Solstice AutoClient 2.0 Administration Guide

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

Part No: 802-5630-10
Revision A, May 1996

SunSoft
A Sun Microsystems, Inc. Business

Please
Recycle

Adobe PostScript

Contents

Figures

Tables

About This Book

The *Solstice AutoClient 2.0 Administration Guide* presents the administrative tasks required for the successful operation of the Solstice™ AutoClient™ product. This guide also includes information on how to administer AutoClient systems with Host Manager, an easy-to-use graphical user interface.

Who Should Use This Book

This book is intended for system administrators whose responsibilities include setting up and maintaining systems on a network.

Though much of the book is directed toward novice administrators and other readers who may be new to the Solaris environment, it also contains information useful to experienced system administrators.

How This Book Is Organized

- **Overview information** about AutoClient technology and the Solstice AutoClient product can be found in Chapter 1, "About the AutoClient Technology," and Chapter 2, "About the AutoClient Product."
- Information on **using the Solstice AutoClient product in a name service environment** is in Chapter 3, "Using Solstice AutoClient in a Name Service Environment."

- For information on **Solstice AutoClient security features and setting up a security policy at your site**, see Chapter 4, “Security.”
- Additional information on **using the Host Manager graphical user interface** is in Chapter 5, “Host Manager Reference Information.”
- **Tasks needed to set up your AutoClient systems** are in Chapter 6, “Managing AutoClient Systems.”
- **Booting information** is in Chapter 7, “Booting a System From the Network.”
- **Maintenance tasks** to be performed after your network is set up and running are in Chapter 8, “AutoClient Environment Maintenance.”

Supporting Documentation

You can refer to the following documentation for additional information that may help you set up and maintain your AutoClient systems:

- *SPARC: Installing Solaris Software*
- *x86: Installing Solaris Software*
- *Solaris PowerPC Edition: Installing Solaris Software*
- *System Administration Guide, Volume I*
- *Solstice AdminSuite 2.2 Administration Guide*

Conventions

Table P-1 describes the typographic conventions used in this book.

Table P-1 Typographic Conventions

Typeface or Symbol	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% You have mail.`

`AaBbCc123`	What you type, contrasted with on-screen computer output	`machine_name%` **`su`**
		`Password:`

AaBbCc123	Command-line placeholder: replace with a real name or value	To delete a file, type `rm` *filename*.
AaBbCc123	Book titles, new words or terms, or words to be emphasized	Read Chapter 6 in *User's Guide*. These are called *class* options. You *must* be root to do this.

Shell Prompts in Command Examples

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

Table P-2 Shell Prompts

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

Part 1 — Solstice AutoClient Overview

This part provides an overview of the Solstice AutoClient software and contains these chapters.

1 **About the AutoClient Technology**
Provides information on the AutoClient technology: AutoClient system characteristics, the advantages over other system types, and how the AutoClient technology works.

2 **About the AutoClient Product**
Provides information on what is new with the current product, disk space requirements, configuration issues, limitations, and other product information.

3 **Using Solstice AutoClient in a Name Service Environment**
Provides information on how to use the Solstice AutoClient software in a name service environment.

4 **Security**
Describes security issues and provides suggestions on how to use the Solstice AutoClient software in a manner that conforms to your site security policies.

5 **Host Manager Reference Information**
Provides information on various features of the Host Manager application.

About the AutoClient Technology 1

The Solstice AutoClient product allows you to set up AutoClient systems and provide centralized administration for these systems. An *AutoClient system* is a system type that *caches* (locally stores copies of data as it is referenced) all of its needed system software from a server. AutoClient systems use Solaris™ diskless and cache file system (CacheFS™) technologies.

CacheFS is a general purpose file system caching mechanism that improves NFS™ server performance and scalability by reducing server and network load. (You can also use CacheFS with HSFS file systems.) The AutoClient technology improves ease of administration, enabling system administrators to maintain many AutoClient systems from a server. Changes do not have to be made on each individual system. Users may notice improved performance as well, on both AutoClient systems and servers.

For more information about CacheFS, see *System Administration Guide, Volume I*.

This is a list of the overview information in this chapter.

Note – Throughout this guide, “AutoClient systems” refer to any system that uses the AutoClient technology.

Overview of System Types

System types are basically defined by how they access the root (/) and /usr file systems, including the swap area. For example, standalone and server systems mount these file systems from a local disk, while diskless and dataless clients mount the file systems remotely, relying on servers to provide these services. Table 1-1 lists these and other differences for each system type.

Table 1-1 System Type Overview

System Type	Local File Systems	Local Swap?	Remote File Systems
Server	root (/) /usr /home /opt /export /export/home /export/root	Yes	optional
Standalone System	root (/) /usr /export/home	Yes	optional
Dataless Client	root (/)	Yes	/usr /home
Diskless Client	– *none* –	No	root (/) swap /usr /home
AutoClient System	cached root (/) cached /usr	Yes	root (/) /usr /home

Table 1-2 describes how the other clients compare to a standalone system.

Table 1-2 Comparison of Clients Relative to a Standalone System

System Type	Centralized Administration	Performance	System Disk Usage	Network Use
AutoClient System	better	similar	better	similar
Diskless Client	better	worse	better	worse
Dataless Client	similar	worse	better	worse

Server Characteristics

A server system has the following file systems:

- The root (`/`) and `/usr` file systems, plus swap space
- The `/export`, `/export/swap`, and `/export/home` file systems, which support client systems and provide home directories for users
- The `/opt` directory or file system for storing application software

Servers can also contain the following software to support other systems:

- OS services for diskless clients and AutoClient systems
- Solaris CD image and boot software for networked systems to perform remote installations
- JumpStart™ directory for networked systems to perform custom JumpStart installations

Standalone System Characteristics

A *networked standalone system* can share information with other systems in the network, but it can function autonomously because it has its own hard disk with enough space to contain the root (`/`), `/usr`, and `/export/home` file systems and swap space. The standalone system thus has local access to operating system software, executables, virtual memory space, and user-created files.

A *non-networked standalone system* is a standalone system with all the characteristics listed above except that is not connected to a network.

Dataless Clients

A *dataless client* has local storage for its root (/) file system and swap space. The dataless client cannot function if detached from the network, because its executables (`/usr`) and user files (`/home`) are located across the network on the disk of a server.

Note – SunSoft plans to remove support for dataless clients after Solaris 2.5. You can add this system type now using Host Manager, but in future releases of the Solaris operating environment you will need to choose a different type. It is recommended that you use AutoClient systems instead of dataless clients.

A dataless client places far less demand on the server and the network than a diskless client does. Because dataless clients require less network access, a server can accommodate many more dataless clients than it can diskless clients. Also, since all the user files of all the dataless clients are stored centrally (on a server), they can be backed up and administered centrally.

Diskless Client Characteristics

A *diskless client* has no disk and depends on a server for all its software and storage area. A diskless client remotely mounts its root (/), `/usr`, and `/home` file systems from a server.

A diskless client generates significant network traffic due to its continual need to procure operating system software and virtual memory space from across the network. A diskless client cannot operate if it is detached from the network or if its server malfunctions.

AutoClient System Characteristics

An AutoClient system is nearly identical to a diskless client in terms of installation and administration. It has the following characteristics:

- Requires a 100-Mbyte or larger local disk for swapping and for caching its individual root (/) file system and the `/usr` file system from a server
- Can be set up so that it can continue to access its cache when the server is unavailable
- Relies on servers to provide other file systems and software applications

- Contains no permanent data, making it a field replaceable unit (FRU)

The following figure shows how a server and an AutoClient system work together.

Note – You must obtain a license for each AutoClient system you want to add to your network. See the *Solstice AutoClient 2.0 Installation and Product Notes* for licensing information.

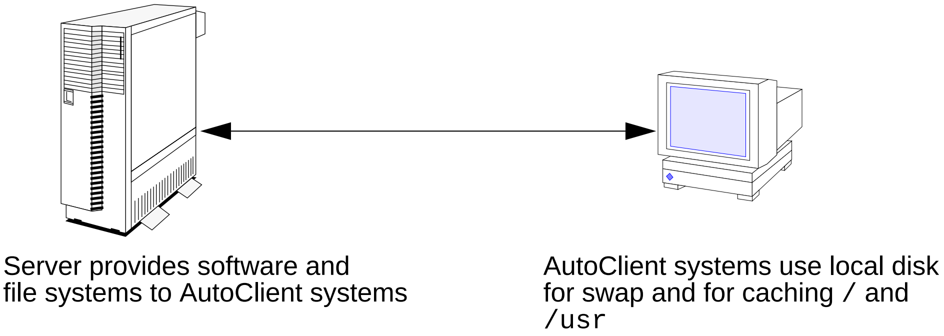

Server provides software and file systems to AutoClient systems
AutoClient systems use local disk for swap and for caching / and /usr

Figure 1-1 AutoClient System Characteristics

Why Use an AutoClient System?

AutoClient technology provides many system administration advantages over existing system types.

Advantages Over Diskless Systems

AutoClient systems:

- Provide better overall scalability in a network environment, which could result in less network load
- Use less disk space on a server than a diskless system (an AutoClient system does not require any swap space on a server)
- Use significantly less network and server bandwidth than a diskless system

Advantages Over Dataless and Standalone Systems

AutoClient systems:

- Require less system administration overhead. The AutoClient system's data is on a server, which enables centralized administration. For example, with AutoClient systems you only need to back up the server(s) that supports the AutoClient systems. To back up dataless systems, you have to perform a backup on each system. Also, you can manipulate AutoClient root file systems from the server, without accessing each system individually.
- Are FRUs, which makes them easy to replace if they fail.
- Are installed by setting up an AutoClient system with the Host Manager. You do not have to use the Solaris installation program to install the Solaris environment on an AutoClient system.

How an AutoClient System Works

The CacheFS technology is the important component of AutoClient systems. A *cache* is a local storage area for data. A *cached file system* is a local file system that stores files in the cache as they are referenced, and subsequent references to the same files are satisfied from the cache rather than again retrieving them from the server. This functionality reduces the load on the network and the server, and generally results in faster access for the AutoClient system. Note that when the cache becomes full, space is reclaimed on a least recently used (LRU) basis. Files that have been unreferenced for the longest time are discarded from the cache to free space for the files that are currently being referenced.

An AutoClient system uses its local disk for swap space and to cache its individual root (`/`) file system and the `/usr` file system from a server's back file systems. Figure 1-2 shows how an AutoClient system works.

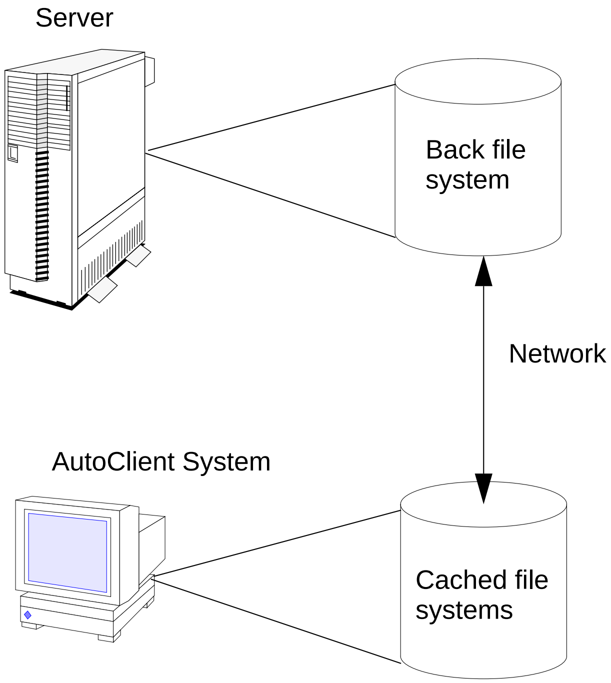


Figure 1-2 How an AutoClient System Works

How an AutoClient System’s Cache Is Updated

An AutoClient system uses *consistency checking* to keep a cached file system synchronized with its back file system. The following descriptions show how consistency checking is done for an AutoClient system:

- By default, files that are updated in the server’s back file systems are updated on the AutoClient system’s cached file systems within 24 hours. However, if the update needs to occur sooner, you can use the `autosync` command. The `autosync(1M)` command initiates consistency checking that updates (synchronizes) an AutoClient system’s cached file systems with its server’s back file systems.

  For more information about the `autosync` command, see Chapter 8, “AutoClient Environment Maintenance.” You can also refer to the `autosync(1M)` man page.

- Each time an AutoClient system is booted with the `-f` option, the AutoClient system’s cached file systems are checked for consistency and updated with its server's back file systems.

Note – Consistency checking for an AutoClient system is different from a system running CacheFS. AutoClient files (`/` and `/usr`) are not likely to change very often, so consistency checking does not need to occur as frequently on an AutoClient system as it does on a system running CacheFS. This reduces traffic on your AutoClient network. See *System Administration Guide, Volume I* for detailed information about CacheFS consistency checking.

Also, if you add new files to an AutoClient system, its server's back file systems are updated immediately, because an AutoClient system uses a *write-through* cache. A write-through cache is one that immediately updates its back file system as data is changed or added to the cache.

About the AutoClient Product 2

The Solstice AutoClient product allows you to set up AutoClient systems and administer changes to them. This chapter provides information regarding the AutoClient product so that you can successfully complete the tasks discussed in the subsequent chapters.

This is a list of the overview information in this chapter.

What's New in the Solstice AutoClient 2.0 Product

The Solstice AutoClient 2.0 product provides the following new features:

- The "disconnectable" option

  The "disconnectable" option enables AutoClient systems running the Solaris 2.5 or later software to continue running by using their cached file systems when the server is unavailable. Implementing this feature eliminates "NFS server *servername* not responding" error messages.

  For more information on this feature, see the online help in Host Manager or "Fields on the Add Window for the Solstice AutoClient System Type" on page 71.

- Additional disk configuration options

  In addition to the 1disk and 2disk configuration options, you can now choose the local200 and local400 configuration options for disks that are a minimum of 300 Mbytes or 500 Mbytes, respectively. The local200 and local400 configuration options set up a scratch file system mounted on `/local`, in addition to your cached file system.

Caution – It is recommended that you choose the 1disk or 2disk option. Choosing the local200 or local400 option means that your system is no longer a FRU, which is a primary feature of AutoClient systems. Data written to the scratch file system is *not* written back to the server.

  For more information on this feature, see the online help in Host Manager or "Fields on the Add Window for the Solstice AutoClient System Type" on page 71.

- A command-line interface (CLI) for Host Manager

  The CLI allows you to complete most of the administrative operations you can perform with Host Manager from the command line.

  For more information on this feature, see "Command-Line Equivalents of Host Manager Operations" on page 18. Examples are provided, where appropriate, with the tasks described in Chapter 6, "Managing AutoClient Systems."

- Mixed-mode name service support

  The new admtblloc command enables you to configure the name service policy for the Solstice AutoClient product. You can now choose a mixture of name services for the Solstice AutoClient product to populate.

  For more information on this feature, see "Selecting a Name Service Environment" on page 24.

- Patching clients

  The new admclientpatch command provides a way to add patches to diskless clients and AutoClient systems in a more efficient and centralized way. It also provides a way to ensure that the same patches are applied to all your diskless clients and AutoClient systems. Patches are automatically applied when clients are created.

  For more information on this feature, see "Managing Patches on AutoClient Systems" on page 123.

- PowerPC support

  The Solstice AutoClient product can now be installed and run on PowerPC systems that are running the Solaris 2.5.1 or later software.

- Solaris 2.5.1 support

  The Solstice AutoClient product now supports SPARC, x86, and PowerPC clients and servers running the Solaris 2.5.1 operating system.

- Improved NIS support

  The Solstice AutoClient product can now view and modify NIS maps on Solaris 2.x NIS master servers that have been set up with the Name Services Transition Kit 1.2.

  For more information on this feature, see "Selecting a Name Service Environment" on page 24.

- Remote halting and booting commands

  Two new commands, admhalt and admreboot, can be used to remotely reboot and halt systems.

  For more information on these commands, see the admhalt(1M) and the admreboot(1M) man pages.

- Automatically booting x86 systems from the network

  You can create an MDB floppy that enables your x86 AutoClient systems to automatically boot from the network in case you are not present to boot it, such as after a power failure.

  For more information on this feature, see “x86: How to Set Up a System to Automatically Boot From the Network” on page 112.

- Operation logging

  You can create a log file that collects the date, time, server, user ID (UID), and operation, for each major operation completed with Host Manager.

  For more information on this feature, see “Logging Host Manager Operations” on page 58.

- Mixed OS support for servers

  For more information AutoClient interoperability, see “Solstice AutoClient Interoperability Support” on page 13.

Solstice AutoClient Interoperability Support

Table 2-1 describes the server-client configurations that are supported by the Solstice AutoClient 2.0 software.

Table 2-1 Supported Server-Client Configurations

If You Have A ...	You Can Add OS Services and Support For ...	For the Following Releases ...
SPARC server running Solaris 2.3 – 2.5.1	SPARC clients	Solaris 2.4 – 2.5.1
	x86 clients	Solaris 2.4 – 2.5.1
	PowerPC clients	Solaris 2.5.1
x86 server running Solaris 2.4 – 2.5.1	SPARC clients	Solaris 2.4 – 2.5.1
	x86 clients	Solaris 2.4 – 2.5.1
	PowerPC clients	Solaris 2.5.1
PowerPC server running Solaris 2.5.1.	SPARC clients	Solaris 2.4 – 2.5.1
	x86 clients	Solaris 2.4 – 2.5.1
	PowerPC clients	Solaris 2.5.1

Disk Space Requirements for AutoClient Servers and AutoClient Systems

Table 2-2 lists the disk space requirements for AutoClient servers and AutoClient systems.

Table 2-2 Disk Space Requirements for AutoClient Servers and AutoClient Systems

System Type	File System	Minimum Disk Space Requirements

Servers of AutoClient systems	root (/)
	`/usr`
	`/var`
	`/export`
	`/export`
		1 Mbyte
		4 Mbytes
		7.5 Mbytes
		17 Mbytes per OS service
		20 Mbytes for each AutoClient system (typically in `/export`)
		Note: When you add an AutoClient system to a server, the `/export/root` directory is specified by default to store the 20 Mbytes for each system. However, you can specify any directory that has available disk space. See “Adding AutoClient Systems” on page 71 for detailed information.

AutoClient systems	cache for root (/) and shared `/usr`	Minimum of 70 Mbytes

Caution – The AutoClient configuration uses the entire disk(s) on the system. (For more information on AutoClient disk configurations, see Table 6-3 on page 73.) If data already exists on the disk(s), it will be overwritten. You should preserve the data elsewhere by backing it up before you add and boot a system. (See “Adding AutoClient Systems” on page 71.)

Configuration and Transition Issues

In the Solaris 2.5 environment you can add new AutoClient systems to your network or you can make the following AutoClient system conversions.

Table 2-3 AutoClient System Conversions

You Can Convert A ...	To A ...
Generic System	AutoClient System
Standalone System	AutoClient System
Dataless System	AutoClient System
AutoClient System	Standalone System

Caution – If you plan to convert existing generic, dataless, or standalone systems to AutoClient systems, you should consider this process a re-installation. Any existing system data will be overwritten when the AutoClient system is booted for the first time.

Note – Supported configurations for AutoClient systems are systems with one or two disks only. Other disk configurations are not recommended for the AutoClient system type. Depending on the disk configuration you choose, all of one disk or all of two disks could be overwritten by the AutoClient product. (Disk configuration options are described in Table 6-3 on page 73.)

- If your standalone system contains local mail (in `/var/mail`), copy these directories from the local disk before using the local disk as a cache. In your AutoClient configuration, set up a central mail spool directory on your server for ease of administration.
- If your network has local file systems (other than the Solaris distribution file systems) on your standalone systems, you need to save these files before converting these systems to AutoClient systems. AutoClient systems that maintain local file systems lose the significant advantages of being FRUs, and of not requiring system backup.

- When an AutoClient system is set up using Host Manager, the /opt directory will be empty. On the server, you should establish a uniquely-named /opt file system for each platform that it will support (for example, sparc_opt or x86_opt), so that the AutoClient systems can mount the appropriate file system.

  You should use Storage Manager to create and maintain your file systems. See *Solstice AdminSuite 2.2 Administration Guide* for more information on Storage Manager.

Solstice AutoClient Product Limitations

When you set up your network with AutoClient systems, you need to consider the following limitations:

- The /usr file system is read-only for AutoClient systems; systems cannot make any modifications to the /usr file system. AutoClient systems make use of the /usr file system in the same way as diskless and dataless systems (mounted read-only).
- The pkginfo(1) command will not reflect all the software that is available to an AutoClient system. In particular, the package database for an AutoClient system will contain only the packages that were installed in the system's root directory. The pkginfo(1) command will not reflect that all of the software in /usr is available.
- Normally, booting an AutoClient system as an NIS system will not work if the network has an NIS+ server running that already knows about the AutoClient system; the AutoClient system will be automatically set up as an NIS+ system. However, you can override this by modifying your bootparams file and adding the ns key for your AutoClient system. For more information on the ns key, see bootparams(4).
- If an AutoClient system is running the Solaris 2.4 software, and the AutoClient server is unavailable, the AutoClient system will see the message in its console "NFS server *servername* not responding." Only AutoClient systems running the Solaris 2.5 or later software can be set up to use the file systems in the cache when the server is unavailable. For more information on the disconnectable feature, see Table 6-2 on page 71 or online help.
- Once you have successfully added OS services to an OS server, they cannot be deleted using Host Manager.

- The AutoClient product does not support Power Management™ software, which conserves the amount of power that a system consumes. For more information on Power Management software, see *Using Power Management*.

The Relationship Between AutoClient Systems and Host Manager

AutoClient systems are installed, configured, and maintained with the command-line interface or with Host Manager. Host Manager is a graphical user interface that allows for greater efficiency and ease of use in administering your AutoClient systems in a network environment. Host Manager enables system administrators to perform the following tasks:

- Add, modify, display, or remove AutoClient system information in a network environment
- Convert existing generic, standalone, and dataless systems to the AutoClient system type
- Change information about multiple AutoClient systems in one operation

Note – Host Manager does not set up an AutoClient system's `/opt` directory. For more information, see "Configuration and Transition Issues" on page 15.

Host Manager has the following features:

- **Easy conversion to the AutoClient system type** – You can easily add AutoClient systems to your network, and convert some existing system types to AutoClient systems.
- **Easy Modification** – You can modify an AutoClient system by using the Modify screen. You do not have to delete and then re-add the system, unless you already saved changes.
- **Global browsing** – You can look at the systems in your local network on one screen.
- **Batching** – You can add, delete, and modify many AutoClient systems in one work session.
- **Progress/status indication** – At the bottom of the main menu is a display area that shows you how many systems have been added, deleted, or modified within a work session.

- **Viewing and scrolling capabilities** – Scroll bars enable easy viewing of system information. Host Manager also provides a search mechanism.
- **Viewing error messages** – If an error occurs during an operation, a pop-up window appears. You can also open the window manually from the View menu.

You can find more information on these features in Chapter 5, “Host Manager Reference Information,” and in Chapter 6, “Managing AutoClient Systems,” as these features pertain to individual tasks.

Note – This book focuses on using Host Manager to maintain AutoClient systems. For more information on other Host Manager functionality, use online help or see the *Solstice AdminSuite 2.2 Administration Guide*.

Command-Line Equivalents of Host Manager Operations

Table 2-4 lists the commands that provide the same functionality as Host Manager and can be used without running an X Window System™, such as the OpenWindows™ environment. Many of the tasks in Chapter 6, “Managing AutoClient Systems,” provide corresponding examples using the command-line equivalents.

Table 2-4 Command-Line Equivalents of Host Manager

Command	**Description**
`admhostadd`	Adds support for a new system or OS server.
`admhostmod`	Modifies an existing system or OS server. You can also add OS services to an existing OS server.
`admhostdel`	Deletes an existing system or OS server.
`admhostls`	Lists one or more system entries in the selected name service.
`admhostls -h`	Lists hardware information of one or more system entries in the selected name service.

Files Modified by Host Manager

Table 2-5 describes the system files that may be modified by Host Manager when adding and maintaining your AutoClient systems.

Table 2-5 Files Modified by Host Manager

System File	Where Modified	Description
`bootparams`	`/etc` files, NIS, or NIS+	A database listing the servers that provide the paths to a client's boot and installation software and a client's root and swap areas
`/etc/dfs/dfstab`	Server providing the file services	A file containing a series of `share` commands that make file resources available to the client system
`ethers`	`/etc` files, NIS, or NIS+	A database containing the client's Ethernet address
`hosts`	`/etc` files, NIS, or NIS+	A database containing the client's host name and associated IP address
`timezone`	`/etc` files, NIS, or NIS+	A database containing the client's time zone
`/export/root`	Server providing the file services	A default directory that contains root files for a diskless client or AutoClient system
`/export/swap`	Server providing the file services	A default directory that contains the swap file for a diskless client
`/var/sadm/softinfo`	Solaris 2.3 server providing OS services	A directory containing a list of OS services available on Solaris 2.3 server
`/var/sadm/system/admin/services`	Solaris 2.4 - 2.5.1 server providing OS services	A directory containing a list of OS services available on a Solaris 2.4 - 2.5.1 server
`/tftpboot`	Server providing the boot services	A directory containing SPARC and PowerPC client booting information
`/rplboot`	Server providing the boot services	A directory containing x86 client booting information
`/etc/inetd.conf`	Server providing the boot services	A system file that starts the `tftp` and `rpl` boot daemons
`cred.org_dir`	NIS+	A NIS+ table used to store the host's DES and LOCAL credentials

Using Solstice AutoClient in a Name Service Environment 3

The Solstice AutoClient software can be used in different name service environments. When you use each application or command-line equivalent, you must specify the name service environment data you wish to modify.

This is a list of the overview information in this chapter.

Available Name Service Environments

The Solstice AutoClient software can be used to manage information on the local system or across the network using a name service. The sources of information that can be managed by the Solstice AutoClient software are described in Table 3-1.

Table 3-1 Available Name Service Environments

Name Service	Select This Name Service To Manage ...
NIS+	NIS+ table information. This requires sysadmin group (group 14) membership and the appropriate ownership or permissions on the NIS+ tables to be modified.
NIS	NIS map information. You must be a member of the sysadmin group. If the NIS master server is running the Solaris 1.x OS Release, you must have explicit permissions on the NIS master server to update the maps. This means an entry for your host name and user name must reside in root's `.rhosts` file on the NIS master server. This entry is not required if the NIS master server is running the Solaris 2.x OS Release and the Name Services Transition Kit 1.2 software.
None	The `/etc` files on the local system. You must be a member of the sysadmin group on the local system.

See "Setting Up User Permissions to Use the Solstice AutoClient Software" on page 25 for information on using the Solstice AutoClient software with or without a name service environment.

The /etc/nsswitch.conf File and the Solstice AutoClient Product

The Solstice AutoClient software allows you to select which name service databases will be updated (written to) when you make modifications with Host Manager. However, the /etc/nsswitch.conf file on each system specifies the policy for name service lookups (where data will be read from) on that system.

Caution – It is up to the user to make sure that the name service they select from Host Manager is consistent with the specifications in the /etc/nsswitch.conf file. If the selections are not consistent, Host Manager may behave in unexpected ways, resulting in errors or warnings. See “Selecting a Name Service Environment” on page 24 for an example of the window from which you select a name service.

The /etc/nsswitch.conf file has no effect on how the system configuration files get updated. In the /etc/nsswitch.conf file, more than one source can be specified for the databases, and complex rules can be used to specify how a lookup can be performed from multiple sources. There is no defined syntax for using the rules in the /etc/nsswitch.conf file to perform updates.

Because of this, updates are controlled by the name service selection that is made when the Host Manager is started. The administrator must decide where the update is to take place.

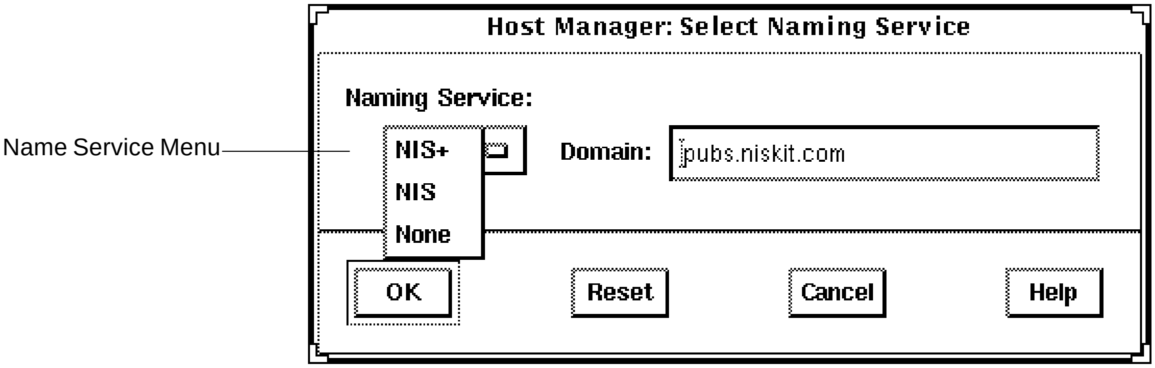
When using Host Manager, administrative operations can take place on multiple systems with a single operation. It is possible that each of these systems could have a different /etc/nsswitch.conf configuration. This situation can make it very difficult to administer your network. It is recommended that all of the systems have a consistent set of /etc/nsswitch.conf files and that the Solstice AutoClient software is used to administer the primary name service specified in the standard /etc/nsswitch.conf file.

With this release of the Solstice AutoClient product, you can define a more complex update policy for Host Manager by using the admtblloc command. For more information on this command, refer to the admtblloc(1M) man page and see “The admtblloc Command” on page 30.

Selecting a Name Service Environment

After you start the Solstice Launcher and click on an application icon, a window is displayed prompting you to select a name service. Select the name service that is appropriate for your environment.

This example is from Host Manager's Load window.



pubs.niskit.com

Working with the Name Services Transition Kit 1.2

The Name Services Transition Kit 1.2 is designed to allow you to support a NIS server running Solaris 2.x. Installing the software and setting up the Solaris 2.x NIS servers is described in the *Name Services Transition Kit 1.2 Administrator's Guide*. The Solstice AutoClient software can manage information using the NIS name service supported by Solaris 2.x NIS servers installed with the Name Services Transition Kit 1.2 software.

On NIS servers installed with the Solaris 2.x OS Release, the Name Service Transition Kit 1.2, and the Solstice AutoClient software, the configuration files stored in `/etc` directory are modified by the Solstice AutoClient applications (these files are in turn automatically converted to NIS maps). If the NIS server is not installed with the Solstice AutoClient software, then the directory location specified by the `$DIR` variable in the `/var/yp/Makefile` is used.

Setting Up User Permissions to Use the Solstice AutoClient Software

To use the Solstice AutoClient software, membership in the sysadmin group (group 14) is required. See “Adding Users to the sysadmin Group” on page 26 for more information.

Following are additional requirements to use the Solstice AutoClient software for each name service.

User Permissions in the NIS+ Environment

The requirements for using the Solstice AutoClient software are:

- Membership in the NIS+ admin group.
- Modify permissions on the NIS+ tables to be managed. These permissions are usually given to the NIS+ group members.

See *NIS+ and FNS Administration Guide* for information on adding users to a NIS+ group and granting permissions on NIS+ tables.

User Permissions in the NIS Environment

The requirements for using the Solstice AutoClient software are:

- An entry for your host name and user name in root’s `.rhosts` file on the NIS master server if the server is running the Solaris 1.x OS Release. If the NIS master server is running the Solaris 2.x OS Release and Name Services Transition Kit 1.2 software, this entry is not required.

- Running `ypbind` with the `-broadcast` option, which is the default form, if you want to manage NIS map information in domains other than your own.

Adding Users to the sysadmin Group

The following procedures describe how to add users to the sysadmin group for each name service. If you have access to the Solstice AdminSuite software, you should use Group Manager instead of these procedures to add users to the sysadmin group.

▼ How to Add a User to the sysadmin Group Using NIS+

1. **Log in to a system in your NIS+ domain as an authorized user with read and write access rights to the group table.**
2. **Save the group table to a temporary file.**

```
$ niscat group.org_dir > /var/tmp/group-file
```

3. **Edit the file, adding the users you want to authorize to use the Solstice AutoClient software.**

   The following sample shows users added to the sysadmin entry in the group file.

```
.
.
.
sysadmin::14:user1,user2,user3
nobody::60001:
noaccess::60002:
```

In this example,

user1,user2,user3 — Represent the user IDs you are adding to the sysadmin group.

4. **Merge the file with the NIS+ group table.**

```
$ /usr/lib/nis/nisaddent -mv -f /var/tmp/group-file group
```

The results of the merge are displayed.

5. **Remove the temporary file.**

```
$ rm /var/tmp/group-file
```

Verification of Adding Users to the sysadmin Group

Verify that the user is a member of the sysadmin group by entering the following commands. Perform this step for each user you added to the file.

```
# su - user1
$ groups
staff sysadmin
$ exit
```

▼ How to Add a User to the sysadmin Group Using NIS

1. **Log in as root on the NIS master server.**

2. **Edit the `group` file (the default directory location is `/etc`).**
   Add a comma-separated list of members to the sysadmin group.

```
.
.
.
sysadmin::14:user1,user2,user3
```

Note – The directory location of the `group` file is specified in the NIS `makefile` using the `$DIR` variable. Consult this file if you are uncertain of the location of the `group` file.

3. **Change directory to the location of the NIS `makefile` (the default is `/var/yp`) and remake the NIS map.**

```
# cd /var/yp
# make group
```

Note – Depending on the size of the NIS map, it may take several minutes or several hours to update the maps and propagate the changes throughout the network.

---

4. **(Optional) If the NIS master server is running the Solaris 1.x OS Release, create a `.rhosts` entry in the root (`/`) directory on the NIS master server for users authorized to modify NIS maps. Use the following format:**

```
host-name user-name
```

▼ How to Add a User to the sysadmin Group Without a Name Service

Use this procedure if you will use the Solstice AutoClient software on the local system only.

1. **Become root on your system.**

2. **Edit the `/etc/group` file.**
   Add a comma-separated list of members to the sysadmin group.

```
.
.
.
sysadmin::14:user1,user2,user3
```

Setting Up Solstice AutoClient Name Service Policy

A name service policy is used to specify the location of system and network information managed by the Solstice AutoClient software. This information can be located in the `/etc` directory for a local system, or in the NIS+ or NIS name service.

The Solstice AutoClient software supports a *mixed-mode* name service policy. A mixed-mode name service policy enables you to specify different name services for configuration information.

You can use the `admtblloc(1M)` command to choose a mixture of name services for the Solstice AutoClient tools to populate. For example, you can set up Host Manager to populate local `/etc` files for `bootparams` information and to populate the NIS+ tables for the other host configuration information, as shown in Figure 3-1.

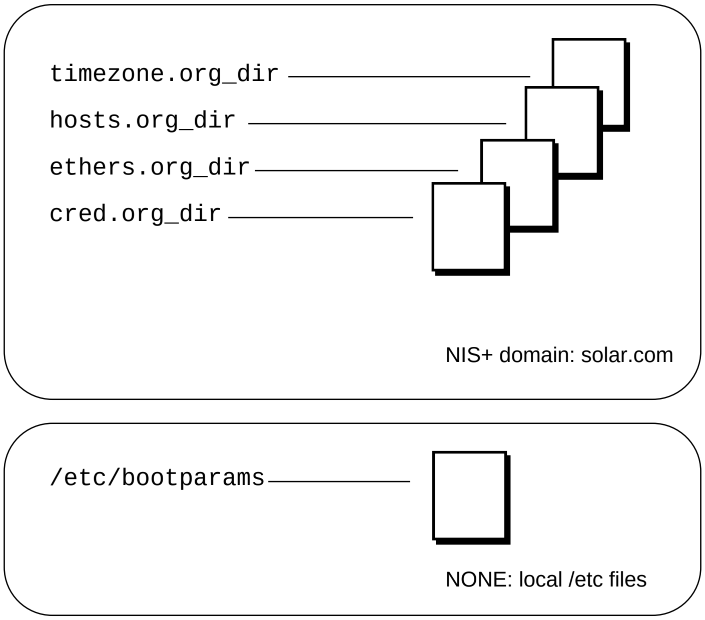


Figure 3-1 Example Mixed-Mode Name Service Policy

Caution – If you choose to implement a mixed-mode name service policy, you must run the Solstice AutoClient software from the system containing information in the `/etc` directory.

The `admtblloc` *Command*

The `admtblloc` command is used to implement a mixed-mode name service policy in the Solstice AutoClient software. To use this command, you must have permission to use the software for each name service as described in “Setting Up User Permissions to Use the Solstice AutoClient Software” on page 25.

Note – The `admtblloc` command has no relation to the `/etc/nsswitch.conf` file used to set the system-wide name service selection policy in the Solaris 2.x operating environment. The `admtblloc` command is used to set the policy for all users of the Solstice AutoClient software graphical user interface tools or command line interfaces.

Specifying the Name Service Policy Using `admtblloc`

This example shows how to specify the name service policy specified in Figure 3-1 using the `admtblloc` command:

```
$ admtblloc -c NIS+ -d solar.com bootparams NONE
```

In this example,

`- c NIS+ -d solar.com` The NIS+ domain `solar.com` is the name service *context* (the name service and domain name specified in the Load window).

`bootparams` `bootparams` is the configuration file to set the name service policy for.

`NONE`	`NONE` specifies that the host running the Solstice AutoClient tool or command line interface must use the `bootparams` file found in the local `/etc` directory.

After setting the mixed-mode name service policy specified in Figure 3-1, the Solstice AutoClient software will use the `bootparams` information stored in the `/etc` directory on the current host running the Solstice AutoClient tool whenever the name service (specified in the Load window) is NIS+. The name service policy for the other configuration files (`hosts`, `ethers`, `timezone` and `credential`) is NIS+, unless you specify otherwise using `admtblloc` again. The mixed-mode name service policy remains in effect for all users of the Solstice AutoClient software in the name service until you change it using the `admtblloc` command once again.

Note – If you specify that the name service location of a configuration file is NONE using the `admtblloc` command, the `/etc` file on the current host running the Solstice AutoClient application or command-line interface is modified. You should log in to the host where you want to use the local `/etc` file and perform operations using the Solstice AutoClient on that system.

Viewing the Name Service Policy Using admtblloc

This example shows how to display the name service policy using the admtblloc command:

```
$ admtblloc
Name           Name Service  Path

Aliases        NIS+
Hosts          NIS+
Group          NIS+
Netgroup       NIS+
Protocols      NIS+
Bootparams     NONE
Auto.home      NIS+
RPC            NIS+
Timezone       NIS+
Netmasks       NIS+
Ethers         NIS+
Passwd         NIS+
Services       NIS+
Networks       NIS+
Locale         NIS+
```

In this example output,

`Name` Is the name of the configuration file.

`Name Service` Specifies the name service used to access the configuration file.

`Path` (Optional) Specifies the path to the ASCII source file on NIS servers in the NIS name service. The default is the `/etc` directory.

By default, the admtblloc command displays the policy for the name service to which the current host belongs. To display the name service policy for a different name service, specify the name service context.

This example shows how to display the name service policy for the NONE or local /etc files name service context domain using the admtblloc command:

```
$ admtblloc -c NONE
Name           Name Service  Path
Aliases        NONE
Hosts          NONE
Group          NONE
Auto_home      NONE
Netgroup       NONE
Protocols      NONE
Bootparams     NONE
RPC            NONE
Timezone       NONE
Netmasks       NONE
Ethers         NONE
Passwd         NONE
Services       NONE
Networks       NONE
Locale         NONE
```

In this example,

-c — Specifies the name service context.

NONE — Is the local /etc files name service.

You can also use the admtblloc command to display the name service policy for a specified configuration file. This example shows how to display the name service policy for the hosts file in the default name service:

```
$ admtblloc Hosts
Hosts          NIS+
```

Note – The configuration file names are case-sensitive.

Configuration Files Supported by the `admtblloc` *Command*

Following is a list of the configuration files the Solstice AutoClient software can use in a mixed-mode name service environment.

- `Aliases`
- `Hosts`
- `Group`
- `Auto_home`
- `Credentials`
- `Netgroup`
- `Protocols`
- `Bootparams`
- `Rpc`
- `Timezone`
- `Netmasks`
- `Ethers`
- `Passwd`
- `Services`
- `Networks`
- `Locale`

Note – The `admtblloc` command can be used to set the name service policy for only the configuration files present in this list.

Refer to the `admtblloc(1M)` man page for more information about how to use this command.

Security 4

An important part of using the Solstice AutoClient software is understanding its security features and setting up security policies to protect your administrative data.

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

Security Information

The Solstice AutoClient software uses the distributed system administration daemon (`sadmind`) to carry out security tasks when you perform administrative tasks across the network. The `sadmind` daemon executes the request on the server on behalf of the client process and controls who can access the Solstice AutoClient software.

Administering security involves *authentication* of the user and *authorization* of permissions.

- Authentication means that the `sadmind` daemon must verify the identity of the user making the request.

- Authorization means that sadmind verifies that the authenticated user has permission to execute the Solstice AutoClient software on the server. After the user identity is verified, sadmind uses the user identity to perform authorization checks.

  If you have permission to use the Solstice AutoClient software, you also need to have create, delete, or modify permission before you can change an NIS+ map. See *NIS+ and DNS Setup and Configuration Guide* for a description of NIS+ security.

User and group identities are used for authorization checking as follows:

- **Root identity** – The root identity has privileges (to access and update data) only on the local system. If the server is the local system (in other words, if the user has logged in as root on the server), the user will be allowed to perform Solstice AutoClient functions on the server under the root identity.
- **User who is a member of sysadmin group (**group 14**)** – Solstice AutoClient permissions are granted to users who are members of the sysadmin group (group 14). This means that a user modifying administration data must be a member of the sysadmin group on the system where the task is being executed.

Security Levels

Each request to change administration data contains a set of credentials with a UID and a set of GIDs to which the user belongs. The server uses these credentials to perform identity and permission checks. Three levels of authentication security are available.

The security levels are described in Table 4-1.

Table 4-1 Solstice AdminSuite Security Levels

Level	Level Name	Description
0	NONE	No identity checking is done by the server. All UIDs are set to the `nobody` identity. This level is used mostly for testing.
1	SYS	The server accepts the original user and group identities from the client system and uses them as the identities for the authorization checks. There is no checking to be sure that the UID of the user represents the same user on the server system. That is, it is assumed the administrator has made the UIDs and GIDs consistent on all systems in the network. Checks are made to see if the user has permission to execute the request.
2	DES	Credentials are validated using DES authentication, and checks are made to be sure that the user has permission to execute the request. The user and group identities are obtained from files on the server system by mapping the user's DES network identity to a local UID and set of GIDs. The file used depends on which name service is selected on the server system. This level provides the most secure environment for performing administrative tasks and requires that a `publickey` entry exists for all server systems where the `sadmind` daemon is running, and for all users accessing the tools.

Note – Level 1 is the default security used by `sadmind`.

Changing the Security Level

You can change the security level from Level 1 to Level 2 by editing the `/etc/inetd.conf` file on each system, and adding the `–S 2` option to the `sadmind` entry. If you do this, make sure that the servers in the domain are set up to use DES security.

You do not need to maintain the same level of security on all systems in the network. You can run some systems, such as file servers requiring strict security, at security Level 2, while running other systems at the default Level 1 security.

See the description of how to set up security for NIS+ in *NIS+ and FNS Administration Guide*.

Name Service Information

The `sadmind` daemon uses information held by the name service. The three sources of information are:

- Files in the `/etc` directory such as `passwd`, `group`, and `shadow`, referred to as the keyword `files`
- The NIS name service referred to as the keyword `nis`
- The NIS+ name service referred to as the keyword `nisplus`

On each system, the `/etc/nsswitch.conf` file lists several administrative files, followed by a list of one or more keywords that represent the name services to be searched for information. If more than one keyword is listed, they are searched in the order given. For example, the entry

```
group:  files nisplus
```

indicates that the security mechanism looks first in the local `/etc/group` file for an entry. If the entry exists, the security mechanism uses the information in this entry. If the entry doesn't exist, the NIS+ `group` file is searched.

By default, systems running the Solaris 2.4 and higher OS release have an entry for `group 14` in the local `/etc/group` file. If you want to set up your system to use network-wide information, do not add members to the sysadmin group on the local system. Instead, update the group14 entry found in the group table stored in the name service.

When running under Level 2 security, the security mechanisms use the public/private key information. Make sure that the entry for `publickey` is followed by either `nis` or `nisplus` (depending on which name service you are using), and remove the `files` designation. See *NIS+ and FNS Administration Guide* for more information about the `nsswitch.conf` file.

Things to Consider When Creating a Security Policy

Consider the following when creating a security policy for using the Solstice AutoClient software in a name service environment.

- Determine how much trust is needed.

  If your network is secure and you do not need to use authentication security, you can use the Solstice AutoClient software with the default Level 1 security.

  If you need to enforce a higher level of security, you can set the security level of `sadmind` to Level 2.

- Determine which name service will be used.

  The name service determines where the security methods get information about user and group identities. The name services are designated in the `/etc/nsswitch.conf` file (see "Name Service Information" on page 38).

- Decide which users have access to the Solstice AutoClient software.

  Decide which users will perform administrative functions over the network with the Solstice AutoClient software. List these users as members of group14 accessed by the server system. The `group 14` must be accessible from each system where administration data will be updated by the Solstice AutoClient software. The `group 14` can be established locally on each system or can be used globally within a name service domain, depending upon the policy established by the administrator.

- Determine global and local policies.

  The *global policy* affects all hosts in the network. For example, you can add members to `group 14` in the NIS or NIS+ `group` file. Members of this group will have permission to perform administrative tasks on all server systems that list the network name service as the primary source of information. The name services are listed in the `/etc/nsswitch.conf` file. For more information about the `nsswitch.conf` file, see "Name Service Information" on page 38.

  A user can establish a local policy that is different from the global policy by creating a `group 14` in the local `/etc/group` file and listing the users who have access to the local system. The members of this group will have permission to manipulate or run the Solstice AutoClient software methods on the user's local system.

Note – Setting up a local policy does not disable a global policy. Name service access is determined by the nsswitch.conf file.

- Set up permissions for NIS+ management.

  You need the proper permissions when using the Solstice AutoClient software to modify or update the NIS+ files. In addition to the permissions required by the Solstice AutoClient software, the NIS+ security mechanisms impose their own set of access permissions. The NIS+ security mechanisms are described in *NIS+ and FNS Administration Guide*.

- Set up access for NIS management.

  If the NIS master server is running the Solaris 1.x operating system, a user must have a .rhosts entry on the NIS master server to modify the NIS files. If the NIS master server is running the Solaris 2.x operating system and the Name Services Transition Kit 1.2, then no entry is required. The NIS updates will be authorized using the standard group 14 mechanism.

Creating a Level 2 DES Security System

Creating a level 2 DES security system requires a number of steps that depend upon your system configuration. The following sections describe how to set up your system to have level 2 DES security for systems using `/etc`, `NIS`, and `NIS+` name services.

▼ How to Create Level 2 DES Security for Systems Using `/etc` Name Service

1. **On each system that runs the** `sadmind` **daemon, edit the** `/etc/inetd.conf` **file.**
   Change this line (or one similar to this):

```
100232/10 tli  rpc/udp wait root /usr/sbin/sadmind sadmind
```

   to:

```
100232/10 tli  rpc/udp wait root /usr/sbin/sadmind sadmind -S 2
```

2. **On each system that runs the sadmind daemon, set the** `/etc/nsswitch.conf` **entry for** `publickey` **to** `files`**.**
   Change this entry (or one similar to this):

```
publickey:nis [NOTFOUND=return] files
```

   to:

```
publickey:files
```

3. **Create credentials for all** `group 14` **users and all of the systems that will run** `sadmind -S 2`**.**

   a. **Log in as root to one of the systems that will run** `sadmin -S 2`**.**

b. Run the following command for each user that will run admintool**.**

```
# newkey -u username
```

Note – You must run this command even for users who are not in group 14. If you are not in group 14 and do not have credentials, you are not a user according to sadmind; you will not be able to run any methods, even those that do not require root. You will have to supply the user's password to the newkey program.

c. Run the following command for every host that you have configured to run secure sadmind**.**

```
# newkey -h hostname
```

You will have to provide the root password for each of these hosts to the newkey program.

d. Copy the /etc/publickey **file on this system to each of the hosts (put this file in** /etc/publickey**).**
This file contains all the credentials for each user and each host.

Note – Do not run newkey on each of the systems. This seems to create a different public/private key pair, and the public key will not be valid across the network. You must create this file on one machine and then copy it to all the others.

e. As root, enter the following command on each system to put root's private key in /etc/.rootkey**.**

```
# keylogin -r
```

By doing this, you will not have to keylogin as root on every system every time you want to run admintool; this creates an automatic root keylogin at boot time.

4. **Create an `/etc/netid` file for each user and each system; put this file on all of the systems.**

   a. **For each user in the `publickey` file, create an entry in `/etc/netid` that looks like the following:**

   ```
   unix.uid@domainnameuid:  uid: gid,gid, ...
   ```

   b. **List every group that this user is a member of; `sadmind -S 2` and files look to `netid` rather than `/etc/group` to determine `group 14` membership.**

   c. **For each host in the `publickey` file, create an entry in `/etc/netid` that looks like the following:**

   ```
   unix.hostname@domainname                    0:hostname
   ```

   d. **Copy this file to every system in `/etc/netid`.**

5. **Reboot all of the machines.**

6. **On each system that you want to run the application on, log in and then `keylogin`. (You must be a member of `group 14`.)**
   After the `keylogin`, you can safely log out; your key is stored in the `keyserv` daemon until you explicitly `keylogout` or the system reboots.

▼ How to Create Level 2 DES Security for Systems Using NIS Name Service

1. **On each system that runs the** `sadmind` **daemon, edit the** `/etc/inetd.conf` **file.**
   Change this line (or one similar to this):

```
100232/10 tli  rpc/udp wait root /usr/sbin/sadmind sadmind
```

   to:

```
100232/10 tli  rpc/udp wait root /usr/sbin/sadmind sadmind -S 2
```

2. **On each system that runs the** `sadmind` **daemon, set the** `/etc/nsswitch.conf` **entry for** `publickey` **to** `nis`**.**
   Change this entry (or one similar to this):

```
publickey:nis [NOTFOUND=return] files
```

   to:

```
publickey:nis
```

3. **Create credentials for all** `group 14` **users and all of the systems that will run** `sadmind -S 2`**.**

   a. **Log in as root on the nis server.**

   b. **Run the following command for each user that will run** `admintool`**.**

```
# newkey -u username -s files
```

Note – You must run this command even for users who are not in group 14. If you are not in group 14 and do not have credentials, you are not a user according to sadmind; you will not be able to run any methods, even those that do not require root. You will have to supply the user's password to the newkey program.

c. **Run the following command for every host that you have configured to run secure** sadmind**.**

```
# newkey -h hostname
```

You will have to provide the root password for each of these hosts to the newkey program.

d. **Copy the** /etc/publickey **file on this system to the source file that is specified in** /var/yp/Makefile; **remake and push the** nis **maps.**

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

4. **Verify that you are a member of** group 14 **in the group/nis maps.**

a. **Login as root.**

b. **Change directories to the source file specified in** /var/yp/Makefile**.**

c. **Manually edit the group file and add yourself to Group14, just as you did in the** /etc/group **file.**

d. **Change directories to** /var/yp **and run** make**.**

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

You should see the group map pushed; a message appears indicating that this action has occurred.

Note – The security system looks in the NIS maps for your group14 access and will fail if you do not have group14 specified there, regardless if your /etc/nsswitch.conf file has group files nis.

When sadmind is running in -S 2 mode, it uses the publickey entry to determine which name service to look at for user credentials. When the entry in /etc/nsswitch.conf is nis, it looks in the nis group map to ensure that the user is a member of group 14.

5. **As root, enter the following command on each system to put root's private key in** /etc/.rootkey**.**

```
# keylogin -r
```

   By doing this, you will not have to keylogin as root on every system every time you want to run admintool; this creates an automatic root keylogin at boot time.

6. **Reboot all of the workstations; verify that the** nscd **gets flushed.**

7. **On each system that you want to the application to run on, log in and then** keylogin**. (You must be a member of** group 14**.)**
   After the keylogin, you can safely log out; your key is stored in the keyserv daemon until you explicitly keylogout or the system reboots.

▼ How to Create Level 2 DES Security for Systems Using NIS+ Name Service

1. **On each system that runs the** sadmind **daemon, edit the** /etc/inetd.conf **file.**
   Change this line:

```
100232/10 tli  rpc/udp wait root /usr/sbin/sadmind sadmind
```

   to:

```
100232/10 tli  rpc/udp wait root /usr/sbin/sadmind sadmind -S 2
```

2. **On each system that runs the** `sadmind` **daemon, set the** `/etc/nsswitch.conf` **entry for** `publickey` **to** `nisplus`**.**
   Change this entry (or one similar to this):

   ```
   publickey:nisplus [NOTFOUND=return] files
   ```

   to:

   ```
   publickey:nisplus
   ```

3. **Log in as root on the NIS+ master server; create credentials for all** `group 14` **users and all of the systems that will run** `sadmind -S 2`**.**

   a. **Create local credentials for the user.**

   ```
   # nisaddcred -p uid username.domainname. local
   ```

   b. **Create** `des` **credentials for the user.**

   ```
   # nisaddcred -p unix.uid@domainname -P username.domainname. des
   ```

4. **Log in as root on the NIS+ master server; add all of the users for the** `admintool` **to the NIS+** `group 14` **using the following command.**

```
# nistbladm -m members=username, username...[name-sysadmin], group.org_dir
```

Note – The use of this function replaces the current member list with the one that is input; therefore, you must include all members you wish to be a part of `group 14`.

5. **As root, add all of the users for the** `admintool` **to the** `NIS+ admin` **group.**

   ```
   # nisgrpadm -a admin username
   ```

   Verify that the `NIS_GROUP env` variable is set to `admin`.

6. **On all the workstations that you intend to run the** `admintool`**, enter the following command.**

   ```
   # keylogin -r
   ```

7. **Reboot all of the workstations; verify that the** `nscd` **gets flushed.**

8. **On each system that you want to the application to run on, log in and then** `keylogin`**. (You must be a member of** `group 14`**.)**
   After the `keylogin`, you can safely log out; your key is stored in the `keyserv` daemon until you explicitly `keylogout` or the system reboots.

Host Manager Reference Information

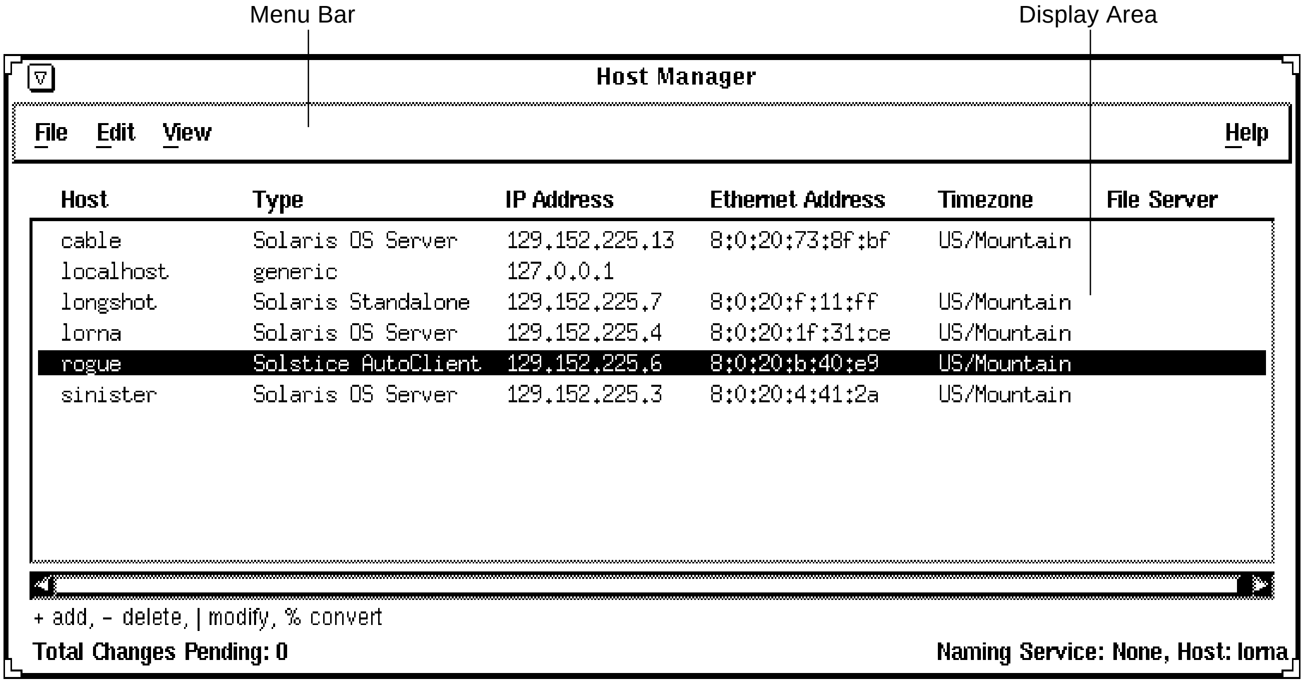
5

This chapter contains reference information for features found in Host Manager.

This is a list of the overview information in this chapter.

Main Window Areas

When you select the Host Manager icon in the Solstice Launcher, the Host Manager's main window is displayed. The areas in the Host Manager's main window are shown in Figure 5-1.



Host	Type	IP Address	Ethernet Address	Timezone	File Server
cable	Solaris OS Server	129.152.225.13	8:0:20:73:8f:bf	US/Mountain	
localhost	generic	127.0.0.1			
longshot	Solaris Standalone	129.152.225.7	8:0:20:f:11:ff	US/Mountain	
lorna	Solaris OS Server	129.152.225.4	8:0:20:1f:31:ce	US/Mountain	
rogue	Solstice AutoClient	129.152.225.6	8:0:20:b:40:e9	US/Mountain	
sinister	Solaris OS Server	129.152.225.3	8:0:20:4:41:2a	US/Mountain	

+ add, – delete, | modify, % convert

Total Changes Pending: 0

Naming Service: None, Host: lorna

Figure 5-1 Host Manager Main Window Areas

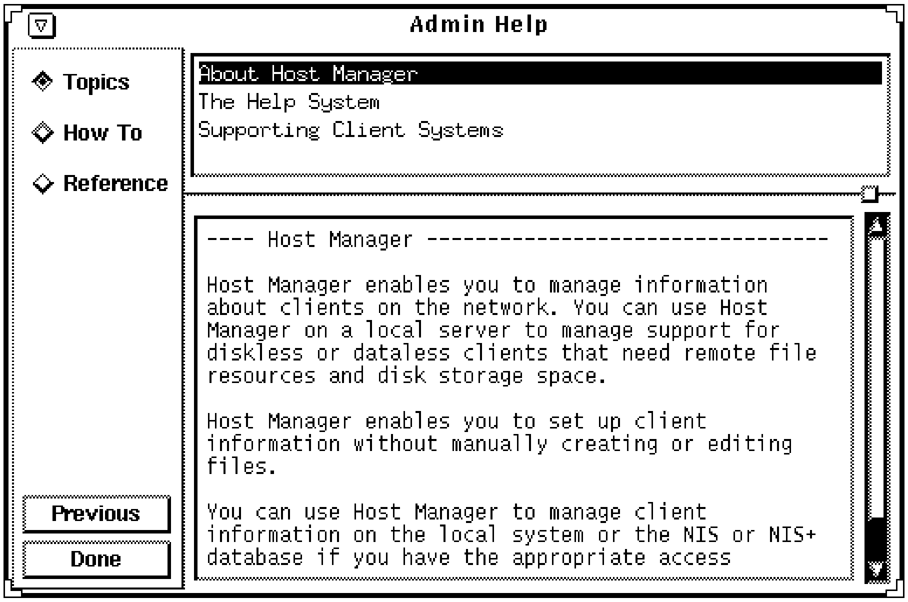
The main window contains two areas: a menu bar and a display area. The menu bar usually contains four menus: File, Edit, View, and Help. For more information on these menus, see the online help (the section "Using Admin Help" on page 51 describes how to access online help).

Using Admin Help

An important part of the Solstice AutoClient software is a Help utility called Admin Help. Admin Help provides detailed information about Host Manager and its functions.

- To access Admin Help from the Host Manager main window, choose "About Host Manager" from the Help menu.
- To access the online help from a command window, click on the Help button.

Figure 5-2 shows the Admin Help window.



About Host Manager
The Help System
Supporting Client Systems

```
---- Host Manager ---------------------------------

Host Manager enables you to manage information
about clients on the network. You can use Host
Manager on a local server to manage support for
diskless or dataless clients that need remote file
resources and disk storage space.

Host Manager enables you to set up client
information without manually creating or editing
files.

You can use Host Manager to manage client
information on the local system or the NIS or NIS+
database if you have the appropriate access
```

Figure 5-2 Admin Help Window

The titles displayed in the top window pane identify the list of topics available for each level of help.

The text displayed in the bottom window pane describes information about using the current menu or command.

Use the scroll bars to the right of each pane to scroll through the help information displayed.

On the left side of the Admin Help window are buttons used to find information and navigate through the help system. The buttons are described in Table 5-1.

Table 5-1 Admin Help Buttons

This Button ...	Is Used To ...	Notes
Topics	Displays a list of overview topics.	Click on a title in the top window pane to view the accompanying help text.
How To	Displays a list of step-by-step procedures.	Click on a title in the top window pane to view the accompanying help text.
Reference	Displays a list of more detailed information.	Click on a title in the top window pane to view the accompanying help text.
Previous	Returns to the last accessed help text.	The help viewer automatically returns to the previous help selection.
Done	Exits the help system.	The Admin Help window is closed.

Filtering System Entries

To view specific system entries in Host Manager's main window, choose Set Filter from the View menu. The Filter window is displayed and you have the option of setting from one to three filtering characteristics, as shown in Figure 5-3.

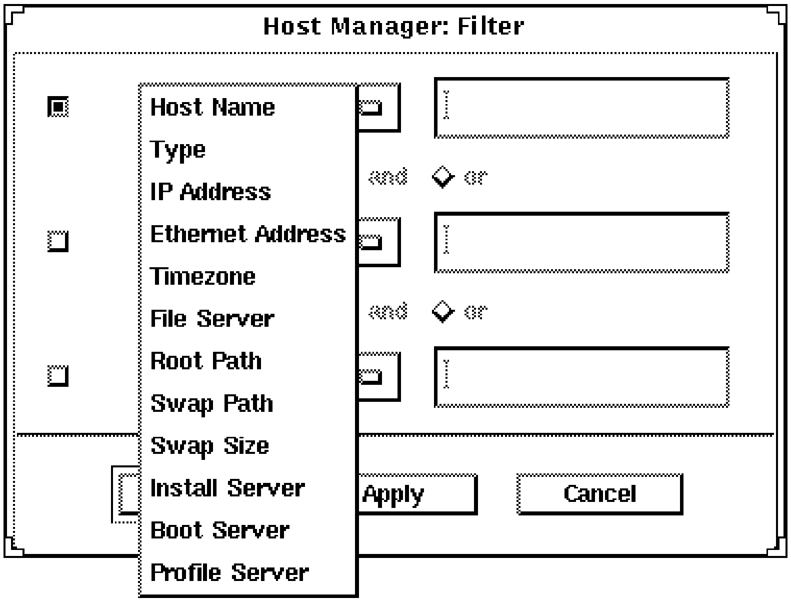


Figure 5-3 Filtering System Entries With Host Manager

After you have chosen a method for filtering the entries that are displayed in the main window, click on OK.

Buttons

Table 5-2 describes the common window buttons used in Host Manager.

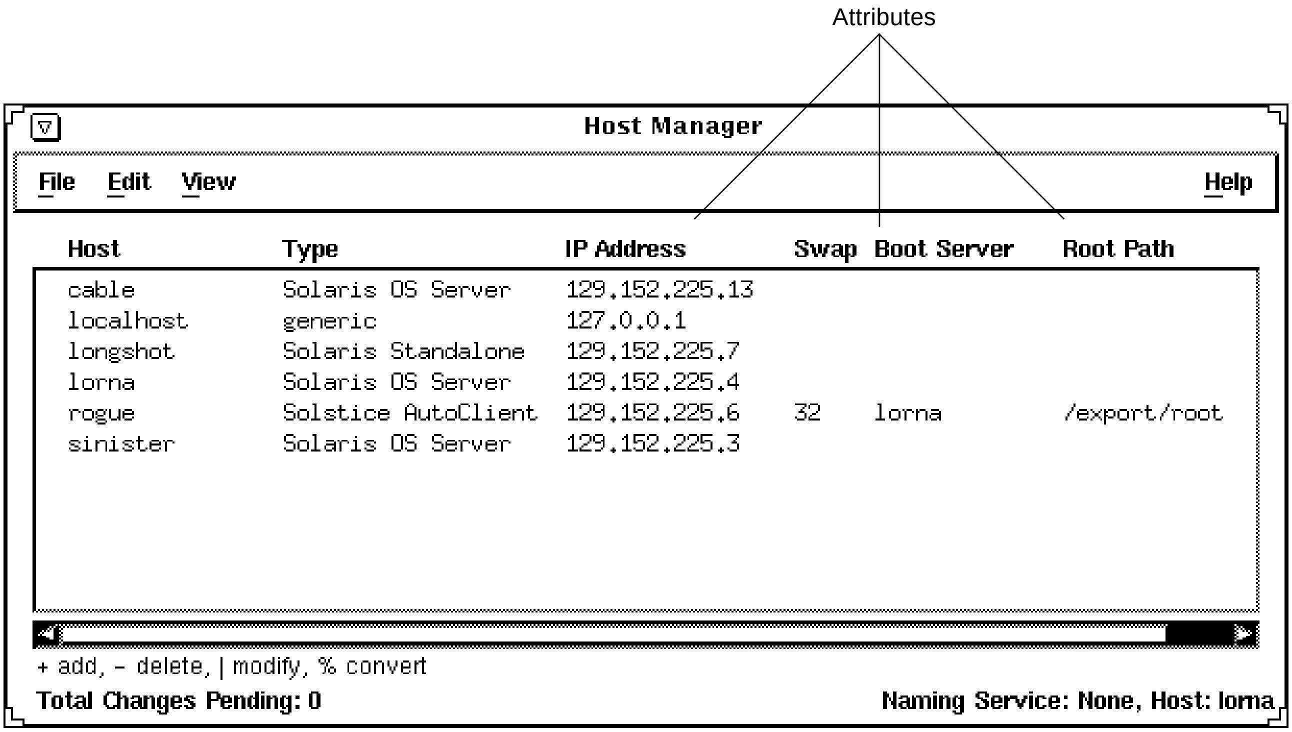
Table 5-2 Common Window Buttons in Host Manager

This Button ...	Is Used To ...
OK	Complete a task so that it can be processed. The window is closed after the task is completed.
Apply	Complete a task but leave the window open. (Not available on all windows.)
Reset	Reset all fields to their original contents (since the last successful operation).
Cancel	Cancel the task without submitting any changes and close the window. Fields are reset to their original contents.
Help	Access Admin Help.

Caution – Clicking on OK after clicking on Apply might cause a duplicate operation, resulting in an error. Click on Cancel after clicking on Apply to dismiss the window.

Global Browsing Capabilities

Host Manager enables you to see most system attributes in the main window, shown in Figure 5-4. Choose Customize from the View menu to change your attribute viewing options.



Attributes

Host	Type	IP Address	Swap	Boot Server	Root Path
cable	Solaris OS Server	129.152.225.13			
localhost	generic	127.0.0.1			
longshot	Solaris Standalone	129.152.225.7			
lorna	Solaris OS Server	129.152.225.4			
rogue	Solstice AutoClient	129.152.225.6	32	lorna	/export/root
sinister	Solaris OS Server	129.152.225.3			

+ add, - delete, | modify, % convert

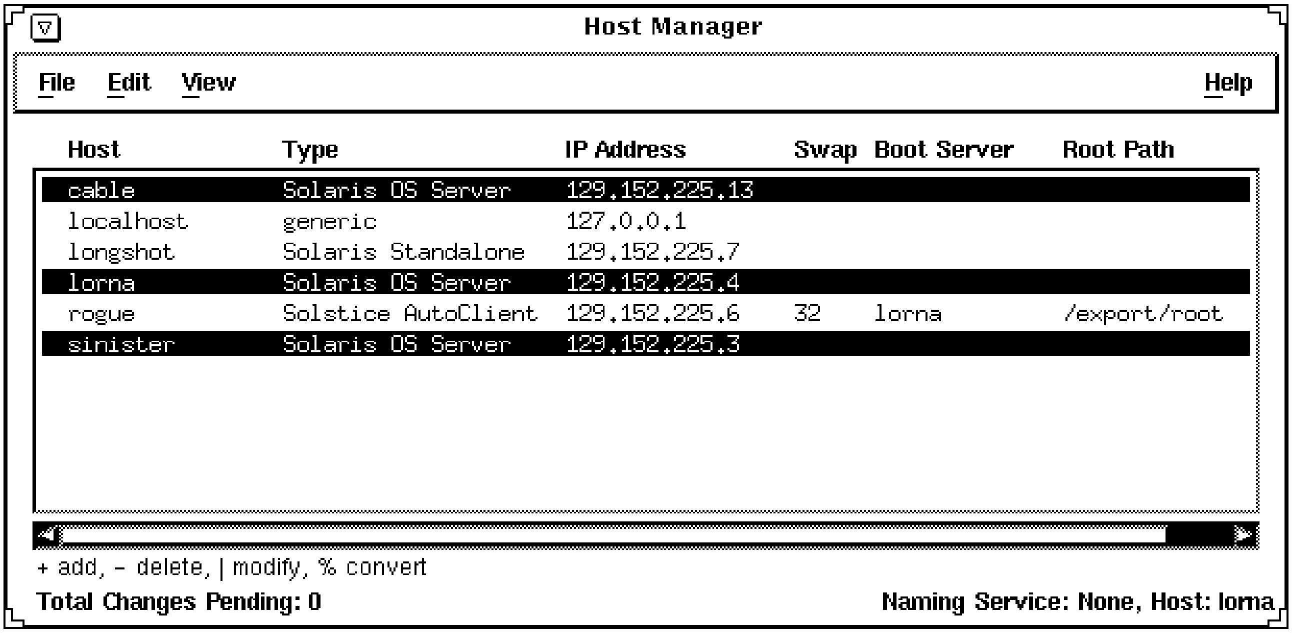
Total Changes Pending: 0

Naming Service: None, Host: lorna

Figure 5-4 Global Browsing Capabilities With Host Manager

Batching Operations

Host Manager enables you to add, delete, modify, convert, and revert more than one system at the same time, which is called *batching.* The scrolling and highlighting capabilities of the main window enable you to select multiple systems, as shown in Figure 5-5. To select more than one system, click SELECT (by default, the left mouse button) on the first system. Then select each subsequent system by pressing the Control key and clicking SELECT.



Host	Type	IP Address	Swap	Boot Server	Root Path
cable	Solaris OS Server	129.152.225.13			
localhost	generic	127.0.0.1			
longshot	Solaris Standalone	129.152.225.7			
lorna	Solaris OS Server	129.152.225.4			
rogue	Solstice AutoClient	129.152.225.6	32	lorna	/export/root
sinister	Solaris OS Server	129.152.225.3			

+ add, - delete, | modify, % convert

Total Changes Pending: 0

Naming Service: None, Host: lorna

Figure 5-5 Selecting Multiple Entries Within Host Manager

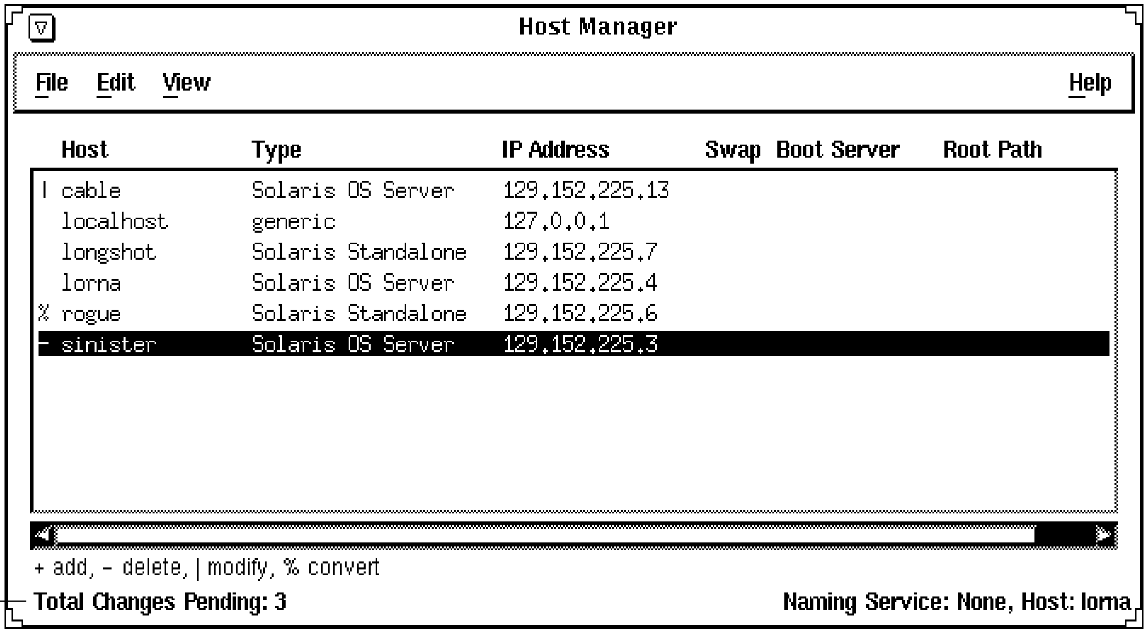
See Chapter 6, “Managing AutoClient Systems,” for information on completing add, delete, modify, convert, and revert operations.

Status Area

“Main Window Areas” on page 50 describes two areas of Host Manager’s main window: a menu bar area and a display area. The Host Manager main window also has a status area in the bottom of the window, which is shown in Figure 5-6.

In the left corner, the status area displays status information about pending changes, such as how many systems are waiting to be added, deleted, modified, and converted. In the right corner, the status area displays the current name service you are modifying with Host Manager.

The message “Total Changes Pending” reflects the number of systems that are waiting to be added, deleted, modified, and converted when you choose Save Changes from the File menu. After you choose “Save Changes” from the File menu, this message changes to “All Changes Successful.” If any changes did not succeed, a message is written to the Errors pop-up window.



```
  Host          Type                 IP Address       Swap  Boot Server    Root Path
| cable         Solaris OS Server    129.152.225.13
  localhost     generic              127.0.0.1
  longshot      Solaris Standalone   129.152.225.7
  lorna         Solaris OS Server    129.152.225.4
% rogue         Solaris Standalone   129.152.225.6
- sinister      Solaris OS Server    129.152.225.3
```

+ add, – delete, | modify, % convert

Total Changes Pending: 3

Naming Service: None, Host: lorna

Status Area

Figure 5-6 Status Information Within Host Manager

Logging Host Manager Operations

You can set up a log file to record each major operation completed with Host Manager or its command-line equivalents. After you enable logging, the date, time, server, user ID (UID), and description for every operation are written to the specified log file.

You need to follow the procedure described in “How to Enable Logging of Host Manager Operations” on page 58 on each server where you run the Host Manager and want to maintain a logging file.

▼ How to Enable Logging of Host Manager Operations

You do not need to quit Host Manager or the Solstice Launcher, if they are already started.

1. **Become root.**
2. **Edit the** `/etc/syslog.conf` **file and add an entry at the bottom of the file that follows this format:**

   ```
   user.info filename
   ```

   Note that *filename* must be the absolute path name of the file, for example: `/var/log/admin.log`.

3. **Create the file,** ***filename*****, if it does not already exist:**

   ```
   # touch filename
   ```

4. **Make the changes to the** `/etc/syslog.conf` **file take effect by stopping and starting the syslog service:**

```
# /etc/init.d/syslog stop
Stopping the syslog service.
# /etc/init.d/syslog start
syslog service starting.
#
```

Solstice AdminSuite operations will now be logged to the file you specified.

Example of a Host Manager Log File

```
Nov 30 10:34:23 lorna Host Mgr: [uid=100] Get host prototype
Nov 30 10:34:52 lorna Host Mgr: [uid=100] Adding host: frito
Nov 30 10:35:37 lorna Host Mgr: [uid=100] Get host prototype
Nov 30 10:35:59 lorna Host Mgr: [uid=100] Deleting host frito
Nov 30 10:36:07 lorna Host Mgr: [uid=100] Modifying sinister with
sinister
Nov 30 14:39:21 lorna Host Mgr: [uid=0] Read hosts
Nov 30 14:39:43 lorna Host Mgr: [uid=0] Get timezone for lorna
Nov 30 14:39:49 lorna Host Mgr: [uid=0] Get host prototype
Nov 30 14:40:01 lorna Host Mgr: [uid=0] List supported
architectures for lorna dirpath=/cdrom/cdrom0/s0
```

Part 2 — Setting Up and Maintaining AutoClient Systems

This part provides instructions on setting up and maintaining AutoClient systems. This part contains these chapters.

6 **Managing AutoClient Systems**
Provides instructions for how to set up AutoClient systems using Host Manager and describes how to add AutoClient support (that is, OS services) to a server.

7 **Booting a System From the Network**
Provides instructions on how to manually boot your AutoClient systems from the network and how to set them up to automatically boot from the network.

8 **AutoClient Environment Maintenance**
Provides instructions for how to update your AutoClient systems' caches with their back file system, replace faulty AutoClient systems, log Host Manager operations, and patch AutoClient systems.

Managing AutoClient Systems 6

This chapter describes how to use the Host Manager application to perform specific tasks for managing AutoClient systems in your network. The overall process includes:

- Making additions/changes to your network
- Viewing additions/changes on the Host Manager main window
- Saving changes

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

Note – This book focuses on using Host Manager to maintain AutoClient systems. For more information on other Host Manager functionality, use online help or see the *Solstice AdminSuite 2.2 Administration Guide*.

6

Starting Host Manager

Prerequisites

Be sure your network meets all the requirements identified in the *Solstice AutoClient 2.0 Installation and Product Notes,* and that you have completed the installation tasks described in the *Solstice AutoClient 2.0 Installation and Product Notes.* These tasks are summarized here:

- You have a system running the appropriate Solaris 2.x software.
- You have a bit-mapped display monitor connected to the system you are using, or you have the DISPLAY environment variable set to an appropriate display system.
- Your system is running an X Window System.
- You have the required access privileges such as root access to the local system or membership in group 14 (sysadmin group).
- You have the necessary name service permissions if you are using a name service.
- You have installed the Solstice AutoClient 2.0 license on the license server.

▼ How to Start Host Manager

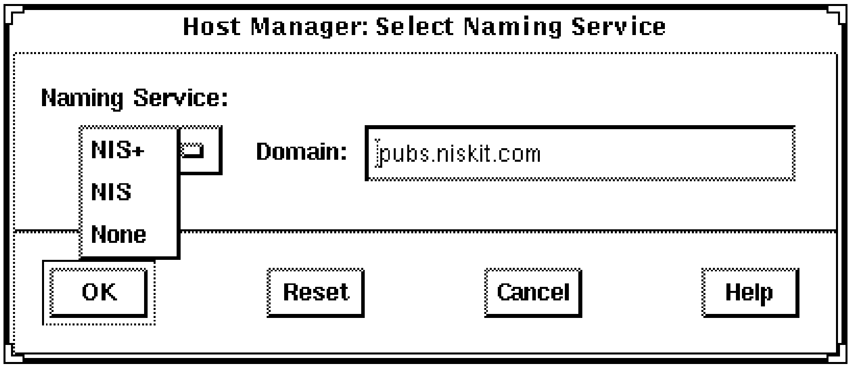
1. **Verify that the prerequisites summarized in “Prerequisites” on page 64 are met.**

2. **On the AutoClient server, type the following command to start the Solstice Launcher.**

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

3. **Click on the Host Manager icon.**

The Host Manager Select Naming Service window is displayed. If you are using a name service, it shows the server's domain name, or if you are using local files, the system name is displayed.



4. **Choose a name service and click on OK.**

Note – You should choose the appropriate name service based on your site policy. For more information on setting up a name service policy, see Chapter 3, "Using Solstice AutoClient in a Name Service Environment."

If you choose NIS or NIS+ as your Naming Service, and then type a different domain name in the Domain field, the system you are running Host Manager on needs to have permission to access the specified domain.

You will see a message box telling you that the software is gathering the system data.

6

Supporting AutoClient Systems

A Solaris *OS server* is a server that provides OS services to support AutoClient systems that have a kernel architecture different from the server's kernel architecture. For example, if a server with a Sun4c kernel architecture needs to support an AutoClient system with a Sun-4™ kernel architecture, client support for the Sun-4 kernel architecture must be added to the server.

Note – When using Host Manager to set up and maintain AutoClient systems, the AutoClient server is the file server and OS server for the AutoClient systems.

To support clients of a different platform group, or clients that require the same or a different Solaris release than the OS server, you must add the particular OS service to the OS server. You must have the appropriate Solaris CD image to add OS services.

For example, if you have an OS server running Solaris 2.4 and you want it to support diskless clients running Solaris 2.5, you must add the Solaris 2.5 OS services to the OS server. In this example, both the server and client are the same platform group, but they are running different versions of the OS.

▼ How to Add OS Services to an OS Server

Note – This procedure assumes that the AutoClient server is already set up to be an OS server. For information on adding an OS Server or converting an existing system to an OS Server, see the online help or the *Solstice AdminSuite 2.2 Administration Guide.*

1. **Start Host Manager from the Solstice Launcher and select the name service, if not done already.**
   See "Starting Host Manager" on page 64 for more information.
2. **Select the OS server to which you want to add services from the Host Manager main window.**
3. **Choose Modify from the Edit menu.**
   The Modify window is displayed.

4. **Click on Add in the OS Services window to add services.**
   If this is the first time you have added services in the current Host Manager session, the Set Media Path window is displayed, so continue with Step 5. If you have already added services in the current Host Manager session, the Add OS Services window is displayed, so skip to Step 7.

5. **Fill in the Set Media Path window.**
   After choosing the system containing the Solaris CD image, which must be minimally set up as a managed system, complete the remaining fields as shown in Table 6-1.

Table 6-1 Setting the Media Path

If You Are Using ...	And ...	Then Enter the Path ...
The Solaris CD as the Solaris CD image	The Solaris CD is managed by Volume Management	`/cdrom/cdrom0` or `/cdrom/cdrom0/s0` or `/cdrom/cdrom0/s2`
	The Solaris CD is not managed by Volume Management	Where you mounted the Solaris CD
A copy of the Solaris CD on the install server's hard disk (set up by using the `setup_install_server` command)		To the Solaris CD image

6. **Click on OK.**
   The Add OS Services window is displayed.

7. **(Optional) Click on Set Path to change the path to the Solaris CD image from which to add the client services.**
   If you previously entered a media path, the software will use this path as the default. If the path is incorrect, you need to complete this step.

8. **Choose the distribution type.**
   The default distribution type is Entire Distribution.

9. **Select a service you want to add and click on Add.**
   The Add OS Services window closes. If you want to add more services, repeat Step 4 through Step 9.

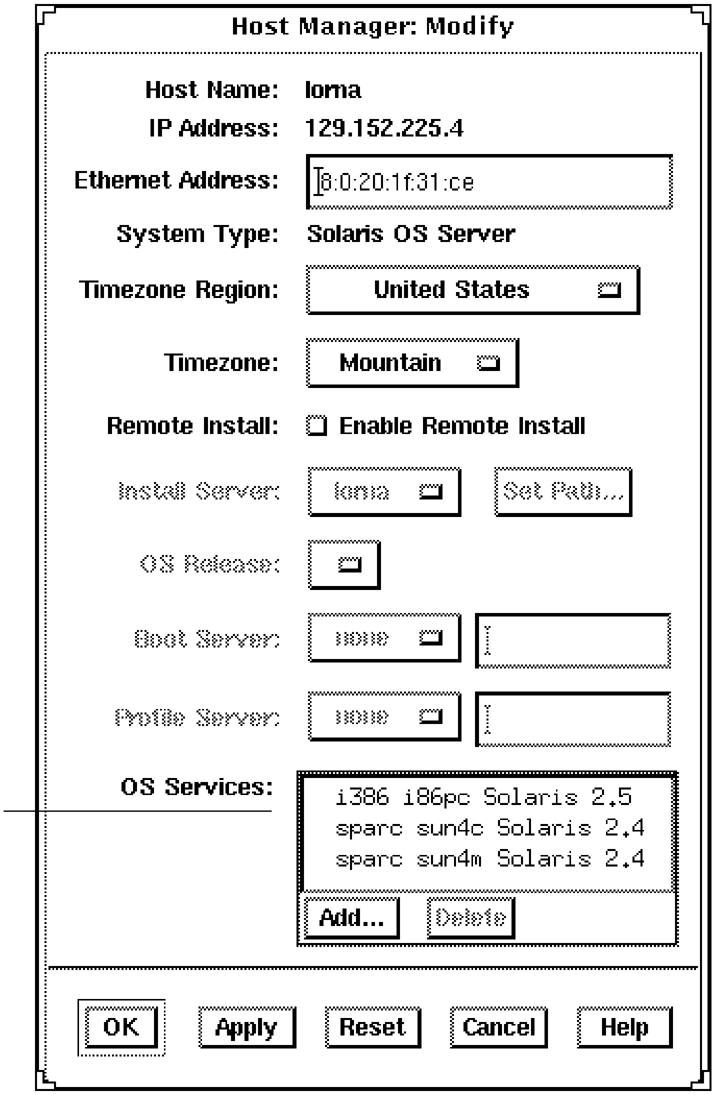
Note – Once you have successfully added OS services to an OS server, they cannot be deleted using Host Manager.

10. **Click on OK.**
    The Modify window closes.

11. **Choose Save Changes from the File menu to add services.**

Example of a Completed Add OS Services Window

The following example shows a completed Modify window for an OS server, lorna, where services are being added (see the OS Services field).



Host Manager: Modify

Host Name: lorna

IP Address: 129.152.225.4

Ethernet Address: 8:0:20:1f:31:ce

System Type: Solaris OS Server

Timezone Region: United States

Timezone: Mountain

Remote Install: □ Enable Remote Install

Install Server: lorna

OS Release:

Boot Server: none

Profile Server: none

OS Services:

```
i386 i86pc Solaris 2.5
sparc sun4c Solaris 2.4
sparc sun4m Solaris 2.4
```

List of OS services for each platform, platform group, and OS Release

Verification

To verify that all the OS services have been added, make sure the status line at the bottom of the main window says "All changes successful."

Example of a Command-Line Equivalent for Adding Services to an OS Server

The following command is equivalent to using Host Manager to add OS services to an OS server.

```
% admhostmod -x mediapath=jupiter:/cdrom/cdrom0/s0 \
-x platform=sparc.sun4c.Solaris_2.5 lorna
```

In this command,

`-x mediapath=jupiter:/cdrom/cdrom0/s0` Specifies that the Solaris CD image is on a mounted CD on a remote system named `jupiter`. Note that the remote system must be minimally set up as a managed system.

`-x platform=sparc.sun4c.Solaris_2.5` Specifies the services to be installed; in this case, the Solaris 2.5 services for a SPARC Solaris, Sun4c kernel architecture.

`lorna` Specifies the name of the OS server.

Adding AutoClient Systems

The procedure in this section explains how to add individual or multiple AutoClient systems to a server. When you add AutoClient systems to the server, the systems themselves may be up and running or powered down.

You will be required to provide the information shown in Table 6-2 when adding an AutoClient system to your network.

Table 6-2 Fields on the Add Window for the Solstice AutoClient System Type

Field Name	**Default/Specifications**
Host Name	No default. 1 to 255 alphanumeric characters. You can also use dashes, underscores, or periods. Do not begin or end the host name with a dash.
IP Address	No default. Enter an IP address in the form of *n.n.n.n*, where *n* is any number from 0 to 255. It must be a valid class A, B, or C IP address.
Ethernet Address	No default. Enter a hexadecimal Ethernet address in the form of *n:n:n:n:n:n* where *n* is 00 to ff. Valid characters are 0-9, a-f, and A-F.
System Type	The System Type should be Solstice AutoClient.
Timezone Region	The default is the server's time zone region.
Timezone	The default is the server's time zone.
File Server	The default is the server specified in the Set Defaults window. If none is specified, the local system is the default. For more information on setting defaults for Host Manager, see the *Solstice AdminSuite 2.2 Administration Guide* or the online help.
OS Release	The default is the OS release specified in the Set Defaults window.
Root Path	The default is the root path specified in the Set Defaults window.

Table 6-2 Fields on the Add Window for the Solstice AutoClient System Type *(Continued)*

Field Name	Default/Specifications
Swap Size	The default is the size specified in the Set Defaults window.
Disk Config	The default is 1disk. See Table 6-3 on page 73 for disk configuration options. Do not assume you can use the default. You must make sure that the disk configuration you choose is correct for this system.
Disconnect-able	The default is that the disconnectable feature is disabled, which means that users cannot use their cached file systems if the server is unavailable. Turning the disconnectable feature on (enabling disconnectability) means that when the AutoClient system's server is unavailable, users can continue to use their cached file system, and will not see "NFS server not responding" error messages. The AutoClient system must be running the Solaris 2.5 or later software.

Note – The swap size default is the minimum amount of swap created. It is possible that you will have more swap space than you requested. If you choose 2disks as your configuration option, the entire second disk is used for swap. Always leave swap size at its default value if you choose the 2disks option.

Table 6-3 describes the various disk configuration options for AutoClient systems. You will need to choose one of these options for each AutoClient system.

Table 6-3 Disk Configuration Options

Disk Configuration Options	Meaning
1disk	Use the whole disk as the cache. Swap is a file on that disk.
2disks	Use one disk for the cache and one disk for swap.
local200	Use only with system disks that are 300 Mbytes or larger. Creates a 200Mbyte cache (including swap), and the rest of the system disk is used for a file system that is mounted on `/local`.
local400	Use only with system disks that are 500 Mbytes or larger. Creates a 400Mbyte cache (including swap), and the rest of the system disk is used for a file system that is mounted on `/local`.

Note – If you choose the 1disk or 2disk option, and the system has more than one disk or two disks respectively, the software will randomly pick which disk(s) to use.

⚠ **Caution –** The local200 and local400 disk configuration options allow you to set up a scratch file system on your AutoClient system. This file system can be used to store files that are *not* written back to the server. Since the files are not written back to the server, it is possible to lose this information if the system malfunctions. If you choose the local200 or local400 disk configuration option, and your system disk is smaller than 300Mbytes or 500Mbytes respectively, you could get a runtime error when the AutoClient system first boots.

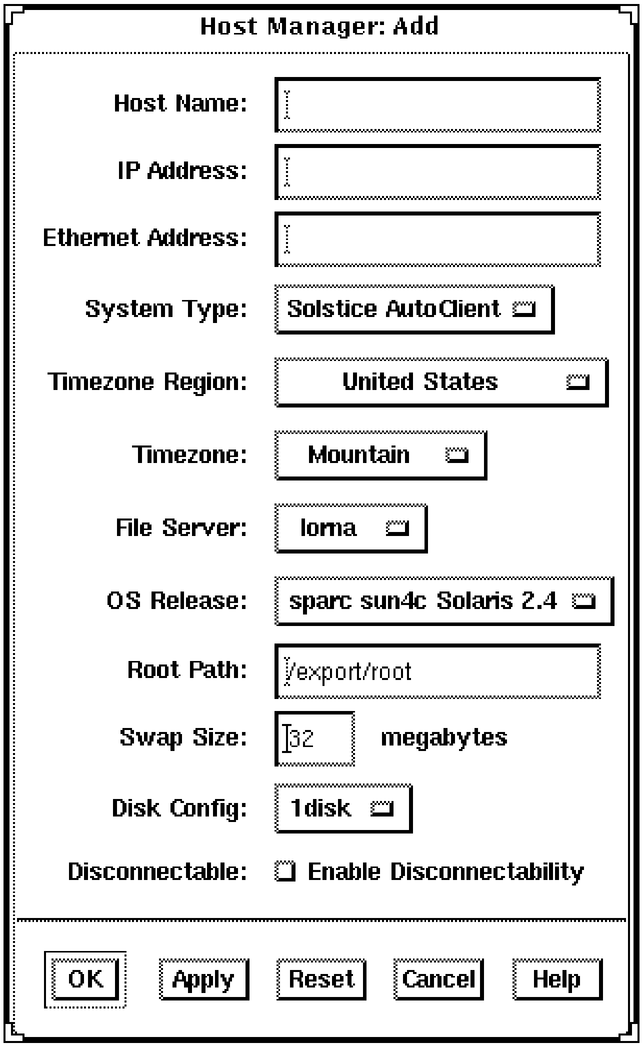
▼ How to Add an AutoClient System to a Server

Note – This procedure assumes that the AutoClient server is already set up to be an OS server and is already installed with the kernel architectures of the AutoClient system(s) to be added. For information on adding an OS Server or converting an existing system to an OS Server, see the online help or the *Solstice AdminSuite 2.2 Administration Guide*.

1. **Start Host Manager from the Solstice Launcher and select the name service, if not done already.**
   See "Starting Host Manager" on page 64 for more information.

2. **Choose Add from the Edit menu.**
   The Add window is displayed. Note that the default system type is Solaris Standalone.

3. **Choose Solstice AutoClient from the System Type menu.**
   The Add window for a Solstice AutoClient system is displayed.



Host Manager: Add

Host Name:	
IP Address:	
Ethernet Address:	
System Type:	Solstice AutoClient
Timezone Region:	United States
Timezone:	Mountain
File Server:	lorna
OS Release:	sparc sun4c Solaris 2.4
Root Path:	/export/root
Swap Size:	32 megabytes
Disk Config:	1disk
Disconnectable:	□ Enable Disconnectability

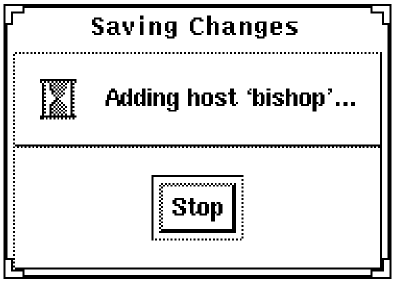
4. **Fill in the system information for the AutoClient system.**

5. **After entering the required information, click on OK.**
   If you have not enabled licensing for the Solstice AutoClient feature, you will see a message saying that the software was unable to check out a license. For information on enabling licensing, see the *Solstice AutoClient 2.0 Installation and Product Notes.*

   The AutoClient system becomes part of the list of AutoClient systems to add, and it is displayed on the Host Manager main window with a plus sign (+) next to it. The + means that the system is a "pending add."

6. **Repeat Step 2 through Step 5 to add subsequent AutoClient systems to your "batch" of pending changes.**
   The "Total Changes Pending" status will be incremented each time you add a system.

7. **When you are ready to confirm addition of all the AutoClient systems listed in the window, choose Save Changes from the File menu.**
   The Saving Changes message window appears. All of the AutoClient systems are added when you choose Save Changes from the File menu.



Adding host 'bishop'...

Adding each client takes several minutes, depending on server speed, current load, and the number and type of patches that will be automatically added.

As each AutoClient system is successfully added (as shown in the Saving Changes window), its corresponding entry no longer appears as a pending add in the Host Manager main window (that is, the + no longer appears next to the host name).

Caution – For the AutoClient system to work properly, it needs root access to its `/export/root` directory. If Host Manager displays a message that the `/export` directory is already shared and has different share options than required, you need to allow root access to the client root area before the AutoClient system will function properly. The access mode for the client root is normally `rw=`*clientname*, `root=`*clientname*.

If Host Manager displays a message that the `/usr` directory is already shared, it is because it tried to share `/usr` read-only. If you have it shared with read-write permissions, it is okay and you do not have to make any modifications.

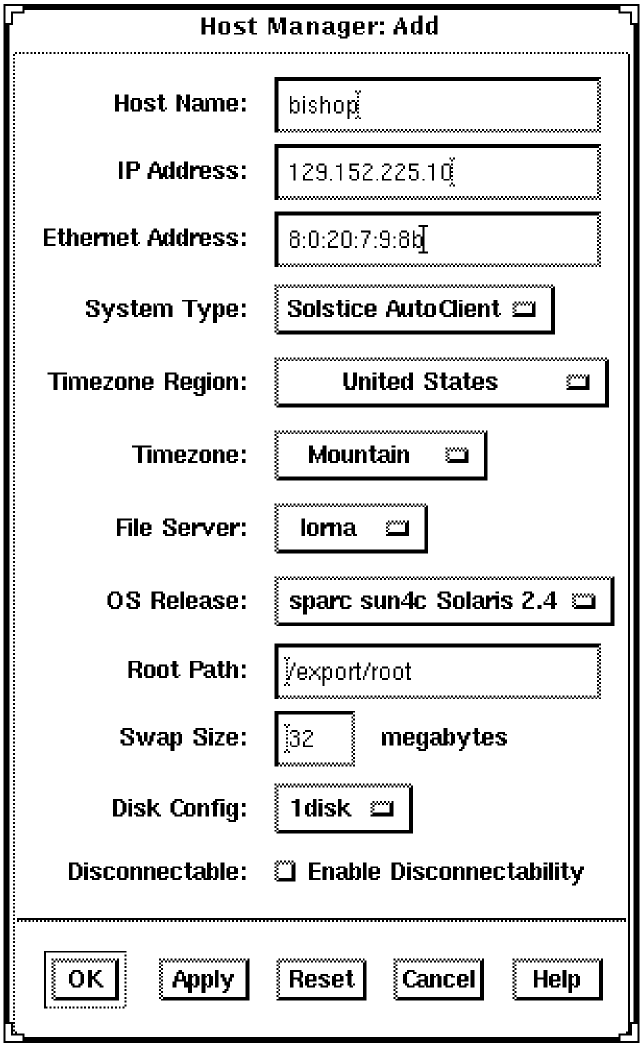
8. **Boot your AutoClient system(s) from the network.**
   For more information about booting your AutoClient systems, see Chapter 7, “Booting a System From the Network.”

9. **Provide system configuration information for the AutoClient system during the initial boot process, if prompted.**

10. **Create a root password when prompted.**

Example of a Completed Add Window

The following example shows a completed Add window for the Solstice AutoClient system type.



Host Manager: Add

Host Name: bishop

IP Address: 129.152.225.10

Ethernet Address: 8:0:20:7:9:8b

System Type: Solstice AutoClient

Timezone Region: United States

Timezone: Mountain

File Server: lorna

OS Release: sparc sun4c Solaris 2.4

Root Path: /export/root

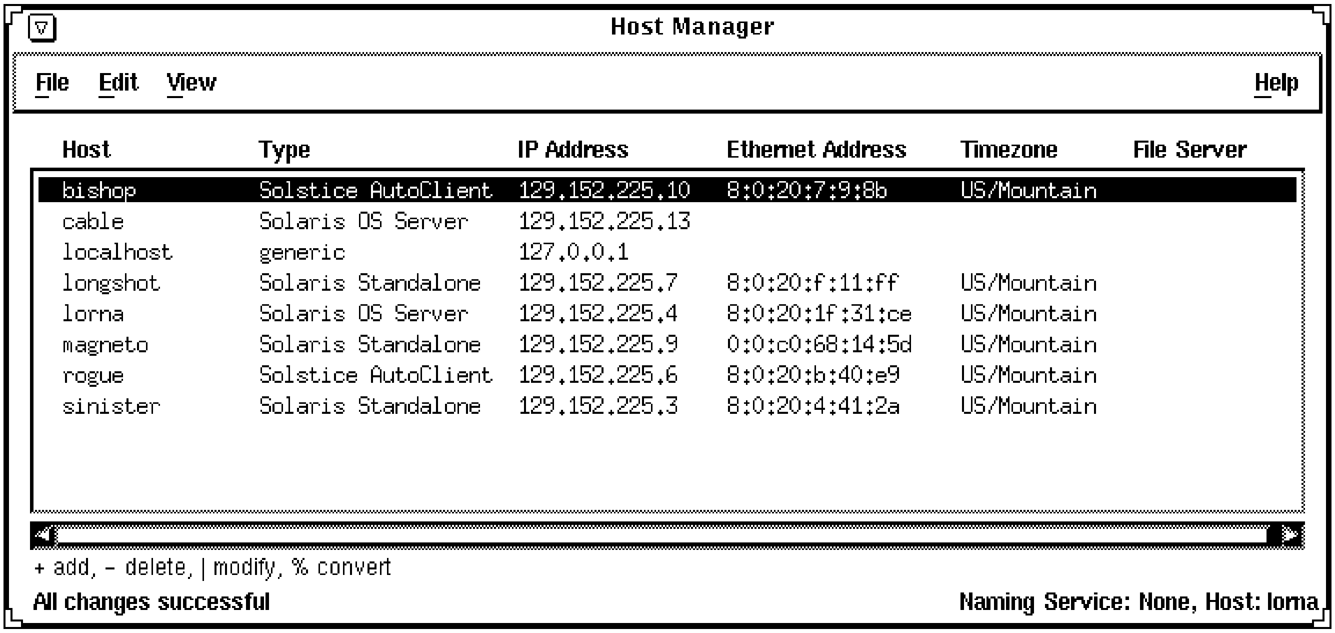
Swap Size: 32 megabytes

Disk Config: 1disk

Disconnectable: □ Enable Disconnectability

Verification

To verify that all the systems have been added, make sure the status line at the bottom of the main window says “All changes successful.”



Host	Type	IP Address	Ethernet Address	Timezone	File Server
bishop	Solstice AutoClient	129.152.225.10	8:0:20:7:9:8b	US/Mountain	
cable	Solaris OS Server	129.152.225.13			
localhost	generic	127.0.0.1			
longshot	Solaris Standalone	129.152.225.7	8:0:20:f:11:ff	US/Mountain	
lorna	Solaris OS Server	129.152.225.4	8:0:20:1f:31:ce	US/Mountain	
magneto	Solaris Standalone	129.152.225.9	0:0:c0:68:14:5d	US/Mountain	
rogue	Solstice AutoClient	129.152.225.6	8:0:20:b:40:e9	US/Mountain	
sinister	Solaris Standalone	129.152.225.3	8:0:20:4:41:2a	US/Mountain	

All changes successful

Naming Service: None, Host: lorna

Example of a Command-Line Equivalent for Adding an AutoClient System

The following command is equivalent to using Host Manager to add support for an AutoClient system.

```
% admhostadd -i 129.152.225.10 -e 8:0:20:7:9:8b \
-x type=AUTOCLIENT -x tz=US/Mountain -x fileserv=lorna \
-x os=sparc.sun4c.Solaris_2.4 -x root=/export/root \
-x swapsize=32 -x disconn=N -x diskconf=1disk bishop
```

In this command,

`-i 129.152.225.10`	Specifies the IP address of the AutoClient system.
`-e 8:0:20:7:9:8b`	Specifies the Ethernet address of the AutoClient system.
`-x type=AUTOCLIENT`	Specifies the type of system being added, in this case an AutoClient system.
`-x tz=US/Mountain`	Specifies the system’s timezone.
`-x fileserv=lorna`	Specifies the name of the OS server.
`-x os=sparc.sun4c.Solaris_2.4`	Specifies platform, kernel architecture, and software release of the AutoClient system.
`-x root=/export/root`	Specifies the root path of the AutoClient system.
`-x swapsize=32`	Specifies the size of the swap file.
`-x disconn=N`	Specifies whether the disconnectable option is enabled, in this case it is not enabled.
`diskconf=1disk`	Specifies the AutoClient system’s disk configuration.
`bishop`	Specifies the name of the AutoClient system.

Troubleshooting Adding Systems

If you receive any error messages indicating that any AutoClient systems failed to be added, use Table 6-4 to troubleshoot the problem.

Table 6-4 Troubleshooting Adding AutoClient Systems

If You Want To ...	Then ...
Stop the add process	Click Stop in the Saving Changes window. Host Manager will stop adding AutoClient systems after it completes adding the current AutoClient system. **Note:** Because Host Manager completes the current operation before stopping the add process, it appears that nothing happens when you click on Stop. Just click on Stop *once*, and the add process will stop after the current operation is completed.
Modify an AutoClient system that failed to be added	**1)** Click on the specific AutoClient system in the main window. **2)** Choose Modify from the Edit menu, or double-click on the selected system. The Modify window is displayed with the selected AutoClient system's information for you to modify. **3)** Modify the information for the AutoClient system and click on Apply. **4)** Repeat steps 1 through 3 to modify additional AutoClient entries. **5)** Choose Save Changes from the File menu.
Ensure you have permission to add clients	Make sure you are a member of sysadmin group 14 on the specified file server, and that you have the appropriate permissions to use Host Manager.

Converting an Existing System to an AutoClient System

In the Solaris 2.5 environment, you can make the AutoClient system conversions shown in Table 6-5.

Table 6-5 AutoClient System Conversions

You Can Convert A ...	To A ...
Generic System	AutoClient System
Standalone System	AutoClient System
Dataless System	AutoClient System
AutoClient System	Standalone System

A *generic* system is one that is not running the Solaris software, or whose type has not yet been updated using Host Manager's Update System Types feature, or uses local or loghost entries in the system management databases.

You will be required to provide the following information when converting generic, standalone, or dataless systems to AutoClient systems:

Table 6-6 Required Fields for Conversion to an AutoClient System

Field	Default/Specifications
Timezone Region	The server's time zone region.
Timezone	The server's time zone.
File Server	The file server specified in the Set Defaults window.
OS Release	The OS Release specified in the Set Defaults window.
Root Path	The root path specified in the Set Defaults window.
Swap Size	The size specified in the Set Defaults window.
Disk Config	1disk. See Table 6-3 on page 73 for disk configuration options. Do not assume you can use the default. You must make sure that the disk configuration you chose is correct for this system.
Disconnectable	Disabled.

▼ How to Convert an Existing System to an AutoClient System

The system being converted may be up and running or powered down.

Caution – If you plan to convert existing generic, standalone, or dataless systems to AutoClient systems, you should consider this process as a re-installation. Any existing system data will be overwritten when the AutoClient system is first booted.

1. **Start Host Manager from the Solstice Launcher and select the name service, if not done already.**
   See "Starting Host Manager" on page 64 for more information.

2. **Select a system or systems from the Host Manager main menu.**

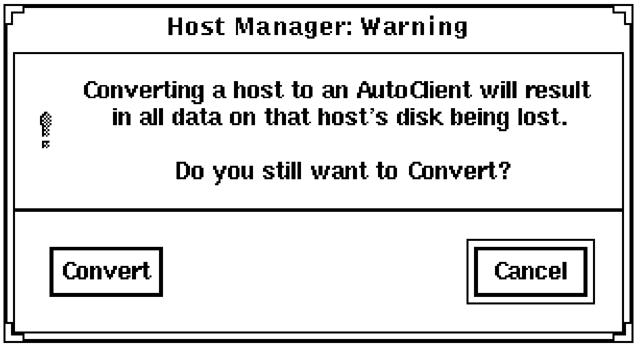
Caution – If you are converting multiple systems in a single operation, make sure they are all of the same kernel architecture.

   To select more than one system, click SELECT (by default, the left mouse button) on the first system. Then select each subsequent system by pressing the Control key and clicking SELECT.

3. **Choose Convert to AutoClient from the Edit menu.**
   The Convert window is displayed with the selected system or systems appearing in the Host Name field.

4. **Fill in the screen by accepting the default or selecting another entry for each field.**
   If you need information to complete a field, see Table 6-6 on page 82 or click on the Help button to see the field definitions for this window.

5. **Click on OK.**
   You will see the following message the first time you use the Convert option in a work session. Subsequent use of the convert option will not generate this message during the same work session. (The duration of a work session is the length of time Host Manager is open. You have to quit and re-start Host Manager to begin a new work session.)



Host Manager: Warning

Converting a host to an AutoClient will result in all data on that host's disk being lost.

Do you still want to Convert?

6. **Click on Convert when you are ready to continue.**
   If you have not enabled licensing for the Solstice AutoClient feature, you will see a message saying that the software was unable to check out a license. For information on enabling licensing, see the *Solstice AutoClient 2.0 Installation and Product Notes.*

7. **Choose Save Changes from the File menu when you are ready to do the conversion(s).**

Caution – For the AutoClient system to work properly, it needs root access to its `/export/root` directory. If Host Manager displays a message that the `/export` directory is already shared and has different share options than required, you need to allow root access to the client root area before the AutoClient system will function properly. The access mode for the client root is normally `rw=`*clientname*, `root=`*clientname*.

If Host Manager displays a message that the `/usr` directory is already shared, it is because it tried to share `/usr` read-only. If you have it shared with read-write permissions, it is okay and you do not have to make any modifications.

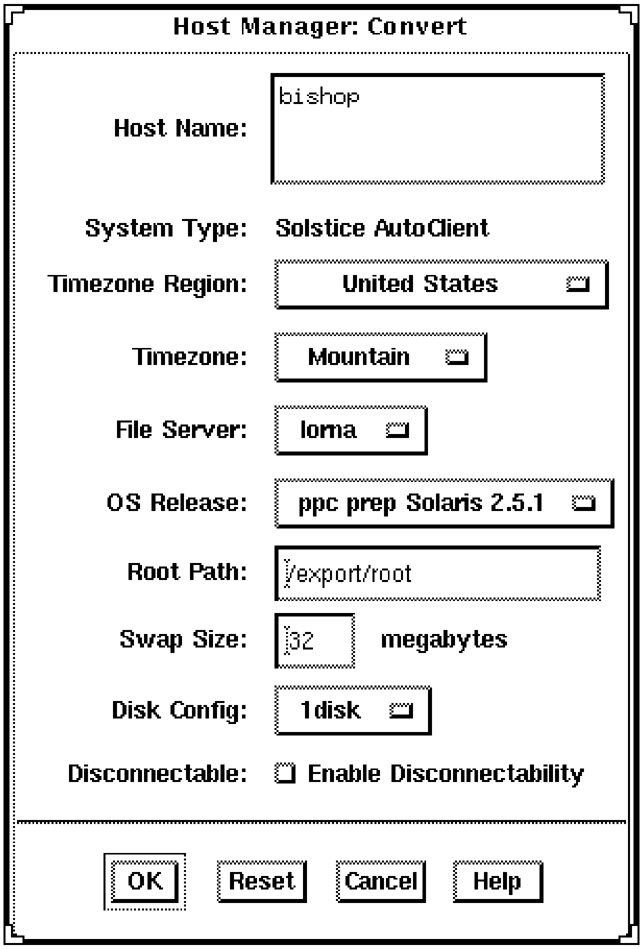
8. **Boot your AutoClient system(s) from the network.**
   For more information about booting your AutoClient systems, see Chapter 7, "Booting a System From the Network."

9. **Provide system configuration information for the AutoClient system during the initial boot process, if prompted.**

10. **Create a root password when prompted.**

Example of a Completed Convert to AutoClient Window

The following shows an example of a completed Host Manager Convert window.



Host Name: bishop

System Type: Solstice AutoClient

Timezone Region: United States

Timezone: Mountain

File Server: lorna

OS Release: ppc prep Solaris 2.5.1

Root Path: /export/root

Swap Size: 32 megabytes

Disk Config: 1disk

Disconnectable: □ Enable Disconnectability

Verification

To verify all the systems have been converted, make sure the status line at the bottom of the main window says “All changes successful.”

Example of a Command-Line Equivalent for Converting a System to an AutoClient System

The following command is equivalent to using Host Manager to convert a system to an AutoClient system.

```
% admhostmod -x type=AUTOCLIENT -x fileserv=lorna \
-x os=i386.i86pc.Solaris_2.5 -x root=/export/root \
-x swapsize=32 -x disconn=N -x diskconf=1disk magneto
```

In this command,

`-x type=AUTOCLIENT` Specifies the type of system after the conversion, in this case an AutoClient system.

`-x fileserv=lorna` Specifies the name of the OS server.

`-x os=i386.i86pc.Solaris_2.5` Specifies platform, kernel architecture, and software release of the AutoClient system.

`-x root=/export/root` Specifies the root path of the AutoClient system.

`-x swapsize=32` Specifies the size of the swap file.

`-x disconn=N` Specifies whether the disconnectable option is enabled, in this case it is not enabled.

`-x diskconf=1disk` Specifies the AutoClient system’s disk configuration.

`magneto` Specifies the name of the system being converted to an AutoClient system.

Converting an AutoClient System to a Standalone System

If you convert an AutoClient system to a standalone system, you will be required to provide the following information:

Table 6-7 Required Fields for Conversion to a Standalone System

Field	Default/Specifications
Timezone Region	The default is the server's time zone region.
Timezone	The default server's time zone.
Remote Install	By default Remote Install is disabled. Click on the selection box if you want to install the Solaris software from remote media. (For more information on remote installation, see *SPARC: Installing Solaris Software*, *x86: Installing Solaris Software*, or *Solaris PowerPC Edition: Installing Solaris Software*.)
Install Server	The default is the install server specified in the Set Defaults window. You must click on Set Path to specify the location of the install image. For more information on setting your media path, see Table 6-1 on page 67.
OS Release	The default is the OS release specified in the Set Defaults window.
Boot Server	The default is none. Choose a boot server and then enter the absolute path for the boot file.
Profile Server	The default is none. Choose a profile server and then enter the absolute path for the autoinstall profile.

An *install server* is a system on the network that provides a Solaris CD image (either from a CD-ROM drive or a copy on hard disk) for other systems to install from. A *boot server* is a system that provides the programs and information a client needs to boot. A *profile server* is a system that contains JumpStart files for systems to perform a custom JumpStart installation.

Caution – If you plan to convert an AutoClient system to a standalone system, you should backup any system data that you might need later (for example, `cron` jobs and calendar data), and *halt the system before completing the convert operation on the server.*

▼ How to Convert an AutoClient System to a Standalone System

Note – This procedure assumes that the install server, boot server, and profile server are already set up. For more information on these tasks, see *SPARC: Installing Solaris Software*.

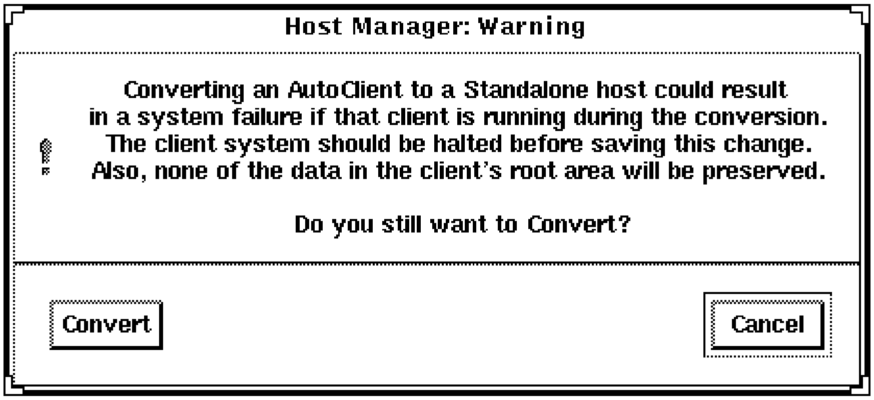
1. **Start Host Manager from the Solstice Launcher and select the name service, if not done already.**
   See "Starting Host Manager" on page 64 for more information.

2. **Select an AutoClient system from the Host Manager main window.**

3. **Choose Convert to Standalone from the Edit menu.**
   The Convert window is displayed.

4. **Fill in the system information.**
   If you need information to complete a field, see Table 6-7 on page 87 or click on the Help button to see the field definitions for this window.

5. **Click on OK.**
   You will see the following message the first time you use the Convert option in a work session. Subsequent use of the convert option will not generate this message during the same work session.



Host Manager: Warning

Converting an AutoClient to a Standalone host could result
in a system failure if that client is running during the conversion.
The client system should be halted before saving this change.
Also, none of the data in the client's root area will be preserved.

Do you still want to Convert?

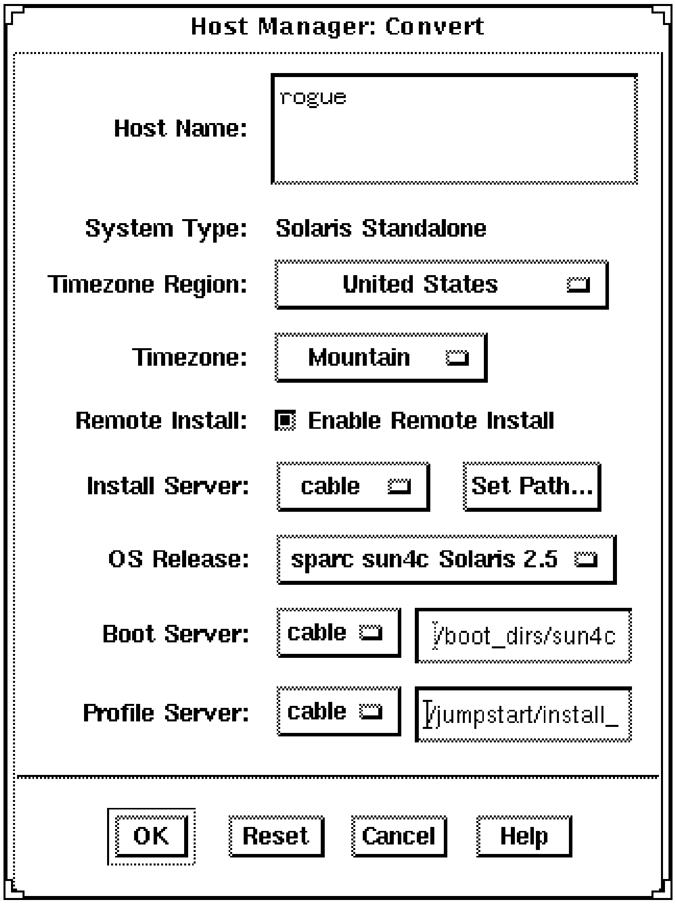
6. **Click on Convert when you are ready to continue.**

7. **Choose Save Changes from the File menu when you are ready to do the conversion.**

8. Boot your standalone system.

Example of a Completed Convert to Standalone Window

The following shows an example of a completed Convert window for converting an AutoClient system to a standalone system.



Host Manager: Convert

Host Name: rogue

System Type: Solaris Standalone

Timezone Region: United States

Timezone: Mountain

Remote Install: ☒ Enable Remote Install

Install Server: cable

OS Release: sparc sun4c Solaris 2.5

Boot Server: cable /boot_dirs/sun4c

Profile Server: cable /jumpstart/install_

Verification

To verify all the systems have been converted, make sure the status line at the bottom of the main window says “All changes successful.”

Example of a Command-Line Equivalent for Converting an AutoClient System to a Standalone System

The following command is equivalent to using Host Manager to convert an AutoClient system to a standalone system. Note that in this example, the boot server, install server, and profile server are also set. (Any remote system must be minimally set up as a managed system.)

```
% admhostmod -x type=STANDALONE -x install=Y \
-x installpath=cable:/cdrom/cdrom0/s0 \
-x os=sparc.sun4c.Solaris2.5 \
-x bootpath=cable:/boot_dirs/boot_sun4c \
-x profile=cable:/jumpstart/install_sample rogue
```

In this command,

`-x type=STANDALONE` Specifies the type of system after the conversion, in this case a standalone system.

`-x install=Y` Specifies that the Solaris software will be installed from remote media.

`-x installpath=cable:/cdrom/cdrom0/s0` Specifies the location of the Solaris software, in this case on a mounted CD on the remote server `cable`.

`-x os=sparc.sun4c.Solaris2.5` Specifies the software to be installed, in this case the Solaris 2.5 software for a SPARC Solaris, sun4c kernel architecture.

`-x bootpath=`
`cable:/boot_dirs/boot_sun4c`
Specifies the boot server and the absolute path of the boot file.

`-x profile=`
`cable:/jumpstart/install_sample`
Specifies the profile server and the absolute path for the autoinstall profile.

`rogue`
Specifies the name of the system being converted.

Modifying an AutoClient System

After configuring an AutoClient system, you may want to change the characteristics of that system. You can make changes both before and after saving the changes; the procedure is the same. However, the information you can modify is different in each situation. See the online help for the field definitions.

▼ How to Modify an AutoClient System

1. **Start Host Manager from the Solstice Launcher and select the name service, if not done already.**
   See “Starting Host Manager” on page 64 for more information.

2. **Select the AutoClient system you want to change in the main window.**
   The system you select should be a pending add.

3. **Choose Modify from the Edit menu.**
   The Modify window appears with fields filled in for the AutoClient system you selected. If you are modifying before saving changes, this Modify window is the same as the Add window for Solstice AutoClient systems.

4. **Change the desired fields in the Modify window.**
   If you need information to complete a field, click on the Help button to see the field definitions for this window.

5. **Click on OK.**
   The changes are implemented when you choose Save Changes from the File menu.

6. **Choose Save Changes from the File menu when you are ready to complete the modification and other pending changes.**

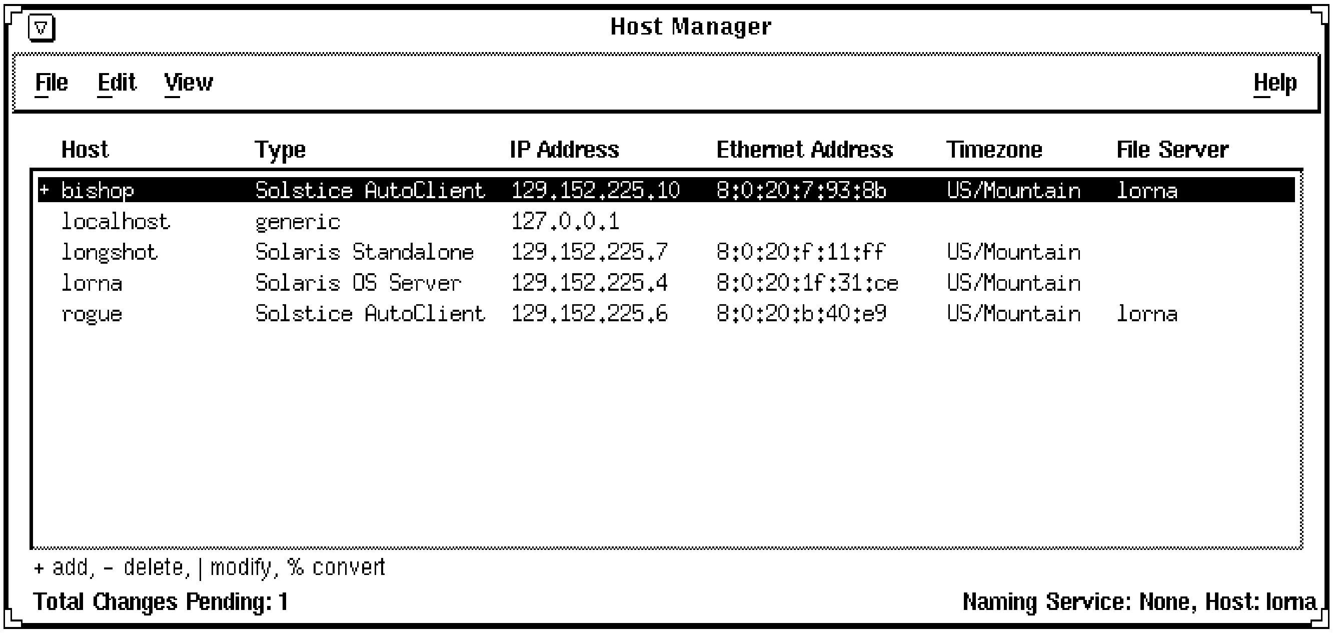
7. **Boot your AutoClient system(s) from the network.**
   For more information about booting your AutoClient systems, see Chapter 7, "Booting a System From the Network."

8. **Provide system configuration information for the AutoClient system during the initial boot process, if prompted.**

9. **Create a root password when prompted.**

Example of a Modify Operation

In this example, the last digit of the IP Address was changed from a 1 to a 10. The operation is still a pending add because the add and modify operations have not yet been saved.



Host	Type	IP Address	Ethernet Address	Timezone	File Server
+ bishop	Solstice AutoClient	129.152.225.10	8:0:20:7:93:8b	US/Mountain	lorna
localhost	generic	127.0.0.1			
longshot	Solaris Standalone	129.152.225.7	8:0:20:f:11:ff	US/Mountain	
lorna	Solaris OS Server	129.152.225.4	8:0:20:1f:31:ce	US/Mountain	
rogue	Solstice AutoClient	129.152.225.6	8:0:20:b:40:e9	US/Mountain	lorna

+ add, - delete, | modify, % convert

Total Changes Pending: 1

Naming Service: None, Host: lorna

Verification

To verify all the systems have been modified, make sure the status line at the bottom of the main window says “All changes successful.”

Example of a Command-Line Equivalent for Modifying an AutoClient System

The following command is equivalent to using Host Manager to modify the IP address on an AutoClient system named `bishop`.

```
% admhostmod -i 129.152.225.10 bishop
```

In this command,

`-i 129.152.225.10`	Specifies the new IP address of the AutoClient system.
`bishop`	Specifies the name of the AutoClient system.

Deleting an AutoClient System

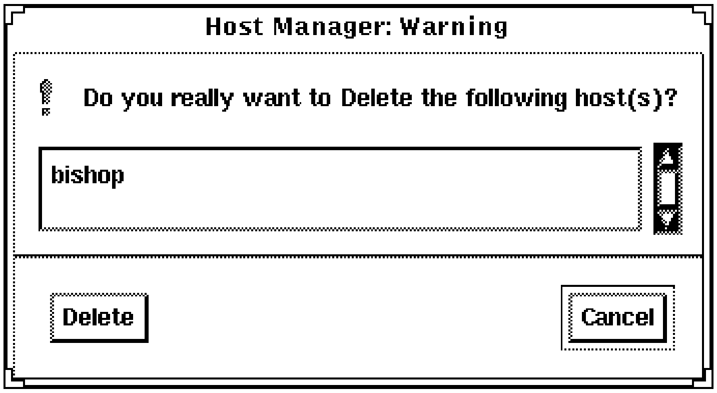
You may need to delete an AutoClient system after it has been added or converted, for example, if the system’s architecture is changing.

▼ How to Delete an AutoClient System

1. **Start Host Manager from the Solstice Launcher and select the name service, if not done already.**
   See “Starting Host Manager” on page 64 for more information.

2. **Select the system or systems you want to delete.**
   To select more than one system, click SELECT (by default, the left mouse button) on the first system. Then select each subsequent system by pressing the Control key and clicking SELECT.

3. **Choose Delete from the Edit Menu.**
   The delete confirmation message appears.



Host Manager: Warning

Do you really want to Delete the following host(s)?

bishop

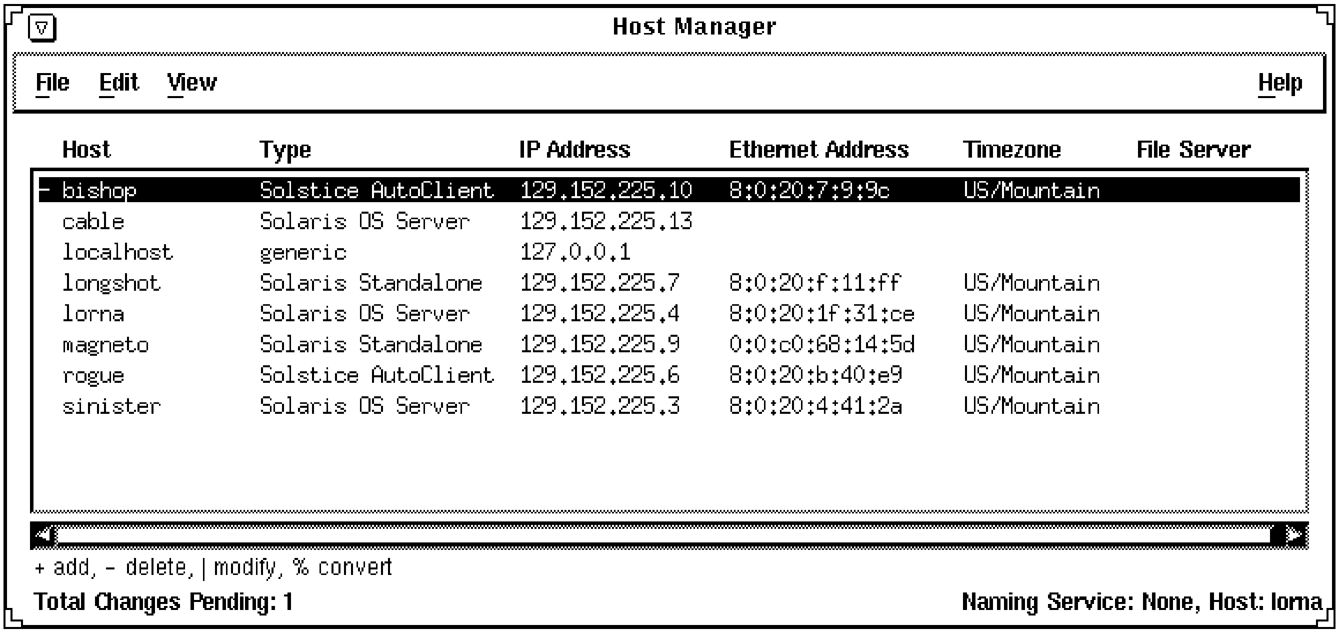
4. **Click on Delete.**

   The system(s) will be marked as a delete change in the main window; you will see a minus sign (-) next to each system. The "Total Changes Pending" status will be incremented for each delete operation.

5. **Choose Save Changes from the File menu when you are ready to delete the system information.**

Example of Deleting an AutoClient System

This example shows a pending delete operation.



Host	Type	IP Address	Ethernet Address	Timezone	File Server
- bishop	Solstice AutoClient	129.152.225.10	8:0:20:7:9:9c	US/Mountain	
cable	Solaris OS Server	129.152.225.13			
localhost	generic	127.0.0.1			
longshot	Solaris Standalone	129.152.225.7	8:0:20:f:11:ff	US/Mountain	
lorna	Solaris OS Server	129.152.225.4	8:0:20:1f:31:ce	US/Mountain	
magneto	Solaris Standalone	129.152.225.9	0:0:c0:68:14:5d	US/Mountain	
rogue	Solstice AutoClient	129.152.225.6	8:0:20:b:40:e9	US/Mountain	
sinister	Solaris OS Server	129.152.225.3	8:0:20:4:41:2a	US/Mountain	

+ add, - delete, | modify, % convert

Total Changes Pending: 1

Naming Service: None, Host: lorna

Verification

To verify all the systems have been deleted, make sure the status line at the bottom of the main window says “All changes successful.”

Example of a Command-Line Equivalent for Deleting an AutoClient System

The following command is equivalent to using Host Manager to delete (that is, remove it from the name service database) an AutoClient system named bishop.

```
% admhostdel bishop
```

Reverting a System to Its Last-Saved State

You may want to revert systems marked with change symbols (|, -, or %) to their last-saved state in the name service database. Reverting these previously existing systems will not affect their presence in the main window.

However, reverting a newly-added (not yet saved) AutoClient system (identified with a +) will result in the entry being deleted from the scrolling list in the main window.

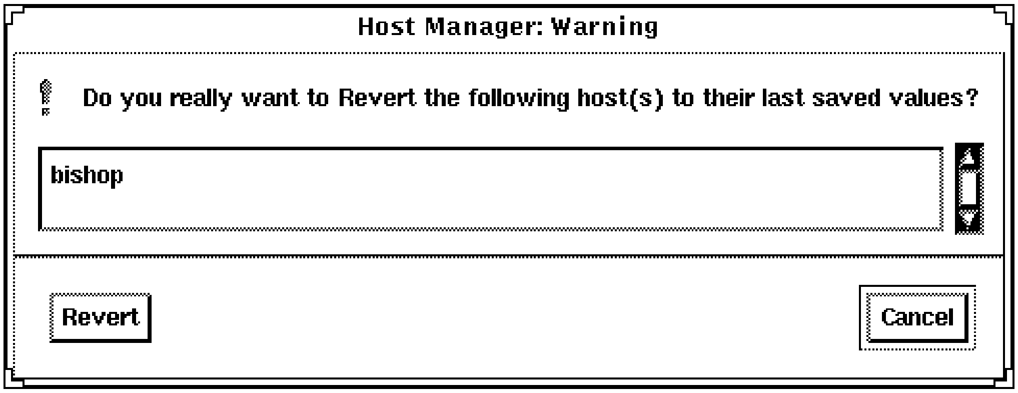
Note that when you select the Revert option, a message asks for confirmation.

▼ How to Revert a System to Its Last-Saved State

1. **Start Host Manager from the Solstice Launcher and select the name service, if not done already.**
   See “Starting Host Manager” on page 64 for more information.

2. **Select the system or systems you want to revert.**
   To select more than one system, click SELECT (by default, the left mouse button) on the first system. Then select each subsequent system by pressing the Control key and clicking SELECT.

3. **Choose Revert from the Edit Menu.**
   The revert confirmation message appears.



Host Manager: Warning

Do you really want to Revert the following host(s) to their last saved values?

bishop

4. **Click on Revert.**
   The revert operation takes effect immediately.

Verification

Make sure the system type displays in the main window as its original type or with its original characteristics.

Using the Command-Line Interface to Automate Setup Tasks

Using the Host Manager command-line equivalents allows you to automate many of the setup tasks associated with creating new diskless and AutoClient systems. This automation is similar to what can be done when using the JumpStart™ product to install Solaris on standalone systems. By writing your own shell scripts and using the command-line equivalents, you can automatically customize the client environment in one operation.

The example in the next section shows how to use the command-line interface to set up an OS server, add OS services, and add a AutoClient system to that server. The server's name is `rogue`, and the AutoClient system is `venus`.

Note – For additional command-line examples, see the command-line equivalent section at the end of most of the procedures in this chapter.

▼ How to Use the Command-Line Interface to Automate Setup Tasks

1. **Convert a standalone system to an OS server.**

```
% admhostmod -x type=os_server rogue
```

2. **Add OS services to the OS server.**

c. **This example adds the Solaris 2.5 End User Cluster OS services for the Sun4m kernel architecture to `rogue`. The Solaris CD image is on a mounted CD on a remote system named `jupiter`. Note that the remote system must be minimally set up as a managed system.**

```
% admhostmod -x mediapath=jupiter:/cdrom/cdrom0/s0 \
-x platform=sparc.sun4m.Solaris_2.5 -x cluster=SUNWCuser \
rogue
```

d. **This examples adds the Solaris 2.5.1 All Cluster OS services for the Sun4m kernel architecture to `rogue`. The Solaris CD image has been copied to hard disk on a remote system, `saturn`, and the automounter is used to access it. Note that the remote system must be minimally set up as a managed system.**

```
% admhostmod -x mediapath=rogue:/net/saturn/export/Solaris_CD \
-x platform=sparc.sun4m.Solaris_2.5.1 -x cluster=SUNWCall \
rogue
```

3. **Add the AutoClient system.**
   This example adds a Sun4m Solaris 2.5.1 AutoClient system named `venus` to the server `rogue`.

```
% admhostadd -i 129.152.225.2 -e 8:0:20:b:40:e9 \
-x type=autoclient -x fileserv=rogue \
-x os=sparc.sun4m.Solaris_2.5.1 \
-x swapsize=40 -x diskconf=1disk -x diskconn=n venus
```

You could use a similar version of this command in a shell script with additional operations to customize the AutoClient system's root as part of setting up the client. The script could be parameterized to accept the IP address, Ethernet address, and host name.

Booting a System From the Network 7

After you add an AutoClient system (see “Adding AutoClient Systems” on page 71) to an AutoClient server, or convert an existing system to an AutoClient system (see “Converting an Existing System to an AutoClient System” on page 82), the AutoClient system is ready to boot and run the Solaris environment.

Note – AutoClient systems must always boot from the network.

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

Note – Systems that you are going to add as AutoClient systems or convert to AutoClient systems may be up and running or powered down during the add and convert operations. They don’t really become AutoClient systems until they are booted. The only exception is when converting an AutoClient system to a standalone system. In this case, the system being converted must be halted prior to completing the convert operation on the server.

SPARC: Booting From the Network

This section provides procedures on how to manually boot your SPARC system from the network, and how to set it up to automatically boot from the network.

You need to read only certain portions of this section. Table 7-1 shows you which task information to read for the type of systems you have on your network.

Table 7-1 System Booting Information

If You Have This System ...	See These Tasks ...	On ...
SPARCstation and above with the Solaris software already running (boot prom prompt) or out of the box (the `ok` prompt)	"SPARC: How to Manually Boot a System From the Network" "SPARC: How to Set Up a System to Automatically Boot From the Network"	page 102 page 104
Sun-4 systems	"SPARC: How to Manually Boot a Sun-4 System From the Network" "SPARC: How to Set Up a Sun-4/3nn System to Automatically Boot From the Network" "SPARC: How to Set Up a Sun-4/1nn, 2nn, or 4nn System to Automatically Boot From the Network	page 104 page 107 page 108

Note – In the Solaris 2.5 environment, only the Sun-4c, Sun-4d, Sun-4m, Sun-4u kernel architectures, and the x86 and PowerPC platforms are supported. The Solaris 2.5 software no longer supports Sun-4 and Sun-4e.

Table 7-2 summarizes the commands you use to manually boot systems from the network for different system models.

Table 7-2 Sun System Boot Commands

System Type	Boot Command
SPARCstation and above	`boot net`
Sun-4/3*nn*	`b le()`
Sun-4/1*nn*, Sun-4/2*nn*, Sun-4/4*nn*	`b ie()`

For more information about the booting process in general, see the *Solaris 2.4 Administration Supplement for Solaris Platforms* for the Solaris 2.4 product, and the *System Administration Guide, Volume I* for the Solaris 2.5 product.

▼ SPARC: How to Manually Boot a System From the Network

Note – If you want to manually boot a Sun-4 system from the network, see “SPARC: How to Manually Boot a Sun-4 System From the Network” on page 104.

1. **Make sure the AutoClient system has been set up as described in “Adding AutoClient Systems” on page 71 or in “Converting an Existing System to an AutoClient System” on page 82.**
2. **Make sure the system is in the prom monitor environment.**
   If the system is not running, power it up. If the system is currently running, use the `init 0` command to get it to the boot prom prompt.
3. **If the screen displays the > prompt instead of the `ok` prompt, type `n` and press Return or Enter.**
   The screen should now display the `ok` prompt. If not, see “SPARC: How to Manually Boot a Sun-4 System From the Network” on page 104.

4. Boot the system from the network.

```
ok boot net
```

Example of Manually Booting a SPARC System From the Network

```
# init 0
> n
ok
.
.
.
ok boot net
Booting from: le(0,0,0)
2bc00 hostname: pluto
domainname: Solar.COM
root server:
root directory: /export/root/pluto
SunOS Release 5.4 Version [2.4_FCS] [UNIX(R) System V Release
4.0]
Copyright (c) 1983-1994, Sun Microsystems, Inc.
configuring network interfaces: le0.
Hostname: pluto
Configuring cache and swap:......done.
The system is coming up. Please wait.
NIS domainname is Solar.COM
starting rpc services: rpcbind keyserv ypbind kerbd done.
Setting netmask of le0 to 255.255.255.0
Setting default interface for multicast: add net 224.0.0.0:
gateway pluto
syslog service starting.
Print services started.
volume management starting.
The system is ready.
login: root
password:
# exit
```

▼ SPARC: How to Manually Boot a Sun-4 System From the Network

1. **Make sure the AutoClient system has been set up as described in "Adding AutoClient Systems" on page 71 or in "Converting an Existing System to an AutoClient System" on page 82.**

2. **Make sure the system is in the prom monitor environment.**
   If the system is not running, power it up. If the system is currently running, use the `init 0` command to get it to the boot prom prompt.

3. **Type the appropriate boot command to boot the system from the network.**

```
> b le()
or
> b ie()
```

▼ SPARC: How to Set Up a System to Automatically Boot From the Network

Note – If you want to set up a Sun-4 system to automatically boot from the network, see "SPARC: How to Set Up a Sun-4/3nn System to Automatically Boot From the Network" on page 107, or "SPARC: How to Set Up a Sun-4/1nn, 2nn, or 4nn System to Automatically Boot From the Network" on page 108.

1. **Make sure the AutoClient system has been set up as described in "Adding AutoClient Systems" on page 71 or in "Converting an Existing System to an AutoClient System" on page 82.**

2. **Make sure the system is in the prom monitor environment.**
   If the system is not running, power it up. If the system is currently running, use the `init 0` command to get it to the boot prom prompt.

3. **If the screen displays the > prompt instead of the `ok` prompt, type `n` and press Return or Enter.**
   The screen should now display the `ok` prompt. If not, see "SPARC: How to Set Up a Sun-4/3nn System to Automatically Boot From the Network" on page 107, or "SPARC: How to Set Up a Sun-4/1nn, 2nn, or 4nn System to Automatically Boot From the Network" on page 108.

4. **Determine the version number of the boot prom with the** `banner` **command. The following is an example:**

```
ok banner
SPARCstation 2, Type 4 Keyboard

ROM Rev. 2.0, 16MB memory installed, Serial # 289
Ethernet address 8:0:20:d:e2:7b, Host ID: 55000121
```

5. **Set the boot device.**

   If the boot prom is version 2.0 or greater, type the following command.

```
ok setenv boot-device net
boot-device=net
```

   If the boot prom version is less than 2.0, type the following command.

```
ok setenv boot-from net
```

   For more information about boot proms, see the *OpenBoot 2.x Command Reference Manual* or the *OpenBoot 3.x Command Reference Manual*.

6. **Boot the system automatically from the network by using the** `boot` **command.**

```
ok boot
```

SPARC: How to Display Existing Boot Device Values on Sun-4 Systems

This procedure describes how to display the current boot device values, if you need to record them before changing them.

1. **Display the values of the system's current booting devices.**

```
> q18
```

The system displays the first EEPROM value.

2. **Write down the EEPROM number and value.**
For example, you might see `EEPROM 018:12?`. The EEPROM number is 018 and the value is 12.

3. **Press Return to display the next value.**

4. **Repeat steps 2 and 3 until the last value is displayed.**
The last value is 00.

5. **Quit the EEPROM display mode.**

```
EEPROM 01B: 00? q
```

Example of Displaying Existing Boot Device Values on Sun-4 Systems

```
> q18
EEPROM 018: 12?
EEPROM 019: 69?
EEPROM 01A: 65?
EEPROM 01B: 00? q
>
```

Entering `q18` and pressing Return three times displays the three values. You should retain this information. The last `q` entry returns you to the `>` prompt.

SPARC: How to Set Up a Sun-4/3*nn* System to Automatically Boot From the Network

1. **Make sure the AutoClient system has been set up as described in "Adding AutoClient Systems" on page 71 or in "Converting an Existing System to an AutoClient System" on page 82.**

2. **Make sure the system is in the prom monitor environment.**

3. **(Optional) Perform the procedures in "SPARC: How to Display Existing Boot Device Values on Sun-4 Systems" on page 106 if you want to record the current boot device values.**

4. **At the command prompt, enter the following boot device code sequence.**

   ```
   > q18 12 6c 65
   ```

   This is the code for `le` (the Lance Ethernet).

   What you are doing for any of the Sun-4 architectures is programming the EEPROM (or NVRAM) by entering `q` followed by the hexadecimal address in the EEPROM. This sets the appropriate operating system boot device.

5. **Boot the system automatically from the network.**

   ```
   > b
   ```

Example of Setting Up a Sun-4/3nn System to Automatically Boot From the Network

```
> q18 12 6c 65
EEPROM 018 -> 12
EEPROM 019 -> 6C
EEPROM 01A -> 65
>
```

If the system output looks like the example above, you set the codes successfully. If the output looks similar to the following:

```
> b

EEPROM boot device... ie(0,0,0)

Invalid device = ‘ie’
```

you set the wrong code for the specific system architecture, and the system will not boot. You need to reset the codes. In the above example output, a Sun-4/3*nn* was set up with the wrong device code (`ie` instead of `le`).

▼ SPARC: How to Set Up a Sun-4/1*nn*, 2*nn*, or 4*nn* System to Automatically Boot From the Network

1. **Make sure the AutoClient system has been set up as described in “Adding AutoClient Systems” on page 71 or in “Converting an Existing System to an AutoClient System” on page 82.**

2. **Make sure the system is in the prom monitor environment.**

3. **(Optional) Perform the procedures in “SPARC: How to Display Existing Boot Device Values on Sun-4 Systems” on page 106 if you want to record the existing boot device values.**

4. At the command prompt, enter the following boot device code sequence.

```
> q18 12 69 65
```

This is the code for `ie` (the Intel Ethernet).

What you are doing for any of the Sun-4 architectures is programming the EEPROM (or NVRAM) by entering `q` followed by the hexadecimal address in the EEPROM. This sets the appropriate operating system boot device.

5. Boot the system automatically from the network.

```
> b
```

Example of Setting Up a Sun-4/1nn, 2nn, or 4nn System to Automatically Boot From the Network

```
> q18 12 69 65
EEPROM 018 -> 12
EEPROM 019 -> 69
EEPROM 01A -> 65
```

If the system output looks like the example above, you set the codes successfully. If the output looks similar to the following:

```
> b

EEPROM boot device... le(0,0,0)

Invalid device = ‘le’
```

you set the wrong code for the specific system architecture, and the system will not boot. You need to reset the codes. In the above example output, a Sun-4/1*nn*, 2*nn*, or 4*nn* was set up with the wrong device code (`le` instead of `ie`).

Where to Go Next

If you have problems booting your AutoClient system, see "Troubleshooting Problems When Booting an AutoClient System" on page 117. Otherwise, go on to Chapter 8, "AutoClient Environment Maintenance."

x86: Booting From the Network

The following procedures apply to x86 systems. Booting an x86 system uses these two subsystems:

- Solaris boot diskette (contains the program that provides booting from the network)
- Secondary boot subsystem

The Solaris boot diskette, also known as the MDB diskette, provides a menu of bootable devices such as disk, network, or CD-ROM. (The system probes currently connected devices and displays the devices in the MDB menu.) AutoClient systems must boot from the network so you would always enter the code for the network device.

The second boot subsystem menu displays available boot options. The system automatically boots to run level 3 if you do not select an option within 60 seconds. The other options enable you to specify boot options or enter the boot interpreter (see `boot(1M)`).

▼ x86: How to Manually Boot a System

This procedure describes how to manually boot your x86 system from the network. Screen displays will vary based on system configurations.

1. **Make sure the AutoClient system has been set up as described in "Adding AutoClient Systems" on page 71 or in "Converting an Existing System to an AutoClient System" on page 82.**

2. **Insert the Solaris boot diskette into the drive.**

3. **Press the reset button.**
   The Primary Boot Subsystem menu is displayed after a short time.

```
Solaris 2.4 for x86                     Multiple Device Boot, vsn 2.1

                Solaris/x86 Multiple Device Boot Menu

           Code   Device    Vendor    Model/Desc          Rev
           ====================================================

           10     DISK      MAXTOR    LXT-535S            8.75
           11     CD        SONY      CD-ROM CDV-8012     3.1d
           12     NET       SMC/WD    I/O=300 IRQ=5

           Enter the boot device code:

30
```

The Solaris boot diskette provides a menu of bootable devices such as disk, network, or CD-ROM. (The system probes currently-connected devices and displays the devices in the MDB menu.)

Note – The number 30 displayed in the bottom left corner counts down, indicating the number of seconds left to set the boot device code. If you do not specify the boot device code within 30 seconds, the system will attempt to boot from the C drive, which is the default device.

---

4. **Enter the boot device code to boot from the network.**
   In this example the boot device code is 12.

   The Secondary Boot Subsystem menu is displayed after a short time.

```
Solaris 2.4 for x86                  Secondary Boot Subsystem, vsn 2.11

                    <<< Current Boot Parameters >>>
 Boot path: /eisa/dpt@5c88,0/cmdk@0,0:a
 Boot args: /kernel/unix

 Type    b [file-name] [boot-flags] <ENTER>   to boot with options
 or      i <ENTER>                            to enter boot interpreter
 or      <ENTER>                              to boot with default

                     <<< timeout in 60 seconds >>>

 Select (b)oot or (i)nterpreter:
```

5. **Type** `b` **or** `boot` **to boot the system and press Return.**

Note – Use the `-f` option of the `boot` command (or the `b` command) to re-create the cache on the AutoClient system. You need to re-create the cache if you get any booting errors (see “Troubleshooting Problems When Booting an AutoClient System” on page 117) or if the server’s file systems had to be restored from backup.

▼ x86: How to Set Up a System to Automatically Boot From the Network

This procedure describes how to create an x86 multiple device boot (MDB) diskette so that your x86 AutoClient system will always boot from the network—so you do not have to be there to boot it. Otherwise, if the master MDB diskette is inserted into the drive, an x86 system will attempt to boot off the C drive after a power cycle (for more information see “x86: Booting From the Network” on page 110).

Note – Before following these steps to create an MDB boot diskette, obtain the master MDB diskette for the x86 system and a blank 1.44 Mbyte diskette. The blank diskette will be formatted, so do not use a diskette with data on it.

1. **Become root on your server.**
2. **Change your working directory.**

   ```
   # cd /opt/SUNWadm/2.2/floppy
   ```

3. **Create the MDB boot diskette.**

   ```
   # ./mk_floppy
   ```

   The script prompts you when to insert the MDB master diskette and the blank diskette, and provides additional status information.

   ```
   Please insert the master MDB floppy and press Return:
   Please insert a blank floppy and press Return:
   Formatting 1.44 MB in /dev/rdiskette
   .............................................................
   ...................
   fdformat: using "./mdboot" for MS-DOS boot loader
   Successfully created the AutoClient floppy.
   #
   ```

4. **Insert the MDB boot diskette into the diskette drive of the x86 system.** You must leave this boot diskette in the diskette drive so that the system will automatically boot from the network if a power cycle occurs.

Where to Go Next

If you have problems booting your AutoClient system, see “Troubleshooting Problems When Booting an AutoClient System” on page 117. Otherwise, go on to Chapter 8, “AutoClient Environment Maintenance.”

7

PowerPC: Booting From the Network

This section provides procedures on how to manually boot your PowerPC system from the network, and how to set it up to automatically boot from the network. The steps are similar to those for booting a SPARC system from the network.

▼ PowerPC: How to Manually Boot a System

1. **Make sure the AutoClient system has been set up as described in "Adding AutoClient Systems" on page 71 or in "Converting an Existing System to an AutoClient System" on page 82.**

2. **Make sure the system is in the prom monitor environment.**
   If the system is not running, power it up. If the system is currently running, execute the following commands to get it to the boot prom prompt.

   a. **Become root.**

   b. **Halt the system by using the** `init 0` **command.**

   ```
   # init 0
   ```

   c. **Press any key to stop the automatic boot process after the** `Hit any key to abort` **prompt.**

   ```
   Automatically booting in 5 seconds. Hit any key to abort.
   ```

   The `ok` prompt is displayed.

3. **If this is the first time you are booting the system as an AutoClient system, insert the Solaris boot diskette into the diskette drive.**

4. **Boot the system from the network.**

   ```
   ok boot net
   ```

PowerPC: How to Set Up a System to Automatically Boot From the Network

1. **Make sure the AutoClient system has been set up as described in "Adding AutoClient Systems" on page 71 or in "Converting an Existing System to an AutoClient System" on page 82.**

2. **Make sure the system is in the prom monitor environment.**
   If the system is not running, power it up. If the system is currently running, execute the following commands to get it to the boot prom prompt.

   a. **Become root.**

   b. **Halt the system by using the `init 0` command.**

   ```
   # init 0
   ```

   c. **Press any key to stop the automatic boot process after the `Hit any key to abort` prompt.**

   ```
   Automatically booting in 5 seconds. Hit any key to abort.
   ```

   The `ok` prompt is displayed.

3. **Verify that the `auto-boot?` parameter is set to `true` by using the `printenv` command.**
   If the `auto-boot?` parameter is set to `true`, skip to Step 6.

   ```
   ok printenv auto-boot?
   auto-boot?="true" (default value = "true")
   ```

4. **If the `auto-boot?` parameter is set to `false`, change it to `true` by using the `setenv` command.**

   ```
   ok setenv auto-boot? true
   ```

5. **Verify that the** `auto-boot?` **parameter has been changed.**

```
ok printenv auto-boot?
auto-boot?="true" (default value = "true")
```

6. **Change the** `boot-device` **parameter by using the** `setenv` **command to boot from the network rather than from disk.**

```
ok setenv boot-device boot net
```

7. **Verify the change by using the** `printenv` **command.**

```
ok printenv boot-device
boot-device="boot net" (default value =
"pci/pci1000,1@1/disk@6,0:\solaris.elf")
```

8. **If this is the first time you are booting the system as an AutoClient system, insert the Solaris boot diskette into the diskette drive.**

9. **Boot the system automatically from the network.**

```
ok boot
```

Where to Go Next

If you have problems booting your AutoClient system, see "Troubleshooting Problems When Booting an AutoClient System" on page 117. Otherwise, go on to Chapter 8, "AutoClient Environment Maintenance."

Troubleshooting Problems When Booting an AutoClient System

Table 7-3 provides a list of the most common error messages that may be displayed when you try to boot an AutoClient system. Each error message is followed by a description of why the error occurred and how to fix the problem.

Table 7-3 Booting Error Messages

Error Message	Reason Error Occurred	How to Fix the Problem

`ERROR: Insufficient file system space configuration`
`Slice/partition does not fit in disk segment.`
`Not enough space on disk.`
	You may have specified a swap size that is too large, or you selected the wrong disk configuration.
	Note: If you have x86 AutoClient systems, the free space on your DOS partition may be too small.
		Use Host Manager to set up the AutoClient system again, this time making sure that the disk config size is at least as large as swap space + 24 Mbytes.
		Note: For x86, the disk size is the Solaris partition.

`Could not create /.cache/swap file`
or
`Could not clear existing swap entries from /etc/vfstab`
	System failure.	Reboot the system using the `-f` option of the `boot` command. If you receive the error again, call your service representative.

Note – The first three messages have similar reasons why the error occurred. They have the same method of fixing the problem. All of the messages are followed by this flag: `FATAL: Error in disk configuration`.

You may receive error messages that contain a `FATAL` flag. If you do, you should reboot the system by using the `-f` option of the boot command. If you receive the `FATAL` flag error again, use Host Manager to set up the AutoClient system again.

You need to re-create the cache if you get any booting errors or if the server's file systems had to be restored from backup. To re-create the cache on the AutoClient system, type `boot` followed by the `-f` option. The `-f` option re-creates the cache.

Note – Some SPARC booting problems not related to AutoClient systems can be corrected if you use the `reset` command at the `ok` prompt before booting the AutoClient system. If the system begins to boot from somewhere other than the network after the AutoClient system resets, you must reboot the system. Then you can proceed to boot the AutoClient system with the appropriate boot command.

AutoClient Environment Maintenance 8

After you have set up your AutoClient system network using Host Manager, you will need to perform certain maintenance tasks.

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

8

Overview of AutoClient Patch Administration

In its simplest form, you can think of a patch as a collection of files and directories that replace or update existing files and directories that are preventing proper execution of the software. The existing software is derived from a specified *package* format, which conforms to the Application Binary Interface. (For details about packages, see the *System Administration Guide, Volume I*.)

On diskless clients and AutoClient systems, all software resides on the server. For example, when you add a software patch to an AutoClient system, you don't actually install the patch on the client, because its local disk space is reserved for caching. Instead, you add the patch either to the server or to the client's root file system (which resides on the server), or both. An AutoClient system's root file system is typically in `/export/root/`*hostname* on the server.

Applying patches to clients is typically complicated because the patch may place software partially on the client's root file system and partially on the OS service used by that client.

To reduce the complexity of installing patches on diskless clients and AutoClient systems, the Solstice AutoClient product includes the `admclientpatch` command. Table 8-1 summarizes its options and use.

Table 8-1 `admclientpatch` Options and Use

Option	**Use**
`-a` *patch_dir/patch_id*	Add a patch to a spool directory on the server.
`-c`	List all diskless clients and AutoClient systems that are served by this server and list all patches applied to the OS service file system for those clients.
`-p`	List all currently spooled patches.
`-r` *patch_id*	Remove the specified *patch_id* from the spool directory.
`-s`	Synchronize all clients so that the patches they are running match the patches in the spool directory.

Guidelines for AutoClient Patch Administration

The general procedure for maintaining patches on AutoClient systems is as follows:

- Use `admclientpatch -a` or `-r` to create or update a spool directory of all appropriate patches on the local machine.
- On any client server, use `admclientpatch -s` to synchronize those patches installed on clients with those patches in the spool directory.

This general procedure for maintaining patches assumes the OS server (that is, the server providing OS services to clients) is the same system with the patch spool directory. If, however, your site has several OS servers for your AutoClient systems, you may want to use a single file server for the patch spool directory, and then mount that directory on the OS servers.

If this is the way you choose to configure your site, you will have to do all updates to the patch spool directory directly on the file server. (You can't successfully run `admclientpatch -a` or `-r` from one of the OS servers because the patch spool directory is shared read-only.) When mounting the patch spool directory from a single file server, the general procedure for maintaining patches on AutoClient systems is as follows:

- On the file server, use `admclientpatch -a` or `-r` to update a spool directory of all appropriate patches on the file server.
- On all OS servers that mount the patch directory from the file server, use `admclientpatch -s`.

Caution – Do not manually add or remove patches from the spool directory. Instead use the `admclientpatch` command for all of your patch administration tasks.

What Happens When You Add a Patch With the `admclientpatch -a` Command

The `admclientpatch -a` command copies patch files from the patch directory to a spool directory on the local system. The spool directory is `/opt/SUNWadmd/2.2/Patches`. If the patch being added to the spool directory makes any existing patches obsolete, `admclientpatch` archives the old patches in case they need to be restored.

What Happens When You Remove a Patch With the `admclientpatch -r` *Command*

The `admclientpatch -r` command removes an existing patch from the spool directory and restores the previous version of that patch—if it exists. (Patches made obsolete by a new patch in the spool area are archived so that they can be restored.)

What Happens When You Synchronize a Patch With the `admclientpatch -s` *Command*

The `admclientpatch` command is a front-end to the standard patch utilities, `installpatch` and `backoutpatch`. Using these utilities, installing a patch and backing out a patch are distinct tasks. However, by using `admclientpatch -s`, you do not need to be concerned whether you are installing or backing out a patch. The `-s` option ensures that `admclientpatch` will take the appropriate actions. It either installs the patch on the server and in the client's own file systems on the server, or it backs out the patch from the clients and server and re-installs the previous version of that patch. This is what is meant by *synchronizing* patches installed on the clients with patches in the patch spool directory.

How Host Manager Uses the Patch Spool Directory

When you use Host Manager to add new diskless clients and AutoClient systems to a network's configuration files, it will automatically set up those new clients with the patches in the patch spool directory. Host Manager may detect that the installation of a patch in an OS service area may have made all other clients of that service out of sync with the patch spool directory. If so, Host Manager will issue a warning for you to run `admclientpatch -s` on other servers to synchronize the patches installed on them with the patches in the patch spool directory.

For More Information on Patch Administration

For details about what happens when you add or remove a patch and how patches are distributed, see the *System Administration Guide, Volume 1*. For more details about how to use `admclientpatch`, refer to the `admclientpatch(1m)` man page.

Managing Patches on AutoClient Systems

▼ How to Copy Patches to an OS Server's Patch Spool Directory

1. **Make sure you have your `PATH` environment variable updated to include `/opt/SUNWadm/2.2/bin`. For details, refer to the *Solstice AutoClient 2.0 Installation and Product Notes*.**

2. **Log in to the OS server and become root.**

3. **Copy patches to the default spool directory with this command.**

```
# admclientpatch -a patch_dir/patch_id
```

In this command,

patch_dir Is the source directory where patches reside on a patch server. The patch server can be the local or a remotely available machine.

patch_id Is a specific patch ID number, as in 102209-01.

This completes the procedure for copying a patch to the default spool directory on the OS server.

Verification of Copying Patches to a Patch Spool Directory

To verify the selected patches have been added to the default patch spool directory for the Solstice AutoClient product, use the `admclientpatch -p` command to see the list of currently spooled patches.

Examples of Copying Patches to a Patch Spool Directory

The following example copies the patch ID 100974-02 from a patch server named `cable` to the spool directory on the local (OS server) system, using the automounter:

```
# admclientpatch -a /net/cable/install/sparc/Patches/100974-02
Copying the following patch into spool area: 100974-02 . done
```

The following example copies the patch ID 102113-03 from a patch server named `cable` to the spool directory on the local (OS server) system, by mounting the patch server's patch directory on the local system:

```
# mount cable:/install/sparc/Patches /mnt
# admclientpatch -a /mnt/102113-03
Copying the following patch into spool area: 102113-03 . done
```

▼ How to Back Out a Patch from the OS Server's Patch Spool Directory

1. **Make sure you have your** `PATH` **environment variable updated to include** `/opt/SUNWadm/2.2/bin`**. For details, refer to the *Solstice AutoClient 2.0 Installation and Product Notes*.**

2. **Log in to the OS server and become root.**

3. **Back out patches to the default spool directory with this command:**

```
# admclientpatch -r patch_id
```

In this command,

patch_id Is a specific patch ID number, as in 102209-01.

This completes the procedure for backing out a patch from the default spool directory on the OS server.

Verification of Backing Out Patches from a Patch Spool Directory

To verify the selected patches have been backed out from the default patch spool directory for the Solstice AutoClient product, use the `admclientpatch -p` command to see the list of currently spooled patches.

Example of Backing Out Patches from a Patch Spool Directory

The following example backs out the patch ID 102209-01 from the default Solstice AutoClient spool directory.

```
# admclientpatch -r 102209-01
Unspooling the following patch: 102209-01
Removing the following patch from the spool area: 102209-01 .
```

▼ How to Synchronize Patches Installed on AutoClient Systems with Patches Spooled on the OS Server

1. **Make sure you have your `PATH` environment variable updated to include `/opt/SUNWadm/2.2/bin`. For details, refer to the *Solstice AutoClient 2.0 Installation and Product Notes*.**

2. **Log in to the OS server and become root.**

3. **Synchronize patches on clients with patches in the spool directory on the OS server.**

   ```
   # admclientpatch -s
   ```

   Using the `-s` option either installs or backs out patches running on clients, whichever is appropriate.

Note – It may be necessary to reboot your AutoClient systems after installing patches. If so, you can use the remote booting command, `admreboot`, to reboot the systems. For more information on this command, see the `admreboot(1M)` man page.

This completes the procedure synchronize patches on all clients.

Verification of Synchronizing Patches Running on Clients

To verify that the patches in the Solstice AutoClient patch spool directory are running on diskless clients and AutoClient systems, use the `admclientpatch` command with the `-c` option.

```
# admclientpatch -c
Clients currently installed are:
        rogue                   Solaris, 2.5, sparc
            Patches installed :  102906-01
OS Services available are:
        Solaris_2.5
            Patches installed :  102906-01
```

Example of Synchronizing Patches on Running Clients

The following command synchronizes all clients with the patches in the OS server's patch spool directory. The `-v` option reports whether `admclientpatch` is adding new patches or backing out unwanted patches.

```
# admclientpatch -s -v
Synchronizing service: Solaris_2.5
    Installing patches spooled but not installed
        102939-01      .....skipping; not applicable
Synchronizing client: rogue

All done synchronizing patches to existing clients and OS
services.
```

Updating Cached File Systems With Back File Systems

With the AutoClient technology, a new cache consistency mode has been added to the CacheFS consistency model. This consistency mode is called `demandconst`, which is a new option to the `cfsadmin(1M)` command. This mode assumes that files are generally not changed on the server, and that if they ever are changed, the system administrator will explicitly request a consistency check. So no consistency checking is performed unless a check is requested. There is an implied consistency check when a CacheFS file system is mounted (when the AutoClient system boots), and an AutoClient system is configured by default to request a consistency check every 24 hours. This model helps AutoClient performance by imposing less network load by performing less checking.

The risk of inconsistent data is minimal since the system's root area is exported only to that system. There is no cache inconsistency when the system modifies its own data since modifications are made through the cache. The only other way a system's root data can be modified is by root on the server.

The `/usr` file system is similar in that the server exports it as read-only, so the only way it could be modified is by the system administrator on the server. Use the `autosync(1m)` command to synchronize a system's cached file system with its corresponding back file systems.

You can update individual AutoClient systems, all local AutoClient systems in your network, or all AutoClient systems in a designated file, to match their corresponding back file systems. You should do this update when you add a new package in the shared `/usr` directory or in one or more system `/` (root) directories, or when you add a patch. The following procedures show how to use the `autosync(1M)` command. The command is issued from the server.

Requirements for Using the `autosync` Command

To use the `autosync` command, you need to be a member of the UNIX group, sysadmin (GID=14).

If you need to create the sysadmin group, see "Setting Up User Permissions to Use the Solstice AutoClient Software" on page 25.

▼ How to Update All AutoClient Systems With Their Back File Systems

Use the `autosync` command with no options to update all cached file systems on all the local AutoClient systems in your network.

```
% autosync
```

The system responds with the names of any systems that failed to be updated. No system response means the updates were all successful.

Example of Updating All AutoClient Systems With Their Back File Systems

The following example shows an update that failed on systems `pluto`, `genesis`, and `saturn`.

```
% autosync
pluto:: failed:
genesis:: failed:
saturn:: failed:
```

Verification

If there is no system response, all updates are successful.

How to Update a Single AutoClient System With Its Back File System

Use the `autosync` command with the `-h` option to update all cached file systems on a specified AutoClient system in your network:

```
% autosync -h hostname
```

In this command,

`-h` Specifies one system.

hostname Is the name of the system whose cache you want to update.

Example of Updating One AutoClient System With its Back File System

The following example shows how to update all cached file systems on the AutoClient system `pluto`:

```
% autosync -h pluto
```

If the system failed to be updated, you would get the following system response:

```
% autosync -h pluto
pluto:: failed:
```

Verification

If there is no system response, all updates are successful.

▼ How to Update a Specific File System on an AutoClient System

Use the `autosync` command as follows to synchronize a specific file system on an AutoClient system with its back file system:

```
% autosync -h hostname cached-filesystem
```

In this command,

`-h`	Specifies one system.
hostname	Is the name of the system whose cache you want to update.
cached-filesystem	Is the name of the cached file system you want to update.

Example of Updating a Specific File System on an AutoClient System

The following example shows how to update the cached file system `/usr/share` on the AutoClient system `foo`:

```
% autosync -h foo /usr/share
```

▼ How to Update More Than One AutoClient System With Its Back File System

1. **Create a file containing the names of the systems you want to synchronize with their back file systems.**
   The file can be located anywhere. For example, you could put the file in `/tmp` or `/home`. If you run the `autosync` command without arguments and several systems fail to update, put the names of the systems that failed to update in this file. For example, enter one name per line.

2. **Use the `autosync` command as follows to update all AutoClient systems in the *host_file* file.**

   ```
   % autosync -H host_file
   ```

   In this command,

   `-H` Specifies a file containing the names of all AutoClient systems to update.

   host_file Is the name of the file containing the names of all AutoClient systems in the network you want to update.

Example of Updating More Than One AutoClient System Using a File

The following example shows how to update all AutoClient systems in the host file `net_hosts`:

```
% autosync -H net_hosts
```

For example, the contents of `net_hosts` might be:

```
mars
jupiter
saturn
```

▼ How to Update an AutoClient System From the System Itself

Use the autosync command as follows to update all cached file systems on a AutoClient system. This command is used on the system itself, and not the server:

```
% autosync -l
```

You can also specify a particular file system on the system that requires updating.

Example of Updating an AutoClient System From the System Itself

The following example shows how a client requests update of its own /usr/share file system:

```
% autosync -l /usr/share
```

Replacing a Faulty AutoClient System

Since an AutoClient system contains no permanent data, it is an FRU. An FRU can be physically replaced by another compatible system without loss of permanent data. So, if an AutoClient system fails, you can use the following procedure to replace it without the user losing data or wasting a lot of time.

Note – If you replace only the disks or another part of the system, and the Ethernet address stays the same, you must use the `boot -f` command to reboot the system so that the cache is reconstructed.

FRU Restrictions

You cannot switch kernel architectures or OS releases from the original configuration.

▼ How to Replace a Faulty AutoClient System

1. **If the system is currently running, use the `halt` command to get it to the prom monitor environment and turn it off.**
2. **Disconnect the faulty AutoClient system from the network.**
3. **Connect the replacement AutoClient system onto the network.**
   The replacement AutoClient system must have the same kernel architecture as the faulty AutoClient system.
4. **Start Host Manager from the Solstice Launcher on the AutoClient system's server, and select the name service, if not done already.**
   See "How to Start Host Manager" on page 64 for more information.
5. **Select the faulty AutoClient system you wish to modify from the main window.**
6. **Choose Modify from the Edit menu.**
   The Modify window appears with fields filled in specific to the AutoClient system you selected.
7. **Modify the Ethernet address and the disk configuration to be that of the new AutoClient system.**

8. **Click on OK.**

9. **Choose Save Changes from the File menu.**

10. **Turn on the new system.**

11. **If the screen displays the > prompt instead of the `ok` prompt, type `n` and press Return.**
    The screen should now display the `ok` prompt.

Note – This step is not required for Sun-4 systems, because they do not have the `ok` prompt.

---

12. **Boot the AutoClient system with the following command:**

If the AutoClient System Is A ...	Then Enter ...
Sun4/3*nn*	`b le()`
Sun4/1*nn* Sun4/2*nn* Sun4/4*nn*	`b ie()`
x86	See “x86: Booting From the Network” on page 110.
All other Sun systems	`boot net`

13. **After the AutoClient system boots, log in as root.**

14. **Set the AutoClient system’s default boot device to the network by referring to “SPARC: How to Set Up a System to Automatically Boot From the Network” on page 104.**

Note – This step is necessary for an AutoClient system, because it must always boot from the network. For example, an AutoClient system should automatically boot from the network after a power failure.

Example of a Command-Line Equivalent for Replacing a Faulty AutoClient System

The following command is equivalent to using Host Manager to modify the Ethernet address for an AutoClient system.

```
% admhostadd -e ethernet_address host_name
```

The following command is equivalent to using Host Manager to modify the disk configuration for an AutoClient system.

```
% admhostadd -x diskconf=disk_config host_name
```

For more information on disk configuration options, see Table 6-3 on page 73.

Packing Files in the Cache

You can use the `cachefspack` command to pack an AutoClient system's cache with specific *cached* files and directories, which means that they will always be in the system's cache and not removed when the cache becomes full. The files and/or directories that you pack in your cache must be from a cached file system, which means they must be under the root (`/`) or `/usr` file systems for AutoClient systems.

Note – If you set up your AutoClient system with the disconnectable option, you will have the added benefit of continued access to your cache and the packed files if the server becomes unavailable. For more information on the disconnectable option, see Table 6-2 on page 71.

▼ How to Pack Files in the Cache

Pack files in the cache using the `cachefspack` command.

```
$ cachefspack -p filename
```

In this command,

`-p` Specifies that you want the file or files packed. This is also the default.

`filename` Specifies the name of the cached file or directory you want packed in the cache. When you specify a directory to be packed, all of its subdirectories are also packed. For more information about the `cachefspack` command, see the man page.

Examples

The following example specifies the file `cm` (Calendar Manager) to be packed in the cache.

```
$ cachefspack -p /usr/openwin/bin/cm
```

The following example shows several files specified to be packed in the cache.

```
$ cachefspack -p /usr/openwin/bin/xcolor /usr/openwin/bin/xview
```

The following example shows a directory specified to be packed in the cache.

```
$ cachefspack -p /usr/openwin/bin
```

Unpacking Files

You may need to unpack a file from the cache. For example, if you have other files or directories that are a higher priority than others, you can unpack the less critical files.

▼ How to Unpack Files in the Cache

Unpack individual files in the cache using the `-u` option of the `cachefspack` command.

```
$ cachefspack -u filename
```

In this command,

`-u` Specifies that you want the file or files unpacked.

`filename` Is the name of the file or files you want unpacked in the cache. For more information about the cachefspack command, see the man page.

Unpack all the files in a cache directory using the `-U` option of the `cachefspack` command.

```
$ cachefspack -U cache_directory
```

In this command,

`-U` Specifies that you want to unpack all packed files in the specified cached directory.

cache_directory Is the name of the cache directory that you want unpacked from the cache. For more information about the `cachefspack` command, see the man page.

Examples

The following example shows the file /usr/openwin/bin/xlogo specified to be unpacked from the cache.

```
$ cachefspack -u /usr/openwin/bin/xlogo
```

The following example shows several files specified to be unpacked from the cache.

```
$ cachefspack -u /usr/openwin/bin/xview /usr/openwin/bin/xcolor
```

The following example uses the -U option to specify all files in a cache directory to be unpacked.

```
$ cachefspack -U /usr/openwin/bin
```

You cannot unpack a cache that does not have at least one file system mounted. With the -U option, if you specify a cache that does not contain mounted file systems, you will see output similar to the following:

```
$ cachefspack -U /local/mycache
cachefspack: Could not unpack cache /local/mycache, no mounted
filesystems in the cache.
```

Displaying Packed Files Information

You may want to view information about the files that you've specified to be packed, and what their packing status is.

▼ How to Display Packed Files Information

To display information about packed files and directories, use the `-i` option of the `cachefspack` command, as follows:

```
$ cachefspack -i cached-filename-or-directory
```

In this command,

`-i` Specifies you want to view information about your packed files.

cached-filename-or-directory Is the name of the file or directory for which to display information.

Examples

The following example shows that a file called `ttce2xdr.1m` is marked to be packed, and it is in the cache.

```
# cachefspack -i /usr/openwin/man/man1m/ttce2xdr.1m
cachefspack: file /usr/openwin/man/man1m/ttce2xdr.1m marked
packed YES, packed YES
.
.
.
```

The following example shows a directory called `/usr/openwin`, which contains a subdirectory `bin`. Three of the files in the `bin` subdirectory are: `xterm`, `textedit`, and `resize`. The file `textedit` is specified to be packed, but it is not in the cache. The file `textedit` is specified to be packed, and it is in the cache. The file `resize` is specified to be packed, but it is not in the cache.

```
$ cachefspack -i /usr/openwin/bin
.
.
.
cachefspack: file /bin/xterm marked packed YES, packed NO
cachefspack: file /bin/textedit marked packed YES,packed YES
cachefspack: file /bin/resize marked packed YES,packed NO
.
.
.
```

Glossary

authentication

A process where the sadmind daemon must verify the identity of the user making a system administration request across the network.

authorization

A process where the sadmind daemon verifies that the authenticated user has permission to execute the Solstice AutoClient software on the server. After the user identity is verified, the sadmind daemon uses the user identity to perform authorization checks.

AutoClient system

A system with a monitor and keyboard, CPU and memory, Ethernet hardware, and a small disk (at least 100-Mbyte) to cache its root (/) and /usr file systems from a server on a network. This system gets it other file resources from a server on the network. Has the advantage of eliminating the need to administer the local disk, and results in good performance for end users.

back file system

A term used in the Solaris CacheFS environment to describe the file system on the server that is mounted onto a client's disk cache.

boot server

A system that provides the programs and information a client needs to boot.

bootparams **file**

A file containing entries that are used to enable client systems to boot from the network.

cache

A local storage area for data.

cached file system

A local file system that stores files in the cache as they are referenced.

CacheFS

An optional Solaris file system type used to improve the speed of access to remote or slow file systems. Accesses file systems from the server on an as-needed basis and caches them to a specified part of the local disk drive. CacheFS is a standard feature of the Solaris 2.3 and later releases.

consistency checking

The process of ensuring that the two copies of data—the copy on the server and the copy on the client—are the same. The Solstice AutoClient product maintains consistency in two ways: by passing disk write operations through the AutoClient system's cache, and by periodically passing updated server file systems to the cache.

dataless client

A system with a monitor and keyboard, CPU and memory, Ethernet hardware, and small local disk for the swap area and the root (`/`) file system. This system gets it other file resources from a server on the network.

default

An assumed value, or an action taken automatically unless you specify otherwise.

device

A hardware component, such as a printer or disk drive, acting as a unit to perform a specific function.

diskless client

A system with a monitor and keyboard, CPU and memory, and Ethernet hardware. This system gets it file resources and swap space from a server on a network.

Ethernet address

A system's hardware address. The Ethernet address can be displayed using the `banner` command from the PROM level.

`ethers` **file**

A file containing Ethernet addresses of network client systems.

file server

A system that shares file resources and disk storage space for network clients.

file system

A hierarchy of files and directories in the Solaris operating environment.

front file system

A term used in the CacheFS environment to describe the file system on the client.

generic system

A system that is not running the Solaris software, or whose type has not yet been updated using Host Manager's Update System Type feature, or uses local or loghost entries in the system management databases.

group

A collection of users who share files and other system resources. Each user belongs to a primary group (listed in the user's `passwd` entry), and optionally, one or more secondary groups.

`group` **file**

A file containing entries for UNIX groups. The `group` file is accessed from Group Manager.

group ID (GID)

A group identification number used by the system to identify a user's primary group. Group ID numbers for users usually range from 100 to 60000.

Group Manager

One of Solstice AdminSuite's applications used to manage group information in the `group` file.

Host Manager

One of Solstice AdminSuite's applications used to manage network client services.

host name

A unique name that identifies a system.

install server

A system on the network that provides a Solaris CD image (either from a CD-ROM drive or a copy on hard disk) for other systems to install from.

IP address

A system's unique network address.

launcher

See Solstice AdminSuite Launcher.

media server

A system that shares a CD-ROM device for remote installation of software.

name service

Method by which system information is maintained in the network. There are three selections in Solstice AdminSuite: NIS, NIS+, and None.

- **NIS** – Name service shipped with the SunOS 4.1.x operating system (Solaris 1). Designated systems, called NIS servers, contain maps that store information about the network, its clients, and its users.
- **NIS+** – Name service shipped with the Solaris software. Making use of true databases (instead of two-column maps that simply associate one variable with another), NIS+ stores more information than NIS.
- **None** – Method for administrators of networks that do not use a network name service; administrators usually select one system on the network on which to maintain a master copy of the `/etc` configuration files.

name service domain

A group of systems and the information served to those systems.

network client

A system that uses remote resources from a server.

`nsswitch.conf`

A file that contains an entry for each system file and a corresponding name service source to search for the system file information. The name service sources are designated as keywords— `nis`, `nisplus`, or `files`. If more than one name service source is listed, they are searched in the order given.

OpenWindows

A windowing system based on the OPEN LOOK graphical user interface.

OS server

A server that provides OS services to support diskless clients, dataless clients, and AutoClient systems.

OS services

OS software that you can add to an OS server for it to support clients of other platform groups and Solaris releases. You can also add services for clients that are the same platform group and require the same Solaris release as the OS server.

profile server

A system that contains JumpStart files for systems to perform a custom JumpStart installation.

PROM

A programmable read-only memory chip with a program called the monitor that runs a quick self-test procedure and checks such things as the hardware and memory on the system. If no errors are found, the system begins the automatic boot process.

PROM prompt

The prompt displayed when the system halts; either > or ok.

root

A user who has access to all parts of the system. This is usually the system administrator. Also known as *superuser*.

sadmind **daemon**

A distributed system administration daemon that carries out security tasks when administrative tasks are performed across the network.

secondary group

Membership in this group is defined by the group identifier listed in the group file with a list of users as members.

shell

A command-line interpreter program that accepts and executes commands that you type. There are several varieties of shell programs, and three are included in the Solaris software: Bourne, Korn, and C.

Solstice AdminSuite

A graphical user interface used to perform administrative tasks such as managing users, groups, hosts, printers, and serial devices.

Solstice AutoClient

A graphical user interface used to perform administrative tasks on AutoClient systems.

Solstice Launcher

The base window of the interface, used to start the other application tools.

standalone system

A system with a monitor and keyboard, CPU and memory, approximately 200 or more megabytes of disk space, and usually a backup device. It may or may not be connected to a network.

superuser

A user who has access to all parts of the system. This is usually the system administrator. Also known as *root*.

sysadmin group

The UNIX group whose members belong to the sysadmin group (Group 14). Members of the sysadmin group can use Solstice AdminSuite's applications locally or remotely.

`timezone` **file**

A file containing entries for systems and their geographic region and time zone.

user ID (UID)

A number used by the operating system to identify a user. User ID numbers for users usually range from 100 to 60000.

write-through cache

A cache that immediately updates its back file system as data is changed or added to the cache.

`ypbind`

An NIS daemon process that runs on all client systems and allows the client to communicate with an NIS server.

Index

Les logiciels tiers, y compris les technologies relatives aux polices de caractères, inclus dans ce produit sont protégés par des droits d'auteur et exploités sous licences octroyées par les fournisseurs de Sun.

DROITS LIMITES : L'exploitation, la duplication et la distribution de ce produit par le Gouvernement des Etats-Unis sont sujettes aux restrictions décrites à l'alinéa (c) (1) (ii) de la clause Rights In Technical Data and Computer Software des publications DFARS 252.227-7013 et FAR 52.227-19.

Le produit décrit dans ce manuel peut faire l'objet d'un ou de plusieurs brevets américains, brevets étrangers ou demandes de brevet en instance.

MARQUES
Sun, Sun Microsystems, le logo Sun, SunSoft, le logo SunSoft, Solaris, Solstice, SunOS, OpenWindows, DeskSet, ONC, ONC+, NFS, AutoClient, CacheFS, Power Management, Sun-4, et JumpStart sont des marques ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis d'Amérique et dans certains autres pays. UNIX est une marque déposée aux Etats-Unis d'Amérique et dans d'autres pays, exclusivement licenciée par X/Open Company, Ltd. OPEN LOOK est une marque déposée de Novell, Inc. PostScript et Display PostScript sont des marques déposées d'Adobe Systems, Inc. La marque PowerPC est une marque d'International Business Machines. Tous les autres noms de produit, de service ou de société mentionnés dans ce document sont les marques de leurs détenteurs respectifs.

Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. aux Etats-Unis d'Amérique et dans certains autres pays. Les produits portant la marque SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

Les interfaces utilisateur graphiques OPEN LOOK et Sun ont été développées par Sun Microsystems, Inc. à l'intention de ses utilisateurs et détenteurs de licences. Sun reconnaît la contribution originale de Xerox à la recherche et au développement du concept d'interfaces visuelles ou d'interfaces utilisateur graphiques destinées à l'industrie informatique. Sun détient une licence non exclusive octroyée par Xerox permettant l'utilisation de l'interface graphique utilisateur Xerox ; cette licence couvre également les détenteurs de licences Sun qui mettent en application des interfaces utilisateurs graphique OPEN LOOK et adhèrent par ailleurs aux accords de licence écrit de Sun.

Le système X Window est un produit de X Consortium Inc.

CETTE PUBLICATION EST FOURNIE «TELLE QUELLE» SANS AUCUNE GARANTIE EXPRESSE OU IMPLICITE, Y COMPRIS MAIS NE SE LIMITANT PAS AUX GARANTIES IMPLICITES DU CARACTÈRE ADÉQUAT POUR LA COMMERCIALISATION OU UN USAGE PARTICULIER OU DE NON CONTREFAÇON.

CETTE PUBLICATION PEUT CONTENIR DES ERREURS TECHNIQUES OU TYPOGRAPHIQUES. DES MODIFICATIONS SONT CONSTAMMENT APPORTÉES AUX INFORMATIONS CI-INCLUSES. CES MODIFICATIONS SERONT INCORPORÉES AUX ÉDITIONS ULTÉRIEURES DE CETTE PUBLICATION. SUN MICROSYSTEMS, INC. SE RÉSERVE LE DROIT D'APPORTER, À TOUT MOMENT, TOUTE AMÉLIORATION ET/OU MODIFICATION AU(X) PRODUIT(S) ET/OU PROGRAMME(S) DÉCRIT(S) DANS CETTE PUBLICATION.