

# x86: Installing Solaris Software

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



*SunSoft*  
A Sun Microsystems, Inc. Business

© 1994 Sun Microsystems, Inc.  
2550 Garcia Avenue, Mountain View, California 94043-1100 U.S.A.

All rights reserved. This product and related documentation are protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or related documentation 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<sup>®</sup> and Berkeley 4.3 BSD systems, licensed from UNIX System Laboratories, Inc., a wholly owned subsidiary of Novell, Inc., and the University of California, respectively. Third-party font software in this product is protected by copyright and licensed from Sun's Font Suppliers.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the United States Government is subject to the restrictions set forth in DFARS 252.227-7013 (c)(1)(ii) and FAR 52.227-19.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

#### TRADEMARKS

Sun, the Sun logo, Sun Microsystems, Sun Microsystems Computer Corporation, SunSoft, the SunSoft logo, Solaris, SunOS, OpenWindows, DeskSet, ONC, ONC+, Sun-4, IPC, IPX, SLC, ELC, JumpStart, and NFS are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and certain other countries. UNIX is a registered trademark of Novell, Inc., in the United States and other countries; X/Open Company, Ltd., is the exclusive licensor of such trademark. OPEN LOOK<sup>®</sup> is a registered trademark of Novell, Inc. PostScript and Display PostScript are trademarks of Adobe Systems, Inc. DCE<sup>®</sup> is a registered trademark of DCE Group Limited. Internet<sup>™</sup> is a trademark of Internet, Inc. MS-DOS<sup>®</sup> is a registered trademark of Microsoft Corporation. Motif<sup>®</sup> is a registered trademark of Open Software Foundation, Inc. Sony<sup>®</sup> is a registered trademark of the Sony Corporation. Conner<sup>®</sup> is a registered trademark of Conner Peripherals, Inc. NEC<sup>®</sup> is a registered trademark of Nippon Electric Corporation, Inc. Powermate<sup>®</sup> is a registered trademark of NEC Corporation, Inc. All other product names mentioned herein are the trademarks of their respective owners.

All SPARC trademarks, including the SCD Compliant Logo, are trademarks or registered trademarks of SPARC International, Inc. SPARCstation, SPARCserver, SPARCengine, SPARCstorage, SPARCware, SPARCcenter, SPARCclassic, SPARCcluster, SPARCdesign, SPARC811, SPARCprinter, UltraSPARC, microSPARC, SPARCworks, and SPARCcompiler are licensed exclusively to Sun Microsystems, Inc. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK<sup>®</sup> and Sun<sup>™</sup> 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 and product of the Massachusetts Institute of Technology.

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.

THIS PUBLICATION COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THE PUBLICATION. SUN MICROSYSTEMS, INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS PUBLICATION AT ANY TIME.



# *Contents*

---

About This Book .....	vii
<b>1. About Installing Solaris.</b> .....	<b>1</b>
Definition: Installing Solaris Software. ....	1
<b>2. Preparing to Install Solaris</b> .....	<b>5</b>
Why You Should Not Ignore This Chapter. ....	5
Steps to Prepare to Install Solaris. ....	6
<b>3. Preparing to Install Solaris Over a Network</b> .....	<b>15</b>
About Installing Solaris Over a Network .....	15
Servers Required for Network Installation. ....	16
Requirements for Hands-Off Network Installation. ....	18
Files You Should Know About .....	19
Commands You Should Know About .....	20
Creating an Install Server .....	22
▼ How to Create an Install Server. ....	23
▼ How to Create a Boot Server on a Subnet .....	26

---

Setting Up Servers for Network Installation . . . . .	28
▼ How to Set Up Servers for Network Installation of a Standalone, Server, or Dataless Client . . . . .	29
▼ How to Set Up Servers for Network Installation of a Diskless Client. . . . .	39
Preconfiguring Network Information . . . . .	43
▼ How to Preconfigure Network Information. . . . .	43
Preconfiguring the Default Locale . . . . .	46
▼ How to Preconfigure Default Locale Using NIS . . . . .	46
▼ How to Preconfigure Default Locale Using NIS+ . . . . .	49
<b>4. Preparing Custom JumpStart Installations . . . . .</b>	<b>53</b>
Definition: Custom JumpStart Installation . . . . .	53
Reasons to Choose a Custom JumpStart Installation . . . . .	54
Tasks to Prepare for Custom JumpStart Installations . . . . .	55
What Happens During a Custom JumpStart Installation . . . . .	56
Creating a JumpStart Directory on a Diskette . . . . .	59
▼ How to Create a JumpStart Directory on a Diskette for x86 Systems . . . . .	59
Creating a JumpStart Directory on a Server . . . . .	65
▼ How to Create a JumpStart Directory on a Server . . . . .	65
Enabling All Systems to Access the JumpStart Directory . . . . .	68
▼ How to Enable All Systems to Access the JumpStart Directory . . . . .	68
Creating a Profile . . . . .	70
What Is a Profile . . . . .	70
Requirements for Profiles . . . . .	70

---

▼ How to Create a Profile.....	71
Profile Examples .....	72
Profile Keyword and Profile Value Descriptions.....	75
How the Size of Swap Is Determined.....	85
Creating the <code>rules</code> File .....	86
What Is the <code>rules</code> File .....	86
When Does a System Match a Rule.....	86
▼ How to Create the <code>rules</code> File .....	87
Rule Examples .....	90
Important Information About the <code>rules</code> File.....	91
Rule Keyword and Rule Value Descriptions .....	92
How the Installation Program Sets the Value of <code>rootdisk</code> .....	96
Using <code>check</code> to Validate the <code>rules</code> File.....	97
▼ How to Use <code>check</code> to Validate the <code>rules</code> File .....	98
<b>5. Using Optional Custom JumpStart Features .....</b>	<b>101</b>
Overview.....	101
Creating Begin Scripts.....	102
What Is a Begin Script.....	102
Important Information About Begin Scripts .....	102
Ideas for Begin Scripts.....	102
Creating Derived Profiles With Begin Scripts .....	102
Creating Finish Scripts .....	104
What Is a Finish Script .....	104
Important Information About Finish Scripts.....	104

---

Ideas for Finish Scripts .....	104
Adding Files With Finish Scripts .....	105
Customizing the Root Environment .....	106
Setting the System's Root Password With Finish Scripts ..	106
Using <code>pfinstall</code> to Test Profiles .....	108
Why Use <code>pfinstall</code> .....	108
Ways to Use <code>pfinstall</code> .....	108
▼ How to Use <code>pfinstall</code> to Test a Profile .....	109
<code>pfinstall</code> Examples .....	110
Using a Site-Specific Installation Program .....	111
<b>6. Booting and Installing Solaris: Interactive. ....</b>	<b>113</b>
<b>7. Booting and Installing Solaris: Custom JumpStart. ....</b>	<b>121</b>
<b>8. Where to Go After Installing Solaris. ....</b>	<b>127</b>
<b>A. Work Sheets for the Solaris Installation Program. ....</b>	<b>129</b>
<b>B. System Identification Label .....</b>	<b>137</b>
<b>C. Kernel Architectures. ....</b>	<b>141</b>
<b>D. Sample Custom JumpStart Installation .....</b>	<b>143</b>
<b>E. Troubleshooting .....</b>	<b>151</b>
<b>F. Time Zones. ....</b>	<b>161</b>
<b>G. Glossary .....</b>	<b>163</b>
Index .....	173

## *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.

This book does not include instructions for setting up system hardware or other peripherals. Setting up hardware and peripherals is described in the *x86 Device Configuration Guide*.

### **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

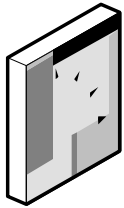
### **After Installing Solaris**

- 8 Where To Go After Installing Solaris



---

## *Related Books*



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

- *x86 Device Configuration Guide*  
Describes how to configure x86 devices before and after installing Solaris software.
- *Solaris 2.4 Open Issues and Late-Breaking News*  
Describes any late-breaking news about installing Solaris software including known problems.
- *Administration Application Reference Manual*  
Describes applications such as Administration Tool that you may use if you're setting up network installations.
- *File System Administration*  
Describes how to back up system files.



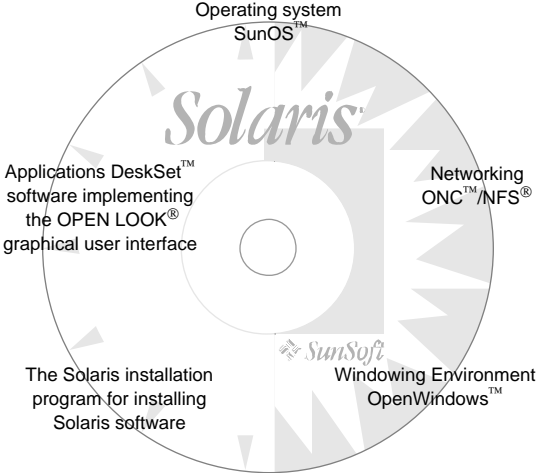
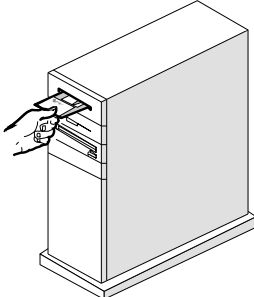
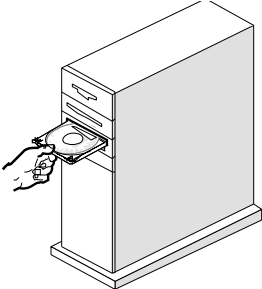
# *About Installing Solaris*

---

**1** 

## *Definition: Installing Solaris Software*

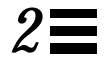
As shown on the following pages, the process of installing Solaris software means copying it from the Solaris CD to a system's local disk.

Stage	Diagram	What Happens
1	 <p>Operating system SunOS™</p> <p><i>Solaris</i></p> <p>Applications DeskSet™ software implementing the OPEN LOOK® graphical user interface</p> <p>Networking ONC™/NFS®</p> <p>The Solaris installation program for installing Solaris software</p> <p>SunSoft Windowing Environment OpenWindows™</p>	<p>You choose a Solaris CD for your hardware platform. The Solaris CD contains the SunOS operating system and other software.</p>
2		<p>You insert the Solaris boot diskette into the system's a: diskette drive.</p>
3		<p>You load the Solaris CD into the CD-ROM drive.</p>

Stage	Diagram	What Happens
4	<pre> SunSoft x.x           Multiple Device Boot  Solaris/x86 Multiple Device Boot Menu  Code  Device  Vendor  Model/Desc      Rev ===== 10   DISK    CONNER  CP330-360MB-3.5  3236 11   CD      SONY    CD-ROM CDU-8012  3.1e 12   NET     SMC/WD  I/O=280 IRQ=10  Enter the boot device code: </pre>	<p>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.)</p>
5	<pre> Select the type of installation you want to perform  1   interactive 2   custom JumpStart  Enter the number of your choice followed by the Enter key. </pre>	<p>You choose a method for installing Solaris software, then the systems boots. During the booting phase, checks are performed on the hardware. (Sample screen shown.)</p>
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>	<p>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.</p>



## 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>Make sure that the system's peripheral devices are properly configured.</i>	<i>page 6</i>
<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 9</i>
<i>If you have a standalone system, determine if you have enough disk space to install Solaris software.</i>	<i>page 10</i>
<i>Back up your system.</i>	<i>page 10</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 11</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 **Make sure that the system's peripheral devices are properly configured.**

Before installing Solaris software, you must configure peripheral devices so they do not conflict with each other, and so that Solaris software can access them. Configuring peripheral devices involves setting jumpers or running a software program under MS-DOS®. See the *x86 Device Configuration Guide* for detailed instructions.



---

**Caution** – If peripheral devices are not correctly configured before installing Solaris software, or if you are using unsupported devices, the Solaris installation program will fail.

---

### 2 **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.

### 3 **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 8, choose a system type from the following lists:

Networked systems:

- *Server*
- *Standalone system*
- *Dataless client*
- *Diskless client*

Non-networked systems:

- *Standalone system*



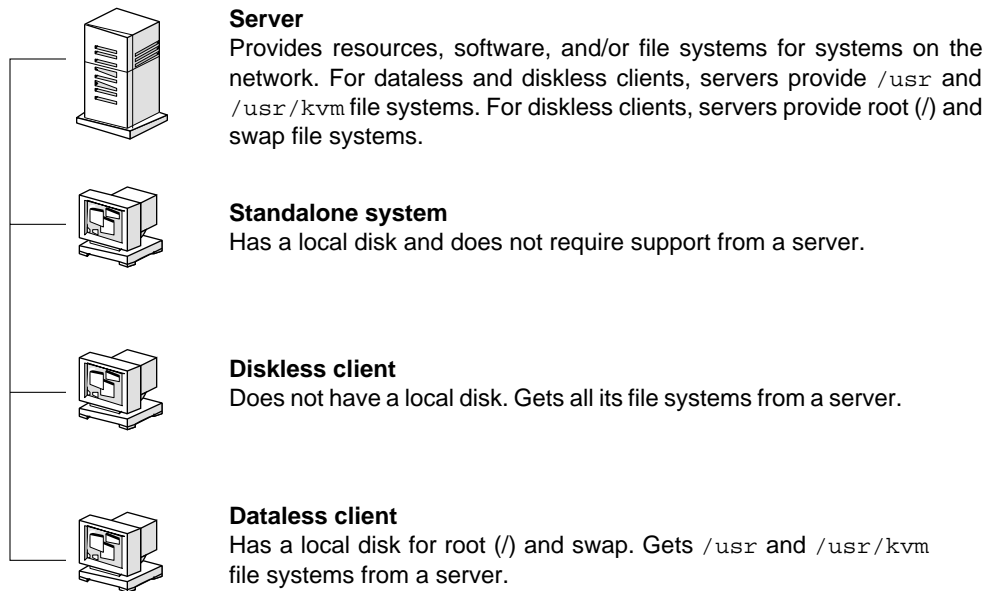
---

**Note** – If you choose a diskless client as the system type, you do not need go any further in this chapter. Go to “How to Set Up Servers for Network Installation of a Diskless Client” on page 39.

---

## Networked systems

---



## Non-networked systems

---

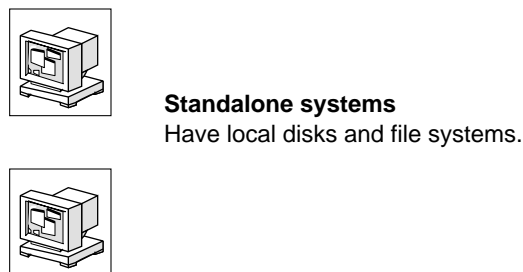


Figure 2-1 System Types

---

**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.

---

#### 4 Determine if you have required hardware.

For a detailed description of hardware requirements, see the *x86 Device Configuration Guide*.

Table 2-1 Hardware Requirements

Hardware Platform	Processor	Minimum Memory	Disk Interfaces	Buses	Device for Installing Solaris
An x86 system of the following type: <ul style="list-style-type: none"><li>• 386</li><li>• 486DX, 486DX2, or 486SX</li></ul>	For 386 systems: <ul style="list-style-type: none"><li>• 387D at 33 MHz ISA/EISA bus</li></ul> For 486D, 486X2, or 486SX systems: <ul style="list-style-type: none"><li>• 25,33,50, or 66 MHz ISA/EISA</li></ul>	16 Mbytes (including 640K base memory)	Integrated SCSI or IDE hard disk	<ul style="list-style-type: none"><li>• ISA</li><li>• EISA</li><li>• MCA</li></ul>	You must have a diskette drive, and one of the following devices for installing Solaris software: <ul style="list-style-type: none"><li>• Local CD-ROM drive</li><li>• Remote CD-ROM drive available over the network</li><li>• Remote hard disk available over the network</li></ul>

**5 If you have a standalone system, determine if you have enough disk space to install Solaris software.**

The software on the Solaris CD is bundled into four *software groups* (see Table 2-2). You must choose one of these software groups when you install Solaris software. Use Table 2-2 to determine approximately how much disk space you'll need. Keep in mind:

- The disk space shown does not account for file system overhead or local file systems required for electronic mail, printer spooling, users' personal file systems, or swap space.

You can remove software packages from these groups to reduce the disk space required.

*Table 2-2* Approximate Disk Space Required for Software Groups

<b>If You Want to Install This Software Group</b>	<b>Which Installs</b>	<b>Then You'll Need Approximately This Much Disk Space</b>
Core System Support	The minimum software required to boot and run Solaris software.	70 Mbytes
End User System Support	The core group plus the recommended software for an end user including OpenWindows and the DeskSet software.	160 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.	240 Mbytes
Entire Distribution	The entire Solaris release (everything on the CD). Compilers and debuggers are not included.	300 Mbytes

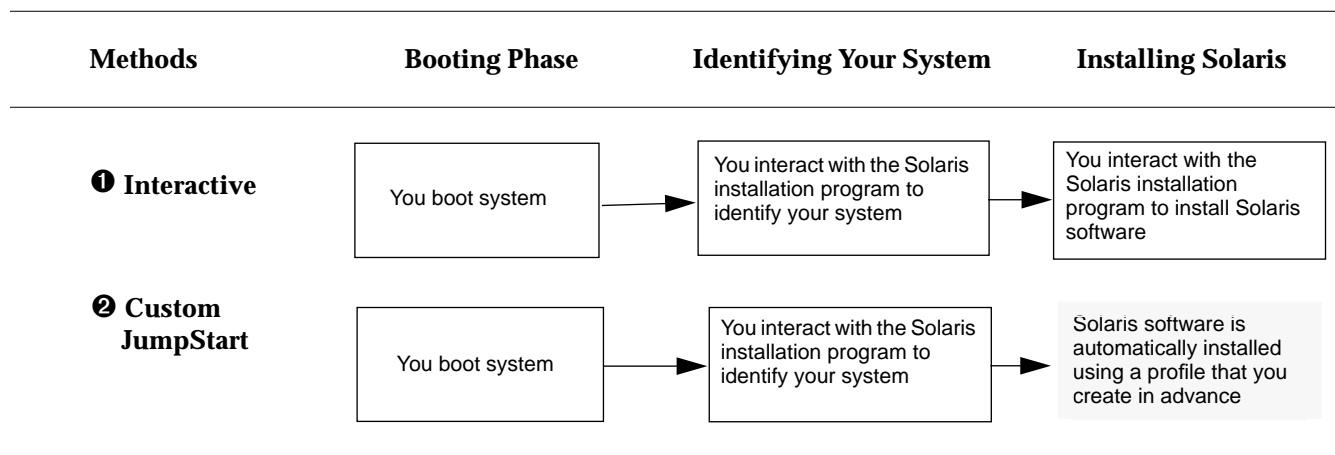
**6 Back up your system.**

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 *File System Administration* for instructions.

## 7 Choose a method for installing Solaris software.

There are two methods for installing Solaris software. One or both methods may be available to you.

- ❶ *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!
- ❷ *Custom JumpStart* (formerly called auto-install) - for the advanced user with experience in Bourne shell scripting and the `vi` editor. 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!



## 8 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. The box marked END points you to the chapter for step-by-step instructions for booting your system and installing Solaris software.

## Interactive Installations

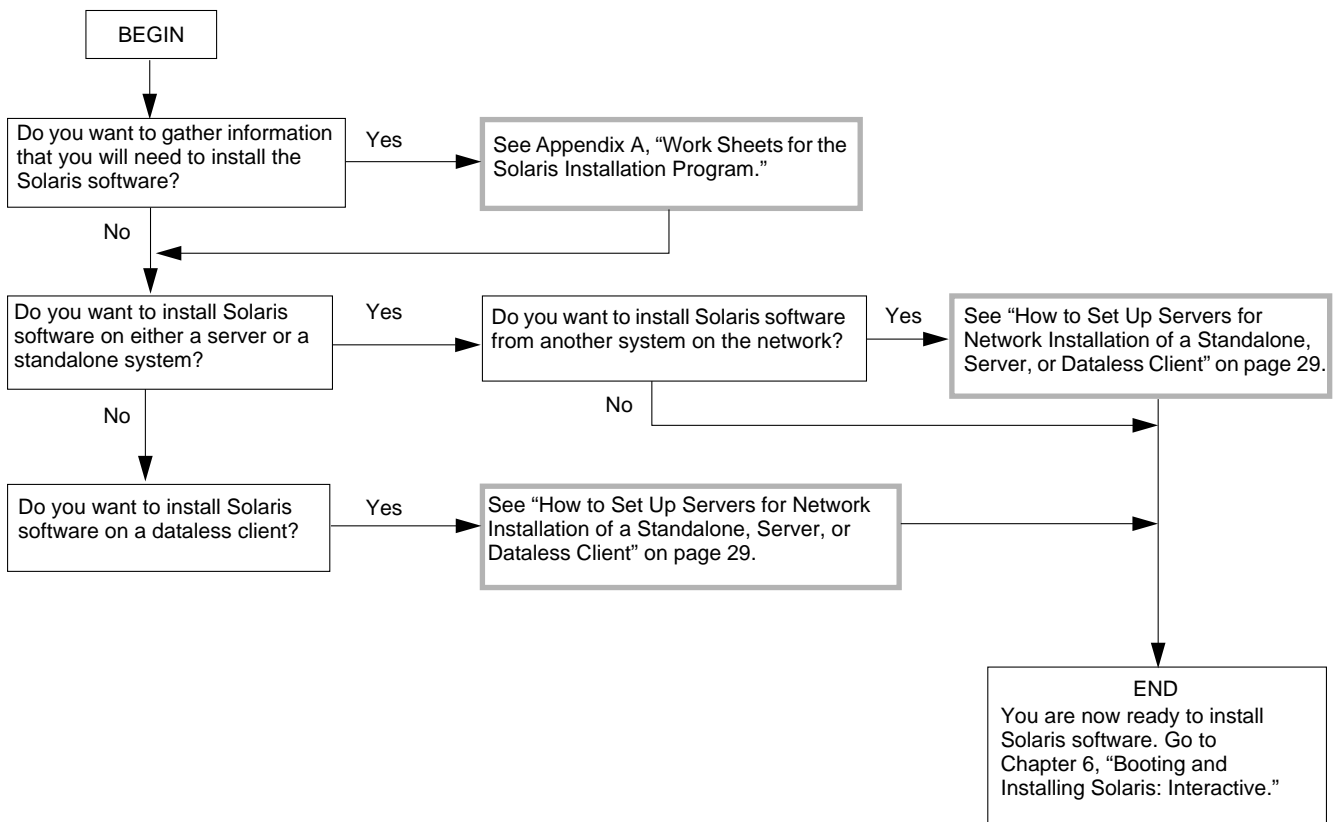


Figure 2-2 Task Map for Interactive Installations

## Custom JumpStart Installations

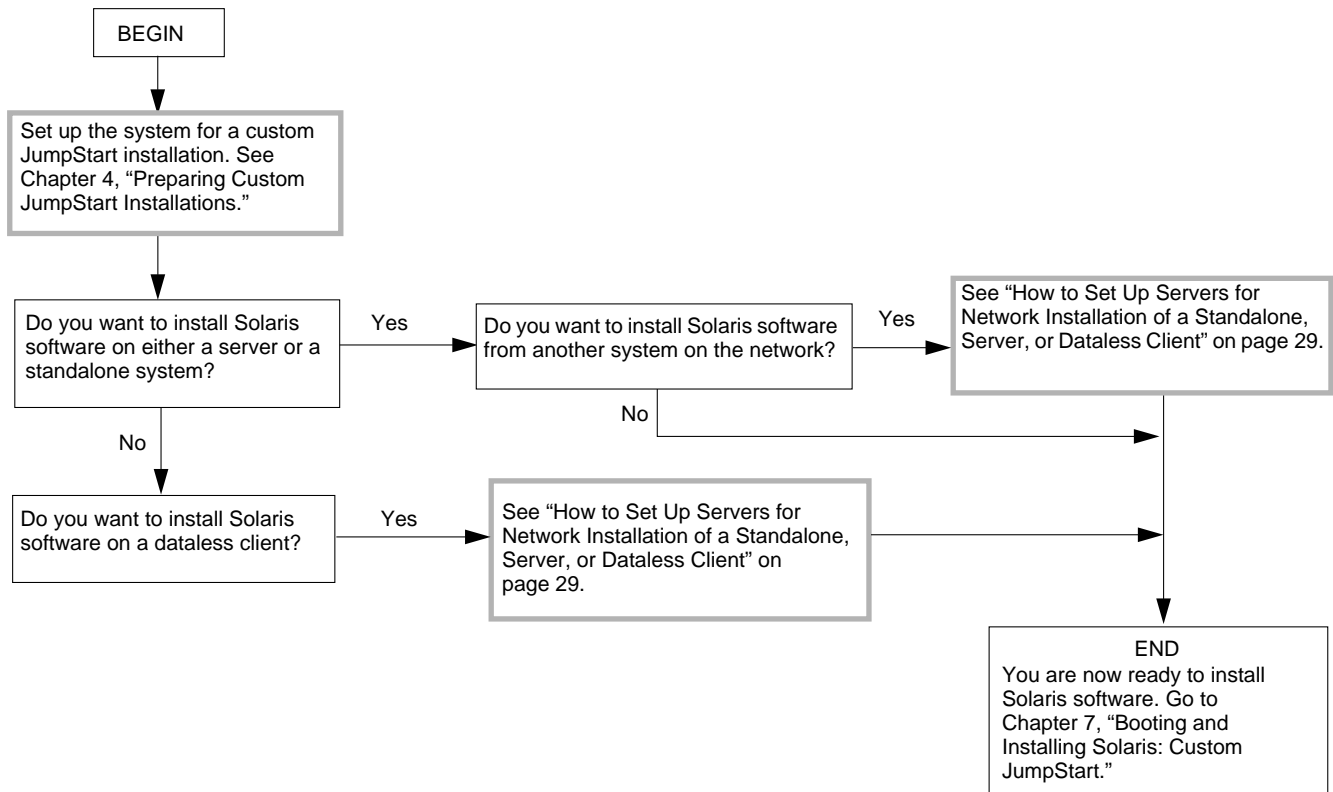


Figure 2-3 Task Map for Custom JumpStart Installations





# Preparing to Install Solaris Over a Network



<i>How to Create an Install Server</i>	<i>page 23</i>
<i>How to Create a Boot Server on a Subnet</i>	<i>page 26</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 Set Up Servers for Network Installation of a Diskless Client</i>	<i>page 39</i>
<i>How to Preconfigure Network Information</i>	<i>page 43</i>
<i>How to Preconfigure Default Locale Using NIS</i>	<i>page 46</i>
<i>How to Preconfigure Default Locale Using NIS+</i>	<i>page 49</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 or SPARC server being used for network installations. An x86 server can provide the Solaris CD image for SPARC systems, and a SPARC server can provide the Solaris CD image for x86 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.

Diskless and dataless clients also require:

- *File server* – a system that provides files for other systems.

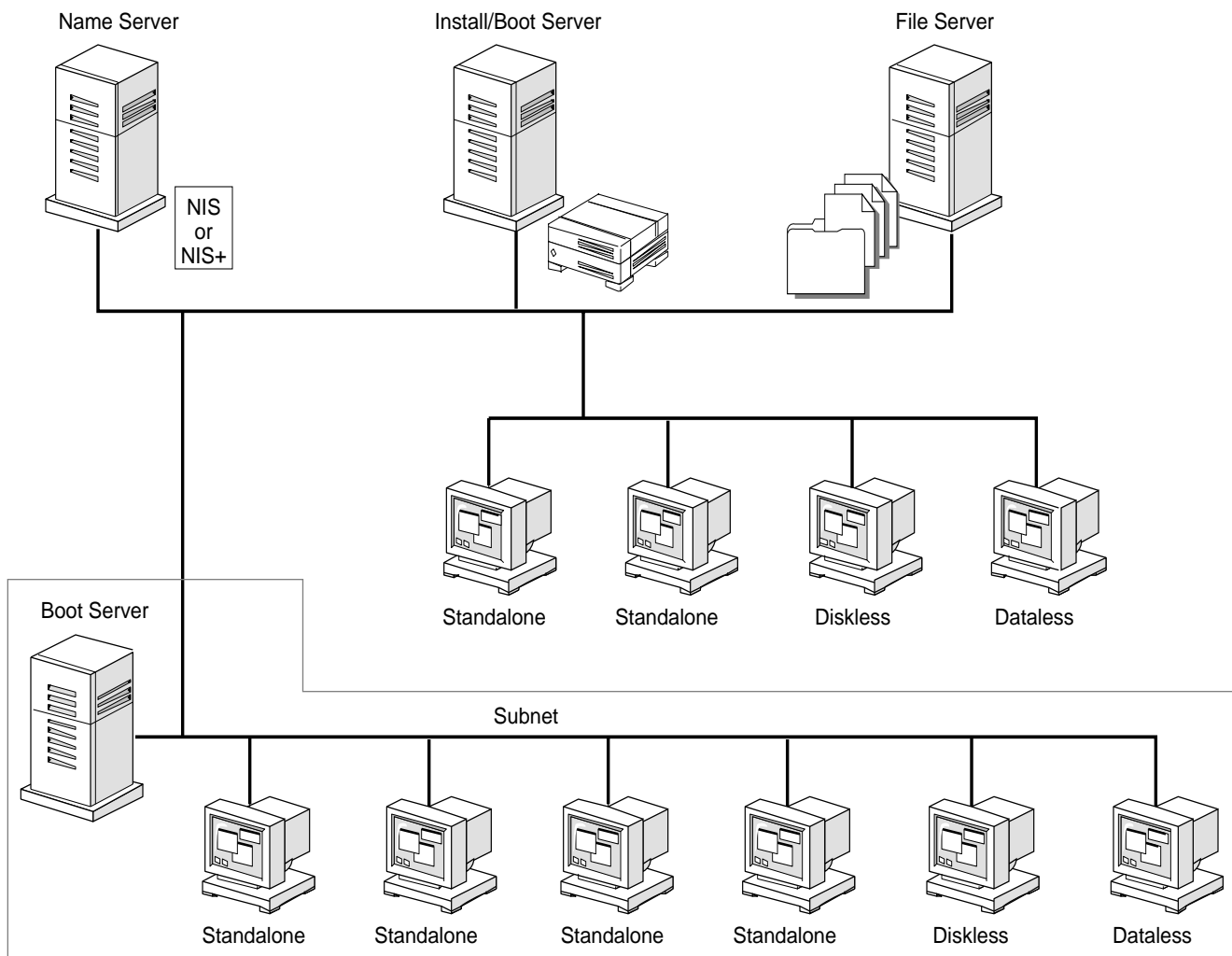


Figure 3-1 Network Installation Servers

### *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. (See “Preconfiguring Network Information” on page 43 for detailed information.) Doing so provides default values that the Solaris installation program can access, which eliminates many prompts to identify the system during an installation.

## Files You Should Know About

Table 3-1 shows files that you may need to modify or monitor to set up a network installation.

Table 3-1 Network Installation Files

Files	Description
/cdrom/cdrom0/s0	File path of the mounted SPARC Solaris CD on a Solaris 2.x system. If more than one CD-ROM drive is attached to the system, the path may vary. (This mount point is created by Volume Management software.)
/cdrom/cdrom0/s2	File path of the mounted x86 Solaris CD on a Solaris 2.x system. If more than one CD-ROM drive is attached to the system, the path may vary. (This mount point is created by Volume Management software.)
/etc/hosts /etc/ethers /etc/bootparams	Files containing information necessary for exchanging information between systems on a network.
/var/nis/ <i>host</i> /hosts.org_dir /var/nis/ <i>host</i> /ethers.org_dir /var/nis/ <i>host</i> /bootparams.org_dir	The NIS+ tables that correspond to the /etc files. These files are only pertinent if the system is using the NIS+ naming service. <i>host</i> is the host name of the NIS+ server.
/etc/nsswitch.conf	A file that you can edit to specify where (in the /etc files or in NIS+ tables) system software should look to find configuration information.
/etc/inet/inetd.conf	A configuration file that has information about daemons that may be required for network installation.
/etc/dfs/dfstab	A configuration file that specifies file systems that will be shared across the network.
/etc/mnttab	A configuration file that specifies file systems that are mounted.

## Commands You Should Know About

Table 3-2 shows programs available for use when setting up for network installation.

*Table 3-2* Network Installation Commands

<b>Program</b>	<b>Description</b>
<code>add_install_client</code>	A script that adds information to the <code>/etc</code> files on the install server to enable network installation. See the <code>add_install_client(1m)</code> man page for more information.
<code>rm_install_client</code>	A script that removes information from the <code>/etc</code> files on the install server to disable network installation. See the <code>rm_install_client(1m)</code> man page for more information.
<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 Administration Tool ( <code>admintool</code> ) program. 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>ttyhstmgr</code>	A text-based equivalent of the Administration Tool Host Manager. See the <code>ttyhstmgr(1m)</code> man page for more information.

---

*Table 3-2 Network Installation Commands (Continued)*

<b>Program</b>	<b>Description</b>
<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 -a</code>	A command for determining system architecture. 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:

- Server
- Standalone
- Dataless

This section describes how to:

- 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 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 as 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	<pre>cd /cdrom/cdrom0/s0 or cd /cdrom/cdrom0/s2</pre> <p><b>Note:</b> 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.</p>
Not running Volume Management	<pre>1) mount -F hsfs -o ro /dev/dsk/c0t6d0s0 /cdrom or mount -F hsfs -o ro /dev/dsk/c0t6d0p0 /cdrom  2) cd /cdrom</pre>

**4. Follow the instructions depending on whether or not you want to copy the Solaris CD to the install server's local disk:**

<b>If You</b>	<b>Then</b>
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,

*install\_dir\_path* 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 -a` 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.**

<b>If Install Server Is</b>	<b>Then</b>
On same subnet as the system to be installed	Go to Task Complete on page 25.
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 26. 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 25.

Task Complete

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 the appropriate section in this chapter:

- “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 architecture 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 as 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. 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 Boot Server Is	Then
Running Volume Management	<pre>cd /cdrom/cdrom0/s0 or cd /cdrom/cdrom0/s2</pre> <p><b>Note:</b> 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.</p>
Not running Volume Management	<pre>1) mount -F hsfs -o ro /dev/dsk/c0t6d0s0 /cdrom or mount -F hsfs -o ro /dev/dsk/c0t6d0p0 /cdrom  2) cd /cdrom</pre>

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

The `setup_install_server` command copies specified kernel architecture information to the local disk. Enter this command once for each type of system to be installed over the network.

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

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 kernel architecture will be copied. You can substitute any directory path.
<code><i>architecture</i></code>	Specifies the kernel architecture of the system to be installed. Sample valid values include <code>sun4e</code> , <code>sun4d</code> , <code>sun4c</code> , <code>sun4m</code> , and <code>i86pc</code> . (For a detailed list of kernel architectures, see Appendix C, “Kernel Architectures.”)

For example, the following command copies kernel architecture information necessary for booting an x86 system to be installed over the network:

```
./setup_install_server -b /export/install i86pc
```

---

**Note** - The `setup_install_server` command will indicate if you do not have enough disk space to copy the kernel architecture information. Use the `df -a` command to determine available disk space.

---

Task  
Complete

The boot server is now set up to boot systems on a subnet. To continue, go to the appropriate section in this chapter:

- “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 <sup>1</sup>	Then You Need This Server Support
Standalone System	Install Server
Server	Install Server
Dataless Client	Install Server and File Server
Diskless Client	File Server

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

---

**Note** – When you install an x86 server, you don't have the option to support SPARC clients. If you want your x86 server to serve SPARC dataless or diskless clients, you must use Software Manager (`swmtool`) and add support for the specific architecture *after* the server has been installed. To do this, select Add Client Support from the `swmtool` Edit menu.

---

Choose from the following procedures, based on the type of systems you are going to install Solaris software on:

- “How to Set Up Servers for Network Installation of a Standalone, Server, or Dataless Client” on page 29
- “How to Set Up Servers for Network Installation of a Diskless Client” on page 39

---

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

The procedure to add standalone systems and servers for installation over the network is the same. The procedure to add dataless clients involves the added step of specifying a file server. This section describes both procedures.

**Overview** – This procedure involves:

- Starting the Administration Tool Host Manager to update the name service and (if necessary) set up a file server
- Using the `add_install_client` command to add information about the standalone, server, or dataless clients to the install server configuration files

---

**Note** – If you have a character-based terminal, you can use `ttyhstmgr` to add systems for network installation. To start `ttyhstmgr`, type `ttyhstmgr` and press Return.

---

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

### 1. Follow the instructions in the following table.

---

<b>If the System to Be Installed Is</b>	<b>And the System Is</b>	<b>Then</b>
Not in the NIS or NIS+ name service	Standalone, server, or dataless client	Go to Step 2.
In the NIS or NIS+ name service	Dataless client	Go to Step 2.
In the NIS or NIS+ name service	Standalone or server	Go to Step 11.

---

- 2. 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 Administration Tool 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.

---

- 3. Start Administration Tool with the following command.**

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

- 4. After the Administration Tool 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 *Name Services Administration Guide*.

---



**5. 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	None. Host Manager will store the information you provide in <code>/etc</code> files. You will need to update NIS maps manually.
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.

**Host Manager: Select Naming Service**

**Naming Service:**

NIS+	Domain Name: <input type="text" value="net.com"/>
NIS	Domain Name: <input type="text" value="pubsnet.com"/>
None	Use /etc files on host: <input type="text" value="sinister"/>

**Show:**  All Hosts

Naming Service: NIS+

**6. On the Host Manager screen, choose Add Host from the Edit menu command.**

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

System To Be Installed Is	Then
Standalone or Server	Go to Step 8.
Dataless Client	Go to Step 10.

**8. On the Host Manager: Add Host screen, complete all fields and click on the Add button.**

The screenshot shows the 'Host Manager: Add Host' dialog box. It contains the following fields and controls:

- Client Type:** A dropdown menu set to 'standalone'. An annotation points to this field: 'Client Type - Select standalone for either a standalone or a server.'
- Host Name:** A text input field.
- IP Address:** A text input field.
- Ethernet Address:** A text input field.
- Timezone Region:** A dropdown menu set to 'United States'.
- Timezone:** A dropdown menu set to 'Mountain'.
- Remote Install:** Two radio buttons, 'Enable' and 'Disable'. The 'Disable' button is selected. An annotation points to it: 'Remote Install - Select Disable, even if you are going to install Solaris software over the network.'
- Media Server:** A dropdown menu set to 'sinister'.
- OS Release:** A dropdown menu.
- Buttons:** 'Add', 'Reset', and 'Help...' buttons.
- Naming Service:** 'NIS+' is displayed at the bottom right.

**9. Go to Step 11.**

Step 10 is required only for adding dataless clients for network installation.

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

The screenshot shows the 'Host Manager: Add Host' dialog box with the following fields and annotations:

- Client Type:**
- Host Name:**
- IP Address:**
- Ethernet Address:**
- Timezone Region:**
- Timezone:**
- File Server:** 
  - Annotation: File Server - Specify the system that you want to provide /usr and /usr/kvm file systems for the dataless client.
- OS Release:** 
  - Annotation: OS Release - Select the OS Release you want installed on the dataless client. The proper OS support must reside on the install server. This support is normally set up when a server is installed. It can also be specified with Software Manager (swmtool) after the server has been installed.
- Remote Install:**  
  - Annotation: Remote Install - Select Disable, even if you are going to install Solaris software over the network.
- Media Server:**
- Buttons: , ,
- Naming Service:** NIS+

**11. Make sure the install server has been set up.**

See “How to Create an Install Server” on page 23 for instructions.

**12. Determine your next step based on whether the standalone system or server is on the same subnet as the install server.**

---

<b>If Standalone System or Server Is</b>	<b>Then</b>
On same subnet as install server	Go to Step 13.
Not on same subnet as install server	Go to Step 16.

---

13. On the install server, change the directory to the mounted Solaris CD or the directory where the Solaris CD image has been copied.
14. Use the `add_install_client` command to add information about the system to be installed to the install server's configuration files.

```
# ./add_install_client [-c server:jumpstart_dir_path] host_name architecture
```

In this command,

<code>-c</code>	Specifies a JumpStart directory for custom JumpStart installations. This option and its arguments are required only for custom JumpStart installations.  <b>Note:</b> This option is unnecessary if you have completed the procedure “How to Enable All Systems to Access the JumpStart Directory” on page 68.
<code>server:jumpstart_dir_path</code>	<code>server</code> is the host name of the server on which the JumpStart directory is located. <code>jumpstart_dir_path</code> is the absolute path of the JumpStart directory.
<code>host_name</code>	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.
<code>architecture</code>	Is the kernel architecture of the system to be installed. Sample valid values include <code>i86pc</code> , <code>sun4e</code> , <code>sun4d</code> , <code>sun4c</code> , and <code>sun4m</code> . (For a detailed list of kernel architectures, See Appendix C, “Kernel Architectures.”)

**15. Go to Task Complete on page 38.**

Step 16 through Step 18 are required only if you are using a boot server.

**16. On the boot server, log in as root.****17. Change the directory to the boot directory.**

The boot directory is created when you set up a system as a boot server, as described in “How to Create a Boot Server on a Subnet” on page 26.

```
# cd /boot_dir_path
```

In this command,

*/boot\_dir\_path*      Is the absolute path to the directory that contains the kernel architecture information necessary for booting a system.

For example, the following command changes to a sample boot directory named `i86pc_boot_dir`.

```
cd /i86pc_boot_dir
```

**18. 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 architecture
```

In this command,

**-c** Specifies a JumpStart directory for custom JumpStart installations. This option and its arguments are required only for custom JumpStart installations.

**Note:** This option is unnecessary if you have completed the procedure “How to Enable All Systems to Access the JumpStart Directory” on page 68.

*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.

**architecture** Is the kernel architecture of the system to be installed. Sample valid values include `i86pc`, `sun4e`, `sun4d`, `sun4c`, and `sun4m`. (For a detailed list of kernel architectures, See Appendix C, “Kernel Architectures.”)

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 an `x86` system, which has an `i86pc` kernel architecture:

```
./add_install_client -s install_server1:/export/install watson i86pc
```

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 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”



## ▼ How to Set Up Servers for Network Installation of a Diskless Client

Technically speaking, diskless clients do not get Solaris software installed over the network. They just receive file services from a file server. However, diskless clients do require boot services of a server on the network. The following procedure describes how to specify:

- Where the diskless client will get its files
- Which server will boot the diskless client

**Overview** – The procedure to add diskless clients for installation involves:

- Logging in to the install server
- Using the Administration Tool Host Manager to add information about the diskless client to the install server configuration files

---

**Note** – If you have a character-based terminal, you can use the `ttyhstmgr` to add systems for network installation. To start `ttyhstmgr`, type `ttyhstmgr` and press Return.

---

Follow this procedure to add a diskless client to the install server.

- 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 Administration Tool 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 Administration Tool with the following command.**

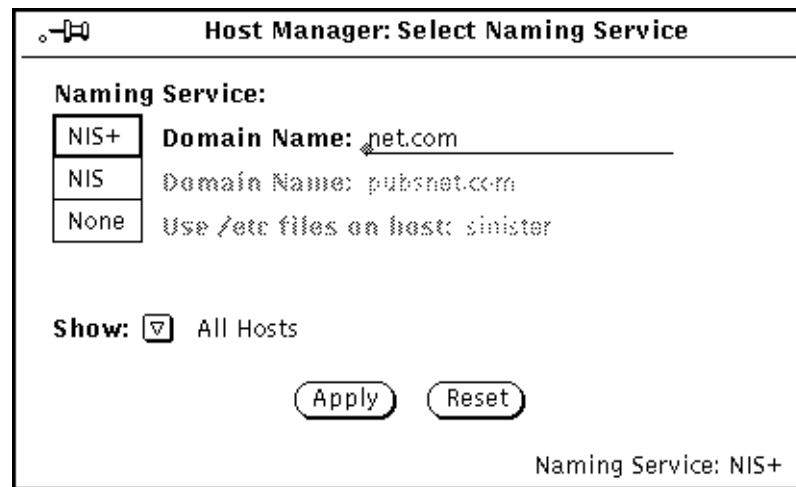
```
$ /usr/bin/admintool &
```

- 3. After the Administration Tool main window appears, click on the Host Manager icon.**

**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	None. Host Manager will store the information you provide in /etc files. You will need to update NIS maps manually.
NIS+	NIS+. Host Manager will update the NIS+ tables.
None	None. Host Manager will store the information you provide in the /etc files. This will provide enough information for systems to boot, and to install Solaris software over the network.



**5. On the Host Manager screen, choose Add Host from the Edit menu command.**

**6. On the Host Manager: Add Host screen, complete all fields and click on the Add button.**

The screenshot shows the 'Host Manager: Add Host' window with the following fields and values:

- Client Type:
- Host Name:
- IP Address:
- Ethernet Address:
- Timezone Region:
- Timezone:
- File Server:  (Annotated: File Server – Specify the server that will provide the software for the diskless client. This system you specify will also be the boot server for the diskless client.)
- OS Release:  (Annotated: OS Release – If the server supports multiple client architectures, select the correct one for the diskless client.)
- Root Path:  (Annotated: Root Path – Specify a directory where the diskless client's individual root file system will reside on the file server. By default, /export/root is specified.)
- Swap Path:  (Annotated: Swap Path – Specify a directory where the diskless client's individual swap space will reside on the file server. By default, /export/swap is specified, but you can specify any directory where there is space available.)
- Swap Size:  /   megabytes
- Terminal Type:

Buttons: , ,

Naming Service: NIS+

**7. On the diskless client, make sure that the peripheral devices are properly configured.**

Before installing Solaris software, you must configure peripheral devices so they do not conflict with each other, and so that Solaris software can access them. Configuring peripheral devices involves setting jumpers or running a software program under the MS-DOS operating system. See the *x86 Device Configuration Guide* for detailed instructions.



**Caution** – If peripheral devices are not correctly configured before installing Solaris or if you are using unsupported devices, the Solaris software installation program will fail.

---

8. On the diskless client, insert the Solaris boot diskette into the system's diskette drive.
9. Power on the system.
10. You will see a screen similar to the following. From that screen, choose to boot the Solaris operating environment from the network (NET) by typing the code number (in this case, 10) and pressing Return.

```
SunSoft Solaris x.x                               Multiple Device Boot, vsn 2.0

                Solaris/x86 Multiple Device Boot Menu

Code   Device   Vendor   Model/Desc           Rev
=====
10     NET       SMC/WD   I/O=280 IRQ=10

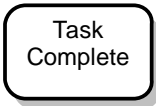
                Enter the boot device code:

30
```



**Caution** – You may need to verify that the bootparams search path is correct in the `/etc/nsswitch.conf` file. If the search path is incorrect, the diskless client will report “timeout waiting for ARP/RARP” messages and the boot will fail. See the `nsswitch.conf(4)` man page for detailed information.

---



The diskless system is now set up to boot over the network. After you turn on the diskless system, it will boot over the network and the proper file systems will be mounted from the specified file server.

---

## Preconfiguring Network Information

Preconfiguring network information provides default values for the following network-wide information:

- Date and time
- Geographic region
- Site subnet mask

Setting up network default values eliminates the need to supply this information after the system boots. After you set up this information in the name service, the Solaris installation program finds it automatically and does not prompt the user for it.

You can also preconfigure the language (called the *locale*) provided by Solaris software. See “Preconfiguring the Default Locale” on page 46 for detailed information.

---

**Note** – To preconfigure a system’s host name and Ethernet address, use the Administration Tool’s Host Manager as described in “How to Set Up Servers for Network Installation of a Standalone, Server, or Dataless Client” on page 29.

---

### ▼ How to Preconfigure Network Information

**Overview** – The procedure to preconfigure network information involves:

- Using the `vi` text editor to edit files in the `/etc` directory
- Updating the name service (NIS or NIS+) files with the information you’ve added to the files in the `/etc` directory

Follow this procedure to preconfigure the date and time, the geographic region, and the subnet mask used at your site. (You can do Step 2, Step 3, or Step 4, one or all, as long as you complete Step 5.)

1. **On the NIS or NIS+ name server, log in as root.**

**2. Edit the `/etc/hosts` file.**

In the `/etc/hosts` file, choose an entry for the server that will supply the date and time for systems being installed over the network. At the end of this entry, add the `timehost` alias. Use the following syntax:

```
IP_address host_name timehost
```

---

**Note** - The `/etc/hosts` file has read-only permissions. Using the `vi` editor, you must exit the file using `:wq!` to force the file to be saved with your changes.

---

**3. In the `/etc/timezone` file, specify the domain that will supply the geographic region for systems being installed over the network.**

To do this, add a valid time zone and the network domain name. Use the following syntax:

```
timezone domain_name
```

For example, the following entry in the `/etc/timezone` file configures all systems in the `xx.yy.zz.COM` domain into the U.S. Central time zone:

```
US/Central xx.yy.zz.COM
```

For information on valid time zone names, see the `/usr/share/lib/zoneinfo` directory.

---

**Note** - The Administration Tool Host Manager sets the geographic region for each system rather than for all systems within a network domain.

---

**4. In the `/etc/netmasks` file, specify the subnet mask to be used by systems being installed over the network.**

To do this, add the network number and the corresponding netmask. Use the following syntax:

```
network_address subnet_mask
```

For example, the following entry in the `/etc/netmasks` file sets the subnet mask for the network address 128.32.0.0:

```
128.32.0.0 255.255.255.0
```

**5. Update the NIS or NIS+ tables with the information you added to the `/etc/hosts`, `/etc/timezone`, and `/etc/netmasks` files.**

If Your Site Uses	Then
NIS	<code>/var/yp/make</code>
NIS+	<code>/usr/lib/nis/nispopulate -F -p /etc hosts timezone netmasks</code>

Task  
Complete

The date and time, geographic region, and netmask are now configured in the network. During a network installation of a system, the Solaris installation program will automatically use the default values you have configured.

## *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` 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 as 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 50 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.**

```
# /var/yp/make
```

Task  
Complete

This completes the procedure to set up a default locale for systems running the NIS name server. 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

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 *Name Services 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*.

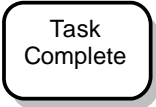
Language	Valid <i>locale</i> Values
English (Solaris default)	c
French	fr
German	de
Italian	it
Latin America	es
Swedish	sw

*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

This completes the procedure to set up a default locale for systems running the NIS+ name server. 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 Systems</i>	<i>page 59</i>
<i>How to Create a JumpStart Directory on a Server</i>	<i>page 65</i>
<i>How to Enable All Systems to Access the JumpStart Directory</i>	<i>page 68</i>
<i>How to Create a Profile</i>	<i>page 71</i>
<i>How to Create the rules File</i>	<i>page 87</i>
<i>How to Use check to Validate the rules File</i>	<i>page 98</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 or SPARC server that is being used for custom JumpStart installations. An x86 server can provide custom JumpStart files for SPARC systems, and a SPARC system can provide custom JumpStart files for x86 systems.

---

## *Reasons to Choose a Custom JumpStart Installation*

You should choose custom JumpStart installations when:

- You have to install the Solaris software on many systems.
- You have to install the Solaris software differently on 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 dataless clients with the end user software group.

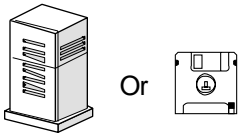
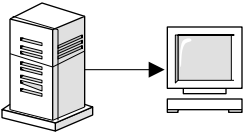
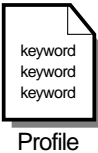
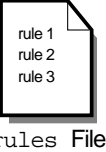
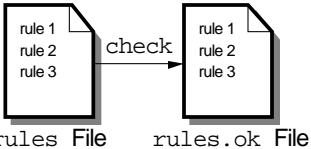
These installations would be time-consuming and tedious if you chose to perform an interactive installation on each system.



## 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 59. If you are going to use a server for custom JumpStart installations, see “Creating a JumpStart Directory on a Server” on page 65.
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 68 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 70 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 86 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 97 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.

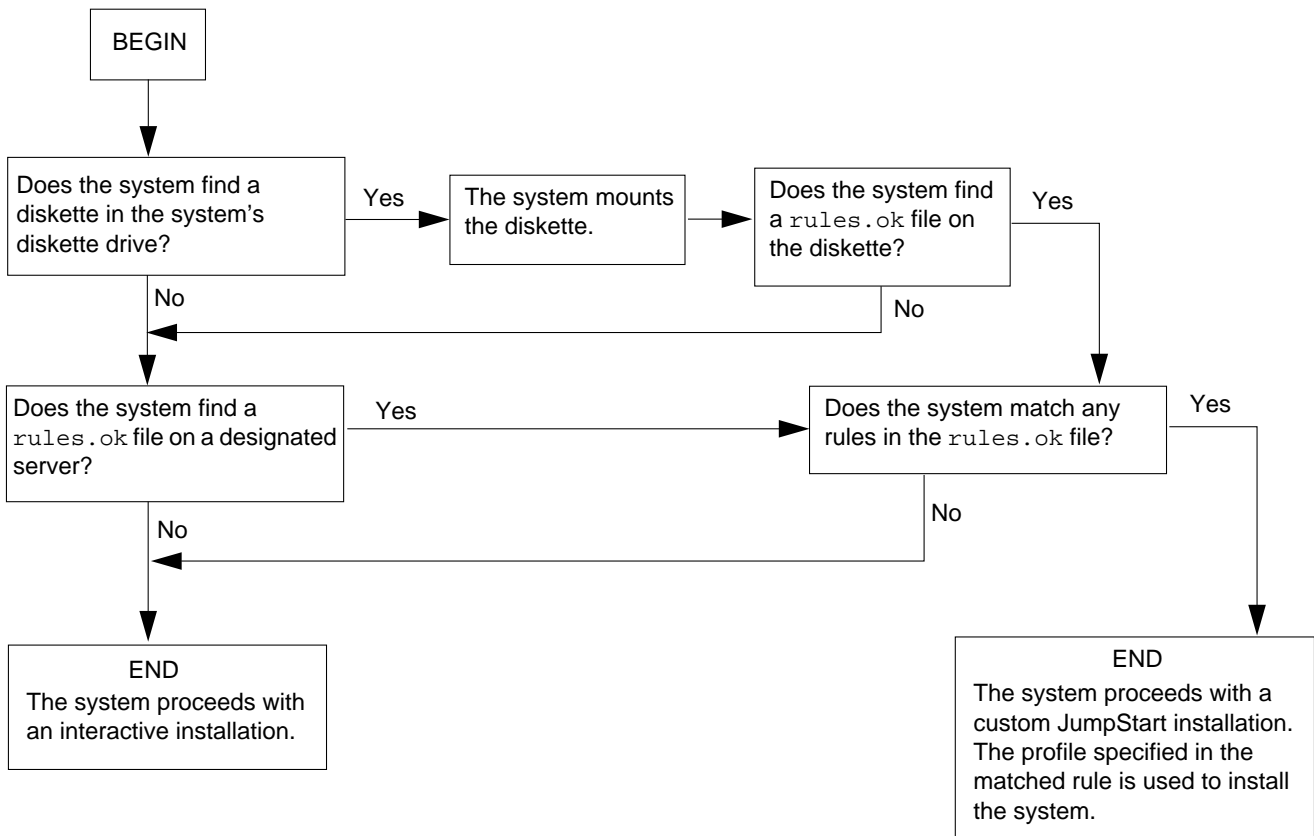


Figure 4-1 What Happens During a Custom JumpStart Installation

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.

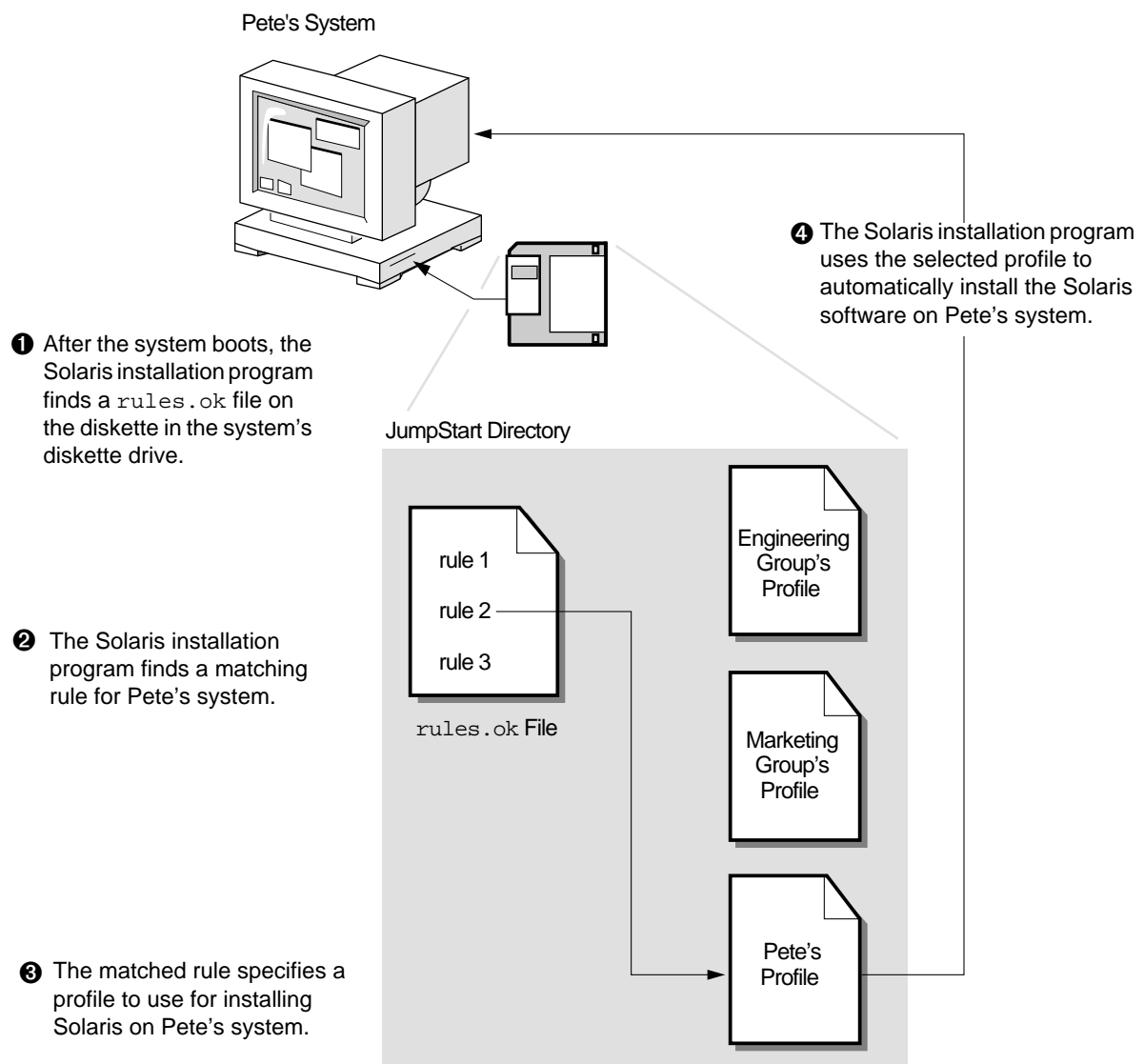


Figure 4-2 How a Custom JumpStart Installation Works: Non-Networked Example

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.

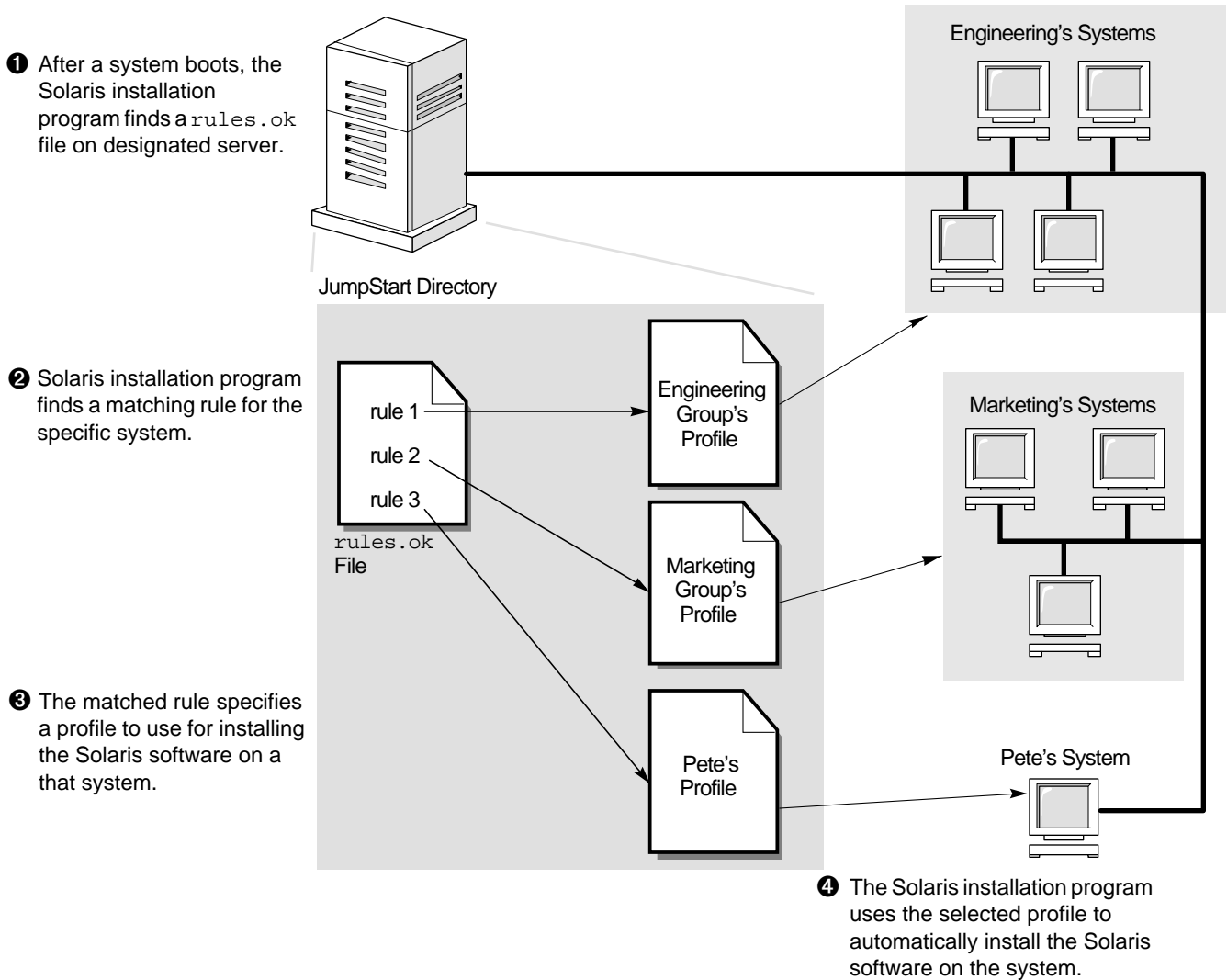


Figure 4-3 How a Custom JumpStart Installation Works: Networked Example

## 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 diskette requirements for the JumpStart directory are different for x86 and SPARC systems. This section has a procedure to only create a JumpStart directory on a diskette for x86 systems. A SPARC system is required to create a JumpStart directory on a diskette for SPARC systems.

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

**Overview** – The procedure to create a JumpStart directory on a diskette for x86 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 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:**

<b>If the System Is</b>	<b>Then</b>
Running Volume Management	Go to Step 4.  <b>Note:</b> 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 systems; you must manually eject the diskette on an x86 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 systems; you must manually eject the diskette on an x86 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:**

---

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

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

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

<b>If the System Is</b>	<b>Then</b>
Running Volume Management	<code>cd /cdrom/cdrom0/s0</code> or <code>cd /cdrom/cdrom0/s2</code>
Not running Volume Management	<ol style="list-style-type: none"> <li>1) <code>mount -F hsfs -o ro /dev/dsk/c0t6d0s0 /cdrom</code>  or  <code>mount -F hsfs -o ro /dev/dsk/c0t6d0p0 /cdrom</code></li> <li>2) <code>cd /cdrom</code></li> </ol>

**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:**

<b>If the System Is</b>	<b>Then</b>
Running Volume Management	<code>cp -r auto_install_sample/* /floppy/floppy0/.</code>
Not running Volume Management	<code>cp -r auto_install_sample/* <i>jumpstart_dir_path</i></code>  <b>Note:</b> <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 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.

---

Task  
Complete

You have completed creating a JumpStart directory on the diskette. To continue, see “How to Create a Profile” on page 71.

## 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	<ol style="list-style-type: none"> <li>1) Insert the Solaris CD into the CD-ROM drive.</li> <li>2) Go to Step 7.</li> </ol>
Solaris CD image on the local disk	<ol style="list-style-type: none"> <li>1) Change the directory to the Solaris image on the local disk. For example: <code>cd /export/install</code></li> <li>2) Go to Step 8.</li> </ol>

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

If the System Is	Then
Running Volume Management	<pre>cd /cdrom/cdrom0/s0 or cd /cdrom/cdrom0/s2</pre> <p><b>Note:</b> 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.</p>
Not running Volume Management	<ol style="list-style-type: none"> <li>1) <code>mount -F hsfs -o ro /dev/dsk/c0t6d0s0 /cdrom</code> or <code>mount -F hsfs -o ro /dev/dsk/c0t6d0p0 /cdrom</code></li> <li>2) <code>cd /cdrom</code></li> </ol>

**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 71.

## *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
```

**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 69.

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 59 or page 65, 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 72 provides some examples of profiles.
- Table 4-2 on page 75 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 108 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 87.

## 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 75 for the list of profile keywords and profile values.

	# profile keywords	profile values
	# -----	-----
❶	install_type	initial_install
❷	system_type	standalone
❸	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
❹	cluster	SUNWCprog
❺	package	SUNWman delete
	package	SUNWolman delete
	package	SUNWxwman delete
	package	SUNWoldem add
	package	SUNWxwdem add
	package	SUNWoldim add
	package	SUNWxwdim add

- ❶ This profile keyword is required in every profile.
- ❷ This profile keyword defines that the system will be installed as a standalone system.
- ❸ 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.
- ❹ The developer software group (SUNWCprog) is installed on the system.
- ❺ 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 kernel architecture than it uses. If you do not specify <code>client_arch</code>, any diskless client must have the same architecture as the server. You must specify <code>client_arch</code> once for each client architecture.</p> <p>Valid values for <i>karch_value</i> are <code>sun4</code>, <code>sun4e</code>, <code>sun4d</code>, <code>sun4c</code>, <code>sun4m</code>, or <code>i86pc</code>. (See Appendix C, “Kernel Architectures” for a detailed list of the kernel architectures of the various systems.)</p> <p><b>Restriction:</b> <code>client_arch</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><b>Restriction:</b> <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 (SPARC only).

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="1"> <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<sup>†</sup></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>. Use Software Manager on an installed system to view detailed information about clusters and their names.</p> <p><b>For Upgrade:</b></p> <ul style="list-style-type: none"> <li>• All clusters already on the system are automatically upgraded.</li> <li>• 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.</li> <li>• 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.</li> </ul>												

<sup>†</sup> Profile keywords that can be used for upgrading (SPARC only).

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><b>Restriction:</b> 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"> <li>• 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>.</li> <li>• 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).</li> <li>• If no fdisk keyword is specified in a profile, the following fdisk keyword is specified during the installation: <ul style="list-style-type: none"> <li><code>fdisk all solaris maxfree</code></li> </ul> </li> <li>• fdisk entries are processed in the order they appear in the profile.</li> </ul> <p><i>disk_name</i> - Choose where the fdisk partition will be created or deleted:</p> <ul style="list-style-type: none"> <li>• <i>cxydz</i> or <i>cydz</i> - A specific disk, for example, <i>c0t3d0</i>.</li> <li>• <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 96).</li> <li>• <i>all</i> - All the selected disks.</li> </ul> <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"> <li>• <i>solaris</i> - A Solaris fdisk partition (SUNIXOS fdisk type).</li> <li>• <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).</li> </ul>

---

† Profile keywords that can be used for upgrading (SPARC only).

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"> <li>• <i>DDD</i> - A decimal fdisk partition. <i>DDD</i> is a decimal number (valid values are 1 through 255). <b>Restriction:</b> This value can be specified only if <i>size</i> is <code>delete</code>.</li> <li>• <i>0xHH</i> - A hexadecimal fdisk partition. <i>HH</i> is a hexadecimal number (valid values are 01 through FF). <b>Restriction:</b> This value can be specified only if <i>size</i> is <code>delete</code>.</li> </ul> <p>The following table shows the decimal and hexadecimal numbers for some of the fdisk types:</p> <table border="1"> <thead> <tr> <th><i>fdisk type</i></th> <th><i>DDD</i></th> <th><i>HH</i></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"> <li>• <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>.</li> <li>• <code>all</code> - An fdisk partition is created on the entire disk (all existing fdisk partitions will be deleted). <b>Restriction:</b> This value can be specified only if <i>type</i> is <code>solaris</code>.</li> <li>• <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). <b>Note:</b> There must be at least one unused fdisk partition on the disk and the disk must have free space, or an error will occur. <b>Restriction:</b> This value can be specified only if <i>type</i> is <code>solaris</code> or <code>dosprimary</code>.</li> <li>• <code>delete</code> - All fdisk partitions of the specified <i>type</i> are deleted on the specified disk.</li> </ul>	<i>fdisk type</i>	<i>DDD</i>	<i>HH</i>	DOSOS12	1	01	PCIXOS	2	02	DOSOS16	4	04	EXTDOS	5	05	DOSHUGE	6	06	DOSDATA	86	56	OTHEROS	98	62	UNIXOS	99	63
<i>fdisk type</i>	<i>DDD</i>	<i>HH</i>																										
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 (SPARC only).



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

Profile Keyword	Profile Values and Description
<code>filesys</code> (use for mounting remote file systems)	<p><code>server:path server_address mount_pt_name [mount_options]</code></p> <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><b>For Dataless Clients:</b> 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> and <code>/usr/kvm</code> file systems. The following lines are 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 filesys sherlock:/export/exec/kvm/Solaris_2.4_sparc.sun4c/usr/kvm - /usr/kvm</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 <code>mount (1M)</code> command) that are added to the <code>/etc/vfstab</code> entry for the specified <i>mount_pt_name</i>.</p> <p><b>Note:</b> 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 (SPARC only).

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

Profile Keyword	Profile Values and Description
filesys (use for creating local file systems)	<p><i>slice size</i> [<i>file_system</i>] [<i>optional_parameters</i>]</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><i>slice</i> - Choose one of the following:</p> <ul style="list-style-type: none"> <li><code>any</code> - The Solaris installation program places the file system on any disk. <b>Restriction:</b> <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>.</li> <li><code>cwtxdysz</code> or <code>cxdsz</code> - The disk slice where the Solaris installation program places the file system, for example, <code>c0t0d0s0</code>.</li> <li><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.</li> </ul> <p><i>size</i> - Choose one of the following:</p> <ul style="list-style-type: none"> <li><code>num</code> - The size of the file system is set to <code>num</code> (in Mbytes).</li> <li><code>existing</code> - The current size of the existing file system is used. <b>Note:</b> 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>.</li> <li><code>auto</code> - The size the file system is automatically determined depending on the selected software.</li> <li><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.</li> <li><code>free</code> - The remaining unused space on the disk is used for the file system. <b>Restriction:</b> 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.</li> <li><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.</li> </ul>

---

† Profile keywords that can be used for upgrading (SPARC only).

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><code>slice size [file_system] [optional_parameters]</code></p> <p><i>file_system</i> - You can use this optional value when <i>slice</i> is specified as <code>any</code> or <code>cwtxdysz</code>. If <i>file_system</i> is not specified, <code>unnamed</code> 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"> <li><i>mount_pt_name</i> - The file system's mount point name, for example, <code>/var</code>.</li> <li><code>swap</code> - The specified <i>slice</i> is used as <code>swap</code>.</li> <li><code>overlap</code> - The specified <i>slice</i> is defined as a representation of a disk region (VTOC value is <code>V_BACKUP</code>). By default, <code>slice 2</code> is an overlap slice that is a representation of the whole disk. <b>Restriction:</b> <code>overlap</code> can be specified only when <i>size</i> is <code>existing</code>, <code>all</code>, or <code>start:size</code>.</li> <li><code>unnamed</code> - 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, <code>unnamed</code> is set by default.</li> <li><code>ignore</code> - 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.</li> </ul> <p><i>optional_parameters</i> - Choose one of the following:</p> <ul style="list-style-type: none"> <li><code>preserve</code> - The file system on the specified <i>slice</i> is preserved. <b>Restriction:</b> <code>preserve</code> can be specified only when <i>size</i> is <code>existing</code> and <i>slice</i> is <code>cwtxdysz</code>.</li> <li><i>mount_options</i> - One or more mount options (<code>-o</code> option of the <code>mount(1M)</code> command) that are added to the <code>/etc/vfstab</code> entry for the specified <i>mount_pt_name</i>.</li> </ul> <p><b>Note:</b> 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>
install_type <sup>†</sup>	<p><code>initial_install   upgrade</code></p> <p><i>install_type</i> defines whether to perform the initial installation option or upgrade option (SPARC only) on the system.</p> <p><b>Restriction:</b> <i>install_type</i> must be the first profile keyword in every profile.</p>

<sup>†</sup> Profile keywords that can be used for upgrading (SPARC only).

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

Profile Keyword	Profile Values and Description																				
locale <sup>†</sup>	<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 2.4 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>sw</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	sw	Taiwanese	zh_TW
Language	<i>locale_name</i>																				
Chinese	zh																				
French	fr																				
German	de																				
Italian	it																				
Japanese	ja																				
Korean	ko																				
Latin American	es																				
Swedish	sw																				
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><b>Restriction:</b> num_clients can be used only when system_type is specified as server.</p>																				

<sup>†</sup> Profile keywords that can be used for upgrading (SPARC only).

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

Profile Keyword	Profile Values and Description
package <sup>†</sup>	<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 Software Manager on an installed system to view detailed information about packages and their names.</p> <p><b>For Upgrade:</b></p> <ul style="list-style-type: none"><li>• All packages already on the system are automatically upgraded.</li><li>• If you specify <i>package_name</i> add, and <i>package_name</i> is not installed on the system, the package is installed.</li><li>• 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.</li><li>• 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.</li></ul>

<sup>†</sup> Profile keywords that can be used for upgrading (SPARC only).

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><b>Restriction:</b> 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><b>Restriction:</b> 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><i>disk_name</i></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 <i>disk_name</i> must be specified in the form <code>ctxydz</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><b>Restriction:</b> 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 (SPARC only).

---

## *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

<b>Physical Memory (in Mbytes)</b>	<b>Maximum Size of Swap (in Mbytes)</b>
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 kernel architecture 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 59 or page 65, 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 92). 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 90 provides some examples of rules.
- Table 4-5 on page 92 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

<b>Field</b>	<b>Description</b>
!	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 kernel architecture ( <code>kernel</code> ), host name ( <code>hostname</code> ), or memory size ( <code>memsize</code> ). It is used with the rule value to match a system with the same attribute to a profile. See Table 4-5 on page 92 for the list of rule keywords.
<i>rule_value</i>	A value that provides the specific system attribute for the corresponding rule keyword. See Table 4-5 on page 92 for the list of rule 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.
<i>begin</i>	A name of an optional Bourne shell script that can be executed before the installation begins. If no <i>begin</i> script exists, you <i>must</i> enter a minus sign (-) in this field. All <i>begin</i> scripts must reside in the JumpStart directory.  See “Creating Begin Scripts” on page 102 for detailed information on how to create <i>begin</i> scripts.

---

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

Field	Description
<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><b>Note</b> - There are optional ways to use the profile field, which are described in “Using a Site-Specific Installation Program” on page 111 and “Creating Derived Profiles With Begin Scripts” on page 102.</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 104 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 98.

## 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 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 i86pc	setup	x86_prof	done
❺	memsize 16-32 && arch sparc	-	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.
- 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. This rule also provides an example of rule wrap, which is defined on page 91.
- ❺ This rule matches if the system has 16-32 Mbytes of memory and its processor type is SPARC. 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 allow 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>arch_value</i>	Matches a system's architecture type.
	<u>Architecture</u> SPARC	If you have a system already installed, the <code>arch</code> command or the <code>uname -p</code> command reports the system's architecture type.
	<i>arch_value</i> sparc	
domainname	x86	i386
	<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>	Matches a system's disk (in Mbytes).
	<i>disk_name</i> - A disk name in the form <code>cxydz</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 96.	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.
	<i>size_range</i> - The size of the disk, which must be specified as a range of Mbytes ( <code>xx-xx</code> ).	<b>Note:</b> 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.

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 96.  <i>version</i> - A version name, such as <code>Solaris_2.2</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>karch_value</i>  Valid values are <code>sun4</code> , <code>sun4e</code> , <code>sun4d</code> , <code>sun4c</code> , <code>sun4m</code> , or <code>i86pc</code> . (See Appendix C, "Kernel Architectures" for a detailed list of the kernel architectures of the various systems.)	Matches a system's kernel architecture type.  If you have a system already installed, the <code>arch -k</code> command or the <code>uname -m</code> command reports the system's kernel architecture type.
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	<p><i>model_name</i></p> <p><u>System</u></p> <p>Sun-4/110</p> <p>Sun-4/2xx</p> <p>SPARCstation 1 (4/60)</p> <p>SPARCstation 1+ (4/65)</p> <p>SPARCstation SLC™ (4/20)</p> <p>SPARCstation IPC (4/40)</p> <p>SPARCstation ELC™ (4/25)</p> <p>SPARCstation IPX (4/50)</p> <p>SPARCstation 2 (4/75)</p> <p>Sun-4/3xx</p> <p>Sun-4/4xx</p> <p>SPARCserver™ 6xx</p> <p>SPARCstation 10</p> <p>SPARCclassic™ (4/15)</p> <p>SPARCstation LX (4/30)</p> <p>SPARCcenter™ 1000</p> <p>SPARCcenter 2000</p> <p>SPARCstation 10 SX</p> <p>SPARCstation 20</p> <p>SPARCstation 5</p> <p>SPARCstation Voyager</p> <p>x86</p>	<p><u>model_name</u></p> <p>Sun 4_100 Series</p> <p>Sun 4_200 Series</p> <p>Sun 4_60</p> <p>Sun 4_65</p> <p>Sun 4_20</p> <p>SUNW,Sun 4_40</p> <p>SUNW,SUN 4_25</p> <p>SUNW,Sun 4_50</p> <p>SUNW,SUN 4_75</p> <p>Sun SPARCsystem 300</p> <p>Sun SPARCsystem 400</p> <p>SUNW,SPARCsystem-600</p> <p>SUNW,SPARCstation-10</p> <p>SUNW,SPARCclassic</p> <p>SUNW,SPARCstation-LX</p> <p>SUNW,SPARCserver-1000</p> <p>SUNW,SPARCcenter-2000</p> <p>SUNW,SPARCstation-10,SX</p> <p>SUNW,SPARCstation-20</p> <p>SUNW,4-80</p> <p>SUNW,S240</p> <p>i86pc</p>	<p>Matches a system's model number, which is system-dependent and varies by the manufacturer. The list shown may not be complete.</p> <p>If you have a system already installed, the <code>prtconf</code> command (line 5) reports the system's model number.</p> <p><b>Note:</b> 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>
network	<p><i>network_num</i></p>	<p>Matches a system's network number, which the 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>	



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

Rule Keyword	Rule Values	Description
totaldisk	<i>size_range</i>  The value must be specified as a range of Mbytes (xx-xx).	<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><b>Example:</b> totaldisk 300-500</p> <p>The example tries to match a system with a total disk space between 300 and 500 Mbytes.</p> <p><b>Note:</b> 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 <math>207000000 / 1048576 = 197</math>. 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]
```

In this command:

`-p path`

Is the path to the Solaris 2.4 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 `check` script. You should use this option if you are using `check` on a system that is running a previous version of Solaris 2.4.

---

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

- 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

5 

*How to Use pinstall to Test a Profile*

*page 109*

## 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 or SPARC server that is being used for custom JumpStart installations. An x86 server can provide custom JumpStart files for SPARC systems and a SPARC system can provide custom JumpStart files for x86 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
mv /a/etc/shadow /a/etc/shadow.orig
nawk -F: '{
    if ( $1 == "root" )
❷     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

❸ # 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 three main 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 Change the root entry in the `/etc/shadow` file for the newly installed system using `$PASSWD` as the password field.
- ❸ Change the entry from 0 to a 1 in the state file, so that the user will not be prompted for the root password. The state file is accessed using the variable `SI_SYS_STATE`, whose value currently is `/a/etc/.sysIDtool.state`. (To avoid problems with your scripts if this value changes, always reference this file using `$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 `initial_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** – Refer to *Solaris 2.4 Open Issues and Late-Breaking News* for the procedure to create a disk configuration file for x86 systems.

---

You must run `pfinstall` on an x86 system to test profiles for x86 systems (using the x86 system's disk configuration or an x86 disk configuration file). Conversely, you must run `pfinstall` on a SPARC system to test profiles for SPARC systems (using the SPARC system's disk configuration or a SPARC 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 actually 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:

`-D`

Tells `pinstall` to use the current system's disk configuration to test the profile against. You must be root to execute `pinstall` with the `-D` 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_2.4/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
```



---

## *Using a Site-Specific Installation Program*

Through the use of begin and finish scripts, sites with unusual 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*

---

**6** 

This chapter provides procedures to boot a system and perform an interactive installation using the Solaris installation program. During the Solaris installation program, the initial installation option is available; the upgrade option is not available. The initial installation option lets you choose defaults and have software automatically laid out for you, or you can customize the software and file system layout.

The procedure in this chapter should be done on the system that is being installed.

1 Make sure that the system's peripheral devices are properly configured.

Before installing Solaris software, you must configure peripheral devices so they do not conflict with each other, and so that Solaris software can access them. Configuring peripheral devices involves setting jumpers or running a software program under MS-DOS. See the *x86 Device Configuration Guide* for detailed instructions.

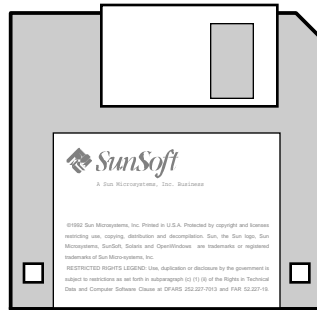


---

**Caution** – If peripheral devices are not correctly configured before you install the Solaris software, or if you are using unsupported devices, the Solaris installation program will fail.

---

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



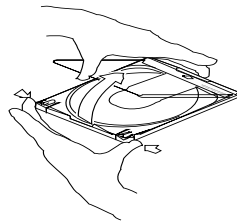
**3 If you are using the system's local CD-ROM drive to install the Solaris software on the system, prepare the CD-ROM drive.**

The instructions that follow cover the most common types of CD-ROM drives. If your CD-ROM drive is not one of the following types, see your hardware manual for instructions.

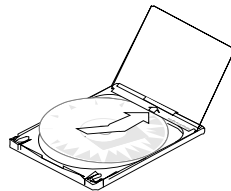
---

**CD-ROM Drive - Caddy Version**

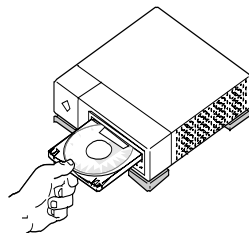
- 1** To open the caddy, pinch the corners while lifting the lid at the center lip.



- 2** Insert the Solaris CD (logo up) and close the caddy.



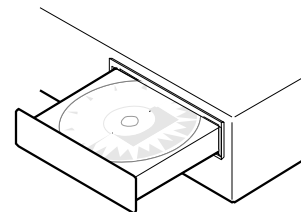
- 3** With the arrow on the caddy pointing toward the CD-ROM drive, insert the caddy.



---

**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.

- 4 Make sure you've reviewed the task map for an interactive installation (shown in Figure 2-2 on page 12).
- 5 Follow the instructions before you boot the system:

---

<b>If The System Is</b>	<b>Then</b>
Off	<ol style="list-style-type: none"><li>1) Turn on the system components in the order recommended in your hardware guide.</li><li>2) Go to Step 6.</li></ol>
On	<ol style="list-style-type: none"><li>1) If the system is running Solaris software, enter the following commands:  <pre>\$ su root # halt</pre></li><li>2) Go to Step 6.</li></ol>

---

- 6 Choose to boot from the local CD-ROM drive (CD) or from another system on the network (NET). (Sample screen shown<sup>1</sup>.)

```
SunSoft Solaris x.x                Multiple Device Boot, vsn 2.0

                Solaris/x86 Multiple Device Boot Menu

Code   Device   Vendor   Model/Desc           Rev
-----
10     DISK      CONNER   CP3300-360MB-3.5    3236
11     CD         SONY     CD-ROM CDU-8012     3.1e
12     NET       SMC/WD   I/O=280 IRQ=10

                Enter the boot device code:

30
```

1. Sony<sup>®</sup> is a registered trademark of the Sony Corporation; Conner<sup>®</sup> is a registered trademark of Conner Peripherals, Inc.

7 Choose the interactive option to install the Solaris software. (Sample screen shown.)

```
Select the type of installation you want to perform

    1  interactive
    2  custom JumpStart

Enter the number of your choice followed by the Enter key.

If you enter anything else, or if you wait for 15 seconds,
an interactive installation will be started
```



---

**Caution** - IDE disk drives do not automatically map out bad blocks like other drives supported by Solaris software. Before installing Solaris on an IDE disk, you may want to perform a surface analysis on the disk. For detailed instructions, go to Appendix E, "Troubleshooting," page 158.

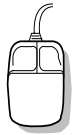

---



## 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, “Work Sheets for the Solaris Installation Program.” There is also a time zone map on page 162 to help you set your system’s time clock. After the Solaris software is successfully installed, you’re ready for Step 9.

If Your System Has a	Then the Interface Is	And the Navigation Device Is
Graphics monitor	<p style="text-align: center;"><b>Graphics-based</b></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;"><b>Mouse</b></p> 
Non-graphics monitor, or an unsupported graphics card	<p style="text-align: center;"><b>Character-based</b></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;"><b>Keyboard</b></p> 

## 9 Start the OpenWindows software to display the desktop:

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

Task  
Complete

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

## *Booting and Installing Solaris: Custom JumpStart*

---

7 

This chapter provides a procedure to boot a system and perform a custom JumpStart installation using profiles you've created to install Solaris software. After you boot (or turn on the system in some cases), the Solaris software is automatically installed.

---

**Note** – The profile you create for a custom JumpStart installation must specify the initial installation option (the upgrade option is not available).

---

The procedure in this chapter should be done on the system that is being installed.

**1 Make sure that the system's peripheral devices are properly configured.**

Before installing Solaris software, you must configure peripheral devices so they do not conflict with each other, and so that Solaris software can access them. Configuring peripheral devices involves setting jumpers or running a software program under MS-DOS. See the *x86 Device Configuration Guide* for detailed instructions.

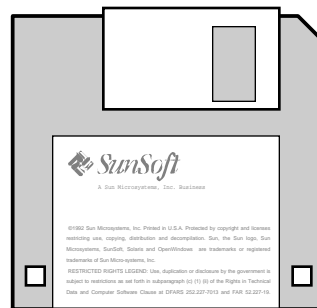


---

**Caution** – If peripheral devices are not correctly configured before you install the Solaris software, or if you are using unsupported devices, the Solaris installation program will fail.

---

**2 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.

---

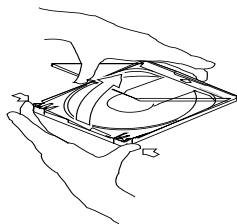
### 3 If you are using the system's local CD-ROM drive to install the Solaris software on the system, prepare the CD-ROM drive.

The instructions that follow cover the most common types of CD-ROM drives. If your CD-ROM drive is not one of the following types, see your hardware manual for instructions.

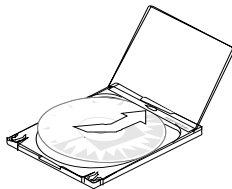
---

#### CD-ROM Drive - Caddy Version

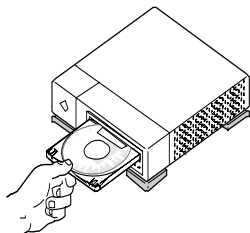
- ❶ To open the caddy, pinch the corners while lifting the lid at the center lip.



- ❷ Insert the Solaris CD (logo up) and close the caddy.



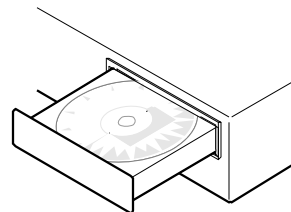
- ❸ With the arrow on the caddy pointing toward the CD-ROM drive, insert the caddy.




---

#### CD-ROM Drive - Tray Version

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



- ❷ Insert the Solaris CD (logo up) into the tray.

- ❸ Push the tray back into CD-ROM drive.

4 Make sure you have reviewed the task map for a custom JumpStart installation (shown in Figure 2-3 on page 13).

5 Follow the instructions before you boot the system:

---

<b>If The System Is</b>	<b>Then</b>
Off	<ol style="list-style-type: none"><li>1) Turn on the system components in the order recommended in the hardware guide.</li><li>2) Go to Step 6.</li></ol>
On	<ol style="list-style-type: none"><li>1) If the system is running Solaris, enter the following commands:  <pre>\$ su root # halt</pre></li><li>2) Go to Step 6.</li></ol>

---

6 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.

7 Choose to boot from the local CD-ROM drive (CD) or from another system on the network (NET). (Sample screen shown<sup>1</sup>.)

```
SunSoft Solaris x.x                Multiple Device Boot, vsn 2.0

                Solaris/x86 Multiple Device Boot Menu

Code   Device   Vendor   Model/Desc           Rev
-----
10     DISK      CONNER   CP3300-360MB-3.5    3236
11     CD         SONY     CD-ROM CDU-8012     3.1e
12     NET       SMC/WD   I/O=280 IRQ=10

                Enter the boot device code:

30
```



**Caution** - IDE disk drives do not automatically map out bad blocks like other drives supported by Solaris software. Before installing Solaris on an IDE disk, you may want to perform a surface analysis on the disk. For detailed instructions, go to Appendix E, "Troubleshooting," page 158.

1. Sony<sup>®</sup> is a registered trademark of the Sony Corporation; Conner<sup>®</sup> is a registered trademark of Conner Peripherals, Inc.

**8 Choose the custom JumpStart option to install the Solaris software. (Sample screen shown.)**

```
Select the type of installation you want to perform

    1  interactive
    2  custom JumpStart

Enter the number of your choice followed by the Enter key.

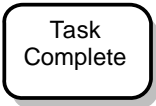
If you enter anything else, or if you wait for 15 seconds,
an interactive installation will be started
```

**9 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.

**10 Start the OpenWindows software to display the desktop:**

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



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 2.4 documentation, refer to the *Solaris 2.4 Introduction*.

Table 8-1 Where to Go After Installing Solaris

Information Needed	Manual Title
Configuring additional devices on your system	<i>x86 Device Configuration Guide</i>
Adding and removing Solaris, third-party, or unbundled software	<i>Software and AnswerBook Packages Administration Guide</i>
Setting up mail accounts	<i>User Accounts, Printers, and Mail Administration</i>
Setting up printers	<i>User Accounts, Printers, and Mail Administration</i>
Setting up user accounts	<i>User Accounts, Printers, and Mail Administration</i>
Performing common administration tasks	<i>Common Administration Tasks</i>
Using system administration tools	<i>Administration Application Reference Manual</i>

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

<b>Information Needed</b>	<b>Manual Title</b>
Increasing your system's performance	<i>Security, Performance, and Accounting Administration</i>
Adding and Maintaining Peripherals	<i>Peripherals Administration</i>
Setting up system security	<i>Security, Performance, and Accounting Administration</i>
Administering file systems	<i>File System Administration</i>
Installing AnswerBooks	<i>Software and AnswerBook Packages Administration Guide</i>

**Note:** The End User AnswerBook is the only AnswerBook installed with the Solaris software. The End User AnswerBook is part of the entire distribution software group.

# *Work Sheets for the Solaris Installation Program*

---



## *Purpose of the Work Sheets*

The work sheets in this appendix list information that may need to be supplied during the Solaris installation program. There is a work sheet for the initial installation option and the upgrade option.

## *Uses of the Work Sheets*

The work sheets can be used before installing Solaris software, or to record your responses as 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 work sheet 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 work sheet to gather information before installing Solaris software, you can ignore shaded areas if you are installing a system as a standalone, non-networked system.

---

## ≡ A

---

### *Work Sheet for the Initial Installation Option*

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

---

<b>Information You May Need When Installing Solaris</b>	<b>Description/Example</b>	<b>Work Space</b>
Boot device	Boot from local CD-ROM or network?	
Method for installing Solaris software	Interactive or custom JumpStart?	
Graphics card	Type of graphics card in your system. Example: Graphics Ultra Pro	
Screen size	Size of the screen on your monitor from the following list: - 14 inch - 16 inch - 19 inch	
Keyboard language	Language your keyboard supports. Example: UK english	
Pointing device	Type of mouse. Example: Logitech MouseMan, serial, port tty00, 3-button	
Graphics adaptor/monitor combination	Screen resolution. Example: 1024 X 768, 1280 X 1024	

---

---

<b>Information You May Need When Installing Solaris</b>	<b>Description/Example</b>	<b>Work Space</b>
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 name service not yet established? Example: Other, DCE	
Domain name	Domain in which the system resides. Example: chandy.West.Arp.COM	
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	

---

## ≡ A

---

Information You May Need When Installing Solaris	Description/Example	Work Space
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, server, or dataless client?	
Diskless clients	Number of diskless clients the server will support. Example: 5 (default)  Number of megabytes of swap space to assign each diskless client. Example: 24 Mbytes (default)	
Diskless client architectures	All the kernel architectures for diskless clients that are <i>different</i> from the server's architecture (see Appendix C, "Kernel Architectures").	

---

---

Information You May Need When Installing Solaris	Description/Example	Work Space
Dataless clients	Host names and IP addresses of the server(s) where dataless clients will mount their <code>/usr</code> and <code>/usr/kvm</code> file systems. Example: <code>morton</code> , <code>112.221.2.1</code> , <code>/export/home</code>	
Languages	Language(s) to select for displaying the user interface after installing Solaris software. Example: French, Spanish	
Software	Which software group to install? <ul style="list-style-type: none"><li>• Core</li><li>• End User System Support</li><li>• Developer System Support</li><li>• Entire Distribution</li><li>• Entire Distribution Plus OEM Support</li></ul> Should the software group be customized?  <b>Note:</b> Removing or adding software often creates software dependencies; system administration knowledge is often required to fix dependencies.	
Disk(s) to install Solaris software on	Disk(s) for installing Solaris. Example: <code>c0t0d0</code> , <code>c0t3d0</code> , <code>c0t4d1</code>	

---

## ≡ A

---

Information You May Need When Installing Solaris	Description/Example	Work Space
Create Solaris partition	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.	
Auto-layout file systems?	Should file systems be laid out on disks automatically or manually?  <b>Note:</b> Manual layout requires advanced system administration skills.	
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	



---

Information You May Need When Installing Solaris	Description/Example	Work Space
File system and disk layout	Accept the layout of file systems as created by auto-layout or manual layout, or customize?  <b>Note:</b> Customizing requires advanced system administration skills.	
Mount remote file systems?	Mount remote file systems from a server?	
Mount remote file systems	Provide remote file system information: <ul style="list-style-type: none"><li>• Local mount point</li><li>• Server's host name</li><li>• Server's IP address</li><li>• File system path</li></ul> 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	

---

≡ A

---

## *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.

<b>Host name:</b> roylat	
<b>System type</b>	
<input checked="" type="checkbox"/>	Standalone
<input type="checkbox"/>	Server
<input type="checkbox"/>	Diskless client
<input type="checkbox"/>	Dataless client
<b>Boot device</b>	network
<b>Graphics card</b>	Graphics Ultra Pro
<b>Screen size</b>	14"
<b>Pointing device</b>	Logitech MouseMan, serial, 3-button
<b>Screen resolution</b>	1024 X 768
<b>Networked?</b>	Yes
<b>Primary network interface</b>	smc0
<b>IP address</b>	129.221.2.1
<b>Name service</b>	NIS+
<b>Domain name</b>	hawthorne.West.Arp.COM
<b>System part of a subnet?</b>	Yes
<b>Netmask</b>	225.225.225.225
<b>Name server and IP address</b>	hudson, 125.222.2.2
<b>Ethernet address</b>	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.

<b>Host name:</b>	
<b>System type</b>	
	<input type="checkbox"/> Standalone
	<input type="checkbox"/> Server
	<input type="checkbox"/> Diskless client
	<input type="checkbox"/> Dataless client
<b>Boot device</b>	_____
<b>Graphics card</b>	_____
<b>Screen size</b>	_____
<b>Pointing device</b>	_____
<b>Screen resolution</b>	_____
<b>Networked?</b>	_____
<b>Primary network interface</b>	_____
<b>IP address</b>	_____
<b>Name service</b>	_____
<b>Domain name</b>	_____
<b>System part of a subnet?</b>	_____
<b>Netmask</b>	_____
<b>Name server and IP address</b>	_____
<b>Ethernet address</b>	_____

**≡ B**

---

## Kernel Architectures



Table C-1 shows the kernel architectures for various hardware platforms. You may need this information when preparing a system for installation of the Solaris software.

*Table C-1* Architecture Types

<b>System</b>	<b>Kernel Architecture</b>
x86	i86pc
Sun-4/110	sun4
Sun-4/2xx	
Sun-4/3xx	
Sun-4/4xx	
SPARCstation 1	sun4c
SPARCstation 1+	
SPARCstation SLC	
SPARCstation IPC	
SPARCstation ELC	
SPARCstation IPX	
SPARCstation 2	
SPARCserver 1000	sun4d
SPARCcenter 2000	
SPARCengine 1E	sun4e

*Table C-1 Architecture Types (Continued)*

<b>System</b>	<b>Kernel Architecture</b>
SPARCstation 10 SPARCserver 6xx SPARCstation LX SPARCclassic SPARCstation 10 SX SPARCstation 20 SPARCstation 5 SPARCstation Voyager	sun4m
Other SPARC systems	Refer to the <i>SPARC Hardware Platform Guide</i> for kernel architecture 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.

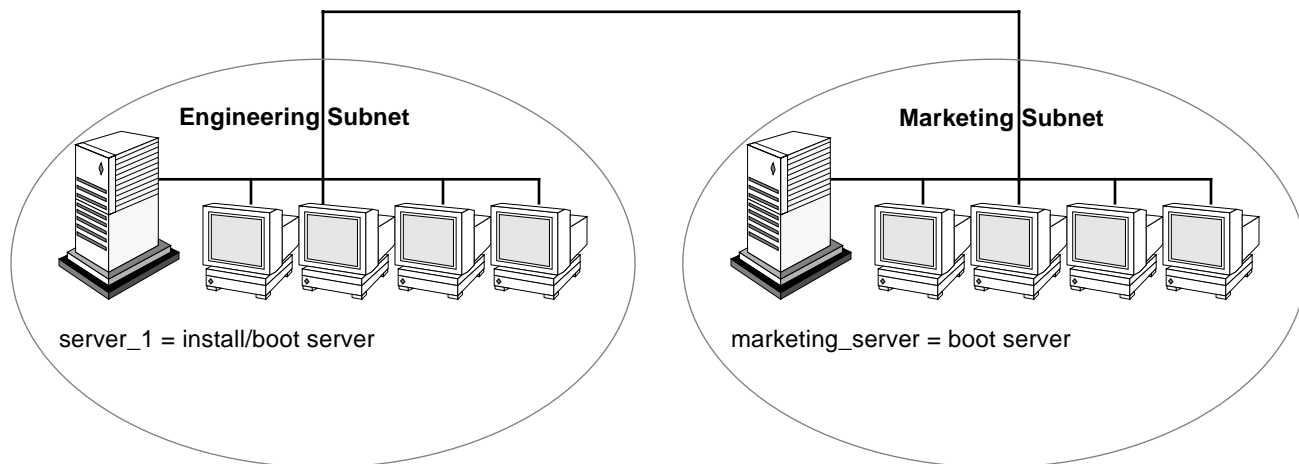


Figure D-1 Sample Site Setup

At this fictitious site:

- The engineering group is on its own subnet. This group uses NEC® Powermate® systems for software development.
- The marketing group is on its own subnet. This group uses Dell 466 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 2.4 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`.
- The network information is preconfigured on the site as described in “Preconfiguring Network Information” on page 43.

## 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	<code>install_type</code>	<code>initial_install</code>
❷	<code>system_type</code>	<code>standalone</code>
❸	<code>partitioning</code>	<code>default</code>
❹	<code>cluster</code>	<code>SUNWCuser</code>
❺	<code>package</code>	<code>SUNWaudmo</code>

- 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:

<code>network</code>	<code>255.222.43.0</code>	-	<code>eng_profile</code>	-
<code>network</code>	<code>255.222.44.0</code>	-	<code>marketing_profile</code>	-

---

**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 92 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 i386pc
# ./add_install_client -c server_1:/jumpstart host_eng2 i386pc
.
.
.
```

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>i386pc</code>	Specifies the architecture of the systems that will use <code>server_1</code> as an install server. (This is the proper kernel architecture for x86 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 i86pc
```

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>i86pc</code>	Specifies the architecture of the systems that will use this boot server. (This is the proper kernel architecture for x86 systems.)

## ≡ D

---

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 i86pc
# ./add_install_client -s server_1:/export/install -c server_1:/jumpstart host_mkt2 i86pc
.
.
.
```

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>i86pc</code>	Specifies the architecture of the systems that will use this boot server. (This is the proper kernel architecture for x86 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. This appendix also suggests possible solutions for some of the problems.

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

<i>Error: Unknown client "host_name"</i>	<i>page 152</i>
<i>ip: joining multicasts failed on tr0 - will use link layer broadcasts for multicast</i>	<i>page 152</i>
<i>le0: No carrier - transceiver cable problem</i>	<i>page 152</i>
<i>Still trying to find a RPL server..</i>	<i>page 153</i>
<i>Requesting Internet address for Ethernet_Address</i>	<i>page 153</i>
<i>RPC: Timed out No bootparams (whoami) server responding; still trying...</i>	<i>page 154</i>

## ≡ E

---

### *Specific Installation Errors*

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

---

<b>Reason Error Occurred</b>	<b>How to Fix the Problem</b>
------------------------------	-------------------------------

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

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.

---

```
ip: joining multicasts failed on tr0 - will use link layer broadcasts for multicast
```

---

<b>Reason Error Occurred</b>	<b>How to Fix the Problem</b>
------------------------------	-------------------------------

You will see this error message when you boot a system with a token ring card. Ethernet multicast and token ring multicast do not work the same way. The driver returns this error message to indicate that an invalid multicast address was given.

Ignore this error message. If multicast doesn't work, IP uses layer broadcasts instead.

---

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

---

<b>Reason Error Occurred</b>	<b>How to Fix the Problem</b>
------------------------------	-------------------------------

This system is not connected to the network.

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

---

```
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. .

```
Still trying to find a RPL server...
```

**Reason Error Occurred**

The system is trying to boot over the network, but the server is not set up to boot this system.

**How to Fix the Problem**

On the install server, execute `add_install_client` for the system to be installed. The `add_install_client` command sets up an `/rplboot` directory, which contains the necessary network boot program.

```
Requesting Internet address for Ethernet_Address
```

**Reason Error Occurred**

The client is trying to boot over the network, but it cannot find a system that knows about it.

**How to Fix the Problem**

Verify the system's host name is in the NIS or NIS+ name service. If the system's host name is in the NIS or NIS+ name service, and the system continues to print this error message, try rebooting.

## ≡ E

---

```
RPC: Timed out
No bootparams (whoami) server responding; still trying...
```

---

**Reason Error Occurred**

The client is trying to boot over the network, but it cannot find a system with an entry in the `/etc/bootparams` file on the install server.

**How to Fix the Problem**

Use `add_install_client` on the install server. This will add the proper entry in the `/etc/bootparams` file, enabling the client to boot over the network.

---

---

## General Installation Problems

---

Problem	How to Fix the Problem
The system boots over the network, but from a system other than the specified install server.	On the name server, update the <code>/etc/bootparams</code> entry for the system being installed. The entry should conform to the following syntax:  <code>install_system root=boot_server:path install=install_server:path</code>

---

---

Problem	How to Fix the Problem
After setting up an install server and configuring the system to install over the network, the system still does not boot.	Be sure the <code>tftpd</code> daemon is running on the install server. Type the following command and press Return:  <code>ps -ef   grep rpld</code>  If this command does not return a line indicating the <code>rpld</code> daemon is running, execute the following command:  <code>/usr/sbin/rpld</code>  After making this change, try booting the system again.

---

## ≡ E

---

---

### 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 -
```

---

---

<b>Problem</b>	<b>How To Fix the Problem</b>
Several processes that normally run when booting the system are not run when the system boots from the CD. (This is done to enable the system to boot and the Solaris installation program to run with only 16 Mbytes of memory.)	<p>To boot from the CD to do disaster recovery (that is, restoring the root file system), you need to add functionality normally provided when booting. To provide this functionality, a script is provided, <code>/sbin/setup_cd</code>, which can be run to perform the additional setup that might be needed to do disaster recovery. <code>setup_cd</code> does the following tasks:</p> <ul style="list-style-type: none"><li>- Finishes setting up the device file system for tapes and ports</li><li>- Enables routing</li><li>- Enables multicast</li><li>- Invokes the <code>statd(1M)</code> daemon</li><li>- Invokes the <code>lockd(1M)</code> daemon</li><li>- Starts the automounter</li></ul> <p>Note that if <code>setup_cd</code> is run on a 16 Mbyte system, it may not be possible to run the Solaris installation program or other programs that need more memory. If a disk with a <code>swap</code> partition is available, the system administrator could avoid this problem by using the <code>swap(1M)</code> command to add swap space to the system before invoking <code>setup_cd</code>.</p> <p>When booting from the Solaris CD during disaster recovery, it is possible to boot the system in single user mode from the CD-ROM. For example, at the menu option to install using Interactive or custom JumpStart, type:</p> <pre>b -s</pre>

---

---

<b>Problem</b>	<b>How to Fix the Problem</b>
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.	<ol style="list-style-type: none"><li>1. Reboot the system from the Solaris CD or from the network.</li><li>2. Choose the upgrade option for installation.</li></ol> <p>The Solaris installation program will determine if the system has been partially upgraded and will continue the upgrade.</p>

---

---

**Problem**

IDE disk drives do not automatically map out bad blocks like other drives supported by Solaris software. Before installing Solaris on an IDE disk, you may want to perform a surface analysis on the disk.

---

**How to Fix the Problem**

---

To perform surface analysis on an IDE disk, follow this procedure:

- 1) Start an interactive installation, as described in Chapter 6, "Booting and Installing Solaris: Interactive." The Solaris installation program will start either a graphical user interface (GUI) or a character user interface (CUI), depending on whether you have a graphics or non-graphics monitor.
  - 2) When either the GUI or CUI program starts, enter information and select the Continue option on the first few screens.
  - 3) When you see the *Installing Solaris - Initial* screen, select the Exit option and exit the installation.
  - 4) If you are using the GUI installation program, open a command tool window for the remaining steps in this procedure. If you are using the CUI installation program, use the system shell for the remaining steps in this procedure.
  - 5) Start the `format` program. Type `format`.
  - 7) Specify the IDE disk drive on which you want to perform a surface analysis.  
**Note:** IDE drives do not include a target number. The IDE drive naming convention is `cx``dy`, where `cx` is the controller number and `dy` is the device number.
  - 6) At the `format>` prompt, type `fdisk`. Use the `fdisk` program to create a Solaris partition on the disk. (If a Solaris `fdisk` partition already exists, leave it alone.)
  - 7) At the `format>` prompt, type `analyze`.
  - 8) At the `analyze>` prompt, type `config`. This will show you the current settings for a surface analysis. If you want to change any settings, type `setup`.
  - 9) At the `analyze>` prompt, type `read`, `write`, or `compare` for the type of surface analysis to be performed. If `format` finds bad blocks, it will remap them.
  - 10) At the `analyze>` prompt, type `quit`.
  - 11) You may want to specify blocks to remap. If so, at the `format>` prompt, type `repair`.
  - 12) Type `quit` to quit the `format` program.
  - 13) Type `installtool` to resume the GUI installation or `suninstall` to resume the CUI installation.
-



---

**Problem**

The Solaris root partition must reside within the first 1024 cylinders of the disk. If it is not, the installation fails after booting.

**How to Fix the Problem**

If the first fdisk partition is primary DOS (PRI DOS), use the `fdisk` program to delete space from it and try booting Solaris again. If the first fdisk partition is extended DOS (EXT DOS) or another operating system, use the `fdisk` program to delete it and try booting Solaris again.

---

≡ *E*

---

## *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.

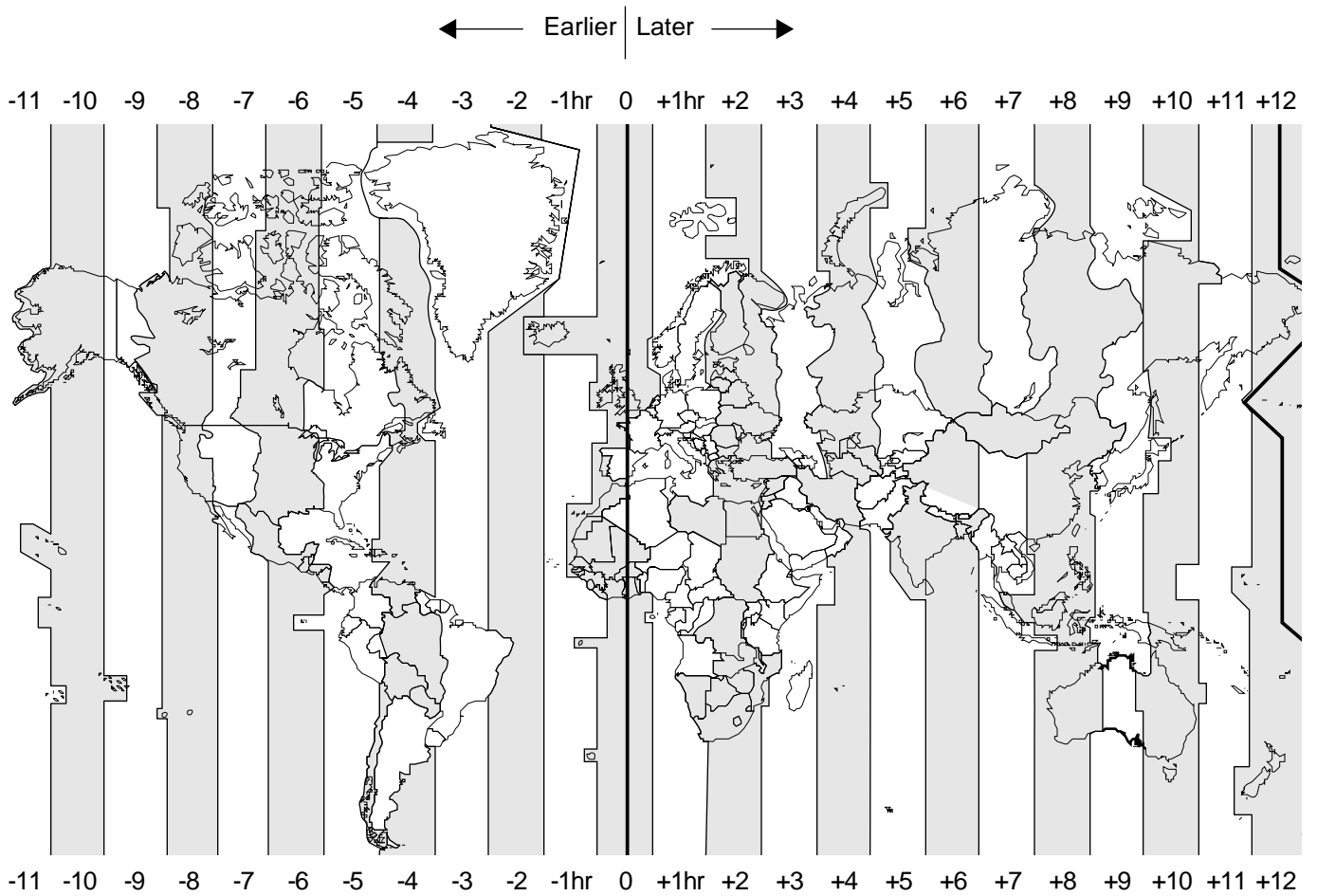


Figure F-1 Greenwich Meantime Map

## Glossary

---



### **architecture type**

The instruction set that a system uses to run the Solaris software.

### **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 a server for its /usr and /usr/kvm file systems.

**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 `pinstall` 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 a 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 a file 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.

**entire distribution plus OEM support**

A software group that contains the entire Solaris release, plus additional hardware support for OEMs (SPARC only). This software group is recommended when installing Solaris software on servers.

**EISA**

Extended Industry Standard Architecture. A type of bus on x86 systems. EISA bus standards are “smarter” than ISA bus systems, and attached devices can be automatically detected when they have been configured via the “EISA configurator” program supplied with the system. See *ISA*.

**/etc**

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

**/export**

A file system on a 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 a file 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).



**ISA**

Industry Standard Architecture. A type of bus found in x86 systems. ISA bus systems are “dumb” and provide no mechanism the system can use to detect and configure devices automatically. See *EISA*.

**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.

**kernel architecture**

The type of kernel that must be run on a particular system because of its set of hardware components.

**locale**

A specific language associated with a region or territory.

**MCA**

Micro Channel Architecture. A type of bus on x86 systems. The MCA bus provides fast data transfer within the computer, and attached devices can be automatically detected when they have been configured using the reference disk provided by the manufacturer. The MCA bus is not compatible with devices for other buses.

**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.

**package**

A functional grouping of files and directories that form a software application. The Solaris software is divided into four main *software groups*, which are each composed of *clusters* and packages.

**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 `rules` file. See *rules file*.

**/ (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 `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. See *profile*.

**rules.ok file**

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* use the `check` script to create the `rules.ok` file.

**server**

A system that provides services to systems on a network. To serve diskless clients, a server must have disk space set aside for each diskless client's root file system and swap space (`/export/root`, `/export/swap`). To serve dataless clients, a server must provide the `/usr` and `/usr/kvm` file systems.

**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.

## ≡ G

---

### **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, entire distribution, or entire distribution plus OEM support (SPARC only).

### **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 a 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. The valid system types are standalone system, dataless client, and diskless client.

### **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 (SPARC only) if at least one disk on your system has a Solaris 2.x root file system on slice 0. An upgrade 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.

## ≡ G

---

# Index

---

## A

- `add_install_client`
  - adding system to boot server 37
  - adding system to install server 35
  - command syntax 35, 37
- adding diskless client 39
- adding systems to boot server 37
- adding systems to install server 35
- administration tool
  - adding diskless clients 39
- alternative installation program 111
- `arch` rule keyword 92
- auto-install, *See* custom JumpStart
- automating installation
  - See* custom JumpStart

## B

- `begin` script
  - creating derived profiles with 102
  - example 103
  - required permissions 102
  - use of 102
- boot procedure for diskless clients 42
- boot procedure for interactive installation 117, 125
- boot server

- adding systems to 37
- defined 16

- booting diskless clients 42
- booting from wrong server 155
- buses supported 9

## C

- `check` script 97, 98
- `client_arch` profile keyword 75
- `client_swap` profile keyword 75
- `cluster` profile keyword 76
- common installation problems 151
- common problems
  - booting from wrong server 155
  - installation fails after booting 159
  - mapping out bad blocks on IDE disk 158
  - OpenWindows doesn't start 156
  - system does not boot 155, 159
- copying boot software with `setup_install_server` 149
- copying Solaris CD to disk 24
- creating a `rules` file 87
- creating profiles 70
- custom JumpStart
  - compared to interactive 11
  - creating a `rules` file 87

---

- defined 53
- from a server 65
- overview of set-up tasks 55
- process described 56, 57, 58
- rules file syntax 87
- sharing the JumpStart directory 65, 145

## D

- dataless client
  - adding for network installation 33
  - adding SPARC support on server 28
  - defined 8
- derived profile, defined 102
- `dfstab` file, sharing the JumpStart directory 65, 145
- disaster recovery procedure 157
- disk interfaces supported 9
- disk requirements for copying Solaris CD to disk 24, 27
- disk space requirements 10
- diskette
  - formatting 61, 62
  - using for custom JumpStart 59
- diskless client
  - adding for network installation 39
  - adding SPARC support on server 28
  - defined 8, 39
- `disksize` rule keyword 92
- DNS, note about use 30
- `domainname` rule keyword 92
- `dontuse` profile keyword 77

## E

- error messages 151

## F

- `fdisk` profile keyword 77, 78
- file server, defined 16
- `filesys` profile keyword 79, 80, 81
- finish script

- adding files with 105
- customizing environment with 106
- examples 106
- required permissions 104
- setting root password with 106
- use of 104
- using `set_root_pw` file 107

- floppy, *See* diskette

- formatting a diskette 61, 62

## H

- hardware platforms supported 9
- host manager
  - adding diskless clients with 39
  - adding standalone, server, or dataless clients with 30
- `hostname` rule keyword 93

## I

- IDE disks, mapping out bad blocks 158
- install server, defined 16
- `install_type` profile keyword 81
- installation methods 11
- interactive installation
  - boot procedure 117, 125
  - compared to custom JumpStart 11
  - decision flowchart 12

## J

- JumpStart directory
  - defined 65
  - required permissions 59, 65
  - sharing from a server 65

## K

- `karch` rule keyword 93
- kernel architectures 141

## L

- languages, valid values 50



- 
- locale
    - defined 46
    - updating NIS+ table 50
    - valid values 50
  - locale profile keyword 82
  - locale.org\_dir, creating NIS+ table 49
  - localization, languages supported 50
- M**
- mapping out bad blocks on IDE disks 158
  - matching order in rules file 90
  - memory requirements for installation 9
  - memsiz rule keyword 93
  - methods for installing Solaris software 11
  - model name rule keyword 94
  - mounting Solaris CD manually 23, 26, 66
- N**
- name server, defined 16
  - naming profiles 71
  - network installation
    - adding diskless clients 39
    - adding standalone systems 29
    - defined 15
    - preconfiguring date and time 43
    - preconfiguring geographic region 43
    - preconfiguring network information 18
    - preconfiguring subnet mask 43
    - recommendations 22
    - requirements 16
  - network rule keyword 94
  - networked systems, defined 8
  - NIS
    - preconfiguring default locale 46
    - pushing maps 45
  - NIS+
    - preconfiguring default locale 49
    - preconfiguring network information 45
  - nispopulate, preconfiguring network information 45
  - nistbladm
    - creating locale.org\_dir table 49
    - updating locale.org\_dir table 50
  - non-networked systems, defined 8
  - num\_clients profile keyword 82
- O**
- OpenWindows, starting 120, 126
  - overview of custom JumpStart 55
- P**
- package profile keyword 83
  - partitioning profile keyword 84
  - PASSWD variable 107
  - pfinstall
    - command syntax 108
    - examples 109, 110
    - testing profiles with 109
    - use of 108
  - populating NIS+ tables 45
  - preconfiguring date and time for network installation 43
  - preconfiguring geographic region for network installation 43
  - preconfiguring network information 18
  - preconfiguring subnet mask for network installation 43
  - profile
    - keywords defined 75
    - values defined 75
  - profile keywords
    - client\_arch 75
    - client\_swap 75
    - cluster 76
    - dontuse 77
    - fdisk 77, 78
    - filesys 79, 80, 81
    - install\_type 81
    - locale 82
    - num\_clients 82
    - package 83
    - partitioning 84

---

- system\_type 84
- usedisk 84
- profiles
  - creating 70
  - examples 72
  - information about editing 71, 87
  - naming 71
  - requirements for 70
  - testing 109

## R

- recovery procedure 157
- requirements for a profile 70
- resolv.conf, note about use 30
- restoring the root file system 157
- root password, setting with finish script 106
- rootdisk 96
- rpld, starting 155
- rule keywords
  - any 92
  - arch 92
  - disksize 92
  - domainname 92
  - hostname 93
  - installed 93
  - karch 93
  - memsize 93
  - model 94
  - network 94
  - totaldisk 95
- rules file
  - comment fields 91
  - creating 87
  - defined 86
  - example rules 90
  - matching order 90
  - name requirements 91
  - required permissions 91
  - rule keywords 92
  - rule matching order 91
  - rule values 92
  - rule wrap 91

- syntax defined 87
- validating 97

## S

- server
  - adding other architecture support on 28
  - defined 8
- setting system clock 161
- setup\_install\_server
  - copying Solaris to disk 24
  - setting up a boot server 27, 149
- sharing JumpStart directory 145
- SI\_CONFIG\_DIR variable 105
- SI\_PROFILE variable 103
- SI\_SYS\_STATE variable 107
- Solaris software
  - buses supported 9
  - character-based interface 119
  - core system support 10
  - developer system support 10
  - disk interfaces supported 9
  - disk space requirements 10
  - end user system support 10
  - entire distribution 10
  - graphical interface 119
  - groups of software to install 10
  - memory requirements for installation 9
  - methods for installing 11
  - processors supported 9
- standalone system
  - adding for network installation 29
  - defined 8
- starting OpenWindows 120, 126
- starting rpld 155
- SUNWCall software group 10, 76
- SUNWCprog software group 10, 76
- SUNWCreq software group 10, 76
- SUNWCuser software group 10, 76
- SUNWCXall software group 76
- surface analysis on IDE disks 158

---

system boot problem 155, 159  
system types 8  
system types and kernel architectures 141  
system\_type profile keyword 84

## **T**

testing profiles with `pinstall` 108  
time zones supported by Solaris 161  
totaldisk rule keyword 95

## **U**

upgrade installation  
    continuing after a failed installation  
    157  
usedisk profile keyword 84

## **V**

validating a `rules` file 97  
validating profiles 109  
variables  
    PASSWD 107  
    SI\_CONFIG\_DIR 105  
    SI\_PROFILE 103  
    SI\_SYS\_STATE 107

## **W**

wrapping lines in `rules` file 91

## **X**

x86 processors supported 9