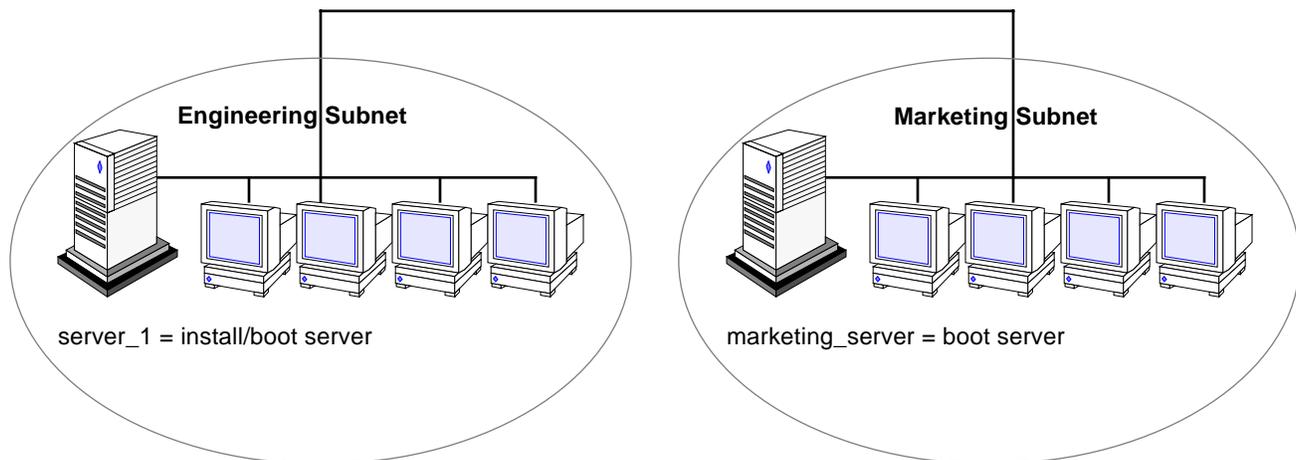


Figure D-1 Sample Site Setup

At this fictitious site:

- The engineering group is on its own subnet. This group uses Motorola Power Stack systems for software development.
- The marketing group is on its own subnet. This group uses IBM ThinkPad Power Series systems for running word processing, spreadsheets, and other office tools.
- The site uses NIS+. The Ethernet addresses, IP addresses, and host names are in NIS+ tables.
- The engineering server named `server_1` has a copy of Solaris 2.5 software on its local disk in a directory named `/export/install`. Both the engineering and marketing groups will install Solaris software over the network from `server_1`.

1 Create a JumpStart directory.

The administrator sets up a JumpStart directory on the install server, `server_1`. This directory will hold files necessary for a custom JumpStart installation of Solaris software. The easiest way to set up this directory is to copy the sample directory from the copy of the Solaris CD that has been put in `/export/install`.

```
# cp -r /export/install/auto_install_sample /jumpstart
```

2 Share the JumpStart directory.

The system administrator shares the `/jumpstart` directory so that the rules file and profiles are accessible to systems on the network. To accomplish this, the administrator adds the following line to the `/etc/dfs/dfstab` file:

```
share -F nfs -o ro,anon=0 /jumpstart
```

Then, at the command line, the administrator uses the `unshareall` and `shareall` commands:

```
# unshareall  
# shareall
```

3 Create the `eng_profile` profile.

The administrator creates a file named `eng_profile` in the `/jumpstart` directory. The `eng_profile` file has the following entries, which define the Solaris software to be installed on systems in the engineering group.

```
❶ install_type  initial_install  
❷ system_type   standalone  
❸ partitioning  default  
❹ cluster       SUNWCprog  
❺ filesys       any 50 swap
```

- ❶ Specifies that the installation will be treated as an initial installation, as opposed to an upgrade.
- ❷ Specifies that the engineering systems are standalone systems.
- ❸ Specifies that the JumpStart software uses default disk partitioning for installing Solaris on the engineering systems.
- ❹ Specifies that the developer's software cluster will be installed.
- ❺ Specifies that each system in the engineering group will have 50 Mbytes of swap space.

4 Create the `marketing_profile` profile.

The administrator creates a file named `marketing_profile` in the `/jumpstart` directory. The `marketing_profile` file has the following entries, which define the Solaris software to be installed on systems in the marketing group.

```
❶ install_type  initial_install
❷ system_type  standalone
❸ partitioning default
❹ cluster      SUNWCuser
❺ package      SUNWaudmo
```

- ❶ Specifies that the installation will be treated as an initial installation, as opposed to an upgrade.
- ❷ Specifies that the marketing systems are standalone systems.
- ❸ Specifies that the JumpStart software will use default disk partitioning for installing Solaris on the marketing systems.
- ❹ Specifies that the end user software cluster is to be installed.
- ❺ Specifies that the audio demo software package is to be added to each system.

5 Edit the `rules` file.

The administrator must define the `rules` file. The Solaris installation program will use the contents of this file to select the proper installation for each department.

At this site, each department is on its own subnet and has its own network address. The administrator uses this information to control how systems are installed. The engineering department is on subnet 255.222.43.0, and marketing is on 255.222.44.0.

In the `/jumpstart` directory, the administrator edits the `rules` file, deletes all of the example rules, and enters:

```
network 255.222.43.0 - eng_profile -
network 255.222.44.0 - marketing_profile -
```

Note – These are sample rules in which an administrator uses a network address to identify which systems will be installed with the `eng_profile` and `marketing_profile`, respectively. The administrator could also have chosen to use host names, memory size, or model type as the rule keyword. See “Rule Keyword and Rule Value Descriptions” on page 82 for a complete list of keywords you can use in a `rules` file.

6 Execute the `check` script.

After the `rules` and profile files are properly set up, the system administrator runs the `check` script to verify the files.

```
# cd /jumpstart
# ./check
```

When `check` finds no errors, it creates the `rules.ok` file.

7 Set up the engineering systems for installation.

After setting up the `/jumpstart` directory and appropriate files, the administrator sets up the install server to install Solaris software on the engineering systems.

The administrator first sets up the engineering systems because they are on the same subnet as the install server. On the install server, the administrator uses the `add_install_client` command:

```
# cd /export/install
# ./add_install_client -c server_1:/jumpstart host_eng1 prep
# ./add_install_client -c server_1:/jumpstart host_eng2 prep
.
.
.
```

In the `add_install_client` command,

<code>-c</code>	Specifies the server (<code>server_1</code>) and path (<code>/jumpstart</code>) to the JumpStart directory
<code>host_eng1</code>	Is the name of a system in the engineering group.
<code>host_eng2</code>	Is the name of another system in the engineering group.
<code>prep</code>	Specifies the platform group of the systems that will use <code>server_1</code> as an install server. (This is the proper platform name for PowerPC systems.)

8 Set up the marketing systems for installation.

Systems cannot boot from an install server on a different subnet, so the administrator sets up a boot server on the marketing group's subnet. On a server on the marketing subnet, the administrator inserts a Solaris CD. The administrator then uses the `setup_install_server` command to copy the boot software from the Solaris CD to the marketing server's local disk.

```
# cd /cdrom/cdrom0/s0
# ./setup_install_server -b /marketing/boot-dir prep
```

In the `setup_install_server` command,

<code>-b</code>	Specifies that <code>setup_install_server</code> will to copy the boot information from the Solaris CD to the directory named <code>/marketing/boot-dir</code> .
<code>prep</code>	Specifies the platform group of the systems that will use this boot server. (This is the proper platform name for PowerPC systems.)

Next, the administrator sets up the marketing systems to boot from the local boot server and install Solaris from the remote install server. The administrator uses the `add_install_client` command on the marketing group's boot server:

```
# cd /marketing/boot-dir
# ./add_install_client -s server_1:/export/install -c server_1:/jumpstart host_mkt1 prep
# ./add_install_client -s server_1:/export/install -c server_1:/jumpstart host_mkt2 prep
.
.
.
```

In the `add_install_client` command,

<code>-s</code>	Specifies the install server (<code>server_1</code>) and the path to the Solaris software (<code>/export/install</code>).
<code>-c</code>	Specifies the server (<code>server_1</code>) and path (<code>/jumpstart</code>) to the JumpStart directory.
<code>host_mkt1</code>	Is the name of a system in the marketing group.
<code>host_mkt2</code>	Is the name of another system in the marketing group.
<code>prep</code>	Specifies the platform group of the systems that will use this boot server. (This is the proper platform name for PowerPC systems.)

9 Boot the systems and install Solaris software.

The administrator boots the engineering systems and marketing systems by using the Solaris boot diskette.

Troubleshooting



This appendix describes problems you may encounter when installing Solaris software, and suggests possible solutions.

The following table shows common error messages and the page number where you can find causes and possible solutions.

<i>Booting "81986E3F.PREP" from server <IP_address> <server_name></i>	<i>page 150</i>
<i>Error: Unknown client "host_name"</i>	<i>page 150</i>
<i>le0: No carrier - transceiver cable problem</i>	<i>page 150</i>
<i>No response to BOOTP request. Load failed.</i>	<i>page 150</i>
<i>The file just loaded does not appear to be executable</i>	<i>page 151</i>
<i>WARNING: getfile: RPC failed: error 5 (RPC Timed out).</i>	<i>page 152</i>



Specific Installation Errors

```
Booting "81986E3F.PREP" from server <IP_address> <server_name>
Read 156kbytes
Trap Vector is 0x300 ...
```

Reason Error Occurred

You are using the wrong boot diskette.

How to Fix the Problem

Get the Solaris boot diskette that matches the version of the Solaris software that you are installing.

```
Error: Unknown client "host_name"
```

Reason Error Occurred

The *host_name* argument in the `add_install_client` command must be a host in the name service.

How to Fix the Problem

Add the host *host_name* to the NIS or NIS+ name service. (Either update the NIS maps or populate the NIS+ tables.) Try executing the `add_install_client` command again.

```
le0: No carrier - transceiver cable problem
```

Reason Error Occurred

This system is not connected to the network.

How to Fix the Problem

If this is a non-networked system, ignore this message. If this is a networked system, make sure the Ethernet cabling is attached securely.

```
No response to BOOTP request. Load failed.
```

Reason Error Occurred

The in.dhcpd daemon on the server is not running.

How to Fix the Problem

Use the Solstice Host Manager to reconfigure the client.

```
The file just loaded does not appear to be executable
```

Reason Error Occurred

This message cannot find the proper media for booting.

How to Fix the Problem

Verify that you are using the correct boot command for your system. If installing from a CD-ROM drive, make sure the Solaris CD is in the CD-ROM drive and that the CD-ROM drive is mounted.



WARNING: getfile: RPC failed: error 5 (RPC Timed out).

Reason Error Occurred

This error occurs when you have two or more servers on a network responding to an install client's boot request. The install client connects to the wrong boot server, and the installation hangs. The following specific problems may cause this error:

- There may be `/etc/bootparams` files on different servers with an entry for this install client.
- There may be multiple `/tftpboot` or `/rplboot` directory entries for this install client.
- There may be an install client entry in the `/etc/bootparams` file on a server and an entry in another `/etc/bootparams` file enabling all systems to access the profile server. Such an entry would look like this:
`* install_config=profile_server:path`

A line like this in the NIS or NIS+ `bootparams` table would also cause this error.

How to Fix the Problem

Examine the network setup:

- Ensure that servers on the network do not have multiple `/etc/bootparams` entries for the install client. If they do, remove duplicate client entries in the `/etc/bootparams` file on all install and boot servers except the one you want the install client to use.
- Ensure that servers on the network do not have multiple `/tftpboot` or `/rplboot` directory entries for the install client. If they do, remove duplicate client entries from the `/tftpboot` or `/rplboot` directories on all install and boot servers except the one you want the install client to use.
- If there's a wildcard entry in the name service `bootparams` map or table (for example, `* install_config=`), delete it and add it to the `/etc/bootparams` file on the boot server.

General Installation Problems

Problem

The system boots over the network, but from a system other than the specified install server.

How to Fix the Problem

On the name server, update the `/etc/bootparams` entry for the system being installed. The entry should conform to the following syntax:

```
install_system root=boot_server:path install=install_server:path
```

Also, ensure there is only one `bootparams` entry on the subnet for the install client.

Problem

OpenWindows is not available to diskless and dataless clients.

How to Fix the Problem

The `/usr/openwin` may be on a separate, mounted file system. This directory is not automatically shared by `admintool`.

The `/usr/openwin` file system must be manually added to the server's `/etc/dfs/dfstab` file and the file system must be shared. The entry looks like this:

```
share -F nfs /usr/openwin
```

Also, an explicit mount of the file system must be made in the client's `/etc/vfstab` file using the following:

```
server_name: /usr/openwin - /usr/openwin nfs - yes -
```

Problem

An installation using the upgrade option fails for reasons beyond your control, such as a power failure or a network connection failure. The system may be in an unbootable state.

How to Fix the Problem

1. Reboot the system from the Solaris CD or from the network.
2. Choose the upgrade option for installation.
The Solaris installation program will determine if the system has been partially upgraded and will continue the upgrade.

Time Zones



The next page shows time zones of the world by hours offset from Greenwich Meantime. This may be useful when setting a system's clock during the Solaris installation program.

Figure F-1 reflects Standard Time. If daylight saving time is in effect, add one hour.

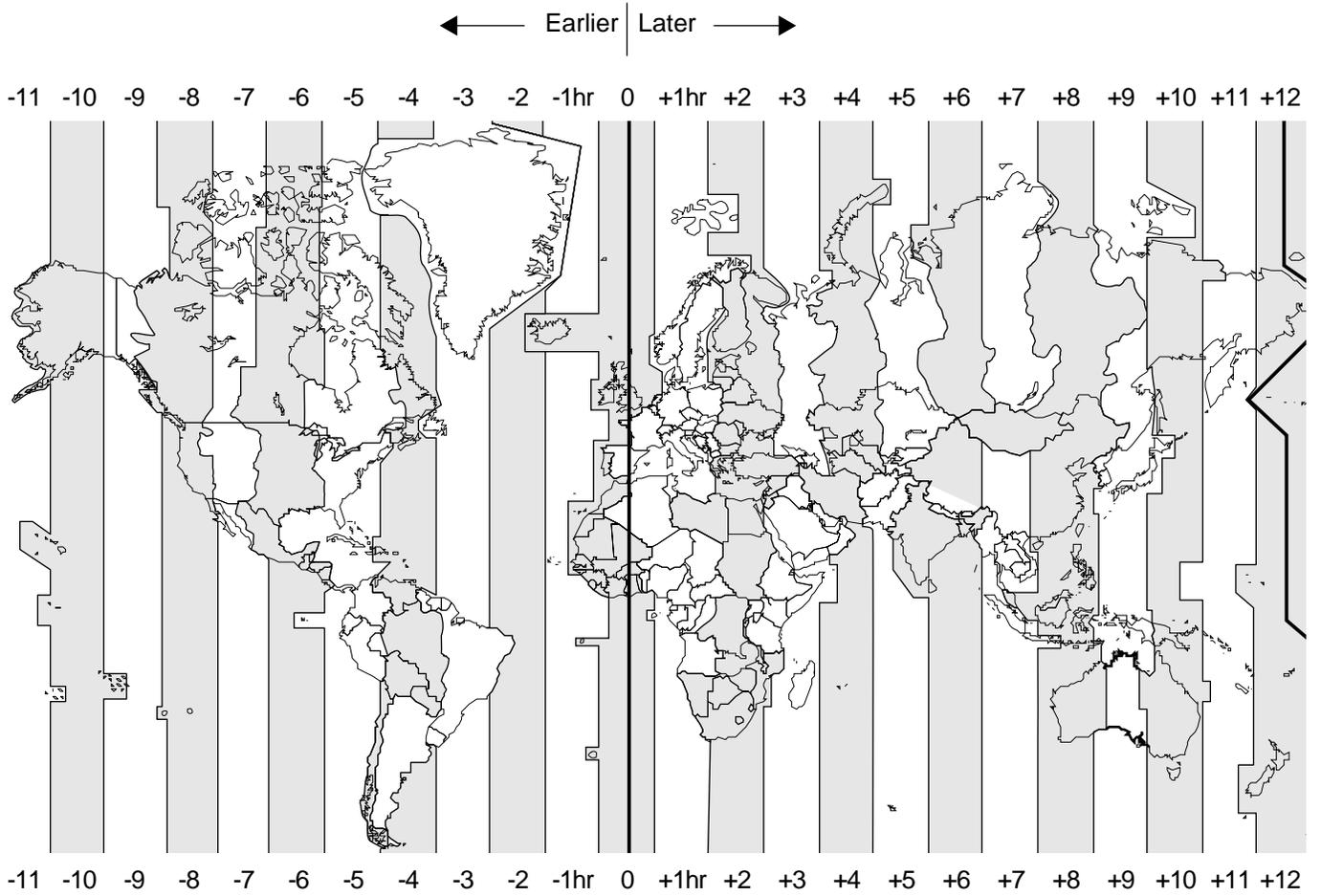


Figure F-1 Greenwich Meantime Map

Glossary

G

AutoClient system

A system type that caches all of its needed system software from an OS server. Because it contains no permanent data, an AutoClient is a field replaceable unit (FRU). It requires a small local disk for swapping and for caching its individual root (/) and /usr file systems from an OS server.

begin script

A user-defined Bourne shell script, specified within the `rules` file, that performs tasks before the Solaris software is installed on the system. Begin scripts can be used only with custom JumpStart installations.

boot server

A server that provides boot services to systems on the same subnet. A boot server is required if the install server is on a different subnet than the systems that need to install the Solaris software from it.

client

A system connected to a network.

cluster

A logical grouping of software packages. The Solaris software is divided into four main *software groups*, which are each composed of clusters and *packages*.

core

A software group that contains the minimum software required to boot and run the Solaris operating environment on a system. It includes some networking software and the drivers required to run the OpenWindows environment; it does not include the OpenWindows software.

custom JumpStart installation

A type of installation in which the Solaris software is automatically installed on a system based on a user-defined profile. You can create customized profiles for different types of users.

dataless client

A networked system that has its own disk on which it maintains its own root (/) file system and swap space. However, a dataless client relies on an OS server for its /usr system.

derived profile

A profile that is dynamically created by a begin script during a custom JumpStart installation.

developer system support

A software group that contains the End User System Support software group plus the libraries, include files, man pages, and programming tools for developing software.

disk configuration file

A file that represents a structure of a disk (for example, bytes/sector, flags, slices). Disk configuration files enable you to use `pfinstall` from a single system to test profiles on different sized disks.

diskless client

A networked system that does not have its own disk, so it relies completely on an OS server for software and file storage. Diskless clients do not have to use the Solaris installation program, because they use the software that is already installed on an OS server.

domain

A part of the Internet naming hierarchy. It represents a group of systems on a local network that share administrative files.

domain name

The identification of a group of systems on a local network. A domain name consists of a sequence of component names separated by periods (for example: tundra.mpk.ca.us). As you read a domain name from left to right, the component names identify more general (and usually remote) areas of administrative authority.

end user system support

A software group that contains the core software group plus the recommended software for an end user, including OpenWindows and DeskSet software.

entire distribution

A software group that contains the entire Solaris release.

/etc

A directory that contains critical system configuration files and maintenance commands.

/export

A file system on an OS server that is shared with other systems on a network. For example, the `/export` file system can contain the root file system and swap for diskless clients and the home directories for users on the network. Diskless clients rely on the `/export` file system on an OS server to boot and run.

fdisk partition

A logical partition of a disk drive dedicated to a particular operating system on x86 systems. During the Solaris installation program, you must set up at least one Solaris fdisk partition on an x86 system. x86 systems are designed to support up to four different operating systems on each drive; each operating system must reside on a unique fdisk partition.

file server

A server that provides the software and file storage for systems on a network.

file system

A collection of files and directories that, when set into a logical hierarchy, make up an organized, structured set of information. File systems can be mounted from your local system or a remote system.

finish script

A user-defined Bourne shell script, specified within the `rules` file, that performs tasks after the Solaris software is installed on the system, but before the system reboots. Finish scripts can be used only with custom JumpStart installations.

host name

The name by which a system is known to other systems on a network. This name must be unique among all the systems within a given domain (usually, this means within any single organization). A host name can be any combination of letters, numbers, and minus sign (-), but it cannot begin or end with a minus sign.

initial installation option

An option presented during the Solaris installation program that overwrites the disk(s) with the new version of Solaris. The initial installation option is presented for upgradable systems; however, the disk(s) that contain the old version of Solaris software (including the local modifications) will be overwritten if you choose the initial installation option.

install server

A server that provides the Solaris installation image for other systems on a network to boot and install from (also known as a *media server*). The Solaris installation image can reside on the install server's CD-ROM drive or hard disk.

interactive installation

A type of installation where you have full hands-on interaction with the Solaris installation program to install the Solaris software on a system.

IP address

Internet protocol address. A unique number that identifies a networked system so it can communicate via Internet protocols. It consists of four numbers separated by periods. Most often, each part of the IP address is a number between 0 and 225; however, the first number must be less than 224 and the last number cannot be 0.

IP addresses are logically divided into two parts: the network (similar to a telephone area code), and the system on the network (similar to a phone number).

JumpStart directory

When using a diskette for custom JumpStart installations, the JumpStart directory is the root directory on the diskette that contains all the essential custom JumpStart files. When using a server for custom JumpStart installations, the JumpStart directory is a directory on the server that contains all the essential custom JumpStart files.

locale

A specific language associated with a region or territory.

media server

See *install server*.

mount

The process of making a remote or local file system accessible by executing the `mount` command. To mount a file system, you need a mount point on the local system and the name of the file system to be mounted (for example, `/usr`).

mount point

A directory on a system where you can mount a file system that exists on the local or a remote system.

name server

A server that provides a name service to systems on a network.

name service

A distributed network database that contains key system information about all the systems on a network, so the systems can communicate with each other. With a name service, the system information can be maintained, managed, and accessed on a network-wide basis. Sun supports the following name services: NIS (formerly YP) and NIS+. Without a name service, each system has to maintain its own copy of the system information (in the local `/etc` files).

network installation

A way to install software over the network—from a system with a CD-ROM drive to a system without a CD-ROM drive. Network installations require a *name server* and an *install server*.

networked systems

A group of systems (called hosts) connected through hardware and software, so they can communicate and share information; referred to as a local area network (LAN). One or more servers are usually needed when systems are networked.

NIS	Network Information Service. A type of name service that is standard on SunOS 3.x, 4.x, and Solaris 1.x systems.
NIS+	Network Information Service, Plus. The replacement for NIS that provides automatic information updating and adds security features such as authorization and authentication. NIS+ is the standard on Solaris 2.x systems.
non-networked systems	Systems that are not connected to a network or do not rely on other systems.
/opt	A file system that contains the mount points for third-party and unbundled software.
OS server	A system that provides services to systems on a network. To serve diskless clients, an OS server must have disk space set aside for each diskless client's root file system and swap space (<code>/export/root</code> , <code>/export/swap</code>). To serve dataless clients, an OS server must provide the <code>/usr</code> file system. To serve autoclients, an OS server must provide everything except the individual root (<code>/</code>) and <code>/usr</code> file systems required for swapping and caching.
package	A functional grouping of files and directories that form a software application. The Solaris software is divided into four main <i>software groups</i> , which are each composed of <i>clusters</i> and packages.
platform group	A vendor-defined grouping of hardware platforms for the purpose of distributing specific software. Examples of valid platform names are <code>i86pc</code> , <code>sun4c</code> .
platform name	The output of the <code>uname -i</code> command. For example, the platform name for the SPARCstation IPX is <code>SUNW,Sun_4_50</code> .
profile	A text file used as a template by the custom JumpStart installation software. It defines how to install the Solaris software on a system (for example, initial installation or upgrade option, system type, disk partitioning, software group), and it is named in the <code>rules</code> file. See <i>rules file</i> .

/ (root)	The file system at the top of the hierarchical file tree on a system. The root directory contains the directories and files critical for system operation, such as the kernel, device drivers, and the programs used to start (boot) a system.
rule	A series of values that assigns one or more system attributes to a profile.
rules file	A text file used to create the <code>rules.ok</code> file. The <code>rules</code> file is a look-up table consisting of one or more rules that define matches between system attributes and profiles. See <i>profile</i> .
rules.ok file	A generated version of the <code>rules</code> file. It is required by the custom JumpStart installation software to match a system to a profile. You <i>must</i> use the <code>check</code> script to create the <code>rules.ok</code> file.
slice	An area on a disk composed of a single range of contiguous blocks. A slice is a physical subset of a disk (except for slice 2, which by convention represents the entire disk). A disk can be divided into eight slices. Before you can create a file system on a disk, you must format it into slices.
Solaris installation program	(1) A menu-driven, interactive program that enables you to set up a system and install the Solaris software on it. (2) Any part of the software that is used to install the Solaris software on a system.
software group	A logical grouping of the Solaris software (clusters and packages). During a Solaris installation, you can install one of the following software groups: core, end user system software, developer system support, or entire distribution.
standalone system	A system that has its own root (/) file system, swap space, and /usr file system, which reside on its local disk(s); it does not require boot or software services from an OS server. A standalone system can be connected to a network.
subnet	A working scheme that divides a single logical network into smaller physical networks to simplify routing.

subnet mask

A bit mask, which is 32 bits long, used to determine important network or system information from an IP address.

swap space

Disk space used for virtual memory storage when the system does not have enough system memory to handle current processes. Also known as the `/swap` or `swap` file system.

system types

The different ways a system can be set up to run the Solaris software. Valid system types are: standalone system, dataless client, and diskless client, AutoClient system, OS server. However, the only system types that are covered in this document are standalone system, dataless client, and OS server.

time zone

Any of the 24 longitudinal divisions of the earth's surface for which a standard time is kept.

upgrade option

An option presented during the Solaris installation program. The upgrade procedure merges the new version of Solaris with existing files on your disk(s), and it saves as many local modifications as possible since the last time Solaris was installed.

/usr

A file system on a standalone system or server that contains many of the standard UNIX programs. A dataless client must share (mount) `/usr` from a file server; it does not have its own `/usr` file system. Sharing the large `/usr` file system with a server rather than maintaining a local copy minimizes the overall disk space required to install and run the Solaris software on a system.

/var

A file system or directory (on standalone systems) containing system files that are likely to change or grow over the life of the system. These include system logs, `vi` files, mail files, and `uucp` files.

Volume Management

A program that provides a mechanism to administer and obtain access to the data on CD-ROMs and diskettes.

Index

Symbols

- (minus sign)
 - in begin and finish scripts, 108
 - in rules, 81
- ! (exclamation mark) rule field, 78
- # (pound sign)
 - in profiles, 62
 - in rules, 81
- && (ampersands) rule field, 78
- ... (ellipsis points) rule field, 78
- = (equals sign) in profile field, 92
- [] (brackets) rule field, 78
- \ (backslash) in rules, 81

Numerics

- 386 system, *See* x86 systems
- 486 systems, *See* x86 systems

A

- add_install_client command
 - custom JumpStart example, 146, 148
 - example, 36
 - install server setup, 35 to 37
 - JumpStart directory access, 59, 61
 - syntax, 36
- adding

- See also* creating
 - clusters when upgrading, 67
 - dataless clients, 29 to 31, 34 to 35
 - files with finish scripts, 95
 - install server configuration
 - information, 35 to 37
 - locale.org_dir table entries, 42
 - OS servers, 29 to 31, 33, 35
 - packages from software groups, 74
 - profile keywords to profiles, 62
 - rules to rules file, 77 to 78
 - standalone systems, 29 to 32, 35
- addresses, specifying, 19
- all, 69
 - location value for fdisk, 68
 - value for filesys, 71
- alternative installation programs, 108
- ampersands (&&) rule field, 78
- AND rule field, 78
- any
 - rule keyword
 - description and values, 82
 - example, 80
 - rootdisk matching, 86
 - slice value for filesys, 71
- arch rule keyword, 82
- architecture, *See* platforms
- auto size value, 71

-
- auto_install_sample directory
 - check script, 88
 - copying files to JumpStart directory, 56, 58
 - set_root_pw finish script, 96 to 98
 - AutoClient systems
 - custom JumpStart installation, 119
 - described, 7
 - identification labels, 133 to 137
 - interactive installation, 113
 - Solaris already installed on OS server, 8
 - auto-install, *See* custom JumpStart
 - auto-install, *See* custom JumpStart installation
- B**
- b option of setup_install_server command, 27, 147
 - backing up
 - before installing Solaris, 11
 - backslash (\) in rules, 81
 - begin rule field
 - described, 79
 - valid entries, 81
 - validation, 87
 - begin scripts
 - creating derived profiles with, 92 to 93
 - overview, 92
 - permissions, 92
 - rule field, 79
 - site-specific installation programs, 108
 - begin.log file, 92
 - binary compatibility package profile
 - example, 65
 - boot diskette, *See* Solaris boot diskette
 - boot server
 - creating on subnet, 25 to 27
 - described, 16
 - requirement for network installation, 16 to 17, 20
 - specifying for network installation, 32, 33
 - boot: Booting "81986E3F.PREP" from server... Trap Vector is 0x300... message, 150
 - boot: no response to BOOTP request. Load failed. message, 150
 - booting the system
 - custom JumpStart installation, 117 to 118
 - interactive installation, 111
 - bootparams file
 - enabling JumpStart directory access, 60
 - updating, 153
 - Bourne shell scripts in rule fields, 79
 - See also* begin scripts; finish scripts
 - brackets [] rule field, 78
 - buses supported, 9
- C**
- C locale name, 42
 - c option
 - add_install_client command, 36, 146, 148
 - pfinstall command, 100
 - cd command, *See* changing directories
 - CD, *See* Solaris CD
 - CD-ROM drives
 - booting from, 118
 - installation, 15 to 16
 - changing directories
 - to JumpStart directory, 88
 - to mounted CD, 58
 - to Solaris CD image on local disk, 55, 58
 - check script
 - comments and, 81
 - derived profiles and, 93
 - directory for, 88
 - rules file validation, 47, 87 to 89

- rules.ok file creation, 87
 - testing rules, 88
- Chinese locale value, 73
- client_arch profile keyword, 66
- client_root profile keyword, 66
- client_swap profile keyword, 66
- clients, dataless or diskless, *See* dataless clients; diskless clients
- clock, time zones for setting, 155 to 156
- cluster profile keyword
 - description and values, 67
 - examples, 63 to 65
- comments
 - in profiles, 62
 - in rules file, 81
- common problems, *See* troubleshooting
- concatenating multiple disk configuration files
 - PowerPC systems, 101, 104 to 107
- configuring
 - disk configuration file creation, 101 to 107
 - hands-off network installation
 - requirements, 18
 - peripheral devices
 - preconfiguring for system installation, 19
 - preconfiguring
 - default locale for network installation, 38 to 43
 - hands-off network installation requirements, 18
- copying
 - boot information, 147
 - disk configuration file to JumpStart directory, 102
 - JumpStart directory files, 95
 - JumpStart installation files from
 - CD, 56, 58
 - Solaris boot diskette, 52 to 56
 - Solaris CD to install server's local disk, 19, 20, 23
- Core System Support software
 - cluster name, 67
- cp command
 - See also* copying
- CPUs (processors)
 - rule keywords, 82
- creating
 - See also* adding
 - boot server on subnet, 25 to 27
 - disk configuration files, 101 to 107
 - /etc/locale file, 40
 - install server, 20 to 24
 - JumpStart directory, 47
 - diskette for x86 and PowerPC systems, 51
 - diskette for x86 systems, 51 to 56
 - server, 57 to 59
 - local file systems, 71 to 72
 - profiles, 47, 61 to 62
 - derived, 92 to 93
 - rules file, 47, 76 to 79
 - rules.ok file, 47, 76, 81, 87
- crontabs documentation, 122
- .cshrc file, 96
- custom JumpStart
 - compared to interactive, 12
- custom JumpStart installation, 115 to 119
 - See also* JumpStart directory; JumpStart installation
 - advantages, 46
 - AutoClient systems, 119
 - booting and installing, 115 to 119
 - booting the system, 117
 - CD-ROM drive preparation, 116
 - configuring peripheral devices, 116
 - starting OpenWindows, 119
 - dataless clients, 119
 - described, 45
 - diskless clients, 119
 - examples, 141 to 148
 - booting and installing, 148
 - check script, 145
 - eng_profile creation, 143
 - engineering systems setup, 146
 - JumpStart directory, 142 to 143

- marketing systems setup, 147 to 148
- marketing_profile
 - creation, 144
- networked, 50
- non-networked, 49
- rules file editing, 144 to 145
- site setup, 141 to 142
- standalone system, 49
- hands-off installation, 18
- JumpStart directory, 36
- optional features, 91 to 108
 - begin scripts, 92 to 93
 - finish scripts, 94 to 98
 - overview, 91
- pfinstall command, 98 to 101
- site-specific installation
 - programs, 108
- overview, 48 to 50
- preparing, 45 to 89
- profile server, 32 to 33

D

- D option of pfinstall command, 100
- d option of pfinstall command, 100
- dataless clients
 - described, 7
 - filesys profile keyword, 70
 - future Solaris releases and, 8
 - network installation server setup, 28 to 35
 - OpenWindows not available, 153
 - software requirements, 8
 - system identification labels, 133 to 137
- dd command, 52 to 55
 - See also* copying
- de locale name, 42, 73
- decimal fdisk partition type, 69
- defaults
 - derived profile name, 93
 - locale, 38 to 43
 - partitioning, 75
 - designating disks, 75

- excluding disks, 68
- SI_CONFIG_DIR variable, 95
- software group installed, 67
- deleting
 - clusters when upgrading, 67
 - fdisk partitions, 69
 - packages from software groups, 74
- derived profiles, 92 to 93
- desktop, 119
 - interactive installation, 113
- Developer system support software
 - cluster name, 67
 - profile example, 63
- dfstab file, 57, 143
- directories
 - changing
 - to JumpStart directory, 88
 - to mounted CD, 56, 58
 - to Solaris CD image on local disk, 55, 58
 - JumpStart
 - adding files, 95
 - copying disk configuration files, 102
 - copying files, 95
 - copying installation files from CD, 56, 58
 - creating, 47, 142
 - creating for x86 and PowerPC systems, 51 to 56
 - creating for x86 systems, 51 to 56
 - enabling access, 47, 59 to 61
 - file name truncation, 56
 - install server setup, 36
 - permissions, 51, 57
 - rules file example, 77
 - sharing, 57, 143
- disaster recovery when booting from CD
 - See also* troubleshooting
- disk configuration files
 - copying to JumpStart directory, 102
 - creating
 - PowerPC multiple disks, 104 to 107

- x86 systems, 101
 - described, 98, 101
- disk drives, *See* CD-ROM drives; diskette drives; hard disks
- disk interfaces, 9
- disk interfaces supported, 9
- disk space requirements, 10
- diskettes
 - copying Solaris boot diskette, 52 to 56
 - ejecting, 53 to 54
 - formatting, 53, 54
 - JumpStart directory
 - access, 59
 - creating for x86 and PowerPC systems, 51 to 56
 - creating for x86 systems, 51 to 56
 - mounting, 55
- diskless clients
 - described, 7
 - OpenWindows not available, 153
 - platforms, 66
 - Solaris already installed on OS server, 8
 - swap space, 66
 - system identification labels, 133 to 137
- disksize rule keyword
 - description and values, 82
 - rootdisk matching, 86
- display
 - interface for interactive installation, 112
 - preconfiguring, 19
 - tip line connection and interactive installation, 111
- displaying
 - mounted file systems, 19
 - platform name, 19
- distribution software group profile
 - examples, 64
- DNS (Domain Name Service), 30
- Domain Name Service (DNS), 30
- domainname rule keyword, 82
- domains
 - default locale, 40
 - rule keyword, 82
- dontuse profile keyword, 68, 75
- DOSHUGE fdisk partition, 68
- DOSOS12 fdisk partition, 68
- DOSOS16 fdisk partition, 68
- dosprimary fdisk partition type, 68

E

- ejecting diskettes, 53 to 54
- ellipsis points (...) rule field, 78
- End user system support software
 - cluster name, 67
- eng_profile example, 143
- English locale value, 42
- Entire distribution plus OEM support software
 - cluster name, 67
- Entire distribution software
 - cluster name, 67
- equals sign (=) in profile field, 92
- erasing, *See* deleting
- es locale name, 42, 73
- /etc files, updating NIS or NIS+ tables, 60
- /etc/bootparams file
 - enabling JumpStart directory access, 60
 - updating, 153
- /etc/dfs/dfstab file, 57, 143
- /etc/locale file, 40
- /etc/resolv.conf file, 30
- /etc/shadow file, 97
- /etc/vfstab file
 - mount options, 72
- Ethernet networks, *See* network installation; networked systems
- exclamation mark (!) rule field, 78
- executing, *See* starting
- existing
 - partitioning value, 75

size value for `filesys`, 71
explicit, partitioning value, 75

F

failed upgrade
 rebooting problems, 153
`fdformat` command, 53, 54
`fdisk` command, 102
`fdisk` profile keyword
 description and values, 68 to 69
 example, 64
file just loaded does not appear
 to be executable
 message, 151
file servers for dataless clients, 34
files and file systems
 begin scripts output, 92
 copying
 JumpStart directory files using
 finish scripts, 95
 JumpStart installation files from
 CD, 56, 58
 Solaris boot diskette, 52
 creating local file systems, 71 to 72
 displaying mounted file systems, 19
 finish scripts output, 94
 mounting remote file systems, 70
 names on PCFS file systems, 56
 preserving data
 existing data, 72
`filesys` profile keyword
 description and values, 70 to 72
 examples, 63 to 64
`finish` rule field
 described, 79
 valid entries, 81
 validation, 87
finish scripts, 94 to 98
 rule field, 79
`finish.log` file, 94
floppy disks, *See* diskettes
formatting diskettes, 53, 54
`fr` locale name, 42, 73

free size value for `filesys`, 71
French locale value, 42, 73

G

German locale value, 42, 73
`getfile: RPC failed: error 5:`
 RPC Timed out message, 60,
 152
Greenwich Meantime map, 156
groups, *See* software groups

H

hands-off installation
 See also custom JumpStart installation
 requirements, 18
hard disks
 copying Solaris CD to install
 server, 20, 23
 local disks in networked systems, 7
 mounting, 19, 70
 partitioning
 deleting `fdisk` partitions, 69
 designating for partitioning
 default, 75
 examples, 63 to 64
 excluding for partitioning
 default, 68
 profile keyword, 75
 rootdisk values, 71, 86 to 87
 size
 root space, 66
 rule keywords, 82, 85
 space available, 23, 27
 space required for installation, 9
 to 11
 swap space
 diskless client, 66
 maximum size, 76
 networked systems, 7
 profile examples, 63, 64
hardware
 See also peripheral devices; *specific*
 hardware

- platform names and groups by system, 139 to 140
- hardware platforms supported, 9
- hardware requirements, 9
- hexadecimal fdisk partition type, 69
- host
 - adding for network Solaris installation, 32 to 34
 - name, 36, 83
- Host Manager
 - adding hosts, 32 to 34
 - described, 19
 - name service selection, 30
- hostaddress rule keyword, 82
- hostname rule keyword
 - description and values, 83
 - example, 80

I

- initial installation
 - interactive installation
 - booting the system, 111
- install server
 - copying Solaris CD to local disk, 19, 20, 23
 - creating, 20 to 24
 - described, 16
 - network installation setup, 35 to 37
 - on subnet, 24 to 27
 - requirement for network installation, 16 to 17
 - specifying for network installation, 32 to 33
 - system types applicable, 20
- install_config command, 60
- install_type profile keyword
 - description and values, 72
 - examples, 63 to 65
 - requirement, 61, 63
 - testing profiles, 98, 99 to 100
- installed rule keyword
 - description and values, 83
 - rootdisk matching, 86

- Integrated Drive Electronics (IDE) interface, *See* IDE interface
- interactive installation, 109 to 113
 - See also* preparing for Solaris installation
 - booting the system, 110 to 111
 - CD-ROM drive preparation, 110
 - compared to custom JumpStart, 12
 - configuring peripheral devices, 110
 - tip line connection and, 111
- IP addresses
 - rule keyword, 82
 - specifying, 19
- it locale name, 42, 73
- Italian locale value, 42, 73

J

- ja locale name, 73
- Japanese locale value, 73
- JumpStart directory
 - adding files with finish scripts, 95
 - copying files
 - disk configuration files, 102
 - installation files from CD, 56, 58
 - using finish scripts, 95
 - creating, 47
 - diskette for x86 and PowerPC systems, 51 to 56
 - diskette for x86 systems, 51 to 56
 - example, 142
 - server, 57 to 59
 - install server setup, 36
 - permissions, 51, 57
 - rules file example, 77
 - sharing, 57 to 59, 143
- JumpStart installation
 - See also* custom JumpStart installation

K

- karch rule keyword, 83
- kdmconfig command, 19
- kernel architecture, *See* platforms

keyboard, preconfiguring for system installation, 19

keywords, *See* profile keywords; rule keywords

ko locale name, 73

Korean locale value, 73

L

labels, system identification, 133 to 137

language (locale)

preconfiguring using NIS name service, 38 to 40

preconfiguring using NIS+ name service, 41 to 43

Latin America locale value, 42, 73

le0: No carrier - transceiver cable problem message, 150

local disks, *See* hard disks

locale file, 40

locale profile keyword
description and values, 73
example, 65

locale, default

preconfiguring using NIS name service, 38 to 40

preconfiguring using NIS+ name service, 41 to 43

locale.org_dir table, adding entries, 42

log files

begin scripts output, 92

finish scripts output, 94

logical AND rule field, 78

M

mail accounts, 121

Makefile file, 39

man pages, 63, 65

marketing_profile example, 144

matching

derived profiles, 93

order for rules, 77, 80

rootdisk values, 86 to 87

maxfree size value for fdisk, 69

memory

minimum required, 9

rule keyword, 80, 83

setting size, 99

swap space size and, 76

virtual, 9

memory requirements for installation, 9

memsize rule keyword

description and values, 83

example, 80

messages, *See* troubleshooting

microprocessors

rule keywords, 82

minimum memory, 9

minus sign (-)

in begin and finish scripts, 108

in rules, 81

model rule keyword

description and values, 84

example, 80

monitor, *See* display

mount command, 19

mounting

begin script caution, 92

diskettes, 55

displaying mounted file systems, 19

OpenWindows file system, 153

remote file systems, 70

Solaris CD, 22, 25, 56, 58

by Solaris installation, 94

multiple disk configuration file

PowerPC systems, 104 to 107

multiple lines in rules, 81

N

name server, 16 to 17

name service

See also NIS; NIS+

hands-off network installation requirements, 18

selecting, 21

specifying, 19
names/naming
 See also name service
 derived profile names, 93
 host name, 36, 83
 PCFS file name truncation, 56
 profile names, 62
 rules file, 77, 81
 software group cluster names, 67
 system model names, 84
 system platform name
 determination, 19, 139

network installation
 See also custom JumpStart installation;
 dataless clients; diskless
 clients; interactive
 installation; JumpStart
 installation
 custom JumpStart installation
 example, 50
 described, 15
 preparing, 15 to 43

network number, 85
network rule keyword
 description and values, 85
 example, 80

networked systems, 6 to 8

NIS
 default locale preconfiguration, 38 to
 40
 selecting, 21
 updating tables corresponding to
 /etc files, 60

NIS+
 default locale preconfiguration, 41 to
 43
 selecting, 21
 updating tables corresponding to
 /etc files, 60

nistbladm command, 41 to 42

No carrier - transceiver cable
 problem message, 150

num_clients profile keyword, 73

O

opening, *See* starting

openwin command
 custom JumpStart installation, 119
 interactive installation, 113
 troubleshooting, 153

OpenWindows software
 not available to diskless or dataless
 clients, 153

starting
 custom JumpStart
 installation, 119
 interactive installation, 113

OS servers
 described, 7, 16
 network installation setup, 28 to 35
 requirement for network
 installation, 16 to 17
 Solaris already installed, 8
 system identification labels, 133 to
 137

osname rule keyword, 85

output files
 begin scripts log, 92
 finish scripts log, 94

overlap value for filesys, 72

P

-p option of check script, 88

package profile keyword
 description and values, 74
 examples, 63, 65

partitioning
 deleting fdisk partitions, 69
 examples, 63 to 64
 excluding disks, 68
 fdisk partitions, 64, 68 to 69
 profile keyword, 75

partitioning profile keyword, 75

PASSWD variable, 97

password, root, 96 to 98

paths
 check script, 88

- install server setup, 36
- PCFS file systems, 56
- peripheral devices
 - configuring
 - custom JumpStart
 - installation, 116
 - interactive installation, 110
 - preconfiguring for system
 - installation, 19
- permissions
 - begin scripts, 92
 - finish scripts, 94
 - JumpStart directory, 51, 57
- `pfinstall` command, 98 to 101
 - See also* disk configuration files
- platform groups, 139 to 140
- platforms
 - diskless client, 66
 - group determination, 139
 - install server setup, 36
 - matching system attributes and
 - profiles, 76, 77, 80
 - name determination, 19, 139
 - rule keywords, 83
 - system model names, 84
 - table of names and groups by
 - system, 139 to 140
- pound sign (#)
 - in profiles, 62
 - in rules, 81
- PowerPC
 - platform names and groups, 139 to 140
- PowerPC systems
 - hardware requirements, 9
 - JumpStart directory creation on
 - diskette, 51 to 56
- preconfiguring, *See* configuring
- preparing for Solaris installation
 - custom JumpStart installation, 45 to 89
 - importance of preparation, 5
 - network preparation, 15 to 43
 - system preparation, 5 to 12
 - work sheets, 125 to 132
- `preserve` value for `filesys`, 72
- preserving data
 - existing data during installation, 72
- printers
 - See also* peripheral devices
- problems, *See* troubleshooting
- processors
 - rule keywords, 82
- profile keywords, 66 to 75
 - adding to profiles, 62
 - case sensitivity, 62
 - `client_arch`, 66
 - `client_root`, 66
 - `client_swap`, 66
 - cluster
 - description and values, 67
 - examples, 63 to 65
 - `dontuse`
 - description and values, 68
 - `usedisk` and, 75
 - `fdisk`
 - description and values, 68
 - example, 64
 - `filesys`
 - description and values, 70 to 72
 - examples, 63 to 64
 - local file systems, 71 to 72
 - remote file systems, 70
 - `install_type`
 - description and values, 72
 - examples, 63 to 65
 - requirement, 61, 63
 - locale
 - description and values, 73
 - example, 65
 - `num_clients`, 73
 - package
 - description and values, 74
 - examples, 63, 65
 - partitioning
 - description and values, 75
 - designating disks, 75
 - examples, 63 to 64
 - excluding disks, 68

- system_type
 - description and values, 75
 - examples, 63 to 64
 - usedisk
 - description and values, 75
 - dontuse and, 68
- profile server, 32 to 33
- profiles
 - comments in, 62
 - creating, 47, 61 to 62
 - derived profiles, 92 to 93
 - described, 47, 61
 - examples, 63 to 65
 - eng_profile, 143
 - marketing_profile, 144
 - matching systems to, 76, 77, 80
 - naming, 62
 - requirements, 61, 62
 - rule field, 79
 - testing, 99 to 100
- prtconf command, 84
- prvtoc command
 - x86 disk configuration file
 - creation, 101

R

- r option of check script, 88
- RAM, *See* memory
- release of Solaris software
 - installed rule keyword, 83
 - osname rule keyword, 85
- remote file systems
 - accessing, 122
 - mounting, 70
- remote Solaris installation, *See* network installation
- removing, *See* deleting
- requirements
 - hard disk space, 9 to 11
 - network installation
 - hands-off configuration, 18
 - servers, 16 to 17, 20
 - profiles, 61, 62
- resolv.conf file, 30
- root (/) file systems
 - networked systems, 7
 - profile example, 64
 - value set by installation program, 86 to 87
- root environment (customizing), 96
- root password, 96 to 98
- rootdisk
 - defined, 86
 - slice value for fileysys, 71
 - value set by installation program, 86 to 87
- rootdisk fdisk partition, 68
- RPC failed: error 5: RPC Timed out message, 60, 152
- RPC Timed out message, 60, 152
- rule keywords, 82 to 85
 - any
 - description and values, 82
 - example, 80
 - rootdisk matching, 86
 - arch, 82
 - disksize
 - description and values, 82
 - rootdisk matching, 86
 - domainname, 82
 - hostaddress, 82
 - hostname, 80, 83
 - installed
 - description and values, 83
 - rootdisk matching, 86
 - karch, 83
 - memsize, 80, 83
 - model, 80, 84
 - network, 80, 85
 - osname, 85
 - totaldisk, 85
 - validation, 87
- rule_keyword* rule field, 78
- rule_value* rule field, 78, 81
- rules
 - derived profiles, 92 to 93
 - examples, 80

- field descriptions, 78 to 79
- matching order, 77, 80
- multiple line rules, 81
- rootdisk matching rules, 86 to 87
- syntax, 78
- testing validity, 88

rules file

- adding rules, 77 to 78
- comments, 81
- creating, 47, 76 to 79
- custom JumpStart example, 144 to 145
- described, 47, 76
- example, 77
- multiple line rules, 81
- naming, 77, 81
- syntax, 78
- testing rules, 88
- validating using check, 47, 87 to 89
 - custom JumpStart example, 145
 - derived profiles and, 93

rules.ok file

- comments and, 81
- creating, 47, 76, 81, 87
- described, 81
- matching order for rules, 77, 80

running, *See* starting

S

- s option of add_install_client command, 36, 148
- saving data during installation, *See* preserving data
- scripts
 - See also* check script; finish scripts
 - begin scripts, 92 to 93, 108
 - Bourne shell scripts in rule fields, 79
 - finish scripts, 94 to 98, 108
- security
 - root password, 96 to 98
- servers
 - See also* boot server; install server; OS servers
 - file servers for dataless clients, 34
 - JumpStart directory creation, 57 to 59
 - name server, 16 to 17
 - network installation setup
 - dataless client installation, 29 to 31, 34 to 35
 - OS server installation, 29 to 31, 33, 35
 - standalone installation, 29 to 32, 35
 - profile server, 32, 33
 - requirements for network installation, 16 to 17
 - root space, 66
- set_root_pw finish script, 96 to 98
- setup, *See* configuring; preparing for Solaris installation
- setup_install_server command
 - boot server setup, 27
 - custom JumpStart example, 147
 - described, 19
 - install server setup, 23
- shaded areas in work sheets, 125
- shadow file, 97
- share command
 - sharing JumpStart directory, 57, 143
 - sharing OpenWindows file system, 153
- shareall command, 58, 143
- sharing
 - JumpStart directory, 57, 143
 - OpenWindows file system, 153
- SI_CONFIG_DIR variable, 95
- SI_PROFILE environment variable, 93
- SI_SYS_STATE variable, 98
- site-specific installation programs, 108
- size
 - fdisk partition, 69
 - hard disk
 - root space, 66
 - rule keywords, 82, 85
 - space available, 23, 27
 - space required, 9 to 11
 - local file system, 71
 - memory, 80, 83, 99

- swap space
 - diskless client, 66
 - maximum size, 76
 - profile examples, 63, 64
 - tip window dimensions, 111
- slices
 - filesys values, 71
 - profile examples, 63 to 64
 - rule keyword, 83
- software groups
 - cluster names for profiles, 67
 - profile examples, 63 to 65
 - specifying packages, 74
 - upgrading, 67
- software. *See* Solaris software; third party software
- Solaris boot diskette, 52
- Solaris CD
 - copying to install server's local disk, 20, 23
 - displaying mounted file systems, 19
 - image on local disk, 55, 58
 - inserting into local drive
 - custom JumpStart installation, 116
 - interactive installation, 110
 - installation on systems without CD-ROM drives, 15 to 16
 - mounting, 22, 25, 56, 58
 - software contained, 2
- solaris fdisk partition type, 68, 69
- Solaris software
 - See also* Solaris CD
 - adding after installation, 121
 - buses supported, 9
 - core system support, 10
 - developer system support, 10
 - disk interfaces supported, 9
 - disk space requirements, 10
 - end user system support, 10
 - entire distribution, 10
 - groups
 - cluster names for profiles, 67
 - profile examples, 63 to 65
 - specifying packages, 74
 - upgrading, 67
 - groups of software to install, 10
 - hard disk space requirements, 9 to 11
 - memory requirements for installation, 9
 - release or version
 - installed rule keyword, 83
 - osname rule keyword, 85
- solstice program. *See* Solstice System Management Base
- Solstice System Management Base
 - Host Manager, 19, 30
 - starting, 29
- SPARC systems
 - platform names and groups, 139 to 140
- speeding network installations, 20
- square brackets [] rule field, 78
- standalone systems
 - custom JumpStart installation
 - example, 49
 - described, 7
 - identification labels, 133 to 137
 - network installation server setup, 28 to 37
 - networked and non-networked systems, 8
 - profile examples, 63 to 64
- Standard Time zones, 155 to 156
- starting
 - check script, 88 to 89
 - OpenWindows software
 - custom JumpStart installation, 119
 - interactive installation, 113
 - Solstice System Management Base, 29
- stty command, 111
- subnet
 - boot server creation on, 25 to 27
 - install server and, 24
- SunOS 4.x systems
 - platform names and groups, 139
- SUNWCall group, 67

SUNWCall software group, 10
SUNWCprog group, 67
SUNWCprog software group, 10
SUNWCreq group, 67
SUNWCreq software group, 10
SUNWCuser group, 67
SUNWCuser software group, 10
SUNWCXall group, 67
sv locale name, 42, 73
swap file systems
 diskless client swap space, 66
 memory size and, 76
 networked systems, 7
 profile examples, 63, 64
 size determination, 76
swap value for filesys, 72
Swedish locale value, 42, 73
SYS_MEMSIZE variable, 99
system accounting utilities profile
 example, 64
system architecture, *See* platforms
system boot, *See* booting the system
system identification labels, 133 to 137
system types, 6 to 8
 See also platforms
system_type profile keyword
 description and values, 75
 examples, 63 to 64

T

Taiwanese locale value, 73
testing
 profiles, 99 to 107
 validating rules files
 custom JumpStart example, 145
 derived profiles and, 93
 testing rules, 88
 using check, 47, 87 to 89
time zones, 155 to 156
timed out RPC error, 152
tip line connection and interactive
 installation, 111

totaldisk rule keyword, 85
transceiver cable problem
 message, 150
troubleshooting, 149 to 153
 booting from wrong server, 153
 general installation problems
 booting the system, 153
truncated file names, 56

U

uname command, 19, 139
Unknown client error message, 150
unnamed value for filesys, 72
unshareall command, 58, 143
upgrade installation
 See also preparing for Solaris
 installation
 custom JumpStart installation, 115 to
 119
 failed upgrade, 153
 interactive installation, 109 to 113
 booting the system, 110 to 111
 profile example, 65
 profile keywords, 65, 67, 72 to 74
usedisk profile keyword
 description and values, 75
 dontuse and, 68
/usr file systems, 7
/usr/bin/solstice program, *See*
 Solstice System Management
 Base
/usr/openwin/bin/openwin
 command
 custom JumpStart installation, 119
 interactive installation, 113
 troubleshooting, 153
/usr/sbin/install.d/pfinstall
 command, *See* pfinstall
 command

V

validating

- profiles, 99 to 107
- rules files
 - custom JumpStart example, 145
 - derived profiles and, 93
 - testing rules, 88
 - using check, 47, 87 to 89
- /var/sadm/begin.log file, 92
- /var/sadm/finish.log file, 94
- /var/yp/make command, 40
- /var/yp/Makefile file, 39
- variables
 - PASSWD, 97
 - SI_CONFIG_DIR, 95
 - SI_PROFILE, 93
 - SI_SYS_STATE, 98
 - SYS_MEMSIZE, 99
- version of Solaris software
 - installed rule keyword, 83
 - osname rule keyword, 85
- vfstab file, 72
- virtual memory requirements, 9
 - See also* swap file systems
- volcheck command, 52
- Volume Management
 - changing directory to mounted
 - CD, 56, 58
 - copying
 - JumpStart installation files from
 - CD, 56
 - Solaris boot diskette, 52 to 53
 - determining whether running, 22, 26
 - mounting diskettes, 55
 - Solaris CD file path and, 22, 26

- JumpStart directory creation on
 - diskette, 51 to 56
 - platform name and group, 139

Z

- zh locale name, 73
- zh_TW locale name, 73
- zones, time, 155 to 156

W

- work sheets, 125 to 132
- wrapping lines in rules, 81

X

- x86 systems
 - file name truncation, 56

Copyright 1996 Sun Microsystems Inc., 2550 Garcia Avenue, Mountain View, Californie 94043-1100, U.S.A. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou de sa documentation associée ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a.

Des parties de ce produit pourront être dérivées du système UNIX® licencié par Novell, Inc. et du système Berkeley 4.3 BSD licencié par l'Université de Californie. UNIX est une marque enregistrée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company Ltd. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Sun, Sun Microsystems, le logo Sun, Solaris, Solstice, SunOS, OpenWindows, ONC, NFS, et DeskSet sont des marques déposées ou enregistrées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC, utilisées sous licence, sont des marques déposées ou enregistrées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc. ELC, IPC, IPX, SLC, SPARCstation, SPARCserver, SPARCclassic, SPARCcenter, SPARCengine, Voyager, Ultra, et UltraServer sont exclusivement licenciées à Sun Microsystems, Inc. Le nom de PowerPC est une marque de International Business Machines Corporation.

Les interfaces d'utilisation graphique OPEN LOOK® et Sun™ ont été développées par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique, cette licence couvrant aussi les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

Le système X Window est un produit du X Consortium, Inc.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" SANS GARANTIE D'AUCUNE SORTE, NI EXPRESSE NI IMPLICITE, Y COMPRIS, ET SANS QUE CETTE LISTE NE SOIT LIMITATIVE, DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DES PRODUITS A RÉPONDRE A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ILS NE SOIENT PAS CONTREFAISANTS DE PRODUITS DE TIERS.