Ignite-UX Administration Guide
for HP-UX 11i

HP Part Number: B3921-90032
Published: September 2010
Edition: 35

Acknowledgements

Intel® Itanium® Logo, Intel, Intel Inside and Itanium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

Java® is a US trademark of Sun Microsystems, Inc.

UNIX® is a registered trademark of The Open Group.

Revision History

Table 1 Revision History

Document Manufacturing Part Number	Operating Systems Supported	Edition No.	Publication Date
B3921–90032	HP-UX 11i v1, 11i v2, 11i v3	35	September 2010
B3921–90006	HP-UX 11i v1, 11i v2, 11i v3	34	March 2010
5992–6584	HP-UX 11i v1, 11i v2, 11i v3	33	September 2009
5992-5309	HP-UX 11i v1, 11i v2, 11i v3	32	November 2008
5992–4731	HP-UX 11i v1, 11i v2, 11i v3	31	September 2008
5992–3336	HP-UX 11i v1, 11i v2, 11i v3	30	March 2008
5992–1959	HP-UX 11i v1, 11i v2, 11i v3	29	December 2007
5992–0602	HP-UX 11i v1, 11i v2, 11i v3	28	September 2007
5991–7999	HP-UX 11.00, 11i v1, 11i v2, 11i v3	27	June 2007
5991-6440	HP-UX 11.00, 11i v1, 11i v2, 11i v3	26	February 2007
B2355-91049	HP-UX 11.00, 11i v1, 11i v2	25	December 2006
B2355-90997	HP-UX 11.00, 11i v1, 11i v2	24	September 2006
B2355-90970	HP-UX 11.00, 11i v1, 11i v2	23	June 2006
B2355-90959	HP-UX 11.00, 11i v1, 11i v2	22	March 2006
B2355-90941	HP-UX 11.00, 11i v1, 11i v2	21	December 2005
B2355-90893	HP-UX 11.00, 11i v1, 11i v1.6, 11i v2	20	September 2005
B2355-90875	HP-UX 11.00, 11i v1, 11i v1.6, 11i v2	19	June 2005
B2355-90872	HP-UX 11.00, 11i v1, 11i v1.6, 11i v2	18	December 2004
B2355-90849	HP-UX 11.00, 11i v1, 11i v1.6, 11i v2	17	September 2004
B2355-90837	HP-UX 11.00, 11i v1, 11i v1.6, 11i v2	16	June 2004
B2355-90834	HP-UX 11.00, 11i v1, 11i v1.6, 11i v2	15	March 2004
B2355-90831	HP-UX 11.00, 11i v1, 11i v1.6, 11i v2	14	December 2003
B2355-90788	HP-UX 11.00, 11i v1, 11i v1.6, 11i v2	13	September 2003
B2355-90829	HP-UX 11.00, 11i v1, 11i v1.6	12	September 2003

Table 1 Revision History *(continued)*

Document Manufacturing Part Number	Operating Systems Supported	Edition No.	Publication Date
B2355-90810	HP-UX 11.00, 11i v1, 11i v1.5, 11i v1.6	11	June 2003
B2355-90772	HP-UX 10.x, 11.00, 11i v1 , 11i v1.5, 11i v1.6	10	March 2003
B2355-90767	HP-UX 10.x, 11.00, 11i v1, 11i v1.5, 11i v1.6	9	December 2002
B2355-90770	HP-UX 10.x, 11.00, 11i v1, 11i v1.5, 11i v1.6	8	October 2002
B2355-90765	HP-UX 10.x, 11.00, 11i v1, 11i v1.5, 11i v1.6	7	September 2002
B2355-90758	HP-UX 10.x, 11.00, 11i v1, 11i v1.5	6	September 2002
B2355-90750	HP-UX 10.x, 11.00, 11i v1, 11i v1.5	5	June 2002
B2355-90749	HP-UX 10.x, 11.00, 11i v1, 11i v1.5	4	March 2002
B2355-90738	HP-UX 10.x, 11.00, 11i v1, 11i v1.5	3	June 2001
B2355-90704	HP-UX 10.x, 11.00, 11i v1	2	December 2000
B2355-90677	HP-UX 10.x, 11.00, 11i v1	1	March 1999

Table of Contents

List of Figures

List of Tables

1 Ignite-UX Overview

Welcome to Ignite-UX!

This chapter contains information for new and experienced users alike.

Introductory information:

Details about Ignite-UX:

Ignite-UX Features

Client and Server Control

The *installation* sessions for multiple targets can be controlled from a single *Ignite-UX server* in a true *client*/server model. A GUI is provided to run on the server and manage multiple simultaneous client installation sessions. Alternatively, a single installation session can be controlled from the client machine. A single Ignite-UX installation server can serve multiple releases of HP-UX for different clients.

Easy-to-Use GUI

The Ignite-UX GUI uses tabs and dialog boxes for task navigation. The Ignite-UX GUI only runs on an *Ignite-UX server*.

Terminal User Interface

Ignite-UX uses a terminal user interface (TUI) with keyboard navigation when run from a *client*. Ignite may also be run in TUI mode from the server.

Command Line Interface

Commands that power Ignite-UX can be executed directly from the operating system's command shell on an *Ignite-UX server* or *client*. For the list of commands, see "Ignite-UX Commands and Manpages" (page 20).

Multi-Sourced Installations

*Installation*s can use multiple Software Distributor (SD) *depot*s in a single installation session. For example, you could install your base OS from one SD depot, a set of patches from another SD depot, and the applications you want from a third SD depot; all in one session.

Multiple Archive Formats

Ignite-UX supports `tar`, `cpio`, and `pax` format *archive*s. (To use the `pax` format with 11i v2, you must have the PAX-Enh product installed. The `pax` format is not available for 11i v1.) Tools are provided to help you create a *golden image* if you wish to install from an archive. You can use one archive along with one or more *depot*s containing patches or additional software.

One-Step Installation

Once you configure a system with a *common configuration* you want replicated to other systems, use Ignite-UX to either manually or automatically install each *client* system. This common configuration can include any supported HP-UX 11i operating system, and you can add any required patches and applications.

Custom Installations

It is easy to create a system that is ready to go as soon as the *installation* session completes. Many of the tasks that are typically done as separate steps after an installation have been incorporated into the installation process. Ignite-UX allows you to specify *kernel* parameters you want set and user-supplied scripts you would like to run as part of the session. In addition, the host and networking information normally supplied at first *boot* can be specified at install time.

Golden Images

A system that has been installed and tuned may be used to create an image. That image may be used as a custom configuration that may be applied in *installation*s to other systems.

Automated Installations

Set up a configuration and then install it on a *client* with no further user interaction. This is possible for both the initial *installation* and the reinstallation cases.

Create a System Manifest

Scan a system and produce a report detailing what hardware is present, how the disks are used, what *kernel* modifications have been made, and what software has been installed. This report can be customized to meet your needs.

Create Custom Installation Media

Construct your own customized, bootable *installation media*. An example script, `make_media_install`, is provided that can help you create bootable media (tapes, CDs, and DVDs) with or without *golden archive*s and SD *depot*s. The example script can be found at `/opt/ignite/data/scripts/examples/make_media_install`.

System Recovery

Ignite-UX provides consistent, reliable recovery in the event of catastrophic hardware or software failure by creating *recovery image*s on tape (with *client* access to a tape drive) or on any *Ignite-UX server* in your environment (with client access to the network).

Support for Multiple Architectures

Ignite-UX supports both the Precision Architecture Reduced Instruction Set Computing (PA-RISC) and the Intel®Itanium® (Itanium®-based) hardware architectures.

Support for HP Servicecontrol Manager

Ignite-UX supports installing HP-UX *client*s in the HP Servicecontrol Manager environment. For details, see the *HP Servicecontrol Manager 3.0 User's Guide*.

Support for New Hardware

Each new release of the Ignite-UX product supports the new hardware included in the corresponding release of HP-UX.

Getting the Ignite-UX Software

Ignite-UX is available in standard SD (Software Distributor) *depot* format from OE and AR media, and from the HP Software Depot Website.

Any Ignite-UX *bundle* is safe to install at any time. None of the *fileset*s in Ignite-UX bundles will cause a reboot to occur.

- **OE and AR Media**

  Ignite-UX released on OE or AR media can only be installed on a server running the HP-UX version supported by the OE or AR media.

  This Ignite-UX is the complete product. (The Ignite complete product is capable of installing and recovering all supported versions of HP-UX.)

  If you require a version of Ignite-UX that can be installed onto any supported version of HP-UX, read the next section about downloading Ignite-UX from the HP Software Depot Website.

- **HP Software Depot Website**

  Follow this link for Ignite-UX on HP Software Depot: **http://www.hp.com/go/ignite-ux-download**

  The Ignite-UX depots available at Software Depot contain the latest Ignite-UX version and can be installed on servers running any supported version of HP-UX.

- **Support for Installation and Recovery of all Supported HP-UX Operating System Versions**

  Each Ignite-UX bundle contains the Ignite-UX tools, plus the data files required to install and recover the particular HP-UX operating systems indicated by the bundle name.

  See the figure below for a list of available bundles and the HP-UX versions the bundles can install and recover.

Figure 1-1 Ignite-UX Bundles Available in the Ignite-UX Product

Depot Name	Bundle Name	HP-UX versions the bundle can install/recover on client		
		11 iv1	11 iv2	11 iv3
Ignite-UX-11-ALL_C.7x.xx.depot (Complete Ignite-UX product)	IGNITE	✓	✓	✓
Ignite-UX-11-11 _C.7x.xx_ HP_-UX_B.11.11 _32+64.depot	Ignite-UX-11-11	✓		
Ignite-UX-11-23_C.7x.xx_ HP_-UX_B.11.11 _32+64.depot	Ignite-UX-11-23		✓	
Ignite-UX-11-31 _C.7x.xx_ HP_-UX_B.11.11 _32+64.depot	Ignite-UX-11-31			✓

NOTE: As of Ignite-UX version C.7.1, the name of the Ignite-UX complete product bundle that installs all supported versions of HP-UX has changed from **B5725AA** to **IGNITE**.

Each bundle can be installed on a server running any version of HP-UX. For example, **Ignite-UX-11-23** can be installed on a server running HP-UX 11i v1 (B.11.11). You can install one or more of the individual **Ignite-UX-11-*xx*** bundles onto your system.

HP recommends you install the complete Ignite-UX product (**IGNITE**) unless you want to block the use of a specific version of HP-UX, increase the download speed from the Software Depot website, or conserve disk space on the server.

As a best practice, do not `swremove` Ignite-UX before updating to a new version. Doing so will cause some files to be reset, including the `INDEX` file, thus you will lose any customizations.

IMPORTANT: Installing individual bundles instead of the complete product might cause problems for Ignite-UX if the complete product was installed previously. Refer to the *Installing and Updating Ignite-UX* white paper if you are unsure of what to install to upgrade Ignite-UX. Links to the Ignite-UX white papers are found at http://www.hp.com/go/ignite-ux-docs.

Ignite-UX Commands and Manpages

The manual pages (manpages) associated with Ignite-UX commands are in the `/opt/ignite/share/doc/` directory, are available in the *HP-UX Reference* at http://www.hp.com/go/ignite-ux-docs, and are listed in Table 1-1 according to the directory the commands are in.

Table 1-1 Ignite-UX Command Manpages

Ignite-UX Command Manpages	Description
Commands in `/opt/ignite/bin`:	
add_new_client(1M)	Add a *client* to an *Ignite-UX server* without requiring a client *boot* from the Ignite-UX server.
auto_adm(1M) *auto_adm*(4)	Manage *logical interchange format* (LIF) *AUTO configuration file*s. Description of `auto_adm` file formats
bootsys(1M)	Reboot and install systems using Ignite-UX.
check_net_recovery(1M)	Compare the files on a running system with a *recovery archive* made with `make_net_recovery`.
check_tape_recovery(1M)	Compare the files on a running system with a *recovery archive* made with `make_tape_recovery`.
copy_boot_tape(1M)	Replicate a PA-RISC *boot* tape.
ignite(5)	Configure, install, and recover HP-UX systems.
instl_adm(1M) *instl_adm*(4)	Manage Ignite-UX *configuration file*s. Description of configuration file syntax.
instl_dbg(1M)	Parse and debug a *client*'s *configuration file*s.
make_boot_tape(1M)	Create a *boot* tape for a PA-RISC system.
make_bundles(1M)	Create Software Distributor (SD) *bundle*s in a *depot*.
make_config(1M)	Generate a *configuration file* for software in an SD *depot*.
make_depots(1M)	Create SD *depot*s from SD *bundle*s for use by Ignite-UX.
make_ipf_tape(1M)	Create a bootable ANSI labeled tape for Itanium-based systems.
make_medialif(1M)	Create bootable Ignite-UX *LIFmedia image* file.
make_net_recovery(1M)	Create *recovery image*s and store them on a network system.
make_tape_recovery(1M)	Create *recovery image*s and store them on tape.
manage_index(1M)	Manage Ignite-UX *INDEX* files without directly editing them.
print_manifest(1M)	Print a system *manifest*.
save_config(1M)	Create hardware *configuration file*.
Commands in `/opt/ignite/lbin`:	
ansitape(1M) *ansitape*(5)	Read and write magnetic tapes conforming to the ANSI standard for magnetic tape labelling. Description of ANSI-labeled tape format.

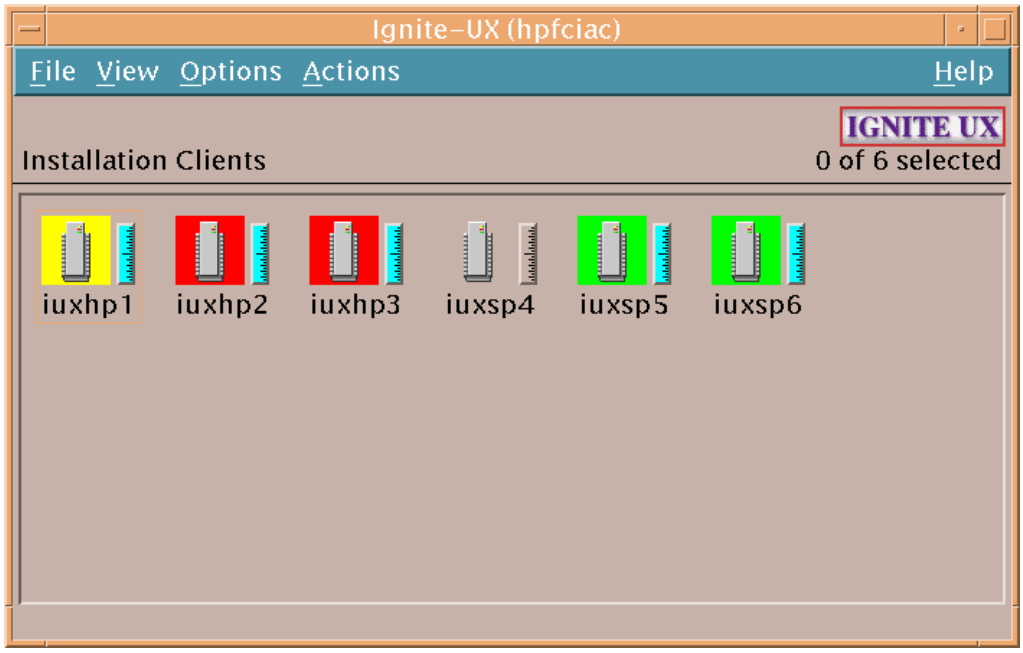
Table 1-1 Ignite-UX Command Manpages *(continued)*

Ignite-UX Command Manpages	Description
archive_impact(1M)	Calculate the per *file system* disk space for tar, cpio, and tar *archive*s, and create the `impacts` statements for use in *configuration file*s.
instl_bootd(1M)	*Boot* protocol server for Ignite-UX *client*s.
instl_combine(1M)	Combine a *LIF volume* and *file system* for use on CD/DVD. This command is used to construct custom, bootable, *installation media*. An example script, `/opt/ignite/data/scripts/examples/make_media_install`, is provided that can help you create bootable media (PA-RISC tapes, CDs, and DVDs) with or without *golden archive*s and/or SD *depot*s included.
pkg_rec_depot(1M)	Create a *depot* containing Ignite-UX recovery *fileset*s.
setup_server(1M)	Perform some administration tasks for an *Ignite-UX server*.
Commands in `/opt/ignite/data/scripts`:	
make_sys_image(1M)	Create an *archive* of a *client*.

Introduction to the Ignite-UX Graphical User Interface

The Ignite-UX GUI workspace provides access to all management tasks using the menu bar and context-sensitive menus.

Figure 1-2 Ignite-UX GUI



The Ignite-UX GUI workspace graphically represents *client*s as icons labeled with the clients' hostnames. You can:

- Click a client icon to select it for further actions.
- Double-click the client icon to display the Client Status dialog box.
- Right-click to activate the Actions menu. You must select the client before right-clicking; any selections made from the Actions menu apply to the selected client.

For more information about these actions, see Chapter 10: "Booting and Installing HP-UX on Clients Using the Server" (page 125), or click **Help**.

Each client's *installation* status is indicated by the colored border around its icon, and the installation gauge shows the relative progress:

- **Green:** The operating system is completely installed, booted, and running with no errors or warnings.
- **Yellow:** A warning condition exists and should be investigated.
- **Red:** An error condition is present. The operating system is partially installed, or the installation has stopped.
- **No color:** Installation has not yet started or the client has been stopped.

Client icons are shown for all booted clients and those that can be used as recovery systems. These systems are known to Ignite-UX by their existence in the `/var/opt/ignite/clients` file.

File Menu

The **File** menu contains basic Ignite-UX functionality:

- **Search -** Find clients that match a text string.
- **Print -** To print a listing of systems, the display must be set with **View->By Properties**
- **Exit -** Quit Ignite-UX.

View Menu

Use the **View** menu to customize the Ignite-UX GUI display:

- **Columns -** Choose which client attributes to display in which column. These selections are apparent only when the object list is displayed by properties.
- **Filter -** View a subset of clients by selected criteria.
- **Sort -** Orders the displayed clients by sort criteria.
- **By Name and Icon -** Displays clients graphically.
- **By Properties -** Displays clients in a text format rather than in the default graphical representation.

TIP: Using the **By Properties** view and sorting the list makes it easier to scan for clients that have finished installing. For example, to view the clients by the percentage of completion, select **View->Sort->% Complete: Descending**. The list of clients will then appear with the clients closest to completion first, as shown in Figure 1-3.

Figure 1-3 Ignite-UX GUI By Properties View

- **Save View as Default -** Saves the current Ignite-UX GUI **View** settings.

Options Menu

Use the **Options** menu to set server configuration variables and to control the refresh rate of the Ignite-UX display.

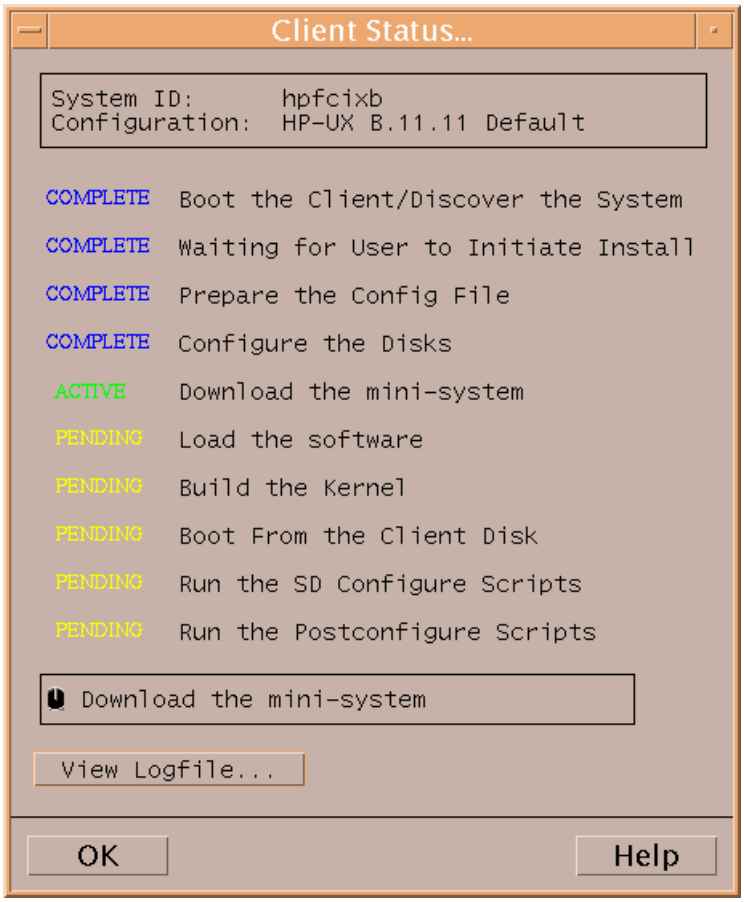
- **Server Configuration -** Identify and set up your installation server. The selections here are covered in detail in "More Server Setup Options" (page 43).
- **Change Refresh Interval -** Select how frequently you want the client display updated.
- **Refresh List -** Update the client display immediately.

Actions Menu

To view available actions for a client, select its icon, then select the **Actions** menu. The actions displayed are dependent on the status of the client, so all actions may not be available. You can use the following actions to manage clients:

- **View Install History... -** Lists details of all successfully installed clients.
- **Boot Client... -** *Boot*s the selected client. If no client is selected, one will be prompted for.
- **Add New Client for Recovery... -** Selects a client to be recovered. For more information, see "Adding Clients for Recovery " (page 227).
- **Run Tutorial/Server Setup... -** Displays the Welcome dialog box. From there, you can run the **Tutorial and Demo**, or click **Server Setup...** to launch the Server Setup Wizard.
- **Client Status... -** The status of the selected client is polled and displayed, as in Figure 1-4 (page 24).

Figure 1-4 Client Status Dialog Box



- **Install Client -** Starts the HP-UX installation process for the selected client. This process is explained in Chapter 10: "Booting and Installing HP-UX on Clients Using the Server" (page 125).
- **Stop Install... -** Stops the installation process on the selected client. After stopping an install, you can reboot or halt the client.
- **Create Network Recovery Archive -** Creates a network *recovery image* using the `make_net_recovery` command. See Chapter 15: "Recovery" (page 209) for more information.
- **Create Tape Recovery Archive -** Creates a recovery image using the `make_tape_recovery` command. See Chapter 15: "Recovery" (page 209) for more information.
- **Move to History... -** Saves critical files for the client, adds them to the history file, and removes the client. The client installation must successfully complete for the configuration to be moved to the history file.
- **Remove Client... -** Deletes the selected client configuration. All client data, except for the *recovery image*, is removed. Recovery information in the client's directories will be removed.
- **View Hardware... -** Lists the hardware associated with the selected client.
- **View/Print Manifest... -** Allows you to view and print the *manifest* for the selected client. The manifest file details the client's installation and is available on the client and Ignite-UX server after the installation. For more information, see "Viewing and Printing a Manifest " (page 166).
- **Change Icon Name... -** Launches a dialog box for renaming the selected icon.

How Ignite Works

When deciding the best way to use Ignite in your data center, it might be useful to understand the structure of Ignite – how it gets started on the client and the functional steps it performs. This section describes the major components of Ignite and where they come from. Ignite *installation* and *recovery* is described in terms of phases, with each phase described in detail.

The Ignite-UX Install Environment

HP-UX installation and recovery is accomplished using the Ignite-UX *install environment*.

The Ignite install environment is a small subset of the HP-UX operating system that allows HP-UX to install itself onto a system. During the initial phases of installation and recovery, the install environment runs in client memory without any disk-based storage. A memory-based RAM disk holds the initial root *file system* needed for HP-UX operation. While operating with a memory-based root disk file system, file system space is very limited. On smaller memory systems, memory for the HP-UX *kernel* and processes might also be limited. Command libraries and other files must be loaded and removed as needed. (Increasing the size of the memory-based root disk to make more space would result in insufficient memory being available for the processes that accomplish installation and recovery.) Once the correct disks are identified, volumes and file systems are created. The install environment then switches to a disk-based file system. When that is completed, some of the RAM disk space is freed.

The Ignite install environment consists of:

- *[W|V|I]INSTALL* – The HP-UX *install kernel*, which is statically linked and includes a wide variety of I/O and other modules so it is able to run on all supported systems.
- *[W|V|I]INSTALLFS* – The initial HP-UX *install file system*, which is copied into the root RAM disk during *boot*. The first 8 KB can contain Ignite-UX configuration content.
- `INSTCMDS` or `INSTCMDSIA`, `SYSCMDS` or `SYSCMDSIA`, and `RECCMDS` or `RECCMDSIA` – *Archive*s of commands, libraries, and other files needed to accomplish installation and recovery, but are not needed to initially get the install environment running. These are loaded as needed during installation and recovery.

The Ignite-UX *install kernel* and *install file system* are loaded into system memory by the standard HP-UX boot loader or virtual system boot loader software. Note that there are a number of boot sources where the Ignite *install environment* may reside. Also, the details of booting vary according to your Ignite data center configuration.

Boot Sources

Ignite always retrieves the *install kernel* and *install file system* from the boot source. By default, Ignite retrieves *INSTCMDS*[IA], *SYSCMDS*[IA], and *RECCMDS*[IA] from that same boot source, but can get these command *archive*s from a different source if requested to. Ignite can determine the boot source by querying the HP-UX kernel.

Ignite can switch its source for command archives and depots if configuration information in the install file system instructs it to, or if instructed to by the Ignite user interface.

Ignite will operate in the same manner, regardless of the boot source.

Installation Versus Recovery

Ignite internally uses the same approach, regardless of whether you are performing an *installation* or *recovery*. The terms “installation” and “recovery” are valuable to describe intended use, but Ignite's internal operation make it possible to blur the distinction between the two, such as when you use *golden image*s.

This design is quite powerful, and allows Ignite to handle significant system differences during recovery by adapting as needed and regressing to more install-like behavior if required.

Network Booting and IP Addresses

When a system boots HP-UX from an Ignite-UX server, it needs an IP address to get the operating system *kernel*. This first IP address is not necessarily the same IP address the system will be assigned for networking when its kernel is up and running. The mechanisms for distributing the first and second IP addresses are sometimes different.

PA-RISC Systems

When a PA-RISC system boots from an Ignite-UX server, the first IP address request is answered by the `instl_bootd` *daemon*. This communication uses ports 1067 and 1068. The file `/etc/opt/ignite/instl_boottab` is referenced to assign the first IP address to the booting system whether it is registered or anonymous.

After HP-UX is running on a PA-RISC system, it requests a second IP address for networking. This request is answered by `bootpd` using ports 67 and 68. The `/etc/bootptab` file is referenced for *registered client*s; *DHCP* services are used for *anonymous client*s.

Itanium-Based Systems

When an Itanium-based system boots from an Ignite-UX server, the first IP address request is answered by the `bootpd` daemon. This communication uses ports 67 and 68. The file `/etc/bootptab` is referenced to assign the first IP address to a registered booting system. If the system is not registered, and you are running HP-UX 11i v2 or HP-UX 11i v3 on the Ignite-UX server, DHCP is used to assign the booting IP address.

When Itanium-based systems request a second IP address for networking, it uses the same daemon, file and ports described above. Configuring DHCP for booting is separate from configuring DHCP for assigning network IP addresses. See "Configuring an Ignite Server to Boot Anonymous Itanium-Based Clients" (page 50) for information about how to configure DHCP for assigning first (boot) and second (networking) IP addresses without conflict.

Phases of Operation

Ignite uses the sequence of high-level phases outlined below to accomplish installation and recovery. Depending on configuration information, some steps within these phases might be skipped. At a very high level, Ignite operates in four phases:

- Startup – The *install environment* is loaded from the *boot* source to the client memory. Ignite runs in client memory. The operation is configured and launched. If the installation or recovery is interactive, the user interface is run to create a configuration.
- Phase 1 – Storage is set up and Ignite moves to the client disk.
- Phase 2 – HP-UX *archive*s and *depot* software are installed. The HP-UX *kernel* is built. A reboot is required to start the final HP-UX kernel and make the new *file system* the root file system.
- Phase 3 – Software is configured. The system is now fully installed or recovered after a reboot or halt.

Startup

Ignite-UX software is started and the Ignite user interface is run to select, create, or modify the configuration that will be used to control installation or recovery. The result of this phase is a detailed system configuration to be used for installation or recovery. Processing for this phase is done on a RAM file system.

1. The *install kernel* and *install file system* are loaded from the boot source to the client memory via boot loader functionality. The HP-UX install kernel is started.
2. The Ignite software is started by the install kernel as an application process running on the install file system.
3. Additional RAM file systems are created to allow enough file system space for loading system setup content.

4. If the system has SAS disks, the I/O configuration is modified as needed to make the mapping between bays and HW paths consistent. This aids consistent installation and recovery. (Improved agile device selection and recovery has eliminated the need for this feature and might result in this step being removed in the future.)
5. Configuration content from the install file system is loaded to determine if the Ignite TUI should be started and if special inventory control is needed. (The Ignite TUI is started by default.)
6. A system I/O inventory is performed. This identifies devices where HP-UX may be installed, and identifies devices and networks used to accomplish installation. Install file system configuration and boot loader option content may be used to control inventory. The boot source is also determined.
7. Unless configuration information directs otherwise, the Ignite TUI is launched on the client.
   - The operation to be performed is set. (Advanced Install is the default operation.)
   - Networking configuration information is determined, if the installation requires the network.
   - The complete set of Ignite *configuration file*s is read and parsed. Note that changing the Configuration or Environment will result in rereading and parsing config content, since these changes generally result in changes to the set of *config file*s.
   - System, software, file system, and other configuration changes are gathered via the interface.
   - When Go! is selected from the user interface, the requested installation or recovery is launched.
   - Configuration sanity checking is performed. If there are problems, you are returned to the TUI.
   - The modified configuration is saved to control installation or recovery processing.
8. If the TUI was not selected to launch, sanity checking is done on the selected config.

Phase 1

Storage is set up and Ignite relocates to the new disk *file system*. The result of this phase is the install or recovery functionality running on what appears to be a normal disk-based file system, and if recovery, an I/O configuration that appears to be restored. Some aspects of the configuration cannot be fully restored until reboot. Processing for this phase is done on a RAM file system.

1. During a recovery, the original I/O configuration is restored if I/O instance data is present in the config. Some aspects of the configuration might be instantly changed. Some aspects are temporarily changed and will be finalized during reboot. If the current system is different, some aspects of the I/O configuration will be impossible to restore.
2. Create disk partitions if needed (Integrity systems only).
3. Create volume manager volumes as needed.
4. Create and mount file systems.
5. Determine software sources and selections.
6. Run pre-config control scripts.
7. Set boot path.
8. Set up and enable swap space.
9. Save final volume configuration data to disk file system.
10. Set locale.
11. Move needed content from RAM file system to disk file system. Load the complete set of commands, libraries, and other files required to accomplish installation and recovery from Ignite command *archive*s to the new disk file system (`SYSCMDS` or `SYSCMDSIA`).
12. Change the root directory to the disk file system with `chroot`.

Phase 2

File content is installed or restored. The result of this phase is the final disk file system and content. Some cleanup and processing that must be done after system reboot is still required. For the HP-UX *install kernel*, the RAM file system is still the root disk. For the commands in this phase, the new disk file systems is the root file system. A reboot is required to change the HP-UX kernel root disk from the RAM disk to the final disk.

1. Release RAM disk space to accommodate software installation and kernel build processes to be done later.
2. Load the archive if indicated in the config (for recovery and golden image installation).
3. Update `mnttab` so it appears to be correct during installation and kernel build.
4. Create device special files.
5. If needed, rename device files to make the I/O configuration appear fully restored.
6. Update `bootconf`.
7. Change I/O configuration files to match final instance config using `ioinit` and `ioscan -M`.
8. Load depot-based software if indicated in the configuration.
9. Save configuration so it is available for reuse.
10. Build final system kernel.
11. Set up the `inittab` file so final Ignite-UX processing will be done after reboot.
12. Reboot system.

Phase 3

Software is configured and final installation or recovery cleanup is done. The result of this phase is a fully installed or recovered system, ready for use after reboot. If configuration has been deferred, the system will be set up to run `FIRST-BOOT set_parms` on initial boot so you may choose the hostname, IP address, and other settings. Processing for this phase is done using the final disk-based file system.

1. Update the *AUTO* boot file.
2. Configure software.
3. Configure final networking.
4. Generate a system *manifest*.
5. Save the installation information for deferred configuration.
6. Perform final cleanup.
7. Reboot or halt system.

Ignite-UX Server Requirements

Hardware Requirements

An Ignite-UX server supporting *boot*, *installation* and *recovery* for clients requires the following hardware:

- **Computer:** An HP 9000 (PA-RISC) system running HP-UX 11i v1, HP-UX 11i v2, or HP-UX 11i v3; or an Itanium®-based system running HP-UX 11i v2 or HP-UX 11i v3 is required.
- **Memory:** Client installation and recovery performance is typically limited by network throughput. Normally, no special consideration for system memory is needed.
- **DVD drive:** A DVD device is recommended to simplify copying HP-UX release depots directly from *installation media* to the Ignite-UX server.
- **Tape device:** As part of your overall disaster recovery plan, you should consider how the Ignite-UX server itself would be recovered. A tape drive allows the Ignite-UX server to use tape media to save the server's own *recovery archive*. Note that depots, saved client recovery

archives, and other client-specific content typically should not be included in the recovery archive saved to tape. This client content should be saved using backup software. Not all systems support tape boot and so require *two-step media recovery*. See "Tape Recovery With No Tape Boot Support — Two-Step Media Recovery" (page 234) and the *Ignite-UX Installation Booting* white paper available at http://www.hp.com/go/ignite-ux-docs for more information.

- **Network interface:** One or more network adapters to support network boot and installation is required. A network adapter directly connected to each supported subnet is preferred. Note that multiple simultaneous network installations and recovery operations can create significant network traffic.
- **Disk space:** An Ignite-UX server might need considerable disk space.
  - Ignite-UX servers must have at least 2 GB of free disk space available in `/opt/ignite` to support installation of all HP-UX releases (B.11.11, B.11.23, and B.11.31).

    To save space, you can support only those HP-UX releases you plan to install or recover.
  - Ignite-UX servers might require significant space in `/var/opt/ignite` to support clients' systems, store software depots, and save *recovery archive*s. Default HP-UX file system sizes are unlikely to be sufficient for an Ignite-UX server. You should consider the number of client systems you intend to support and the maximum number of recovery archives to be saved for each client.

    The size of a recovery archive depends on the content being saved. A recovery archive will normally include at least a full set of HP-UX operating system software.
  - File system space is needed to hold depots required for installation. You should consider how many different OS releases the server might need to support. Note that you might also want to support different OE versions of each HP-UX revision on your Ignite server, such as the HP-UX 11i v2 September 2004 OE release and the 11i v2 September 2006 OE release, or the 11i v3 Base OE (BOE) and the 11i v3 Virtual Server OE (VSE-OE) of a particular release. Space will also be needed to store additional application and patch depots.
  - If you use *golden image*s, file system space is needed to hold them. Consider the size and number of images you will require.
  - See the *HP-UX Installation and Update Guide* available from **`http://www.hp.com/go/hpux-core-docs`** for a detailed description of the disk space required for all Operating Environments for your version of HP-UX.

Other Considerations

An Ignite-UX server might also require software, utilities, and configuration:

- **Use of TFTP:** Ignite-UX transfers some files using TFTP. A list of the minimum directories needed for file transfer is kept in the `/etc/inetd.conf` file. You might need to add directories to the list if you place configuration scripts in nonstandard locations.

  For example, the Ignite server must have the following entry in its `/etc/inetd.conf` file.

  ```
  tftp dgram udp wait root /usr/lbin/tftpd\
     tftpd /opt/ignite /var/opt/ignite
  ```

  If you are using HP Serviceguard clusters or systems with multiple IP addresses on a LAN interface, use the `-s` option with `tftp` and install the patch `PHNE_28762 11.11`.
- **Use of ssh:** With Ignite-UX version C.6.8 and later, `bootsys` can use `ssh`, and `ignite` can use `ssh` for *`make_[tape|net]_recovery`*. With Ignite-UX version C.7.1 and later, the `ignite` command can use `ssh` when calling `bootsys`. To use `ssh`, it must be available on the Ignite server and on the client, and you must have an existing public/private key pair.
- **Optional use of an X11 display server:** An X11 display server allows you to use the GUI to configure and start Ignite. Your Ignite-UX server can use an X server to display the Ignite

GUI, or you can redirect the display to another X terminal by entering the following command:

```
export DISPLAY=system_name:0.0
```

If `DISPLAY` is not set on the server, the Ignite TUI will run.

- **Software:** Get Ignite-UX and any software depots you plan to distribute to clients from the product media (CD or DVD). Ignite-UX can also be downloaded from the web; see "Getting the Ignite-UX Software " (page 19) for more information.
- **Client access to server:** There are multiple methods of having clients contact the Ignite server, each suited to a different environment. See Chapter 2 (page 31) for more details.

Supported Peripherals

Disks and Other I/O

If a disk device is visible, it does not mean it is supported for installation. It is important to verify that the disk is supported by the system, the host bus adaptor (*HBA*), the firmware, the HP-UX release, and the volume manager to be used.

Computer system hardware documentation should be consulted for supported I/O configurations. See the HP Business Support Center http://www.hp.com/bizsupport for HP computer system hardware documentation.

VxVM support is provided for a specific set of devices. The list of supported devices should be consulted – see http://www.hp.com/go/hpux-core-docs. Look for the section *Support Matrixes*, and the document entitled *Device Support Information for Veritas Products on HP-UX*.

LVM supports all the devices HP-UX supports. See the HP-UX Supported Mass Storage Devices Matrix for a table of I/O devices supported for each version of HP-UX.

Firmware

At times you might need new firmware to support a new device or HBA. Ensure that the client's firmware supports the devices and HBAs to be used for boot and root. For example, after the HP9000 rp8400 system was first released, firmware changes enabled the system to boot from disks connected to Ultra 160 HBAs. Check the *Installation and Update Guide* for your HP-UX release, available at http://www.hp.com/go/hpux-core-docs, for instructions on finding firmware information.

Additionally, firmware support for Fibre Channel, tape devices, and LAN cards might vary. In some cases, devices are supported for data use, but device boot is not supported.

Disk Arrays

You can install disk arrays using HP-UX, but Ignite-UX does not directly support configuring an array. The disk array must be configured first; see your array documentation for configuration instructions. In some cases, system firmware may be used to set up disk arrays. The Ignite-UX *install environment* contains tools that are also used to help configure disk arrays. To use these tools in the install environment, you will need to use the *expert recovery* functionality to start an install environment shell. It might also be necessary to load files that are not normally included in the install environment by using the Ignite-UX `loadfile` command. When array configuration is complete, it is necessary to reboot the system in order to use the newly configured disk LUNs during install.

Client Terminals

The Ignite-UX client-side operating system installation tools support VT100 and Wyse 60 terminals, compatible terminal emulators, and all HP terminals. Additional information regarding how to navigate within the Ignite-UX GUI with the keyboard is found in Appendix F (page 277).

2 Making Configuration Decisions for Ignite Servers

Ignite is flexible when configuring networking options and even allows options that don't require networking. Also, you can switch to a source other than the *boot* source for *install content*. These features allow you to choose from a variety of *installation* and *recovery* solutions.

Below are installation solutions, starting with the most simple and progressing to those more complex. This chapter finishes with network booting debugging techniques.

Boot and Install Client from Media

These options do not require a network:

- **Cold install or update a single system directly from media kit DVDs**

  You can use the HP-UX 11i media kit DVDs with Ignite-UX to *cold install* or update a system. For more information, see the *HP-UX Installation and Update Guide* for your version of HP-UX, available at http://www.hp.com/go/hpux-core-docs.

- **Cold install from custom media**

  This option assumes you have already created *custom installation* media. Custom installation media can be a tape or DVD with either a *golden image* or a *recovery image* on it. All *installation media* are bootable. After you boot from media, choose **Media only installation** as a Source Location Option from the Ignite-UX User Interface and Media Options screen. For more information, see Chapter 14 (page 197).

- **Recovery from tape**

  This option assumes you have already created a recovery tape. For more information, see "Creating and Using Recovery Tapes" (page 217).

Simple Network Solutions

These solutions use a single Ignite server that supports network boot, installation, and recovery. The Ignite server and the client systems must be on the same subnet, and no other boot or installation servers can be on that subnet.

Questions you will have to answer when configuring a simple network are:

- Are my clients PA-RISC or Itanium-based?
- Do I want to network boot all my clients?
- Do I want my clients to have their MAC addresses registered with the server to always boot to the same assigned IP address (*registered client*s), or do I want an available IP address assigned to them when they boot (*anonymous client*s) ?
- Do I want the booting IP address to be the same IP address used for networking after installation is complete?
- Do I have *DHCP* running on my subnet?

Decision trees for Ignite-UX server configuration follow. Do not treat them as strictly yes-or-no exercises. Your environment may require choosing multiple methods from the decision trees, and although you may be able to use an option, you might reject it because it is not the best answer for your environment. Also, keep in mind that these decision trees cover booting, so only the initial IP address is affected. For more information, see "Network Booting and IP Addresses" (page 26).

A decision tree for network booting PA-RISC systems is shown in Figure 2-1. A decision tree for network booting Itanium-based systems is shown in Figure 2-2 (page 33). The decision trees assume the network boot clients are on the same subnet as the Ignite-UX server, and that you will always use the `install` option to the *boot console handler (BCH)* `boot` command for PA-RISC systems. Further, the decision tree for network booting Itanium-based systems assumes there is

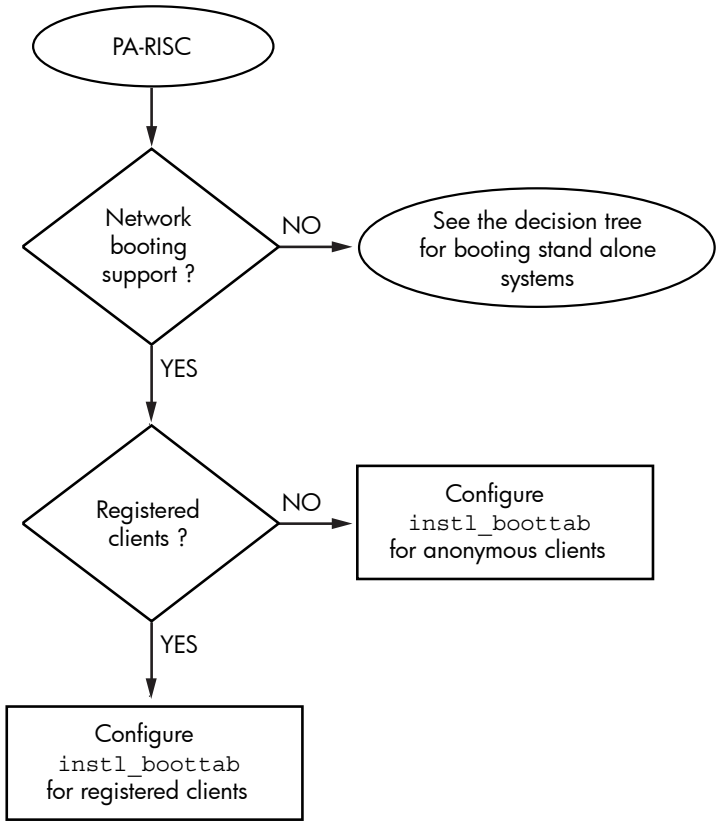
only one DHCP server on your subnet configured to answer boot requests, and that it is running HP-UX.

If you want to boot a system without using the network and your Ignite-UX server, see the decision tree shown in Figure 9-2 "Decision Tree for Booting From Media and Installing HP-UX From the Server ".

NOTE: A lot of clients can only be booted using their built-in LAN interfaces. Other LAN interfaces might not be supported for boot. For more information about LAN interface boot support, consult the hardware documentation for the system or the add-in LAN card.

Use the following decision tree when configuring an Ignite-UX server for PA-RISC clients:

Figure 2-1 Decision Tree When Configuring a Server for Booting PA-RISC Systems



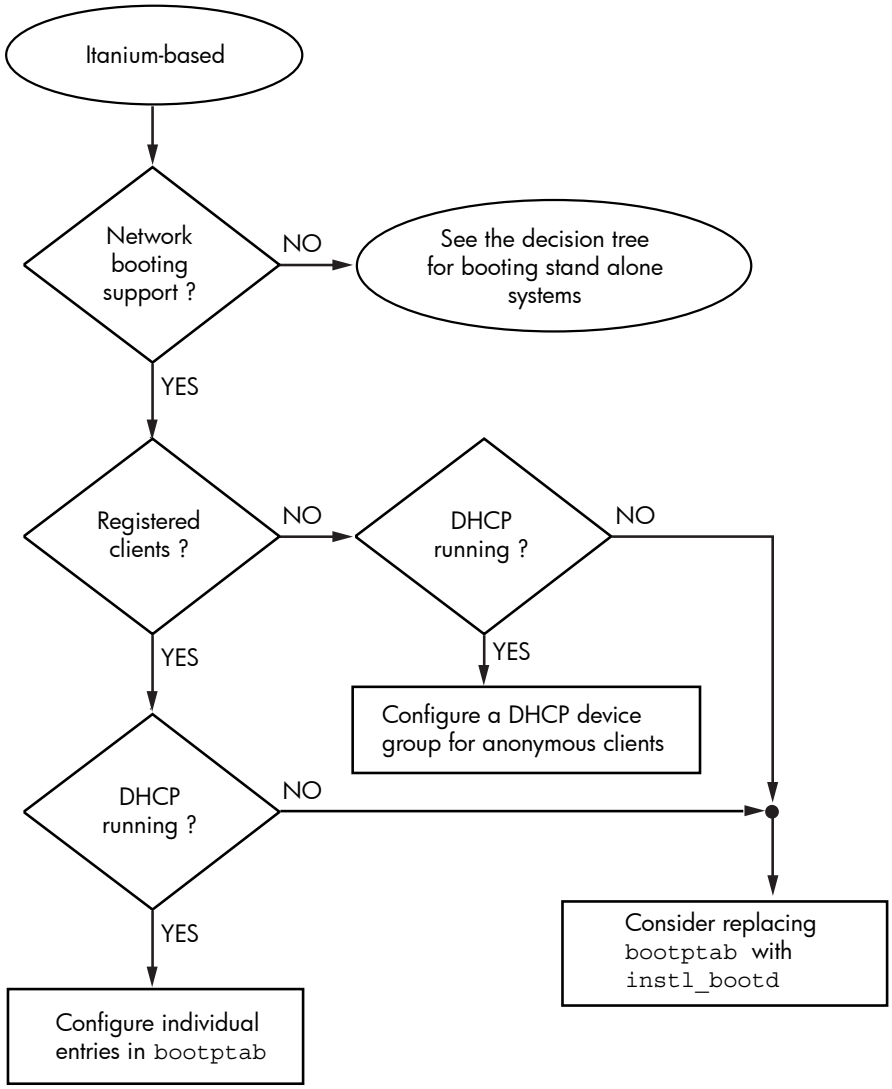
Configure `instl_boottab` for registered clients - To network boot registered PA-RISC clients, the server uses the `instl_bootd` daemon to answer boot requests, and has clients' IP addresses and LAN addresses registered in `/etc/opt/ignite/instl_boottab`. The process of configuring an Ignite-UX server for registered PA-RISC clients is described in "Configuring the Ignite-UX Server for PA-RISC Clients" (page 37). See the *Ignite-UX Quick Start Guide* available at http://www.hp.com/go/ignite-ux-docs if you are new to HP-UX.

Configure `instl_boottab` for anonymous clients - Network booting anonymous PA-RISC clients is similar to booting registered PA-RISC clients; the difference is that some IPs in the `/etc/opt/ignite/instl_boottab` file are not associated with any clients' MAC addresses, and so may be assigned to clients as requests come in. See "Configuring an Ignite Server to Boot Anonymous PA-RISC Clients" (page 49) for more information.

See the decision tree for booting stand alone systems - This decision tree can be found in Figure 9-2.

Use the decision tree below when configuring an Ignite-UX server for Itanium-based clients.

Figure 2-2 Decision Tree When Configuring a Server for Booting Itanium-Based Systems



Configure individual entries in `bootptab` — To network boot registered Itanium-based clients, the server uses the `bootpd` daemon to answer boot requests, and has clients' IP addresses and LAN addresses registered in `/etc/bootptab`. One drawback to this option is that you must configure an entry for every system that needs to boot. The advantage of this method is that it works on all versions of HP-UX. See "Configuring the Ignite-UX Server for Itanium-Based Clients" (page 41) for details. See the *Ignite-UX Quick Start Guide* available at http://www.hp.com/go/ignite-ux-docs if you are new to HP-UX.

Configure a DHCP device group for *anonymous client*s - Configuring an Ignite-UX server to *boot* anonymous Itanium-based clients requires sophisticated considerations; see "Configuring an Ignite Server to Boot Anonymous Itanium-Based Clients" (page 50). This option is only available for Ignite servers running HP-UX 11i v2 and later.

Consider replacing `bootpd` with `instl_bootd`- The `instl_bootd` daemon can support anonymous Itanium-based clients and provide enough information for them to boot. The `instl_bootd` daemon can also be configured for *registered client*s. You might consider running `instl_bootd` if you have PA-RISC and Itanium-based clients on one subnet, since this would allow you to manage client IP addresses in one place: `/etc/opt/ignite/instl_boottab`. The drawback to this arrangement is that an Ignite-UX server configured this way cannot provide DHCP services. This method is not recommended for an environment that consists mainly of Itanium-based systems - use DHCP or `bootpd` in that case. See "Replacing bootpd with instl_bootd" (page 53) for more information.

See the decision tree for booting *stand alone* systems - This decision tree can be found in Figure 9-2.

Alternate Boot with Network Server Installation

A simple way to avoid boot issues in complex network configurations is to avoid network boot. Network installation may be started via non-network boot. The Ignite *install environment* may be booted from a source local to the client system. Regardless of how Ignite-UX is started, it has the same network capabilities once it is running.

- **Use `bootsys` to boot a system already running HP-UX**

  If the client system is already running HP-UX, the Ignite-UX `bootsys` command may be used to copy the *install kernel* and *install file system* to the client system's HP-UX file system. After reboot, the HP-UX boot loader can boot for installation using that copied content. Ignite config content in the install file system may be used to cause Ignite to automatically switch to use the master Ignite server. Because the initial *install environment* is copied from the Ignite server, you can be confident the Ignite versions of the initial *boot content* and software on the Ignite server have the same versions. See "Using bootsys on the Client Console" (page 110).

- **Use DVD media to boot a system for network installation**

  Ignite supports booting for network installation using standard HP *installation media* or custom boot media. The version of the Ignite on the media must match the version of Ignite running on the master Ignite server. The simplest way to ensure the versions match is to use `make_media_install` on the Ignite server to create custom boot media. This custom boot media may be constructed to include *[W|V|I]INSTALLFS* config content, which automatically switches to using the Ignite server on startup. Standard HP-UX installation media may also be used to boot the system, as long as the Ignite version on the media matches the master server Ignite version. Standard HP-UX media config content cannot be modified to automatically switch to your Ignite server. See "Creating a Boot CD/DVD or an Installation DVD" (page 203) and "Tape Recovery With No Tape Boot Support — Two-Step Media Recovery" (page 234).

- **Use vMedia USB DVD to boot a system for network installation**

  Many Integrity systems support Integrated Lights Out (iLO) Virtual Media (vMedia). This feature must be enabled using a license key. Once enabled, a DVD device or an ISO DVD image on a remote system, such as a PC, may be used. In either case, the client system will appear to have a local USB DVD device.

  For more information, see Appendix D (page 263) and the *HP Integrity iLO 2 MP Operations Guide* available at http://www.hp.com/bizsupport.

- **Boot your Integrity system from a USB memory stick device**

  It is possible to configure your Integrity system and a USB flash drive in order to boot HP-UX directly from a memory stick device. Once the system is booted to the HP-UX Ignite-UX install environment, you can perform a variety of installation or recovery actions. See the *Ignite-UX USB Memory Stick Boot* white paper, available at http://www.hp.com/go/ignite-ux-docs, for more information.

Complex Networks

Setting up an Ignite server on a simple network assumes there is a single subnet with only one Ignite server that supports network boot and installation. Often, real network environments are significantly more complex. Configuring an Ignite server to operate correctly while avoiding interference with other boot and installation servers on the network requires special consideration.

For a detailed discussion, see Chapter 5 (page 55).

Diagnosing Network Boot Issues

When configuring a network, sometimes boot and installation will not work at all or will not work as expected. Especially when configuring a complex network, you should expect to spend time diagnosing and resolving issues due to the complexity of the network and interactions between servers. You should also expect that problems might occur in the future as the complex network changes.

This section includes suggested tools and techniques for diagnosing problems.

HP-UX Diagnosing and Debugging

Simple Network Debugging

If network boot is used on a local subnet and the Ignite-UX server is not found, check these items:

- Verify the client is on the same subnet as the Ignite-UX server or boot helper.
- Investigate `instl_bootd` errors in `/var/adm/syslog/syslog.log`.
- In the `/var/adm/inetd.sec` file, ensure the service `instl_boots` exists, and that the IP address 0.0.0.0 is allowed. (Normally, all addresses are allowed via `0.0.0.0`.) The entry should look like:

  ```
  instl_boots allow 0.0.0.0
  ```

- If `/etc/services` comes from NIS, make sure the NIS server has `instl_boot*` entries.

Logging to `syslog.log`

The `bootpd` and `tftpd` daemons have the ability to log requests and responses. The `/etc/inetd.conf` file may be modified to enable logging. The `bootpd -d` option and `tftpd -l` option control logging. For example:

```
# tftp dgram udp wait root /usr/lbin/tftpd tftpd \
     -l /opt/ignite /var/opt/ignite
# bootps dgram udp wait root /usr/lbin/bootpd bootpd -d 9
```

The daemons log to the HP-UX `syslog` file located at `/var/adm/syslog/syslog.log`.

NOTE: Logging should normally be disabled since it can consume a significant amount of disk space.

If the boot configuration includes multiple boot servers (for `bootp` relay, for example) it is often useful to enable logging on all servers.

Using `bootpquery`

To save time when configuring an HP-UX system boot, the `bootpquery` command may be used to simulate a network boot request by requesting `bootpd` to indicate how it would respond to boot requests for a specific MAC address. This is normally much faster and simpler than attempting to boot using a real client system.

To use `bootpquery`, add the `ba` option to the appropriate entries in the `/etc/bootptab` file. Without this option, `bootpd` will send responses only to the client system making the boot request. The `ba` option requests the response be broadcast on the subnet, so any system is able to see the response, including the system where you are using `bootpquery`. For more information, see *bootpquery*(1M).

NOTE: The `ba` option should be removed once testing is completed.

The `bootpquery` output includes valuable debug information:

```
# bootpquery 0011855F549E
Received BOOTREPLY from hpignite.xyzco.com (10.1.1.11)
```

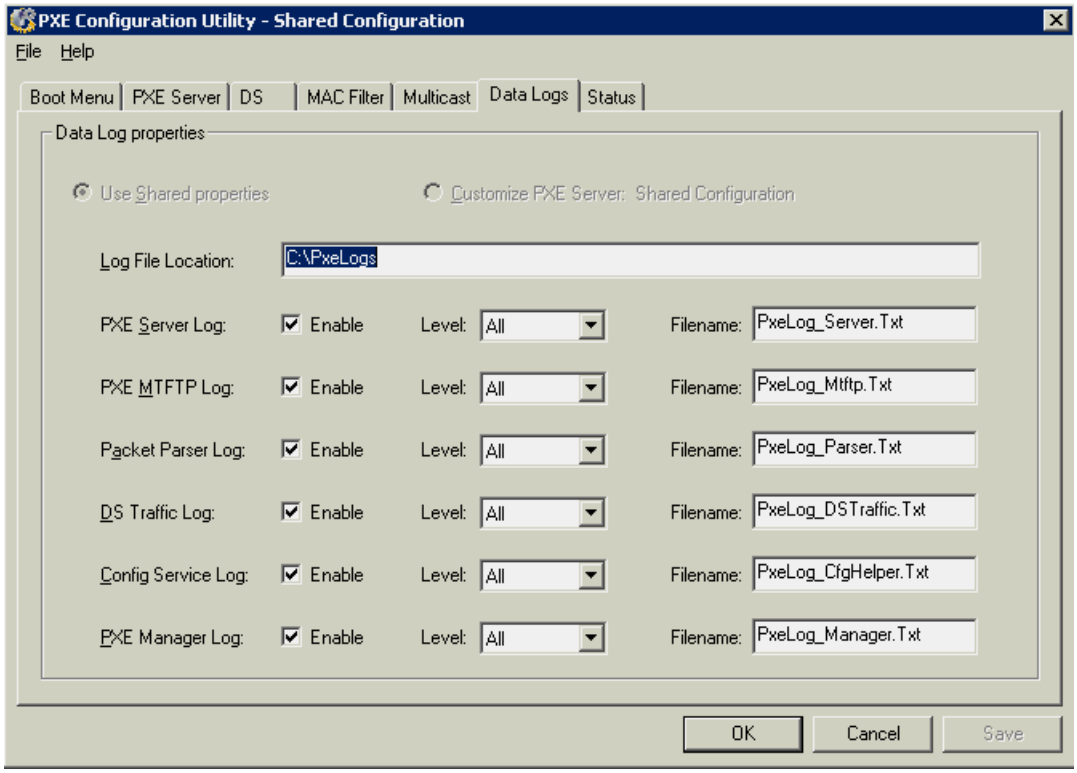
```
Hardware Address:           00:11:85:5f:54:9e
Hardware Type:              Ethernet
IP Address:                 10.1.1.110

Boot file:                  /opt/ignite/boot/nbp.efi

RFC 1048 Vendor Information:
    Subnet Mask:             255.255.255.0
    Gateway:                 10.1.1.1
    Bootfile Size:           24576 512 byte blocks
    Domain Name Server:      10.1.1.1
    Host Name:               hpuxsys1
    Domain Name:             xyzco.com
```

RDP Diagnosing and Debugging

An RDP server can be configured to log PXE boot and TFTP activity. The PXE Configuration Utility may be used to control logging. Logging should be disabled when you are finished diagnosing and debugging.



3 Simple Network: Creating a Server for Registered Clients

This chapter describes how to install a basic *Ignite-UX server* configuration for network booting and installing HP-UX on *client*s registered with the server. This chapter does not discuss support for *anonymous client*s. For information about how to set up anonymous clients, see Chapter 4 (page 49).

See the *Ignite-UX Quick Start Guide* available at http://www.hp.com/go/ignite-ux-docs if you are new to HP-UX.

For PA-RISC clients, a basic server setup will use the `instl_bootd` *daemon* to answer boot requests, will not use *DHCP* for initial system boot, and will register clients' IP addresses and LAN addresses in `/etc/opt/ignite/instl_boottab`.

For Itanium-based clients, a basic server setup will use the `bootpd` daemon to answer boot requests, will not use DHCP for initial system boot, and will have clients' IP addresses and LAN addresses registered in `/etc/bootptab`.

For more information on how systems get IP addresses for booting, see "Network Booting and IP Addresses" (page 26).

Setting up software *depot*s is the same for PA-RISC and Itanium-based systems.

Configuring the Ignite-UX Server for PA-RISC Clients

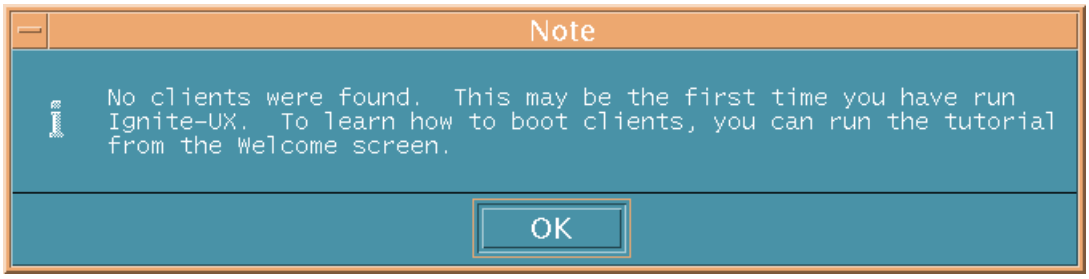
Launch Ignite-UX

As superuser, start Ignite-UX by entering the following command:

```
ignite
```

Because this is the first time Ignite-UX is launched, there are no clients and the message in Figure 3-1 appears. You must boot a client before it can be recognized and managed by Ignite-UX. Acknowledge the message by clicking **OK**.

Figure 3-1 Ignite-UX First Launch Message



The Ignite-UX Welcome dialog box is displayed, as shown in Figure 3-2.

Figure 3-2 Ignite-UX GUI Welcome Dialog Box

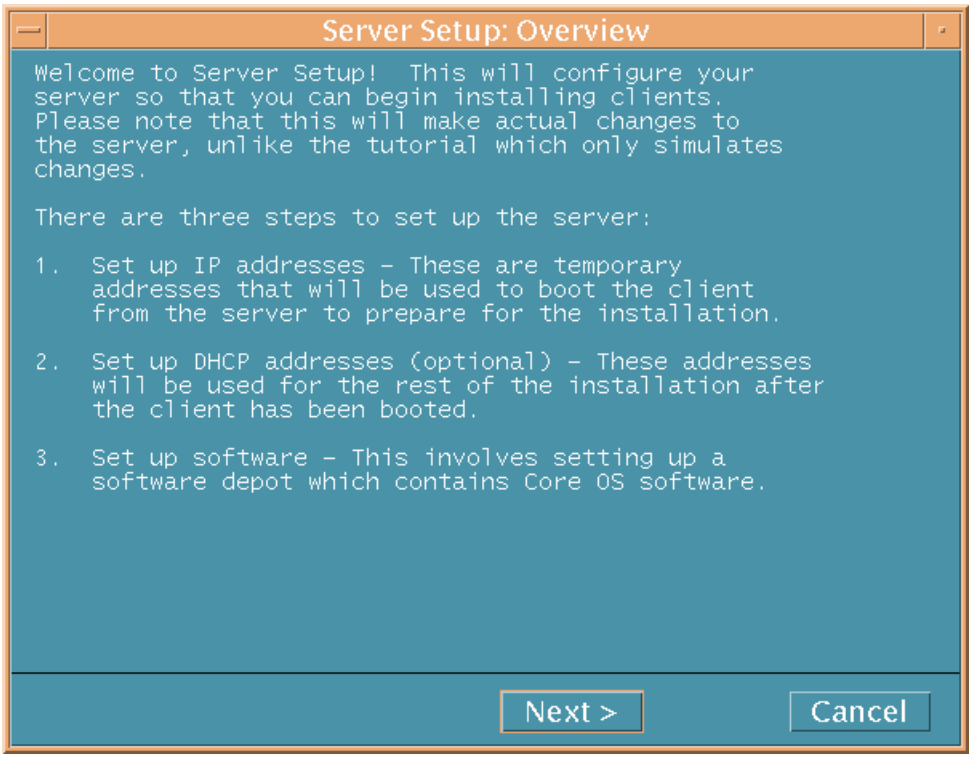
To learn more about the Ignite-UX GUI now, click **Tutorial and Demo...** Once the Ignite server is configured, you can access the tutorial by selecting **Actions→Run Tutorial/Server Setup→Tutorial and Demo** from the Ignite-UX interface.

To bypass this welcome the next time you start Ignite-UX, click the **Do not show this screen again** check box.

Launch the Server Setup Wizard

To begin configuring your Ignite-UX server, click **Server Setup...** to launch the Server Setup Wizard, as shown in Figure 3-3.

Figure 3-3 Server Setup Wizard

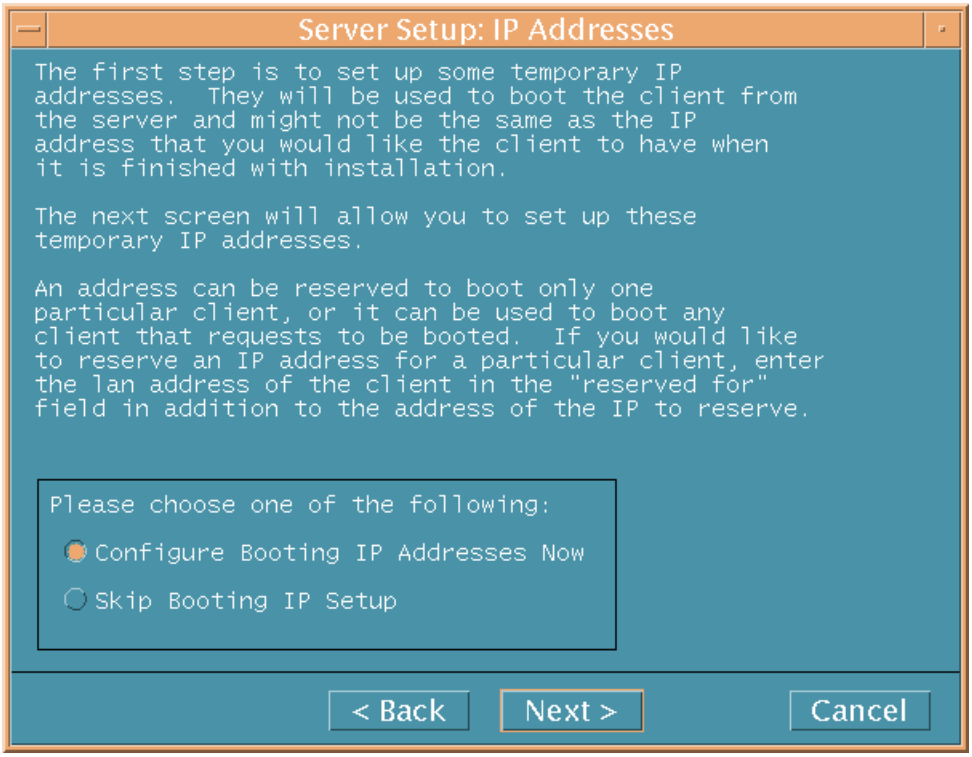


To set up an Ignite-UX server for PA-RISC clients, complete step 1 (Set up IP addresses), skip step 2 (Set up DHCP addresses), and complete step 3 (Set up software).

Click **Next** to advance to the Server Setup: IP Addresses dialog box (Figure 3-4).

NOTE: To end the setup process at any time and leave the system unchanged, click **Cancel**.

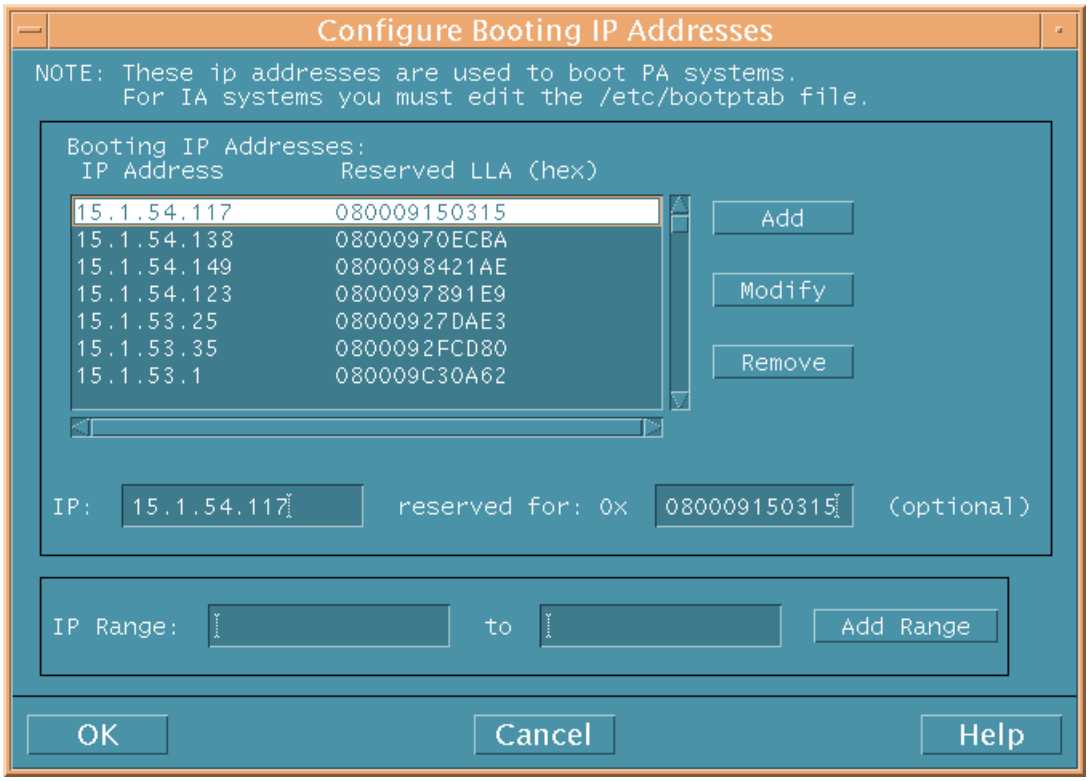
Figure 3-4 Server Setup: IP Addresses



Register the PA-RISC Clients with the Server

Select **Configure Booting IP Addresses Now** from the Server Setup: IP Addresses dialog box (Figure 3-4 (page 39)), then click **Next** to proceed to the Configure Booting IP Addresses dialog box shown in Figure 3-5 (page 40).

Figure 3-5 Configure Booting IP Addresses



Use the Configure Booting IP Addresses dialog box to register client IP addresses with their physical MAC addresses. The IP addresses and corresponding reserved MAC addresses are read from the `/etc/opt/ignite/instl_boottab` file on the server to display in the Booting IP Addresses window. IP addresses with blank reserved MAC addresses are not currently reserved for any client.

If you want to add a new IP address to reserve for a client, click in the **IP** text box and enter the IP address intended for your client. Then click in the **reserved for: 0x** text box and enter the client's MAC address. Click **Add** to enter the IP address/MAC address pair into the Booting IP Addresses window. The MAC address given here must be the MAC address of the network interface to be used to boot the system over the network.

If the IP address you want to reserve for a client is already listed in the Booting IP Addresses window, select that line. The IP address will appear in the **IP** text box, and the current MAC address it is reserved for, if there is one, will appear in the **reserved for: 0x** box. Enter the client MAC address in the **reserved for: 0x** box, then click **Modify**. The IP address will then appear in the Booting IP Addresses window with the client MAC you just entered.

You can remove sets of IP addresses/MACs from the Booting IP Addresses window by selecting the line and then clicking **Remove**.

Continue assigning IP addresses to clients' MACs until all the clients to be booted from the Ignite-UX server are registered. You can modify this information in the future by editing the `/etc/opt/ignite/instl_boottab` file, or via the Ignite-UX GUI under **Options→Server Configuration**.

When you have completed registering clients, click **OK** to write the contents of the Booting IP Addresses window to the `/etc/opt/ignite/instl_boottab` file.

Once you exit the Configure Booting IP Addresses dialog box, a *registered client*'s boot request is answered by `instl_bootd`, and the client will boot to the reserved IP address listed in the Booting IP Addresses window.

NOTE: No intervention is required to have `instl_bootd` pick up changes to the `/etc/opt/ignite/instl_boottab` file. When a boot request is received, `instl_bootd` always checks whether the file was modified since last read, and rereads it before answering any boot request. Care should be taken if you edit the `/etc/opt/ignite/instl_boottab` file manually. For the correct procedure see *instl_bootd*(1M).

Skip DHCP Setup

After exiting the Configure Booting IP Addresses dialog box, the Server Setup: DHCP (optional) dialog box appears. Select **Skip DHCP Setup**, then click **Next**.

A dialog box is displayed to tell you how to configure DHCP services later. Click **OK**.

The Server Setup: Software Depot Setup dialog box is then displayed (Figure 3-6 (page 43)).

Go to the Software Setup Section

Proceed to "Setting Up Software from OE Depots" (page 42) to complete the *Ignite-UX server* setup.

Configuring the Ignite-UX Server for Itanium-Based Clients

Register the Itanium-based Clients with the Server

Registered Itanium-based clients must be entered in the `/etc/bootptab` file manually; they cannot be registered using the Server Setup Wizard. The `/etc/bootptab` file acts as the database for the `bootpd` daemon on the Ignite-UX server. All registered clients you intend to boot from the Ignite-UX server must be entered in the `bootptab` file.

A typical `bootptab` file has a generic, default client specification defined. The individual clients use this default definition and make their specific modifications to it, such as the IP address and the hardware address (MAC address). In the following example, `IADEF` is the default configuration for Itanium-based clients on the subnet, and `iuxclient1` is the specific entry for that particular client.

```
IADEF:\
 ht=ethernet:\
  hn:\
  dn=domain_name.com
  gw=190.1.48.1:\
  sm=255.255.248.0:\
  ds=190.1.48.11 190.1.48.12:\
  vm=rfc1048:\
  bf=/opt/ignite/boot/nbp.efi:\
  bs=48:

iuxclient1:\
  tc=IADEF:\
  ip=15.1.52.204:\
  ha=00306E4A3391
```

The `tc` tag indicates the use of a template for common defaults, so all the values from `IADEF` are assumed for `iuxclient1` unless specifically overridden in the client's definition. The `ip` tag indicates the client's IP address, and the `ha` tag indicates the MAC address. For more information on the `bootptab` file syntax, see *bootpd*(1m).

For each client you intend to boot from the Ignite-UX server, enter their respective IPs and MAC addresses in the `bootptab` file.

From now on, when a registered client's boot request is answered by `bootpd`, it will boot to the reserved IP address you entered in the `/etc/bootptab` file. You can make changes to the `bootptab` file at any time.

IMPORTANT: The server that sends the response to the boot request is the same system from which the client will attempt to `tftp` the boot file. If you are not using an HP-UX system to reply to a request, you must make the required boot files available and current with new releases of Ignite-UX. HP does not provide support for this kind of configuration.

Use the Server Setup Wizard to Proceed to Software Depot Setup

Follow the steps outlined in the section "Configuring the Ignite-UX Server for PA-RISC Clients" (page 37) from "Launch Ignite-UX" (page 37) through "Launch the Server Setup Wizard" (page 38). When you get to the Server Setup: IP Addresses dialog box (Figure 3-4) select **Skip Booting IP Setup**, and then click **Next**.

A note appears instructing you how to configure IP addresses in the future. Click **OK**, and the Server Setup: DHCP (optional) dialog box is displayed. Select **Skip DHCP Setup**, then click **Next**.

A dialog box is displayed to tell you how to configure DHCP services later. Click **OK**.

The Server Setup: Software Depot Setup dialog box now appears (Figure 3-6 (page 43)).

Setting Up Software from OE Depots

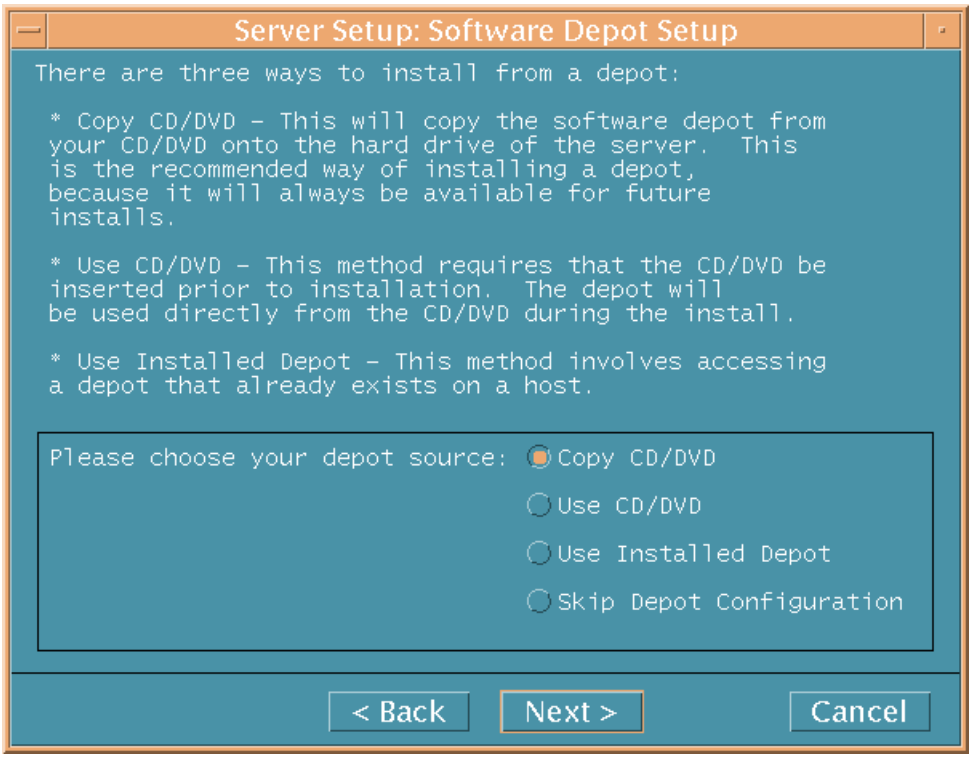
Before starting the software depot installation, you should have on hand either a set of OE media or information about a remote system that contains a previously installed OE depot.

If you are using media, you will need a locally attached optical drive: a DVD-ROM for DVD or CD media, or a CD-ROM drive for CD media. The media should not be mounted before starting if you are planning to create a local depot. The media can be mounted before starting if you are planning to install directly from the media. For performance reasons, HP does not recommend installing directly from media when more than one system will be attempting to access the media.

If you are using a previously installed OE depot, it should have been installed using the process described below, or by using the `make_depots` command.

Regardless of the source of the OE depot, the full OE should be installed, not a subset.

Figure 3-6 Software Depot Setup Page



Select the depot source (media or installed depot) and then click **Next**.

For media, you are prompted to insert the media and select a device.

For an installed depot, you are prompted for the hostname of the system containing the operating system depots. Enter the hostname, then click **Show Depots...**. Select a depot containing a core operating system from the list, and then click **OK**.

You are then asked to confirm your choices. A Server Setup Logfile dialog box is then displayed so you can monitor the depot installation progress. This process is lengthy and can take up to two hours. During this time this dialog box remains active and is updated when new information is written to the log file.

Upon completion, either click **OK** to continue installing additional depots by repeating this process, or click **Finish** to complete the server setup.

NOTE: To install additional software depots once the server is set up, see "Setting Up Additional Software on the Server" (page 46).

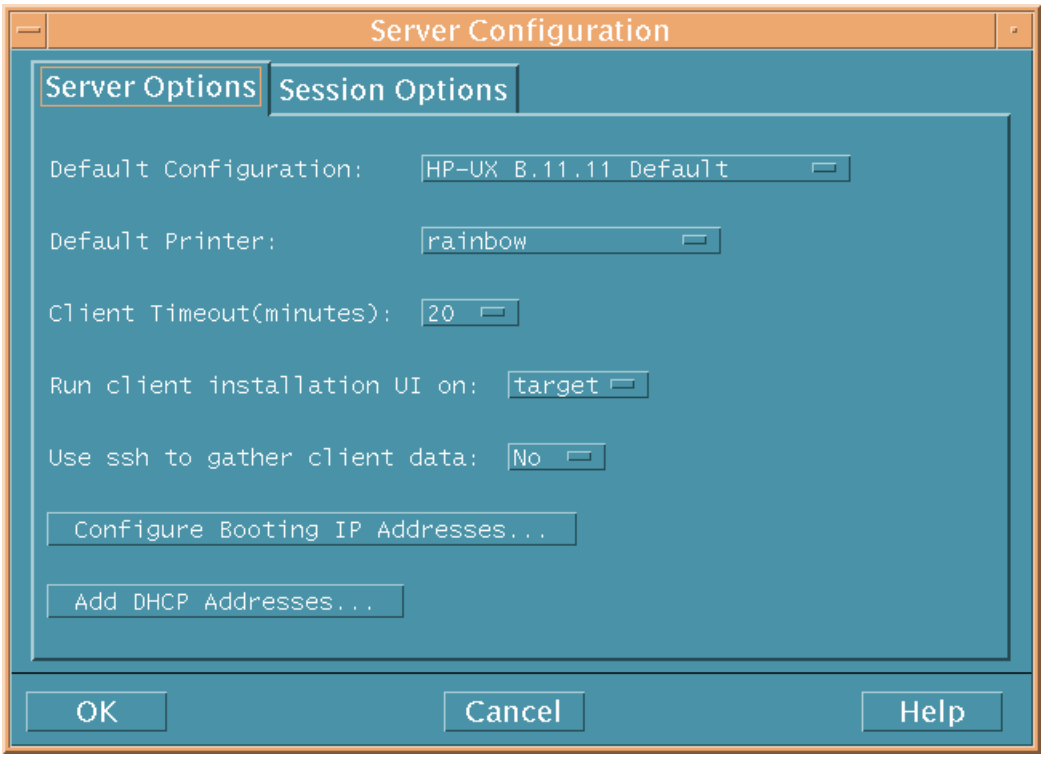
More Server Setup Options

Once your Ignite server is up and running, you can set general options by accessing the Server Configuration dialog box from **Options→Server Configuration**.

Configuring Server Options

General server settings are available on the **Server Options** tab.

Figure 3-7 Ignite-UX Server Configuration Tabs



The following options are available:

- **Default Configuration –** Click the button next to Default Configuration to select from the list of available configurations. The selected configuration is the default that will be used when installing clients. You can override this default setting on a per-client basis with Ignite-UX.
- **Default Printer –** Click the button next to Default Printer, then select one of the available (configured) printers. This is the printer used for printing a *manifest* or installation history. The printer IP address is verified by Ignite-UX before a job is sent.
- **Client Timeout(minutes) –** Click the button next to Client Timeout (minutes):, then select the number of minutes or **off**. Status information is written into the client's `install.log` file during the installation, and this log is actively monitored by Ignite-UX on the server. Setting this value configures Ignite-UX to display a warning message if the `install.log` file has not been updated in the selected number of minutes. HP recommends you use the default value.

  Setting Client Timeout to **off** disables this notification and does not affect the outcome of the installation.

- **Run client installation UI on –** Use the Run client installation UI on: menu to designate where you want to run the client UI for this installation. If you have an Ignite-UX server configured, you can run the client installation interface from the target system using a terminal user interface (TUI), or from the server using whatever UI is set up there (the Ignite-UX GUI or TUI). If the client installation is to be noninteractive (no user intervention), select **none**.

  The default is for the UI to be displayed on the Ignite-UX server.

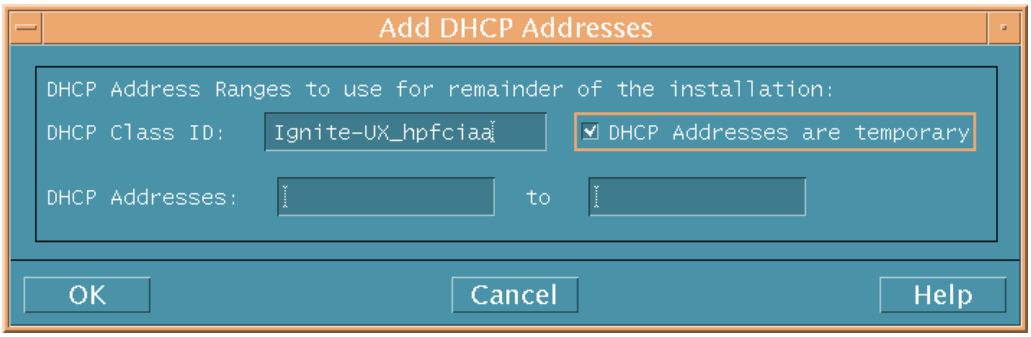
- **Use ssh to gather client data –** For all clients, `ssh` will be used instead of `remsh`, `rlogin`, and `rcp`.

- The **Configure Booting IP Addresses...** button gives you access to the Configure Booting IP Addresses dialog previously described in "Register the PA-RISC Clients with the Server" (page 40).
- **Add DHCP Addresses... –**

  The assignment of DHCP IP addresses for booting is only used for *anonymous client*s. See Chapter 4 (page 49) for more information.

  The IP addresses you provide here are used during boot and installation. These addresses are in use for most of the Ignite-UX download to a client. One address is required for each simultaneous download. For more information see "Network Booting and IP Addresses" (page 26).

  Figure 3-8 Add DHCP Addresses Dialog Box

  This provision of DHCP capability is for the boot and installation only. You will have to coordinate with the administrator of regular DHCP services, which distributes networking IP addresses, to make sure you use a set of available IP addresses that will not cause a conflict with regular DHCP services. See Appendix B (page 255) for information on configuring regular DHCP services on your Ignite-UX server. Unless you are familiar with DHCP services, do not modify the DHCP Class ID field or the **DHCP Addresses are temporary** check box.

  Provide a range of available IP addresses in the DHCP Addresses fields from lowest number to highest:

  ```
  10.2.73.21      10.2.73.40
  ```

  Other ways to set these IP address values are: when prompted by Ignite when it's first run are `instl_adm`, and SMH or SAM.

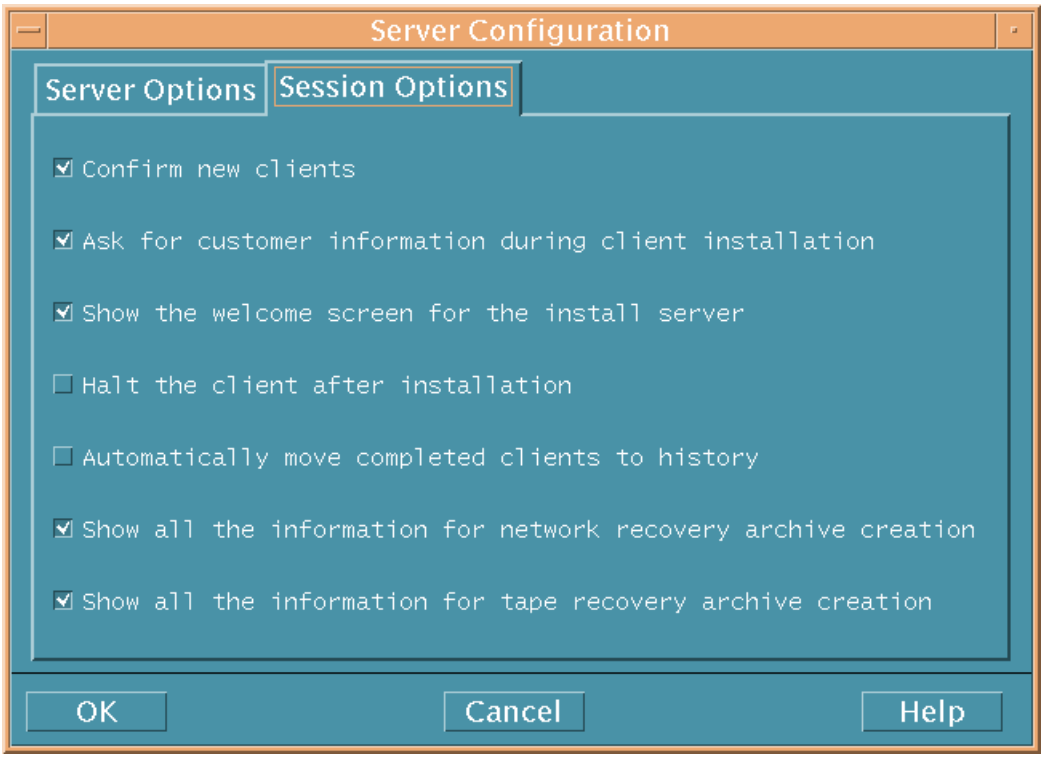
  For more information about setting up DHCP functions, addresses, and class IDs, see "Ignite-UX Server and Boot Helper Setup for DHCP" (page 51), *setup_server*(1M), and *instl_adm*(4).

Configuring Session Options

Ignite-UX allows you to choose how client installation sessions behave. For example, you can decide whether or not to display the Welcome dialog each time you start Ignite-UX, and whether clients are halted on completion of the installation.

The following options are accessible from the Server Configuration dialog box (**Options→Server Configuration**) **Session Options** tab.

Figure 3-9 Session Options Tab



The options you can configure on this tab are explained as follows:

- **Confirm new clients –** Controls whether a confirmation dialog box appears each time a new client is booted from the Ignite-UX server.
- **Ask for customer information during client installation –** Controls whether an input window appears to enable entry of customer name, system serial number, and order number. This information is stored in the `manifest.seed` file in the `/var/opt/ignite/local/manifest` directory. It is used when you are viewing and printing a manifest (see "Viewing and Printing a Manifest " (page 166)) with *print_manifest*(1M). The information entered has no effect on the outcome of an installation.
- **Show the welcome screen for the install server –** Controls whether the WELCOME TO IGNITE UX dialog box appears. The default behavior is to display this dialog box.
- **Halt the client after installation –** Controls whether the client system is halted (rather than rebooted, the default) after installation.
- **Automatically move completed clients to history –** Controls whether completed clients are automatically added to the end of the history log, `/var/opt/ignite/clients/history/history.log`. As part of this action, client configuration and manifest files are automatically moved to the history directory on the Ignite-UX server for future reference. The client icon is removed from the GUI workspace. The client must be `COMPLETE` (fully installed) for this to take place.
- **Show all the information for network recovery image creation –** Controls the amount of information that appears during network *recovery image* creation and installation. The default behavior is to hide this information.
- **Show all the information for tape recovery image creation –** Controls the amount of information that appears during tape recovery image creation and installation. The default behavior is to hide this information.

Setting Up Additional Software on the Server

After you have successfully installed and configured your Ignite-UX server, you might want to set up additional software on the server for installation on clients. Commands written for this

task handle Software Distributor (SD) *depot*s and *bundle*s, but it is possible to configure Ignite to install non-SD software.

The commands that make SD software available for Ignite are capable of sweeping actions, such as packaging an entire software CD/DVD and then making additions to all *configuration clause*s of a specified release. These commands may also be used to fine tune a single configuration clause with a single software addition. Care must be taken when using these commands to get the results you want.

This section is limited in scope and does not attempt to fully address what can quickly become a complex task.

For a complete discussion, see *Ignite-UX Custom Configuration Files* available on http://www.hp.com/go/ignite-ux-docs. Look for the sections "Configuration for software to be installed" and "Installation configurations using Software Distributor depots."

SD Software

Generally speaking, all software supplied by HP for HP-UX is packaged in SD form. Use the steps below to make SD software available to Ignite-UX for installation on clients.

1. If the software is not in a depot, put it in one. The `make_depots` command copies SD bundles to a depot for use by Ignite-UX. See *make_depots*(1M) for more information. For ease of maintenance, HP recommends copying the depots to disk rather than using CD/DVD drives as the source for installation.
2. Run `make_config` on all the depots you plan to use. The `make_config` command creates *configuration file*s for software in a depot. See *make_config*(1M). You must run `make_config` each time you add or modify software in your depot. Be aware that any customizations you've made to a configuration file are lost when you recreate a configuration file with `make_config`.
3. Use `manage_index` to add configuration files to configuration clauses in the `INDEX` file. See *manage_index*(1M).

Example: Create a configuration for compiler software

Given an SD depot of complier software on another server, this example creates a configuration file for that software and adds it to all configuration clauses for the B.11.31 release.

```
# make_config -s server:/depots/compiler \
      -c /var/opt/ignite/data/Rel_B.11.31/compiler_cfg
# manage_index -a -f /var/opt/ignite/data/Rel_B.11.31/compiler_cfg
```

IMPORTANT: Inclusion of multiple versions of Veritas Volume Manager from Symantec (VxVM) in the same installation depot, or in separate depots that are used together in a single cold-installation session, is not supported. Doing so generates errors when attempting to use the installation depot or during reboot when using non-SD depots. For more information, see "Considerations When Using Veritas Volume Manager from Symantec" (page 211).

Non-SD Software

To make non-SD software sources (`tar`, `cpio`, or `pax` *archive*s) available to Ignite-UX, use the example configuration file `/var/opt/data/examples/noncore.cfg`.

1. Make a copy of the `/var/opt/data/examples/noncore.cfg` file and edit it for your particular software. The file contains extensive comments to help you make the changes you need.
2. Use `manage_index` to add configuration files to configuration clauses in the `INDEX` file. See *manage_index*(1M).

IMPORTANT: Do not use *archive*s with files in `/var/adm/sw/*` as software sources. Delivering files to `/var/adm/sw/*` can corrupt the SD Installed Product Database.

4 Simple Network: Creating a Server for Anonymous Clients

This chapter describes how to configure your server to network boot and install HP-UX on *anonymous client*s.

Overview of Anonymous Clients

When booting registered PA-RISC clients, the clients' IP addresses and MAC addresses were entered in the `/etc/opt/ignite/instl_boottab` file. If the clients were Itanium®-based, they were registered in the `/etc/bootptab` file.

An *anonymous client* can be booted from an Ignite-UX server without an IP address previously mapped to its MAC address. Anonymous clients boot using an IP address provided by the server.

Using anonymous client booting on a network is useful when you have many different systems that must be booted, installed, or recovered. It relieves you from the task of configuring for each specific system and eliminates the errors inherent in typing IP addresses and MAC addresses. Such an error could cause IP addresses to be accidentally assigned to more than one computer at a time.

The `/etc/opt/ignite/instl_boottab` file is used to provide PA-RISC systems with anonymous client booting. Within the `instl_boottab` file, if there are IP addresses not assigned to any MAC address, those IP addresses are available to lease to requesting anonymous clients.

Itanium-based clients use DHCP to boot anonymously.

Configuring an Ignite Server to Boot Anonymous PA-RISC Clients

Using the Server Setup Wizard

If you know you want to use anonymous client boot when you start up your Ignite-UX server, you can set it up that way using the Server Setup Wizard.

Start Ignite-UX and the Server Setup Wizard as described in "Configuring the Ignite-UX Server for PA-RISC Clients" (page 37) until you get to the Configure Booting IP Addresses dialog box as shown in Figure 3-5 (page 40).

Enter individual or a range of valid IP addresses. Instead of entering a MAC address in the reserved for: 0x box, leave it blank. When the `instl_bootd` *daemon* requests an IP address for your anonymous PA-RISC client to boot from, it will be given an IP address not registered with any specific MAC address.

Editing the instl_boottab file

You can enter IP addresses in the `instl_boottab` file for anonymous PA-RISC client booting. The `instl_boottab` file contains comments with instructions on syntax. To add an IP address for anonymous PA-RISC booting, simply add that IP address to the file on its own line. Since the IP address is not explicitly marked as reserved or assigned to a MAC address, it is usable by any client. For more information, see *instl_bootd*(1M).

NOTE: No intervention is required to have `instl_bootd` pick up changes to the `/etc/opt/ignite/instl_boottab` file. When a boot request is received, `instl_bootd` always checks whether the file was modified since last read, and rereads it before answering any boot request.

Configuring an Ignite Server to Boot Anonymous Itanium-Based Clients

Working With DHCP

Even on a simple network, there could be devices such as printers requesting network boot. This section describes the challenges involved and solutions for DHCP booting and then acquiring IP addresses for networking.

NOTE: If you are using your Ignite-UX server for DHCP booting, you can set DHCP boot IP addresses from the Ignite-UX GUI by selecting **Options→Server Configuration** as described in "Configuring Server Options" (page 43).

Understanding PXE Booting of Itanium-Based Systems

When an Itanium-based system boots over the network, it sends out a PXE boot request. The PXE protocol is built on top of DHCP. This can cause confusion if there is more than one DHCP server configured to respond to PXE boot requests.

It is not possible for an Itanium-based system to specify the server from which to accept DHCP boot services, ignoring boot offers from all other servers. In other words, there is not an Itanium-based equivalent for the PA-RISC boot command, `boot lan.192.10.10.10 install`, which causes the system to ignore any response except from the IP address 192.10.10.10. This functionality is known as *server selection*.

It is possible for many Itanium-based systems to perform *directed boot*, where server and client networking information is stored in client firmware and DHCP is not used. For more information on directed boot, see "Direct Boot Profiles for Itanium-Based Systems" (page 115).

When an Itanium-based system sends out a PXE boot request, it tries to boot from the first PXE response it gets. If no PXE responses are received within a certain time, the system uses the first DHCP response it gets. If any of these responses are inadequate for network booting, the PXE boot attempt fails and an error message is displayed on the console of the requesting system. The information displayed with PXE errors is usually not explicit enough to determine the cause of the problem (see "Common Network Booting Errors" (page 253)).

For any network where there will be PXE boot requests from Itanium-based systems, only DHCP servers that can supply enough information for a successful boot should be configured to respond. If you have a DHCP server that responds to every DHCP request, regardless of whether it is a PXE request or not, it almost definitely interferes with PXE boot requests from Itanium-based servers. The boot request fails when a normal DHCP response is received in response to a PXE boot request.

In addition to boot failure, the inability to select a boot server can lead to installation of the wrong operating system. Having PXE servers that respond with different *boot content* on the same network can cause confusion. For example, if there is a system supporting Linux boot and a system supporting HP-UX boot on the same network, they can each send a response to a PXE boot request, and the first server to respond will be used. It is not predictable which server would be used for boot.

Interference with a PXE request from a DHCP server is a configuration issue on the DHCP server side. This issue is not specific to HP-UX or Ignite-UX, but rather is related to the way firmware performs a PXE boot.

IMPORTANT: When you configure DHCP servers, make sure there is only one DHCP server on the network that is configured to respond to Itanium-based system PXE boot requests, and that the server is running HP-UX if you want to install HP-UX.

Ignite-UX Server and Boot Helper Setup for DHCP

HP-UX 11i v3 and 11i v2 supports `dhcp_device_group` options that improve anonymous client DHCP booting for Itanium-based clients. The two configuration keywords `re` and `ncid` are used in a DHCP device pool group for this purpose.

Make sure that at a minimum, HP-UX 11i v2 is installed on your Ignite-UX server or *boot helper system*.

Add your device pool group entry to the `/etc/dhcptab` file on your Ignite-UX server or boot helper system.

You should not need to restart `bootpd` if it is already running. When a new `bootp` DHCP request is received, `bootp` checks to see whether it must reread any configuration files. If you want to force `bootp` to reread the configuration file, send it the SIGHUP signal.

The following example DHCP device group is the best way to support anonymous Itanium-based clients:

```
dhcp_device_group:\
    re:\
    ncid:\
    class-id="PXEClient:Arch:00002":\
    lease-time=300:\
    subnet-mask=255.255.255.0:\
    addr-pool-start-address=192.168.1.10:\
    addr-pool-last-address=192.168.1.20:\
    bf=/opt/ignite/boot/nbp.efi
```

The options in the `dhcp_device_group` clause are:

`dhcp_device_group` — Starts a DHCP device pool group for allocating a range of IP addresses to assign to clients with a matching `class-id` in their boot requests.

`re` — A binary option that sets regular expression matching on the `class-id` rather than a default literal match. This is a new option for HP-UX 11i v2.

`ncid` — A binary option that sets removal of the `class-id` from message responses. Since `bootpd` does not support the full Intel Preboot Execution Environment (PXE) protocol, it must not send back a `class-id` in the response. This is a new option for HP-UX 11i v2.

`class-id` — Different kinds of systems may make PXE boot requests. For example, Itanium-based systems and industry standard servers such as HP ProLiant servers may each make a PXE boot request. It is unlikely the same configuration could be used for these different requests. The `class-id` may be used to respond to PXE requests from the correct clients, while ignoring the wrong ones.

All Itanium-based servers send a 32 character PXE boot request in the following format:

`PXEClient:Arch:00002:UNDI:`*`xxxyyy`*

where *xxxyyy* are major and minor numbers for the Universal Network Device Interface revision.

	An industry standard server, such as an HP ProLiant server, sends a PXE boot request in this format: `PXEClient:Arch:00000:UNDI:`*xxxyyy* where *xxxyyy* are the same as described above. The `class-id` in the `dhcp_device_group` example above tells the `bootpd` daemon to respond only to clients with a boot request containing `PXEClient:Arch:00002`. Requests from industry standard servers are ignored. A DHCP server or *boot helper system* configured to respond to any DHCP boot request containing `PXEClient` will respond to both Itanium-based servers and industry standard servers. A PXE response suitable for an industry standard server is unlikely to allow an Itanium-based system to boot.
`lease-time`	How long in seconds the IP address may be used to boot a system. The example value is 300 seconds (5 minutes) but you may need more time if your network is a busy one. Booting on high-traffic networks may take 10 or 15 minutes since the *install kernel* and *install file system* must be downloaded. The problem with increasing the `lease-time` is the possibility of running out of IP addresses to use for booting. If you increase this number, make sure you have enough IP addresses in the pool to accommodate systems that might boot simultaneously.
`subnet-mask`	The subnet mask used by clients.
`addr-pool-start-address`	The first IP address for this address pool.
`addr-pool-last-address`	The last IP address for this address pool.

IMPORTANT: The use of the `ncid` option is critical because it instructs the DHCP server to exclude the DHCP `class-id` in the response to the client's boot request. If a DHCP server responds to a PXE boot request with the DHCP `class-id` in the response, the booting PXE client attempts to communicate with a PXE proxy server on the same host. Since HP-UX does not supply a PXE proxy server, the boot fails. The `ncid` option resolves this issue.

With the device pool group added to the `/etc/dhcptab` file, your HP-UX 11i v2 or 11i v3 Ignite-UX server is now configured to respond to anonymous Itanium-based clients.

IMPORTANT: The server that sends the response to the PXE boot request is the system that the PXE client will attempt to `tftp` the boot file from. If you are not using an HP-UX system to reply to an Itanium-based PXE request, you must make the required boot files available and current with new releases of Ignite-UX. HP does not provide support for this kind of configuration.

Isolating Ignite-UX From Noncontrollable DHCP Servers

Once Ignite-UX starts running, a DHCP request will be used to obtain an IP address used for installation or recovery if needed. Ignite-UX can be configured to specify a class-id for this request.

For more information see Appendix B and *bootpd*(1M).

If you have DHCP servers on your network that you have no control over, it is possible to completely isolate Ignite-UX from them. This is done by adding a `class-id` to the `dhcp_class_id` keyword in the *install file system*. See *instl_adm*(1M) and *instl_adm*(4) for additional information.

When the network boot process completes and the *install kernel* is running, Ignite-UX will use DHCP again to obtain an IP address. This is done because Ignite-UX has no way to determine the IP address used by firmware.

If you are running HP-UX 11i v2 or 11i v3 and have configured a DHCP device group for Itanium-based server PXE requests, you can reuse this device group for isolation purposes. If you added the following into the install file system:

```
dhcp_class_id="IgniteDHCPDeviceGroup",
```

you can change the `class-id` in the DHCP device group that responds to anonymous Itanium–based PXE boot requests to read:

```
class-id="PXEClient:Arch00002|IgniteDHCPDeviceGroup"
```

IMPORTANT:

The `class-id` entry above is a regular expression designed to allow a response to a `class-id` of an Itanium-based system performing a network boot or an `IgniteDHCPDeviceGroup` in `/etc/dhcptab`. This is not a valid `class-id` for use in an Ignite-UX install file system. Systems using a DHCP device group for installing anonymous Itanium-based systems should have `is_net_info_temporary` set to `TRUE` to prevent systems from using the IP address gained via DHCP after installation.

Since regular expression matching is used, | means "or" and allows response to an incoming `class-id` that matches either expression. This example entry would support responding to the initial Itanium-based system boot request as well as subsequent DHCP requests during Ignite-UX operation.

The DHCP servers that respond to any DHCP class-id must be reconfigured or isolated to a different subnet.

The information in this section will not help you isolate a system booting Ignite-UX from other DHCP or PXE boot servers when attempting to network boot from *EFI*. This information does help you stop other DHCP servers from communicating with the installed system after it has already performed a network boot and downloaded an install kernel and install file system.

If you wish to only accept DHCP offers from a specific server after the install kernel and file system loads, consider using the `dhcp_server` keyword in the install file system. The use of the `dhcp_server` keyword has no effect on the EFI/PXE boot process.

Replacing bootpd with instl_bootd

If your Itanium-based system is not running DHCP services, replacing the daemon `bootpd` with the daemon `instl_bootd` allows network booting for registered and *anonymous client*s, and both Itanium-based and PA-RISC clients. See Figure 2-2 “Decision Tree When Configuring a Server for Booting Itanium-Based Systems” and the subsequent discussions for more information.

IMPORTANT: Do not replace `bootpd` with `instl_bootd` if your server is currently providing DHCP services. This procedure configures your Ignite-UX server to run `instl_bootd` instead of `bootpd`. Performing the steps in the following section will prevent the system from providing DHCP services.

Using `instl_bootd` on an Ignite-UX server requires that the `bootpd` daemon is not running on the server. The `instl_bootd` daemon responds to all boot requests from clients. The `instl_bootd` daemon normally runs on a set of unique network ports, 1067/1068, which are used only for booting PA-RISC clients. However, in this implementation, the `instl_bootd` runs on the standard `bootpd` ports, 67/68.

If you are running `bootp` with DHCP on your network, do *not* perform these steps. The `instl_bootd` daemon will answer DHCP requests as if the system were requesting a network boot. Consider other alternatives if you have `bootp` with DHCP running on your network.

Follow these steps to configure your Ignite-UX server to run `instl_bootd` as a replacement for `bootpd`:

1. After you have set up your Ignite-UX server, ensure that `bootpd` is disabled on ports 67/68 by commenting out the following line in the `/etc/inetd.conf` file:

   ```
   bootps dgram udp wait root /usr/lbin/bootpd bootpd
   ```

2. Enable the `instl_bootd` daemon on ports 67/68 by adding the following line to `/etc/inetd.conf`:

   ```
   bootps dgram udp wait root /opt/ignite/lbin/instl_bootd \
   instl_bootd
   ```

3. Restart the Internet daemon, `inetd`, to implement the port changes made in step 2:

   ```
   /usr/sbin/inetd -c
   ```

Your Ignite-UX server is now configured to respond to anonymous clients. For more information, see *instl_bootd*(1M) and *inetd*(1M).

5 Complex Networks: Challenges and Solutions

Most information about Ignite server set up assumes a simple network consisting of one subnet where the server supports network *boot* and *installation*. This simple network configuration is assumed so documentation can be clear and concise.

Often, real network environments are significantly more complex. Configuring an Ignite server to operate correctly in a complex network configuration requires special consideration of network topology.

This chapter identifies some types of complex network challenges and approaches to handle these challenges.

This chapter focuses on Integrity systems only.

How To Use This Chapter

Data centers have unique requirements, constraints, and network topology. It is likely you will have multiple challenges when creating a total solution for system installation and recovery, which will require you to implement multiple solutions for your site.

To help explain network topology, an example complex network diagram will be used that presents multiple challenges. This example network will be referenced throughout the complex networking chapters.

Knowledge of network boot and OS installation steps will help you understand this chapter. Most often, boot and installation is performed by one server. When considering complex network solutions, it sometimes make sense to use separate systems for boot and installation, or to switch servers during the boot process. See the "How Ignite Works" (page 25) section for network boot and OS installation steps information.

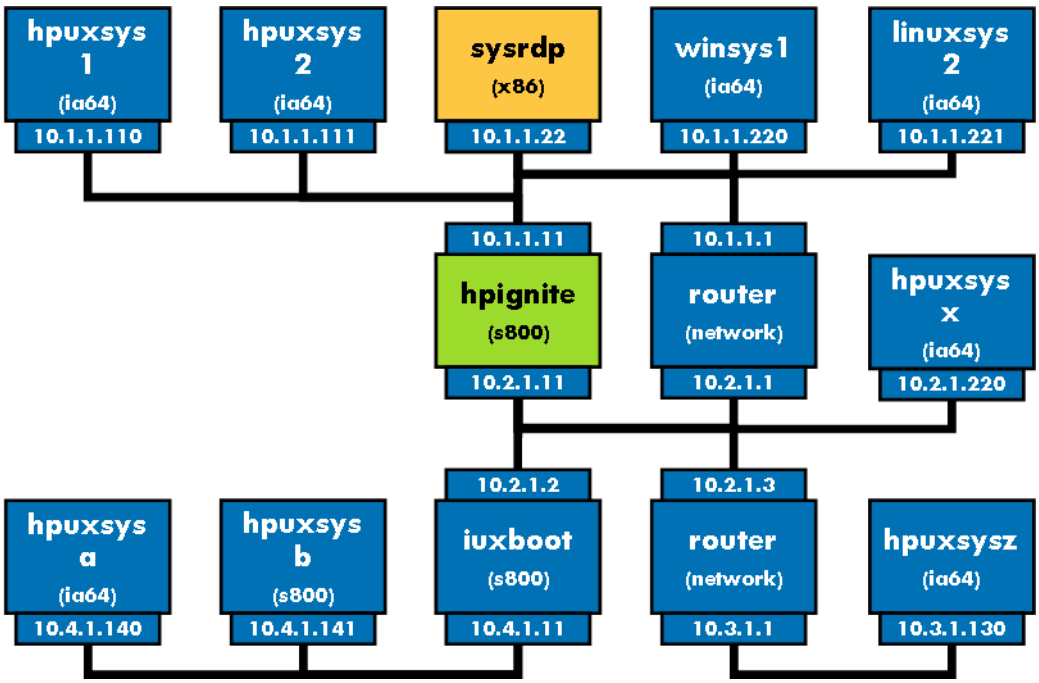
Network boot and installation relies on several protocols that are not detailed here. See "Ignite-UX Server Ports" (page 95) for protocol and port information related to Ignite phases of operation.

It is assumed you have a working knowledge of DHCP, PXE, `bootp`, and TFTP.

Complex Network Challenges

In a complex network configuration, it is often preferable to manage one master Ignite server and use that server to support installation for all subnets. A central server simplifies administration and helps ensure all systems are managed with consistent installation and recovery. The challenge is to have a central Ignite server support network boot for all your required subnets, handle installation, and coexist with any other network boot servers.

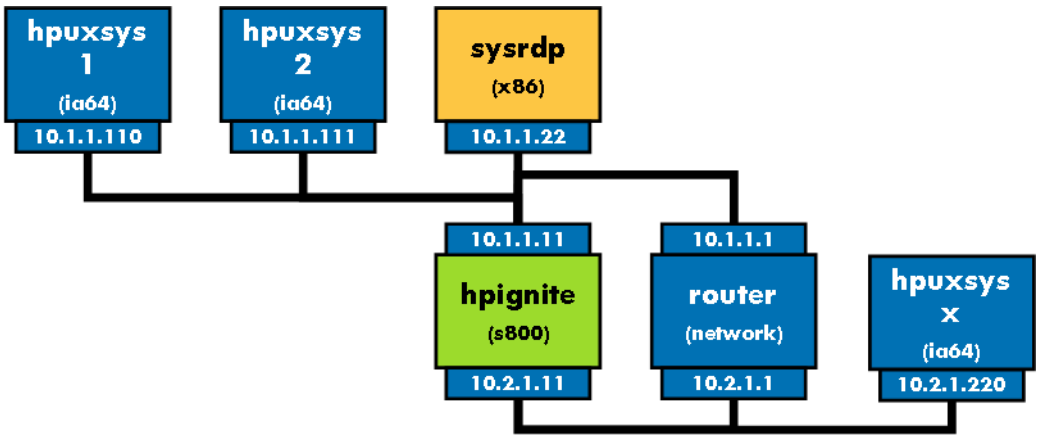
The following diagram illustrates a complex network with multiple subnets (`10.1.1` and `10.2.1`) connected to the Ignite server (`hpignite`), remote systems (`hpuxsysa` and `hpuxsysb`) that use a *boot helper system* (`iuxboot`), a system (`hpuxsysz`) on a separate subnet without a boot helper, and another boot server (`sysrdp`) on the same subnet as the Ignite server. Systems on the same subnet (`10.1.1` or `10.2.1`) as the Ignite server are HP-UX systems (`hpuxsys1`, `hpuxsys2`, and `hpuxsysx`), a Linux system (`linuxsys2`) and a Windows system (`winsys1`). This diagram will be used as an example network configuration throughout the complex network chapters.



Multiple Subnets

The challenge with an Ignite server connected to multiple subnets is ensuring the server is correctly configured to handle client network interfaces for boot and installation on the different subnets. If subnets are isolated or performance is a concern, you will need to ensure that installation traffic is correctly routed to the Ignite server.

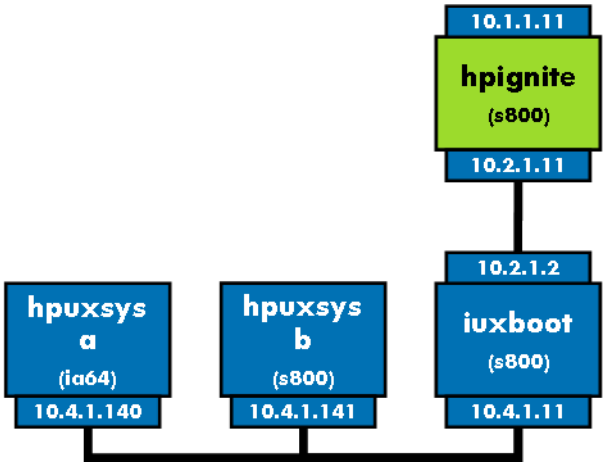
The following diagram shows the example systems used when outlining solutions for a complex network with multiple subnets.



Remote Systems

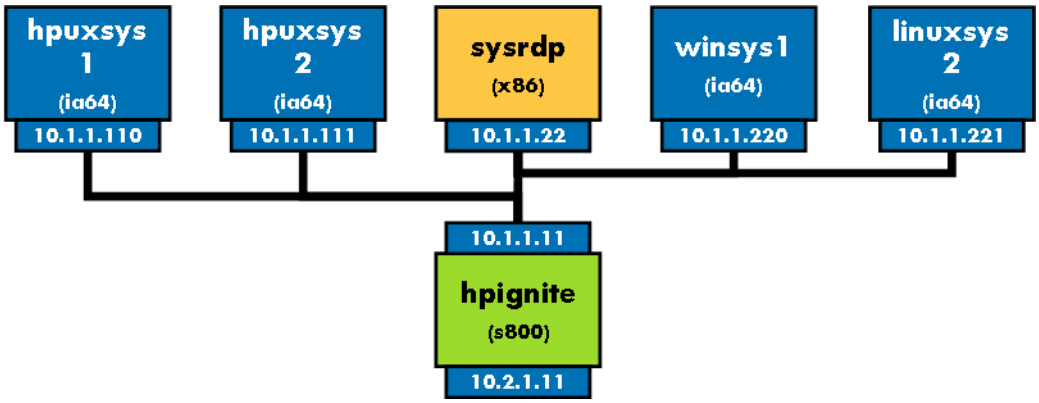
Network boot is based on broadcast protocols. These broadcasts are normally constrained to one subnet. When client systems are on a subnet that is not directly connected to an Ignite server broadcast network, packets used for boot will not be able to reach the Ignite server. If there are remote systems on other subnets (`hpuxsysa` and `hpuxsysb`), you must determine how network boot will be supported on each subnet for these systems. You will also need to ensure that installation traffic is correctly routed.

The following diagram shows the systems that will be referenced when solutions for remote systems are discussed.



Multiple Boot Servers

If there are multiple servers that support boot and installation on a subnet (`sysrdp` and `hpignite`), these systems are very likely to interfere with each other. This is common when systems running different operating systems coexist on the same subnet and network installation is used to manage these systems.



Network boot and installation servers are typically designed with the assumption that they are the only such server on the subnet. Product documentation generally does not include details on how to have multiple servers coexist.

Note that PXE has been designed to assume multiple boot servers provide redundant, identical functionality. The first server to respond to a boot request will be used for system boot. In general, it is not possible to predict which server will respond first.

Often, an administrator wants separate boot and installation servers to provide, for example, different operating systems. In this case, using the correct server is important. As a result, some means of selecting the correct boot and installation server is vital. There is not a simple solution using basic DHCP PXE functionality.

Great care is required to properly set up a network configuration where there are multiple boot servers on a subnet. Each boot server must be configured to correctly coexist with other boot servers and support the desired overall administration solution.

Avoiding Complex Network Issues

The purpose of this section is to provide solutions that avoid the inherent issues in a complex network configuration by modifying the network topology or using boot techniques that avoid boot protocol issues.

An Ignite-UX Server for Each Subnet

If your organization has separate groups that have distinct needs and compute resources, the simplest approach to deal with complex networks might in fact be to manage distinct subnets rather than set up a central Ignite server.

An Ignite server can be placed on each subnet. You may manage each server separately. This avoids the complexities of multiple subnets. Similarly, boot servers for other operating systems can have their own subnets.

Note that newer Integrity systems support HP Virtual Connect technology that permits the remote “rewiring” of network connectivity. This allows systems to be “moved” between subnets via VC profiles, which include network switch configurations. These may be used to avoid issues with managing multiple Ignite servers and subnets.

For information on configuring an Ignite server for a simple subnet, see Chapter 3 (page 37) and Chapter 4 (page 49).

A Multi-Capable Server for Each Subnet

Issues with multiple boot servers on a subnet might be avoided or resolved by having only one boot server on each subnet. Normally, that implies the subnet would have limited boot and installation support for one operating system instead of the ability to support various types of installations.

Depending on the boot server ability for nonstandard configuration, it might be possible to configure a single boot server to initiate or even fully support the installation of a variety of operating systems including HP-UX. Such a configuration is clearly complex and requires expertise in the details of boot and installation support for all the systems and operating systems involved. For more information, see “Multi-Capable Subnet Boot Server” (page 67).

Extend the Local Subnet

In some cases, it is possible to avoid multiple subnet issues by changing the network topology related to network boot functionality. It might be possible to use network tunneling or configure routers to forward some broadcast packets beyond the local subnet.

This results in a larger, single subnet instead of multiple subnets and very effectively avoids issues with multiple subnets. Changing the network topology might work well if data center policies allow and one group manages this larger subnet.

Take care to consider how any network products change network performance and timing, as they might cause boot and installation issues in some cases.

This guide does not include details regarding network infrastructure hardware and software products, and their use. Consult network hardware and software products' information used to extend the local subnet.

Using Virtual LANs Properly for Ignite-UX

If you use Virtual Local Area Networks (VLANs) and you encounter problems during network boot, you need to discover how the VLANs are configured between your Ignite server and client. It is possible to configure VLANs in multiple ways, and some methods might cause issues for Ignite-UX.

The simplest, and possibly the most common, configuration is a single VLAN presented to a single LAN interface where all traffic, including any untagged traffic, travels on the one VLAN. (This method of configuring VLANs is often used to limit the size of a broadcast domain.) This type of configuration does not cause any problems for Ignite-UX because it logically appears as if the client and Ignite server were connected via a switch. The Ignite server will have access to all the network traffic that originates with the client.

A slightly less common, but equally valid, configuration has multiple VLANs configured on a network switch port. Untagged traffic can be presented to only one VLAN, often called the native VLAN. The native VLAN is defined in the configuration of the switch. In this situation, the Ignite server might not have access to the native VLAN of the client.

If the Ignite server does not have access to the native VLAN, it will not have access to any of the untagged traffic from the client. This becomes a problem, since during an Ignite-UX installation or recovery, no network traffic is tagged until the session is complete and the final system is running. This includes *two-step media recovery*.

Problems with VLAN configuration can be difficult to detect, since a system could be configured to use a nonnative VLAN when the operating system is running. This would, for example, allow you to create a *recovery archive* for the client on an Ignite server, but not allow you to recover the client from that same Ignite server.

To remedy this problem, you must provide routing between the VLANs for systems that use nonnative VLANs.

Complex Network Solutions

The approaches outlined in this section can be used to resolve the challenges outlined previously. In some cases, one solution will resolve multiple challenges. Other challenges might require multiple solutions.

Automating HP-UX OS Version Selection

During HP-UX boot for installation, it is necessary to select the HP-UX OS version to be installed. It might be desirable to set up an Ignite server or boot helper to provide different default settings for client systems in order to help automate installation.

The boot loader *AUTO* file controls the menu of HP-UX OS versions and the default selection. The Ignite server may be configured to use separate AUTO files for each HP-UX OS release, as shown below.

```
ln -s /opt/ignite/boot/nbp.efi \
      /opt/ignite/boot/Rel_B.11.23/nbp.efi
ln -s /opt/ignite/boot/hpux.efi \
      /opt/ignite/boot/Rel_B.11.23/hpux.efi
ln -s /opt/ignite/boot/fpswa.efi \
      /opt/ignite/boot/Rel_B.11.23/fpswa.efi
ln -s /opt/ignite/boot/IINSTALL \
      /opt/ignite/boot/Rel_B.11.23/IINSTALL
ln -s /opt/ignite/boot/IINSTALLFS \
      /opt/ignite/boot/Rel_B.11.23/IINSTALLFS
cp    /opt/ignite/boot/AUTO \
      /opt/ignite/boot/Rel_B.11.23/AUTO

ln -s /opt/ignite/boot/nbp.efi \
      /opt/ignite/boot/Rel_B.11.31/nbp.efi
ln -s /opt/ignite/boot/hpux.efi \
      /opt/ignite/boot/Rel_B.11.31/hpux.efi
ln -s /opt/ignite/boot/fpswa.efi \
      /opt/ignite/boot/Rel_B.11.31/fpswa.efi
ln -s /opt/ignite/boot/IINSTALL \
      /opt/ignite/boot/Rel_B.11.31/IINSTALL
ln -s /opt/ignite/boot/IINSTALLFS \
      /opt/ignite/boot/Rel_B.11.31/IINSTALLFS
cp    /opt/ignite/boot/AUTO \
      /opt/ignite/boot/Rel_B.11.31/AUTO
```

The B.11.23 and B.11.31 AUTO files should be edited to have different boot defaults. It is recommended you keep the other boot menu entries so users have the ability to interactively select their desired HP-UX OS version during boot. Alternatively, it is possible to constrain boot options via this approach.

To use this approach, modify `bootptab` entries to use the appropriate boot loader. The boot loader will use the `AUTO` file in the same directory where `nbp.efi` is located.

Ignite requires the HP-UX version of the *install kernel* and *install file system* to match the HP-UX OS version to be installed. By having different defaults in the `AUTO` files, the correct install kernel and file system will be automatically selected.

Limit Network Response by System Class

A subnet might include a variety of systems and devices. Systems that cannot run HP-UX, such as x86 systems and printers, might request network boot. If the subnet includes various systems and devices, HP-UX boot servers should be configured to respond only to device classes they can support.

Note that this approach does not resolve issues with handling client systems that are able to successfully boot and install from different boot servers. The device class is a constant attribute of the system. Thus, this approach will not help resolve issues with multiple boot and installations servers intended to support Integrity systems, since all Integrity systems report the same device class.

For information on how to configure this solution for booting, see "Ignite-UX Server and Boot Helper Setup for DHCP" (page 51).

For information on how to configure this solution for networking IP address allocation, see "Isolating Ignite-UX From Noncontrollable DHCP Servers " (page 52).

Directed Boot

If client system firmware supports directed network boot, that is a simple way to avoid complex network boot issues. *Directed boot* support is not available in some older Integrity systems. Also, directed boot requires interaction with the system console. Finally, directed boot requires specifying the network configuration. In most cases, this configuration needs to be consistent with DHCP or boot server configuration for the client.

Directed boot is provided by the *EFI* commands `dbprofile` and `lanboot`. These commands make it possible to specify the network configuration to be used for boot (IP address, netmask, *gateway*, etc.). For convenience, a `dbprofile` may be associated with one or more EFI boot menu options so subsequent boot for installation from the master Ignite server is simplified. Because the direct boot profile may include a gateway, it is possible to directly boot from an Ignite server on a separate subnet without using a local subnet boot helper.

For detailed information on directed boot, see "Direct Boot Profiles for Itanium-Based Systems" (page 115).

Server Selection

PA-RISC clients can specify the boot server to use for DHCP/`bootp` services and ignore all other boot offers. The client system network configuration is supplied by the boot helper or Ignite server located on the same subnet. It is also possible for PA-RISC systems to "search" for network boot options and build a list that may be used to select the desired boot server. For more information, see "Booting PA-RISC Clients from the Console " (page 111).

Limit Network Boot Response by Network Interface Address

When a system broadcasts a request for *boot*, the request includes the NIC network address (also known as the MAC address).

Boot servers are often configured to respond to all systems that broadcast a boot request since that approach simplifies administration. However, most boot servers have the ability to selectively respond to boot requests.

You can typically configure boot servers to use the network address to decide whether to respond to the client system or not. If the server responds, the network address is typically used to determine the correct client-specific network configuration (IP address, netmask, gateway, etc.).

When using this approach, boot servers typically have configuration content that allows them to respond to a set of MAC addresses. For HP-UX servers, the `/etc/bootptab` file is used to identify the clients to respond to. Listed MAC addresses will receive a response using the client-specific details in the `bootptab` file. For more information, see Chapter 3 (page 37).

Some non-HP-UX boot servers may be configured to ignore a set of network addresses and respond to others. Note that HP-UX does not support this capability.

NOTE: HP Virtual Connect (VC) network technology allows MAC addresses to be changed via profiles. It is possible to allocate a range of MAC addresses for different boot servers, configure an HP-UX server to manage that block of MAC addresses, and then use profiles to select the boot server. Thus, instead of changing the boot server configuration content, the MAC addresses of a system could be changed with VC profiles to effectively choose the correct boot server.

Control Network Boot via Response Timing

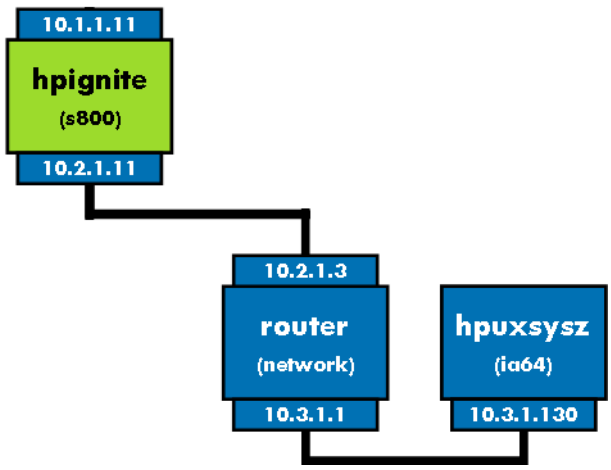
Since a client system uses the first network boot response it receives, response timing may be used to help select the boot server. One boot server that responds to any MAC address may be configured to respond only after a delay, while all other servers are configured to respond to specific MAC addresses.

Because a number of factors might influence network boot response timing, and servers might respond more slowly in some cases, this approach has a risk of intermittent failures caused when the delayed server responds first because the delay is set too short.

Care is required in deciding the appropriate response delay. Note that a one-to-two second delay for other boot server responses might be large enough to cause an HP-UX server responding to a specific MAC address to normally win. However, a larger delay of eight seconds or more is recommended. You will need to decide the correct delay for your subnets and servers. These recommendations are intended to emphasize that the delay should not be set to "just large enough to work" based on limited testing.

Install Remote Clients Through a Network Router

By default, Ignite configures *IINSTALLFS* configuration content without network routing information. If the Ignite server and depot server are on a remote subnet accessed via gateway, and registered IP addresses are used instead of DHCP, the IINSTALLFS configuration content should include network routing information.



The `server` keyword specifies the IP address for your Ignite-UX server. It refers to a specific LAN interface on the Ignite-UX server. The same is true for the `sd_server` keyword that specifies the depot server IP address for any depots needed for installation.

```
server = "10.2.1.11"
sd_server = "10.2.1.11"
netmask[] = "255.255.255.0"
route_gateway[0] = "10.3.1.1"
route_destination[0] = "default"
disable_dhcp = "true"
```

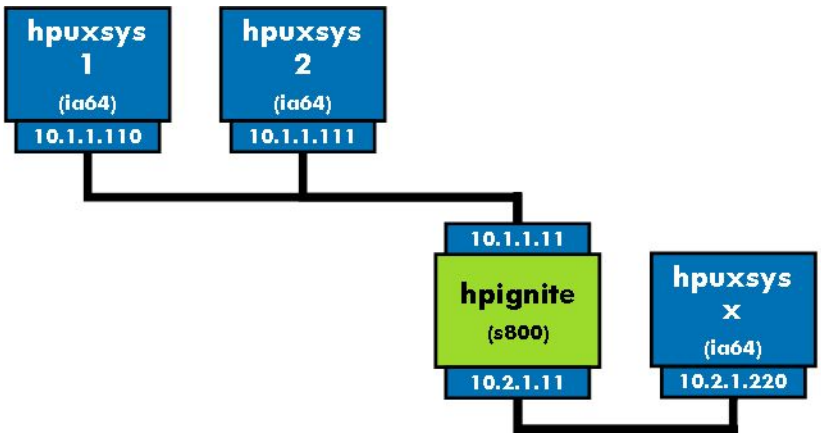
If DHCP is used, this should not be needed since the DHCP server should provide appropriate routing information.

IMPORTANT: The `sd_server` IP address setting in the installation file system will be overridden by `sd_server` settings in configuration files created by `make_config`. To use IINSTALLFS to control the `sd_server` setting you must comment-out or remove the `sd_server` settings within each `sw_source` clause in the affected configuration files.

For more information about changing IINSTALLFS content, see *instl_adm*(1M) and *instl_adm*(4).

Multiple NICs Attach the Ignite Server to Multiple Subnets

It is possible to connect one Ignite server to multiple subnets using multiple network interface cards (NICs). The Ignite server can then be used to support network boot and installation on each of these separate subnets. The following diagram is an example of a multiple subnet complex network.



There are some special Ignite configuration considerations for supporting multiple subnets from one server. You must be careful to configure the server so it sends the client correct networking information for its subnet, and you must configure the client so it contacts the correct Ignite-UX server.

Getting the Client the Correct Networking Information

The HP-UX Ignite server needs to be configured so the client receives a response from the server with the correct network configuration for its subnet.

The `bootptab` content for each MAC address needs to have correct networking configuration for the system's subnet. In particular, IP addresses, netmasks, and gateways need to be valid from the client system's perspective. Thus, `bootptab` will need to include information for all the different subnets used by clients supported by the master Ignite server.

If anonymous network boot is used, care is needed to ensure boot responses are correct. See Chapter 4 (page 49) for more information.

Having the Client Contact the Correct Server

The `server` keyword in the IINSTALLFS configuration information specifies the IP address for your Ignite-UX server. It refers to a specific LAN interface on the Ignite-UX server. The same is true for the `sd_server` keyword that specifies the depot server IP address for any depots needed for installation.

If a client is on a subnet that does not have a route to the IP address specified by `server`, then it will not be able to contact the server after it boots.

For performance reasons, you might want to use the IP address of the Ignite-UX server LAN interface directly connected to the same subnet as the client.

If packets are not routed between different subnets connected to the Ignite-UX server, the client must use the IP address of the Ignite-UX server LAN interface on the same subnet.

Individual clients might use different IP addresses to access the Ignite server due to performance reasons or routing reasons as described in "Install Remote Clients Through a Network Router" (page 61). To customize the IP address used to access the Ignite server on a per client basis, use the `LLA` keyword as described below.

Workarounds to specify the IP address of the Ignite server are:

- Correct the server's IP address on the Ignite-UX network setup screen that appears on the client console when you boot the client.
- Change the configuration content in IINSTALLFS to select the Ignite-UX server and depot server IP addresses connected to their local subnet.

```
server = "10.1.1.11"
sd_server = "10.1.1.11"
LLA[] == "00306E4A03C2" {server = "10.1.1.11" sd_server = "10.1.1.11"}
LLA[] == "00306E4A03C3" {server = "10.2.1.11" sd_server = "10.2.1.11"}
```

IMPORTANT: The `sd_server` IP address setting in the installation file system will be overridden by `sd_server` settings in configuration files created by `make_config`. To use IINSTALLFS to control the `sd_server` setting you must comment-out or remove the `sd_server` settings within each `sw_source` clause in the affected configuration files.

For more information about changing IINSTALLFS content, see *instl_adm*(1M) and *instl_adm*(4).

Ignite-UX `bootp` Boot Helper

An HP-UX server may be used as a boot helper to support boot on a subnet while installation is accomplished via a master Ignite server connected to a different subnet.

A system with Ignite software installed may be located on each subnet to support initial boot. This subnet-local Ignite server may be set up with IINSTALLFS configuration content to specify the single Ignite master server.

Using this approach, all the HP-UX installation and recovery content may be managed on one system. However, each local subnet Ignite boot helper server must be configured to support network boot for all the clients on that subnet. The boot helper server may be configured for promiscuous network boot or via selective MAC address response.

The advantage of this approach is there can be a single Ignite server that handles all HP-UX installation and recovery content. A significant disadvantage of this approach is that Ignite software must be installed on each boot helper. The version of Ignite installed on these boot helpers must always match the version installed on the master Ignite server.

The Ignite product content located in `/opt/ignite/boot` must be present on the Ignite boot helper so it may be used to accomplish network boot.

Follow these steps to set up an Ignite-UX *boot helper system* on the local subnet:

1. Install the Ignite-UX minimum core functionality onto the helper system. The *Ignite-UX_server*:*/depot* is the same Ignite product software depot or media used to set up the master Ignite server.

   ```
   # swinstall -s Ignite-UX_server:/depot Ignite-UX.MinimumRuntime
   ```

2. On the boot helper system, set the default Ignite-UX server. Make sure the correct server is set and any network routing is configured as described in "Having the Client Contact the Correct Server" (page 63) and "Install Remote Clients Through a Network Router" (page 61). This changes the configuration content in *[W|V|I]INSTALLFS*. For more information, see *instl_adm*(1M).

   ```
   # instl_adm -t Ignite-UX_server_IP
   ```

3. Verify that the server is set to the correct Ignite-UX server and gateway for your subnet.

   ```
   # instl_adm -d
   ```

4. If the `gateway` value is incorrect, you can use `instl_adm -g` *`yourgatewayIP`* to correct it. This value is set by `swinstall` when the Ignite-UX product is installed.
5. On the boot helper system, configure the `/etc/opt/ignite/instl_boottab` file as described in "Getting the Client the Correct Networking Information" (page 62).

   For more information, see `/etc/opt/ignite/instl_boottab` on your Ignite-UX server.

HP-UX DHCP PXE Next Server Boot Helper for Integrity Systems

To support Integrity systems, a lightweight Next Server boot helper may be set up on each subnet. A DHCP PXE response includes a Server Address (SiAddr) field that indicates where to get additional network *boot content*. Normally, the value in a response informs the client to get subsequent boot content from the same boot server. The `/etc/bootptab` file can be configured to inform the client to switch to the master Ignite server for other boot content. The `bootptab sa` option is used to indicate the value. This is commonly described as a Next Server value, since the Server Address value is typically only given when the value differs from the initial boot server. The master Ignite server IP address should be used.

In this approach, each subnet must have a DHCP PXE server, but Ignite does not need to be installed on that system. Therefore, there is no need to have multiple systems with the same Ignite software version installed on them. The HP-UX `bootp` server may use the Next Server field to direct the client system to get the HP-UX OS content from an Ignite server on a different subnet.

Using this approach, all the HP-UX installation and recovery content may be managed on one system. However, each local subnet Next Server DHCP PXE boot helper must be configured to support network boot for all the clients on that subnet. The boot helper server may be configured for promiscuous network boot or selective response using client-specific network configurations.

Note that this Next Server boot helper does not have to be an HP-UX system. If it's not an HP-UX system, care must be taken to make sure the PXE response is consistent with the Ignite server. In particular, the `boottab bf` option provided from the Next Server boot helper must be consistent with where the boot content is located on the master Ignite server. Symbolic links may be used to allow a nonstandard location to be supported on the master Ignite server, if needed.

Configuring a Next Server Boot Helper for Integrity systems

A Next Server boot helper does not require Ignite-UX software.

If the DHCP PXE server is an HP-UX system, it must be running 11i v2 (B.11.23) or later. If 11i v2 is used, PHNE_36209 or a superseding `bootpd` patch must be used to enable the configuration of Next Server response.

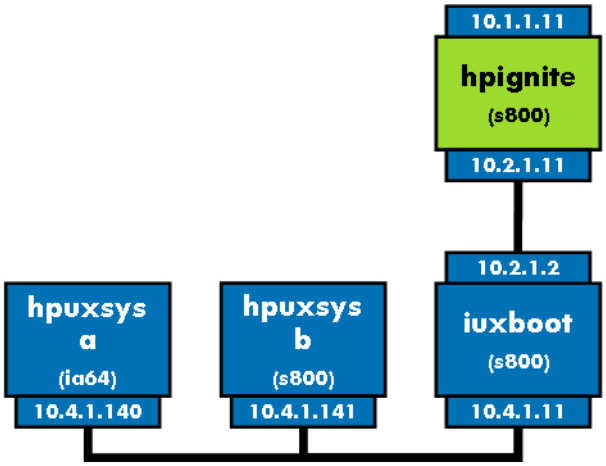
The `nbp.efi` boot loader file must be present on the Next Server boot helper:

```
# cp /usr/lib/efi/EFI/HPUX/nbp.efi /opt/ignite/boot/nbp.efi
```

If the Next Server boot helper is a PA-RISC system, this boot loader file will have to be copied from an Integrity system. Note that the Ignite-UX product may be installed instead of copying this file in place.

The Next Server response is configured in `/etc/bootptab` using the `sa` option. The IP address given with the `sa` option should be the DHCP PXE Next Server (SiAddr) IP address for additional boot content.

This example configuration is for the following complex network diagram.



A sample `/etc/bootptab` for Next Server boot helper configuration follows:

```
next-server:\
     bf=/opt/ignite/boot/nbp.efi:\
     ht=ethernet:\
     hn=:\
     bs=48:\
     dn=xyzco.com:\
     gw=15.1.128.1:\
     sm=255.255.255.0:\
     ds=10.2.1.11

hpuxsysa:\
     tc=next-server:\
     ha=00306E4A03C2:\
     ip=10.4.1.140:\
     sa=10.2.1.11:
```

During DHCP PXE boot, the boot helper server provides the network configuration (IP address, netmask, gateway, etc.). The boot helper also provides the initial boot loader (`nbp.efi`). All other boot content is taken from the master Ignite-UX server. Thus, this boot helper server requires no Ignite-UX product content.

Note that the `bf` option path must match the path where other boot content is located on the master Ignite server. The `bf` path must be valid on the boot helper and the Ignite master server.

Make sure the correct server is set and any network routing is configured as described in "Having the Client Contact the Correct Server" (page 63) and "Install Remote Clients Through a Network Router" (page 61).

Forwarding Boot Requests via `bootp` Relay

The HP-UX `bootp` server has the ability to forward boot requests. With this approach, each subnet must have a `bootp` relay boot helper, but that system does not need to have Ignite software installed. Therefore, there is no need to have multiple systems with the same Ignite software version on them.

When a client system broadcasts a request for network boot, the bootp relay boot helper will forward the request to the master bootp server indicated in the bootptab. The master bootp server will respond to the bootp relay boot helper, which will then forward the response back to the client system. The master Ignite boot server and master bootp server should be the same system.

The bp option must be specified in the bootp relay boot helper's /etc/bootptab file to forward boot requests to the master bootp server. The bp value should be the IP address of the master bootp server. The ip option must not be specified since that value will be provided from the master bootp server. Often an hm option is also specified so a single bootptab relay configuration may be used for many or any clients. The hm option is a MAC address mask used to determine if the MAC address of the requestor matches the MAC address of the ha bootptab option.

```
     ha=000000000000:\
     hm=000000000000:\
     bp=10.2.1.11
```

Note that care is needed if the system is connected to multiple subnets, since the bootp relay helper will respond to boot requests detected on any NIC. Make sure the correct server is set and any network routing is configured as described in "Having the Client Contact the Correct Server" (page 63) and "Install Remote Clients Through a Network Router" (page 61)

See *bootpd*(1M) for details of configuration.

Note that this approach works for PA-RISC systems but might not work for Integrity systems. For Integrity systems, the HP-UX boot loader configures the bootp relay boot helper as a gateway system for network configuration. If that system does not route packets between subnets, this might impede successful use of this approach. If the system routes packets, it will be attached to multiple subnets and therefore respond to boot requests detected on multiple subnets.

Ignite-UX software does not have to be installed on the bootp boot helper.

The following is an example of a local bootp relay boot helper /etc/bootptab content to respond to any client MAC address:

```
bootp-relay:\
     ht=ether:\
     ha=000000000000:\
     hm=000000000000:\
     bp=15.2.1.11
```

The following is an example of selective, MAC-specific /etc/bootptab content:

```
hpfcixa:
     ht=ether:\
     ha=001083352de5:\
     bp=15.2.1.11
```

A block of MAC addresses may be specified using the ha and hm options. This approach might be very useful with HP systems supporting HP Virtual Connect technology.

The master Ignite server needs to have entries for the client system boot requests forwarded from the bootp forward boot helper:

```
ignite-defaults:\
     bf=/opt/ignite/boot/nbp.efi:\
     ht=ethernet:\
     hn:\
     bs=48:\
     dn=xyzco.com:\
     gw=10.4.1.1:\
     sm=255.255.255.0:\
     ds=10.2.1.11

hpfcixa:\
     ht=ignite-defaults:\
     ha=001083352de5:\
```

```
ip=10.4.1.100:\
bp=10.2.1.11
```

To use the `bootp` relay boot helper with PA-RISC systems, boot using standard ports, such as:

```
boot lan.10.2.1.11
```

The installation option to use HP-UX specific network ports might not work:

```
boot lan.10.2.1.11 install
```

Multi-Capable Subnet Boot Server

It is possible to set up a boot server that supports boot for multiple operating systems, including HP-UX. The multi-capable boot server may be an HP-UX system or a non-HP-UX system. If this boot server is an HP-UX system, the challenge becomes: how do I configure the HP-UX boot server to support non-HP-UX boot and installation. If the system is not an HP-UX system, the challenge becomes: how do I set up HP-UX boot and installation on the non-HP-UX system.

The information in this section is intended to provide general information to assist those setting-up a multi-capable boot server to initiate or fully support the installation of a variety of systems, including HP-UX installation. There is no general recipe for configuring these systems. In all cases, expertise is required to adapt these approaches to practical solutions.

For more detailed information about setting up specific types of boot/installation servers to support specific operating systems, see Chapter 6 (page 69).

Non-HP-UX Next Server Boot Helper

If the non-HP-UX boot server supports configuration of the DHCP PXE Server Address (SiAddr) response data, the simplest approach is to have the response specify the master Ignite server (Next Server) to be used for all additional boot content. The non-HP-UX boot server will still need to be configured to determine when boot control should be passed to the Ignite server for HP-UX installation, and when control should be retained to perform other installations. This can be accomplished by using MAC addresses or with a menu of boot options, for example.

Note that the directory where the `nbp.efi` boot loader is located must match the location of other HP-UX boot content on the Ignite master server. If necessary, a symbolic link may be used from the directory path matching the non-HP-UX server location to the standard HP-UX location for boot content.

For more information, see "HP-UX DHCP PXE Next Server Boot Helper for Integrity Systems" (page 64).

Non-HP-UX `bootp` Boot Helper

If the non-HP-UX boot server cannot be configured to support a custom DHCP PXE Server Address value, it is necessary for the server to provide the initial Ignite *install environment* content. To make this approach work, copy content in the `/opt/ignite/boot` directory to the non-HP-UX boot server.

While best practice might be to use the same directory path, there is no particular need for the path to be the same. The path where the Ignite install environment is located on the non-HP-UX boot server must match the DHCP PXE Boot File response value, but does not need to match the default location on an Ignite server. The initial install environment will be entirely taken from the non-HP-UX `bootp` boot helper system.

Note that the version of any Ignite content copied to a non-HP-UX boot server must match the version of the content on the Ignite master server; it will have to be updated when a new version of Ignite is installed.

Also, note that the *AUTO* file and *IINSTALLFS* files include Ignite-UX configuration content. It is important to keep this configuration content consistent with the Ignite-UX server.

Keeping versions and configuration content consistent between these servers can be difficult. If these servers are managed by different groups, ongoing administration might make this approach impractical.

6 Complex Networks: Multi-Capable Servers

This chapter presents a variety of ideas for using servers in a complex network. There is no one solution when configuring servers in a complex environment – look for the solutions that work in your data center.

This chapter focuses on Integrity systems only.

Configuring an RDP Server for Specific MAC Addresses

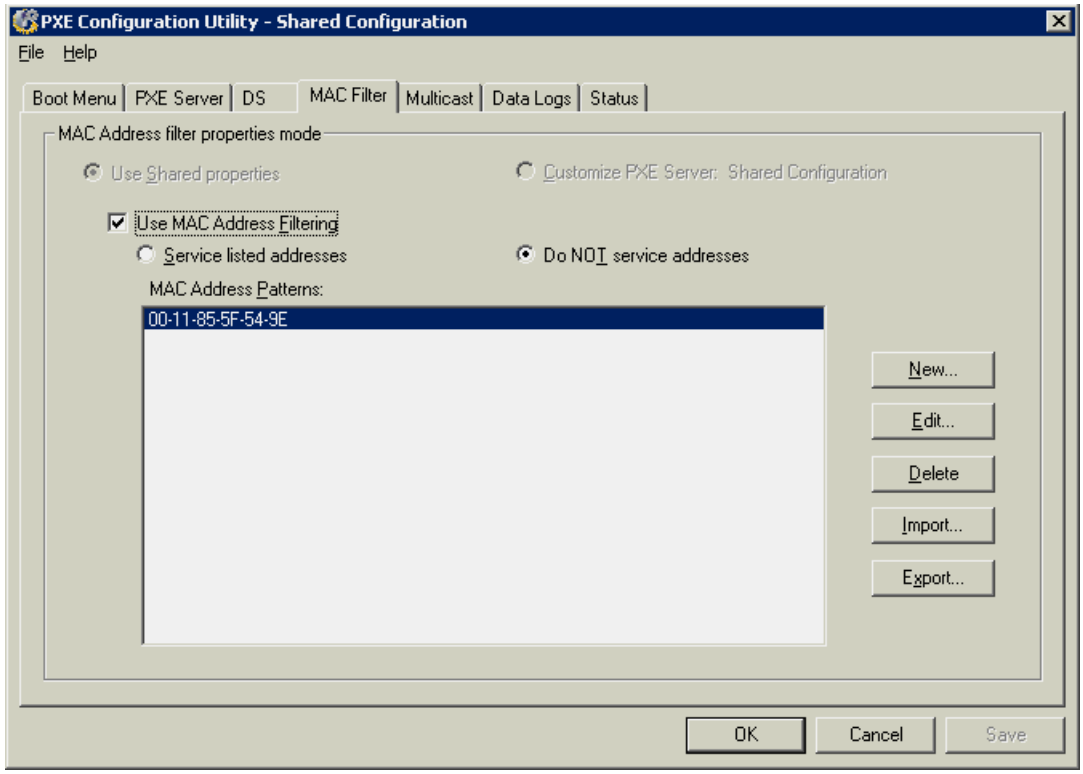
Conflicts between multiple boot servers on a subnet may be avoided if each boot server only responds to the MAC addresses of client systems it should manage. The RDP PXE server may be configured to selectively respond to network boot requests based on the MAC addresses of client systems. If there are more Windows and Linux systems than HP-UX systems, it makes more sense to configure the RDP server to ignore the MAC addresses of HP-UX systems instead of configuring the RDP server to respond to the MAC addresses of its client systems.

To do this, use the **MAC Filter** tab on the PXE Configuration Utility as shown in the figure below.

The Interactive UI may be started using the Windows Start Menu:

Start→All Programs→Altiris→PXE Services→PXE Configuration Utility

It may also be started from the RDP Deployment Solutions Console.



Configuring an RDP Server to Delay PXE Response

The RDP PXE server may be configured to delay responding to network boot requests. This gives a chance for an HP-UX network boot server to consistently respond first. The HP-UX boot server should be configured to only respond to specific MAC addresses via `bootptab` content, allowing the RDP server to manage the non-HP-UX systems.

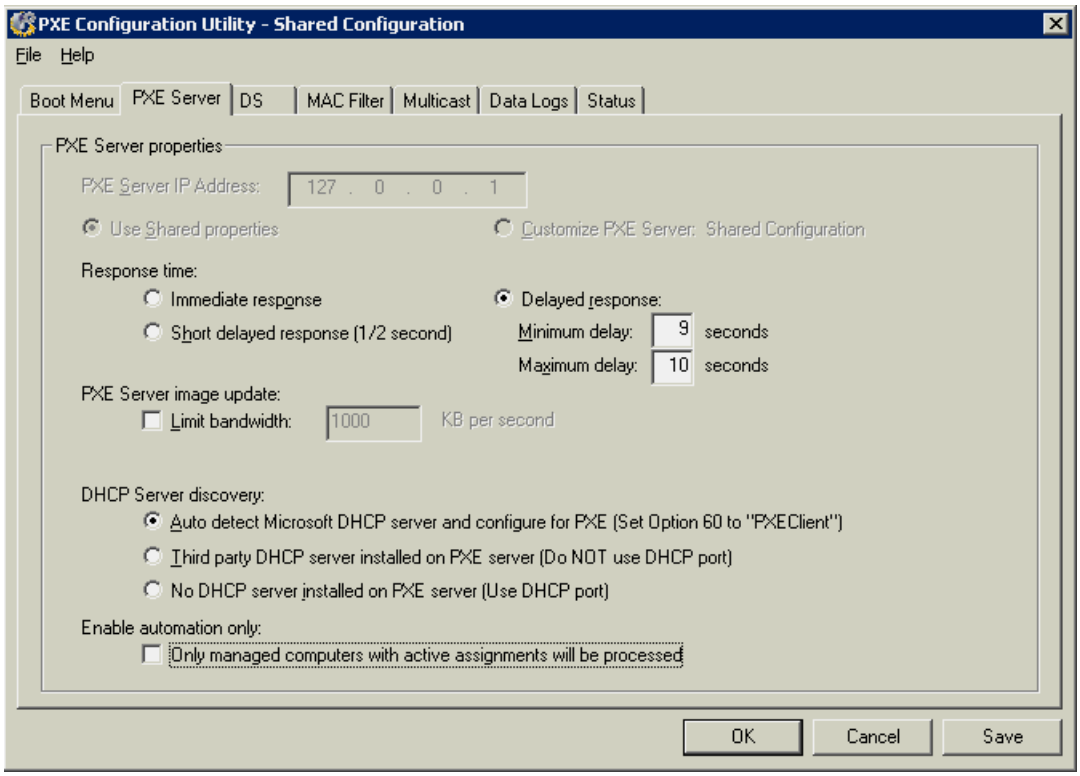
The PXE Configuration Utility on the RDP server may be used to specify the response delay. Use the **PXE Server** tab as shown in the figure below.

The Interactive UI may be started using the Windows Start Menu:

Start→All Programs→Altiris→PXE Services→PXE Configuration Utility

It may also be started from the RDP Deployment Solutions Console.

IMPORTANT: The recommended Minimum delay is 9 seconds and the Maximum delay is 10 seconds. These values should be large enough to provide predictable results considering some possibility of HP-UX boot server delays due to the server being busy and possible network congestion.



Configuring an RDP Server to Initiate HP-UX Installation

An HP ProLiant Essentials Rapid Deployment Pack (RDP) boot server may be configured to act as an HP-UX boot helper. You can do this by adding RDP MenuOptions for the initiation of HP-UX boot for install and placing the Ignite *install environment* on the RDP server. The RDP server may then be used to initiate an Ignite-UX installation of HP-UX.

To set up boot for HP-UX installation, use the PXE Configuration Utility to create the necessary MenuOptions and copy the Ignite install environment content into the proper locations on the RDP server.

The PXE Configuration Utility program is typically located on the RDP server at

```
C:\Program Files\Altiris\eXpress\Deployment Server\PXE\PxeConfig.exe
```

Setting up RDP MenuOptions via Windows Commands

A Windows command line may be used to automate Ignite support set up. To create a MenuOption on an RDP server for HP-UX installation, or to update it with new Ignite *install environment* content, the following command line arguments may be used:

```
PxeConfig.exe -create -other "HP-UX Managed" -ia64 \
     -pathia64 <ignite_content_dir> -save
```

The *ignite_content_dir* directory needs to contain the following HP-UX boot and Ignite install environment files:

- `.\MenuOption160.0` (`nbp.efi` renamed as required by the RDP server)
- `.\MenuOption161.0` (`nbp.efi` renamed as required by the RDP server)
- `.\MenuOption162.0` (`nbp.efi` renamed as required by the RDP server)
- `.\MenuOption163.0` (`nbp.efi` renamed as required by the RDP server)
- `.\Rel_B.11.23`
- `.\Rel_B.11.23\IINSTALLFS`
- `.\Rel_B.11.23\IINSTALL`
- `.\Rel_B.11.31`
- `.\Rel_B.11.31\IINSTALL`
- `.\Rel_B.11.31\IINSTALLFS`
- `.\fpswa.efi`
- `.\hpux.efi`
- `.\AUTO`

The RDP server requires the file name of the initial boot loader file and the name of the MenuOption to match. Normally, `MenuOption160` will be used by the RDP server. However, if there are existing custom MenuOptions, another file name might be required. Copying the small `nbp.efi` boot loader to the first few custom MenuOption names the PXE Configuration Utility uses for MenuOptions should simplify the automatic set up of the RDP server. For HP-UX, this file is actually the `nbp.efi` network boot loader.

The Ignite-UX version for this content on the RDP server must be kept consistent with the Ignite-UX product version on the Ignite server. Note that *IINSTALLFS* includes *config file* content typically modified via use of `instl_adm`. Since these files will reside on the RDP server, a data center administration process for updating these files should be created to keep the RDP files consistent with the Ignite server.

The Ignite *install environment* on the RDP server needs to be updated when:

- The `IINSTALLFS` config content is changed
- A new version of Ignite software is installed on the Ignite server
- Changes are made to the *AUTO* file boot menu
- The HP-UX boot loader is patched

Setting up RDP MenuOptions via Interactive UI

The RDP server PXE Configuration Utility may be used to interactively create a MenuOption for HP-UX boot and Ignite installation.

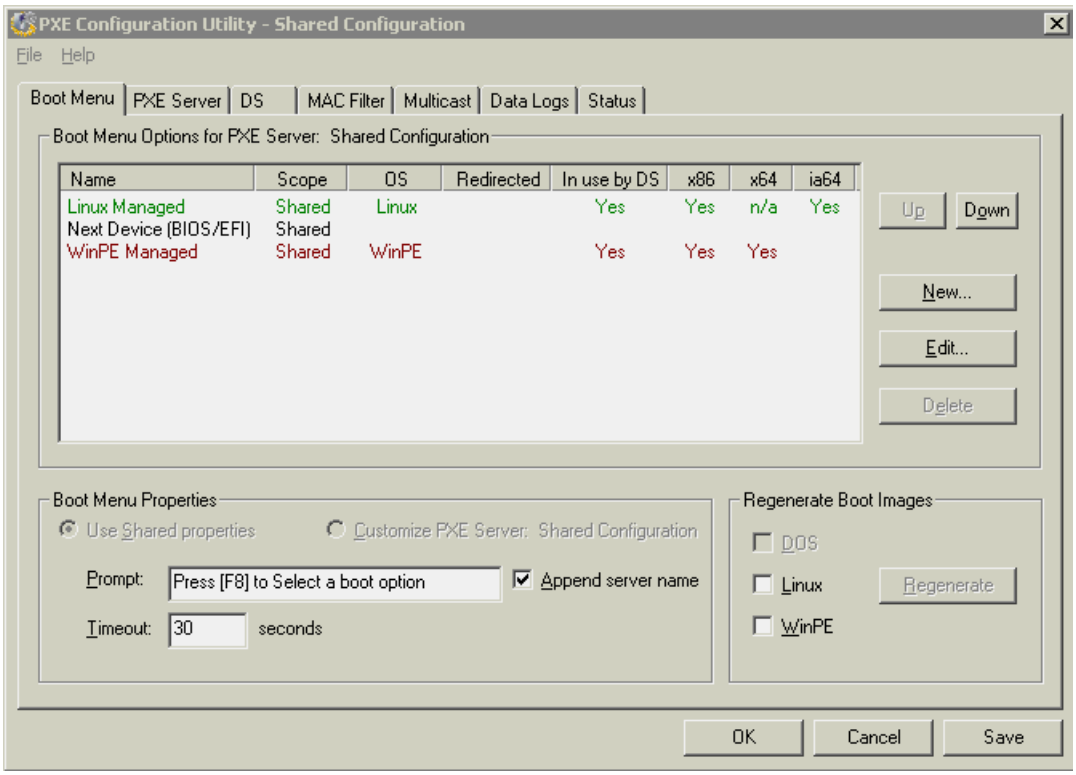
Before starting the interactive UI, put the Ignite-UX boot and *install environment* files in a directory as specified in "Setting up RDP MenuOptions via Windows Commands" (page 71). Any convenient method may be used to transfer the required content from the Ignite server to the RDP server (FTP, Samba share, key chain drive, etc.). You should consider how this content will be updated in the future to stay consistent with the Ignite server.

The Interactive UI may be started using the Windows Start Menu:

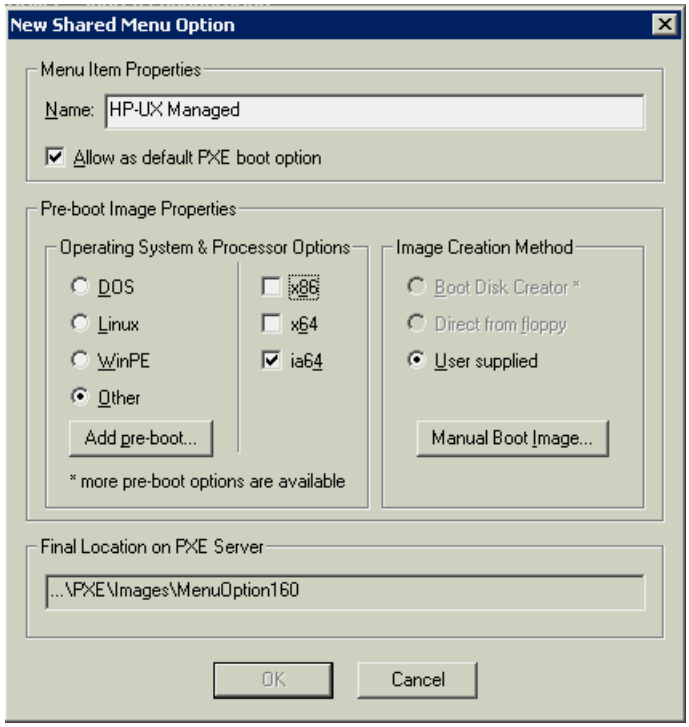
Start→All Programs→Altiris→PXE Services→PXE Configuration Utility

It may also be started from the RDP Deployment Solutions Console.

Once started, the PXE Configuration Utility will display a window similar to the following:



On the **Boot Menu** tab, select the **New...** button to create a MenuOption for HP-UX boot and installation.



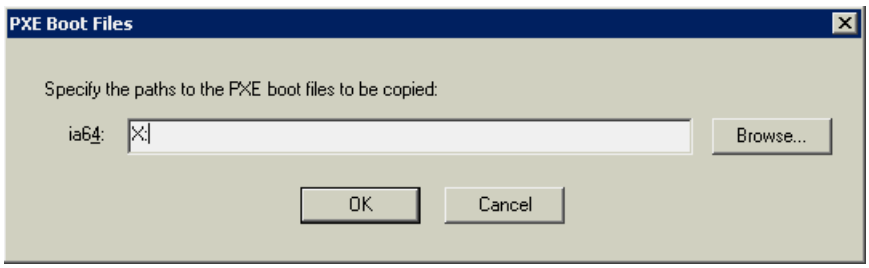
You should validate that the selected MenuOption number in the Final Location on PXE Server text box is consistent with the HP-UX Ignite content provided.

Configure the MenuOption as follows:

1. Enter the name of the MenuOption to be created in the Name: text box.

   For HP-UX, the name "HP-UX Managed" is recommended. If HP-UX release-specific MenuOptions are created by using different *AUTO* content, a name such as "HP-UX 11.31 Managed" is recommended.

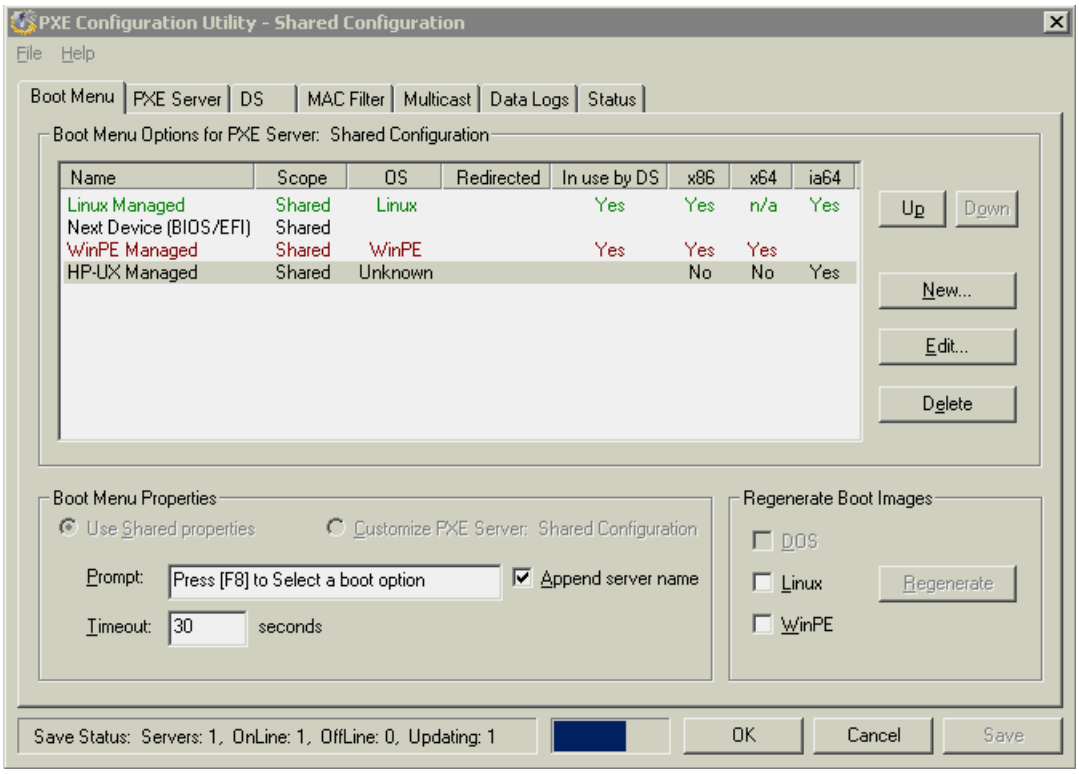
2. Configure the Pre-boot Image Properties.
   - Select Operating System type Other.
   - Select only the ia64 Processor Option, since HP-UX only supports ia64 systems. Make sure x86 and x64 are unselected.
   - Select User supplied for the Image Creation Method, and then select **Manual Boot Image**. In the PXE Boot Files text box, enter the directory containing the Ignite-UX boot and *install environment* files, then select **OK**.



3. Select **OK** from the New Shared Menu Option dialog box.
4. Select **Save** on the Boot Menu tab of the PXE Configuration Utility – Shared Configuration menu.

IMPORTANT: You must select **Save** to enable the new boot menu option before selecting **OK** to exit.

Once set up is complete, the PXE Configuration Utility will show the new MenuOption for HP-UX.



Using an RDP MenuOption for HP-UX

Once the HP-UX MenuOption is set up using either the PXE Configuration Utility command or UI, the RDP PXE server will include “HP-UX Managed” as an option during network boot of ia64 client systems.

When a client system is booted from an RDP PXE server, the system will prompt you to “Press [F8] to select a boot option.” When booting on a serial console, function keys are typically not available. The **m** key may be used to bring up the boot menu instead of the **F8** function key.

```
Loading.: Core LAN Gb A
Running LoadFile()

CLIENT MAC ADDR: 00 30 6E 4C AA A5
CLIENT IP: 10.1.1.100  MASK: 255.255.255.0
DHCP IP: 10.1.1.22  PROXY IP: 10.1.1.22
GATEWAY IP: 10.1.1.1
Press [F8] to Select a boot option: iuxrdp (9)
     Linux Managed
     Next Device (BIOS/EFI)
     HP-UX Managed
```

Linux DHCP PXE Next Server Boot Helper for HP-UX Installation

A Linux boot server, such as an HP Insight Control Environment for Linux (ICE-Linux) server, may be configured to act as an HP-UX boot helper. The Linux server's `dhcpd` *daemon* may be configured to selectively respond to client network requests and to provide a PXE Next Server value that indicates the Ignite server's IP address. This Next Server value will cause boot to switch

to the master Ignite server. Additional HP-UX boot loader and *install content* will be accessed from this Ignite Next Server.

NOTE: DHCP or this PXE boot server is responsible for providing complete network configuration for boot and installation for the client system, including gateway, etc.

The `dhcp.conf` file controls `dhcpd` daemon operation. The `next-server` value is typically associated with specific MAC addresses. For example:

```
allow booting;
allow bootp;
ddns-update-style none;
subnet 10.1.1.0 netmask 255.255.255.0 {
     option routers 10.1.1.1;
     option subnet-mask 255.255.255.0;

     option domain-name "xyzco.com";
     option broadcast-address 10.1.1.255;

     option domain-name-servers 10.1.1.11;

     group {
          host linuxsys2 {
               hardware ethernet 00:0C:29:A1:E9:E5;
               fixed-address 10.1.1.221;
               next-server 10.1.1.11;
               filename "/opt/ignite/boot/nbp.efi";
          }
     }
}
```

Linux server DHCP and PXE boot configuration details for your Linux distribution need to be consulted for correct set up.

Note that the HP-UX network boot loader needs to be copied to the Linux system. This file may be obtained from the Ignite-UX server at `/opt/ignite/boot/nbp.efi`.

This file should be located on the Linux system at `/opt/ignite/boot/nbp.efi`. This file needs to be accessible via TFTP.

Configuring an HP-UX Server to Support Linux Boot and Installation

By using HP-UX network services configuration files, an HP-UX server can be set up to support Linux network boot and installation if you place Linux boot and install content on the HP-UX server. You must acquire the boot and install content from a Linux distribution or Open Source website. Ignite-UX software does not include any Linux support. The information in this section describes use of HP-UX features not directly related to Ignite-UX.

It is not possible to provide all the details for setting up Linux installations here, due to differences between Linux releases. Documentation for network installation for the specific Linux releases being enabled must be consulted for the correct setup.

To enable network boot for Linux installation, proper *boot content* must be placed on the HP-UX server. The elilo boot loader may be obtained from a Linux distribution or the Open Source elilo website at http://sourceforge.net/projects/elilo/. The `IA64` binary is needed for HP-UX Integrity systems.

Linux install kernels and install file system images must be obtained from the appropriate Linux distributions.

The message file `hplinux.msg` controls the appearance of the Linux boot menu. This file must be created with Linux tools.

The boot content may be placed anywhere on the HP-UX system, provided TFTP access is enabled. The location must be consistent with `/etc/bootptab` content. In this example, the Linux boot content is located in the same directory as the HP-UX Ignite boot content.

```
/opt/ignite/boot/LINUX/elilo-ia64.conf
/opt/ignite/boot/LINUX/elilo-ia64.efi
/opt/ignite/boot/LINUX/hpuxlinux.msg
/opt/ignite/boot/LINUX/images/SLES9/initrd
/opt/ignite/boot/LINUX/images/SLES9/Linux
/opt/ignite/boot/LINUX/images/RHEL5UP1
/opt/ignite/boot/LINUX/images/RHEL5UP1/initrd-rhel5.1.img
/opt/ignite/boot/LINUX/images/RHEL5UP1/vmlinuz-rhel5.1
```

The `elilo-ia64.conf` configuration file specifies the set of Linux installation menu options that are provided during network boot. In this example, SuSE 9 and RedHat 5 Update 1 are enabled:

```
chooser=textmenu
message=hplinux.msg
prompt

image=images/SLES9/Linux
     label=SLES9
     description="Install SLES9"
     initrd=images/SLES9/initrd
     append="install=nfs://10.1.1.11/ISOimages/SLES9"
     read-only

image=images/RHEL5UP1/vmlinuz-rhel5.1
     label=RHEL5UP1
     description="Install RHEL5 Update 1"
     initrd=images/RHEL5UP1/initrd-rhel5.1.img
     read-only
```

The Linux distribution content must be made available via NFS or FTP. In this example, the same HP-UX server is used for both boot and installation. Alternatively, Linux install content may be located on another server.

SuSE Linux expects distribution install content to be unpacked as if the distribution media were mounted, and the media contents copied via the *file system* and made available for NFS or FTP access. RedHat expects ISO images to be available as if the distribution media were downloaded or copied to the server from media via *dd*(1) and then made available for NFS or FTP access. For example:

RedHat:

```
/ISOimages/RHEL5UP1
/ISOimages/RHEL5UP1/RHEL5.1-Server-20071017.0-ia64-disc1-ftp.iso
/ISOimages/RHEL5UP1/RHEL5.1-Server-20071017.0-ia64-disc2-ftp.iso
/ISOImages/RHEL5UP1/RHEL5.1-Server-20071017.0-ia64-disc3.ftp-iso
/ISOImages/RHEL5UP1/RHEL5.1-Server-20071017.0-ia64-disc4.ftp-iso
/ISOImages/RHEL5UP1/RHEL5.1-Server-20071017.0-ia64-disc5.ftp-iso
/ISOImages/RHEL5UP1/RHEL5.1-Server-20071017.0-ia64-disc6.ftp-iso
```

SuSE:

```
/ISOimages/SLES9
/ISOimages/SLES9/ARCHIVES.gz
/ISOimages/SLES9/autorun.inf
/ISOimages/SLES9/boot
/ISOimages/SLES9/boot/directory.yast
/ISOimages/SLES9/boot/image
/ISOimages/SLES9/boot/initdisk.gz
/ISOimages/SLES9/boot/rescue
/ISOimages/SLES9/boot/rescuefloppy
/ISOimages/SLES9/boot/root
/ISOimages/SLES9/boot/ChangeLog
/ISOimages/SLES9/boot/content
```

```
/ISOimages/SLES9/boot/control.xml
.
.
.
```

The information presented here gives some general guidance on Linux installation set up. Specific Linux release information must be consulted for getting boot and install content, setting up content on an installation server, and required options in the *EFI* boot configuration file.

The `/etc/bootptab` content to enable Linux boot for installation might look like the following.

```
linuxsys2:\
     ht:=ether:\
     ha=001560045A4A:\
     ip=10.1.1.221:\
     hn:\
     sm=255.255.255.0:\
     gw=10.1.1.1:\
     nt=10.1.1.11:\
     ds=10.1.1.11:\
     bf=/opt/ignite/boot/LINUX/elilo-ia64.efi:\
     bp=10.1.1.11
```

Once all the content has been set up, the HP-UX server should have Linux boot and installation enabled.

Use EFI network boot to start the boot and installation process.

Installation options specified in the `elilo-ia64.conf` file will be displayed as formatted by the `hplinux.msg` file.

IMPORTANT: If the message file is missing or incorrect, the system might display a blank screen when elilo boots. If this happens, you should be able to use **CTRL-c** to exit the boot loader and return to the EFI boot menu.

Select the desired distribution for boot and installation.

```
/-------------------| Linux Install from an HP-UX Server |-------------------\
|                                                                            |
| This image enables Linux installation from an HP-UX system.  Select a      |
| menu option, enter kernel options at the prompt if desired, and press      |
| <return>.  USE AT YOUR OWN RISK!  ^C to go back to EFI; ESC does nothing.   |
|                                                                            |
|            /---------------------------------------------\                 |
|            | Install SLES9                                 |                 |
|            | Install RHEL5 Update 1                        |                 |
|            |                                               |                 |
|            |                                               |                 |
|            |                                               |                 |
|            |                                               |                 |
|            |                                               |                 |
|            |                                               |                 |
|            \---------------------------------------------/                 |
|                                                                            |
| Kernel Options:  ____________________________________________________      |
|                                                                            |
|    ^C now to go back to EFI; ESC does nothing. Once started, there's no    |
|    turning back (i.e. you have to reboot to get back to EFI).              |
\----------------------------------------------------------------------------/
```

RedHat Installation From an HP-UX Server

The RedHat installation process normally starts with language selection.

```
Welcome to Red Hat Enterprise Linux Server
```

```
                    +----------+ Choose a Language +----------+
                    |                                         |
                    | What language would you like to use     |
                    | during the installation process?        |
                    |                                         |
                    |       Catalan                  ^        |
                    |       Chinese (Simplified)     :        |
                    |       Chinese (Traditional)    #        |
                    |       Croation                 :        |
                    |       Czech                    :        |
                    |       Danish                   :        |
                    |       Dutch                    :        |
                    |       English                  v        |
                    |                                         |
                    |                 +----+                  |
                    |                 | OK |                  |
                    |                 +----+                  |
                    |                                         |
                    +-----------------------------------------+
```

```
  <Tab>/<Alt-Tab> between elements | <Space> selects | <F12> next screen
```

Next, the source and method of transport must be selected. In this example, NFS network installation is used.

```
Welcome to Red Hat Enterprise Linux Server

                  +------+ Installation Method +------
                  |                                   |
                  | What type of media contains the   |
                  | packages to be installed?         |
                  |                                   |
                  |            Local CDROM            |
                  |            Hard drive             |
                  |            NFS image              |
                  |            FTP                    |
                  |            HTTP                   |
                  |                                   |
                  |     +----+          +------+      |
                  |     | OK |          | Back |      |
                  |     +----+          +------+      |
                  |                                   |
                  |                                   |
                  +-----------------------------------|

  <Tab>/<Alt-Tab> between elements | <Space> selects | <F12> next screen
```

Then you must specify the location of Linux install content.

```
Welcome to Red Hat Enterprise Linux Server

    +----------------------------+ NFS Setup +----------------------------+
    |                                                                      |
    | Please enter the following information:                              |
    |                                                                      |
    |     o the name or IP number of your NFS server                       |
    |     o the directory on that server containing                        |
    |       Red Hat Enterprise Linux Server for your architecture          |
    |                                                                      |
    | NFS server name:                              10.1.1.11_____________ |
    | Red Hat Enterprise Linux Server directory: /ISOimages/RHEL5UP1_____  |
    |                                                                      |
    |             +----+                          +------+                 |
    |             | OK |                          | Back |                 |
    |             +----+                          +------+                 |
    |                                                                      |
    |                                                                      |
    +----------------------------------------------------------------------+


 <Tab>/<Alt-Tab> / between elements  |  <Space> selects |  <F12> next screen
```

RedHat installation continues presenting additional configuration screens specific to the distribution. Documentation regarding RedHat installation and configuration for client system set up should be consulted.

SuSE Installation From an HP-UX Server

The SuSE installation process normally starts with terminal selection.

```
What type of terminal do you have ?

  1) VT100
  2) VT102
  3) VT220
  4) X Terminal Emulator (xterm)
  5) X Terminal Emulator (xterm-vt220)
  6) X Terminal Emulator (xterm-sun)
  7) screen session
  8) Linux VGA or Framebuffer Console
  9) Other

Type the number of your choice and press Return:
```

The installation continues to present additional configuration screens specific to the distribution. In this case, the location of install content was specified in the elilo configuration file. Documentation regarding SuSE installation and configuration for client system set up should be consulted.

Configuring an HP-UX Server to Support Windows Installation

In theory, it might be possible to configure an HP-UX server to enable Windows installation. However, Windows installation from an HP-UX server has not been investigated. One of the possible challenges for enabling installation is to provide Boot Information Negotiation Layer (BINL) from the server. A solution for Linux servers that might be adaptable is provided via Windows Remote Installation Service (RIS).

7 Managing I/O for Installation and Recovery

This chapter introduces Ignite-UX I/O concepts and describes how multi-path concepts enhance Ignite-UX. This chapter also assists the user moving from the legacy naming model to the agile naming model by touching on changes found in multi-path aware Ignite.

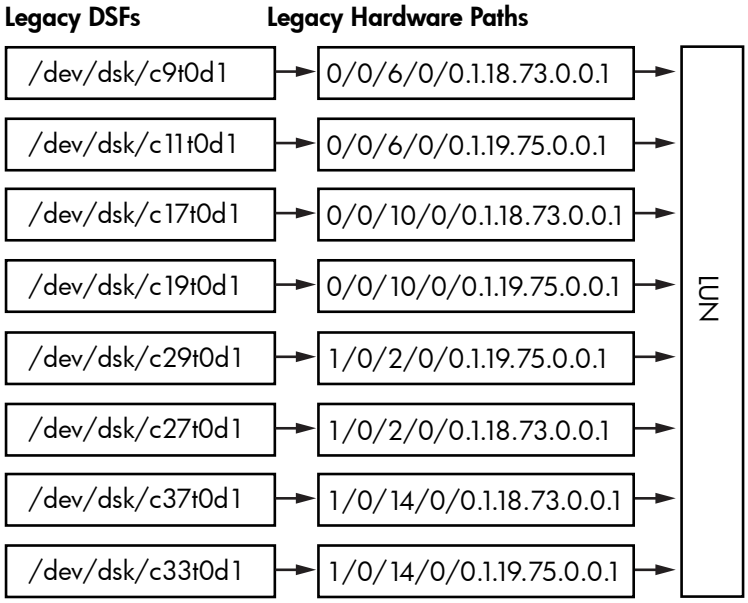
Introducing Multipathing

In its current implementation, beginning with C.7.1.*x*, Ignite-UX is aware of multiple paths to I/O devices. Ignite-UX now supports *agile view* on HP-UX 11i v3.

One of the features of HP-UX 11i v3 and later is the ability to tolerate I/O path changes. For HP-UX 11i v2 and earlier releases of HP-UX, Ignite-UX is aware that multiple device special files (DSFs) and hardware paths may refer to the same device logical unit (LUN).

Previous to HP-UX 11i v3, I/O addressing looks like Figure 7-1, where a *DSF* is specific to one hardware path, which in turn points to an I/O device's LUN.

Figure 7-1 Legacy I/O Stack Addressing Model



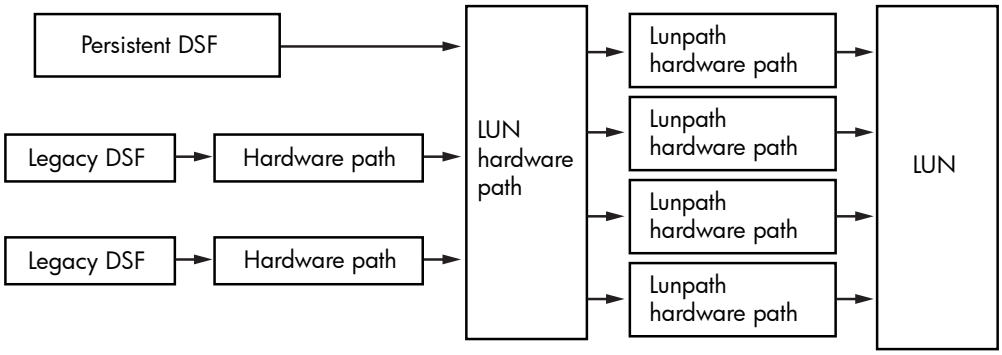
Any of the *legacy DSF*s can be used to access the I/O device. Care must be taken to prevent the simultaneous use of multiple DSFs for conflicting purposes. For example, two DSFs for the same LUN might be used for different volume or *disk group*s. Ignite-UX will detect such an invalid configuration created with the Ignite user interface and prevent *installation*. A final validation is also done during sanity checking, which takes place after starting an installation.

Agile View Concepts

Starting with HP-UX 11i v3, HP-UX is aware of multiple paths to devices and provides *multipathing* functionality automatically. Important new concepts related to this functionality are: *persistent DSF*, *LUN hardware path*, *lunpath hardware path*, *device identifier*, and *agile addressing*.

Agile view I/O addressing logic looks like Figure 7-2.

Figure 7-2 Agile Multiple Path I/O Stack Addressing Model

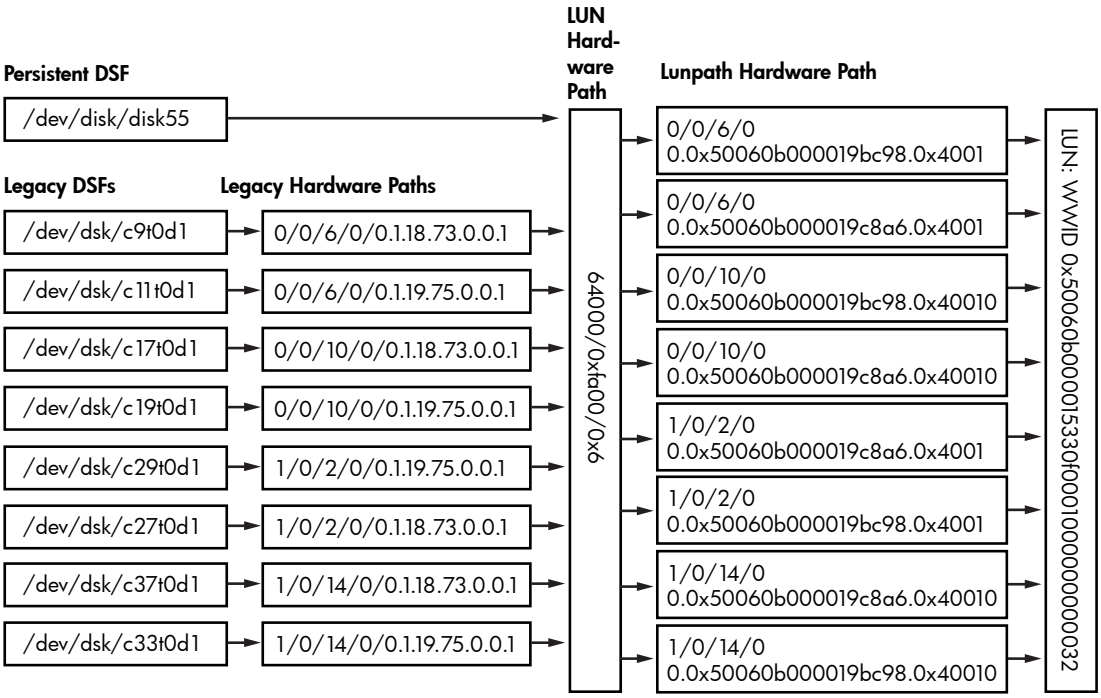


The *persistent DSF* represents the device, regardless of its location in the I/O configuration. With the new model, *legacy DSF*s and hardware paths enjoy multi-path capabilities because after device open, I/Os use every path associated with the I/O device.

The *LUN hardware path* is virtualized, representing all the *lunpath hardware path*s to a device. The lunpath hardware path is the path typically displayed in the Ignite user interface. Lunpath hardware paths do not have device special files - they are associated with a LUN hardware path and are accessed via the *persistent DSF* associated with a LUN hardware path. The Ignite user interface displays the lunpath hardware path so the actual device can be discerned from it; you cannot identify the physical device from looking at the LUN hardware path or the persistent DSF.

Sample *agile addressing* model values for the DSFs and paths are shown below.

Figure 7-3 Agile Naming Example



Identification of devices in a multiple path I/O configuration can be challenging. The I/O stack (driver) identifies devices using *unique LUN ID*s. Often this is a WWID value.

By identifying a device using the unique LUN ID, any of its hardware paths could be used to access it, and *agile addressing* is supported.

The unique LUN ID might be difficult for a user to associate with a specific physical or logical device. For example, often the device WWID is on a device label (e.g. on a sticker) or available via storage management software used to set up a virtual LUN – those values are difficult to

remember and type correctly. It might be easier to select a device using one of its hardware paths. However, if you can remember part of the WWID, you can use the Ignite user interface Disk Selection dialog box Filter text box to limit displayed devices to those with WWIDs containing the text you can remember.

Agile addressing means the hardware path actually used by the system to access a device is independent of the path used to select the device. Selecting a disk via one hardware path might result in the system choosing some other, better hardware path. For example, this can happen when selecting a disk for *boot*, and when volume managers determine the appropriate set of paths to use for device access.

NOTE: For HP-UX 11i v3, the one hardware path used for selection has no special significance in most Ignite-UX user interfaces. Ignite-UX will allow HP-UX system software to select the best path when a particular path is needed. For example, boot paths will be selected by system software when the boot device is selected.

A more user-friendly approach is to identify I/O devices with a *device identifier*. A device identifier is a human-readable device ID defined by the user. It can be written to the device and read back. Data centers may want to create some standard policy for device IDs (e.g. LAB2CAB23LUN15).

In the current implementation, the device ID can be set, checked, and read at installation. The device identifier is stored on the device, so it remains available if the disk is moved to a different system or connected to multiple systems. Not all devices support the use of a device identifier.

See the *scsimgr*(1M) command for more information on how to set and read a device identifier.

When identifying I/O hardware for Ignite-UX *configuration file*s, see Table 7-3 (page 92) for the format of I/O variables.

NOTE: Your data center may use separate processes or groups to administer systems and storage. It is important to record WWID, Device ID, and other details of LUNs assigned to your systems. Access control or protection zones may be used to control the systems permitted to use a LUN; it is important to record which systems have access to LUNs.

For more details on *agile view* HP-UX, see the white paper *Overview: The Next Generation Mass Storage Stack* available at **http://www.hp.com/go/hpux-core-docs**.

Practical Considerations

Ignite-UX uses hardware paths to help you identify I/O devices such as disks, CD/DVD drives and tape drives during *installation* and *recovery*. The format of the hardware path used to identify I/O devices will depend on the version of HP-UX you are using and other factors. Also, depending on your configuration, multiple paths might be displayed for a single device.

For HP-UX 11i v3, Ignite-UX will allow HP-UX system software to select the best path when a particular path is needed. For example, *boot* paths will be selected by system software when the boot device is selected.

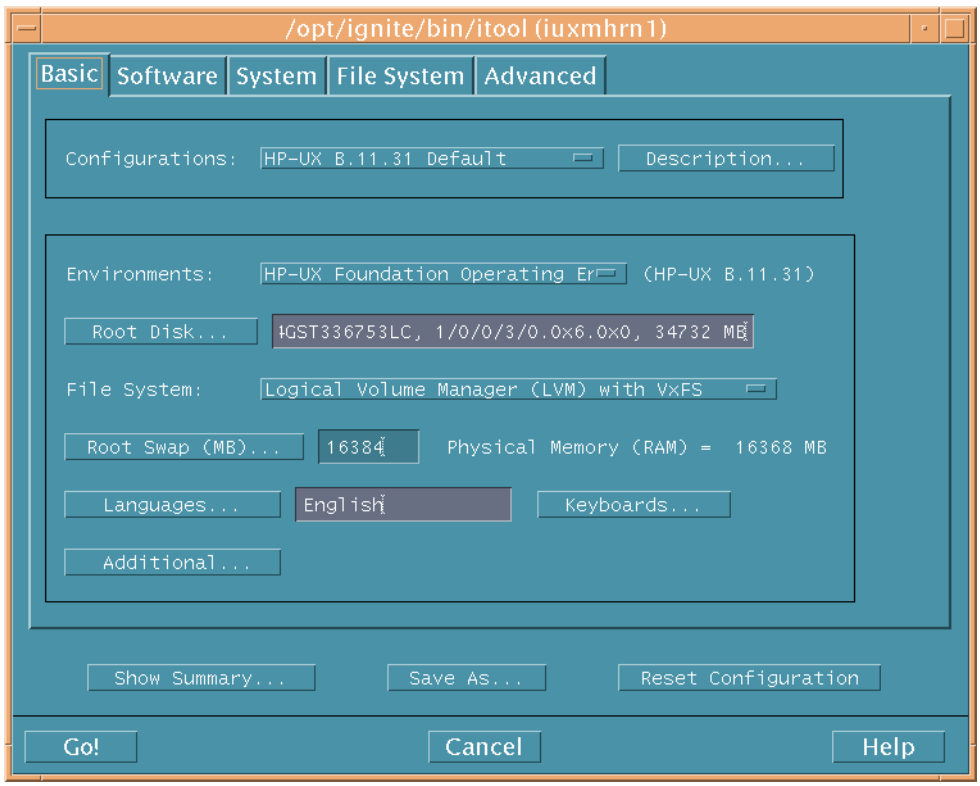
Note that horizontal scrolling might be required to read the entire hardware path and associated data in Ignite-UX GUI screens.

IMPORTANT: Due to application dependencies, HP-UX software deployment tools such as Ignite-UX expect legacy DSFs to be present and the legacy naming model to be enabled. Therefore, HP recommends only partial migration be performed, as detailed in the *HP-UX 11i v3 Persistent DSF Migration Guide*, available at http://www.hp.com/go/hpux-core-docs.

System Installation Configuration

When using Ignite to install HP-UX on a client, a root disk must be identified. Ignite-UX selects a default disk, but it may be changed using the **Root Disk...** button on the Basic tab on the client installation configuration interface shown below.

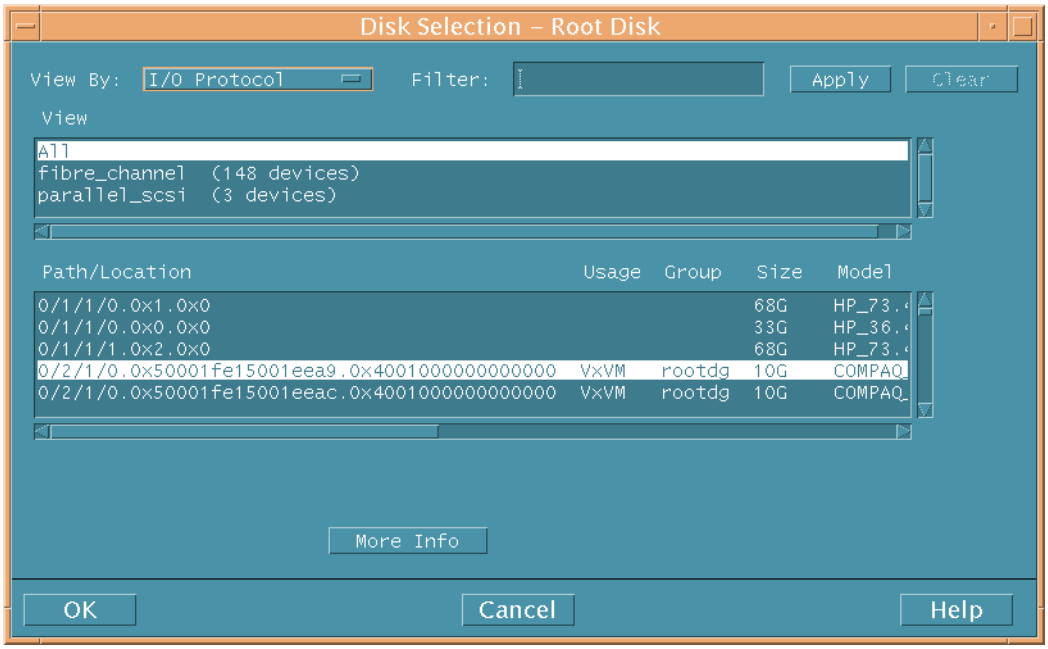
Figure 7-4 Ignite-UX Client Installation Configuration Tabs



The hardware path displayed in the Root Disk text box is the *lunpath hardware path* for HP-UX 11i v3 and later, or the *legacy hardware path* for HP-UX 11i v2 and earlier.

By selecting **Root Disk...**, the Disk Selection – Root Disk dialog box is displayed (Figure 7-5 (page 85)).

Figure 7-5 Disk Selection – Root Disk Dialog Box

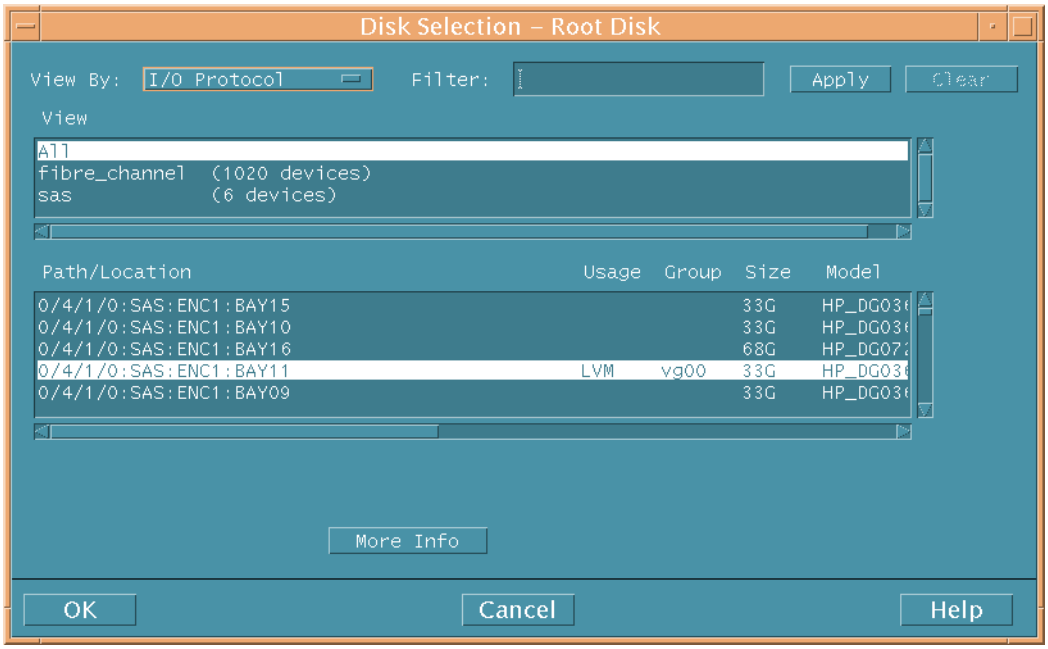


The Disk Selection – Root Disk dialog box displays every path for each disk, therefore disks with multiple paths are listed multiple times. Set View By: to **Disks/Paths** (the default is **I/O Protocol**) to see what paths go to each disk. For more information on using the Disk Selection – Root Disk dialog box, see "Root disk... Button" (page 136).

NOTE: Inventory blocking may be used to reduce the time required to discover devices during Ignite-UX start-up. Devices blocked from inventory will not be listed in the Disk Selection dialog boxes. For more information on using Ignite-UX variables such as *inventory_block_path* and *inventory_block_protocols*, see *instl_adm*(4), "Controlling the I/O Configuration Process" (page 92), and "Additional... Button" (page 140). For more information on the agile naming model, see Figure 7-3 (page 82).

If you have a SAS device, *physical location*s will be displayed in the Disk Selection – Root Disk dialog box as shown in Figure 7-6 (page 86).

Figure 7-6 Disk Selection – Root Disk Dialog Box With Physical Locations



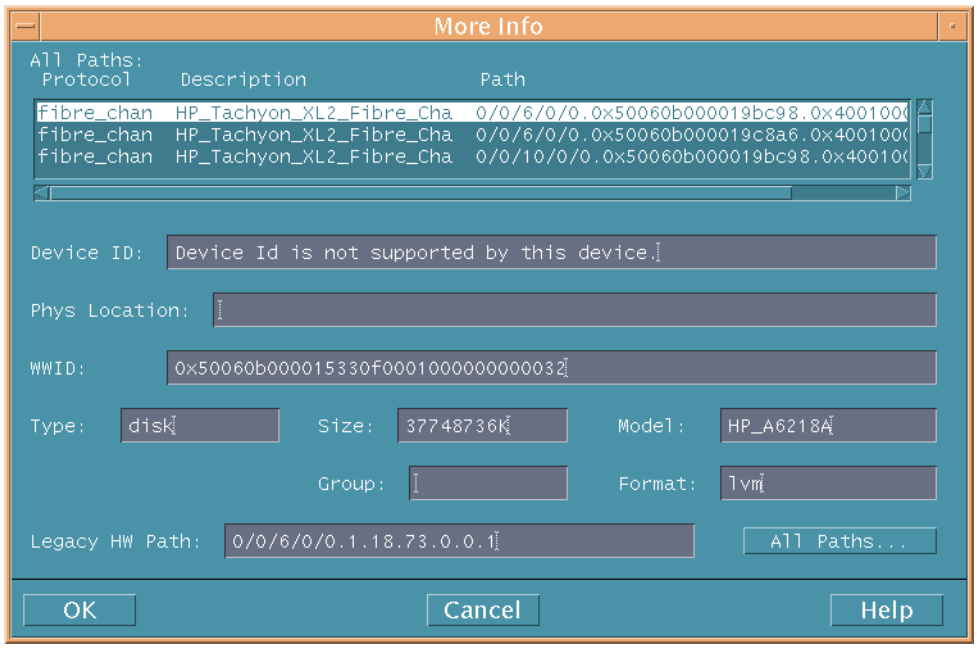
To get a concise listing of all the paths for a single device, select the path of interest and then click the **More Info** button, or use the Disks/Paths View. If a LUN has multiple paths, you may select any of them to get to the same More Info screen. For example, you may select either of the *lunpath hardware path*s below since they both reference the same LUN.

`0/0/6/0/0.0x50060b000019c8a6.0x4001000000000000`

`0/0/10/0/0.0x50060b000019c8a6.0x4001000000000000`

Use the More Info screen to help make the transition from *legacy hardware path*s to the *lunpath hardware path*s. The More Info screen supplies details about a LUN to help you verify that the selected LUN is the one intended. Note that the legacy hardware path is also shown to the very right in the Path/Location window of the Disk Selection dialog box – use horizontal scrolling to see it.

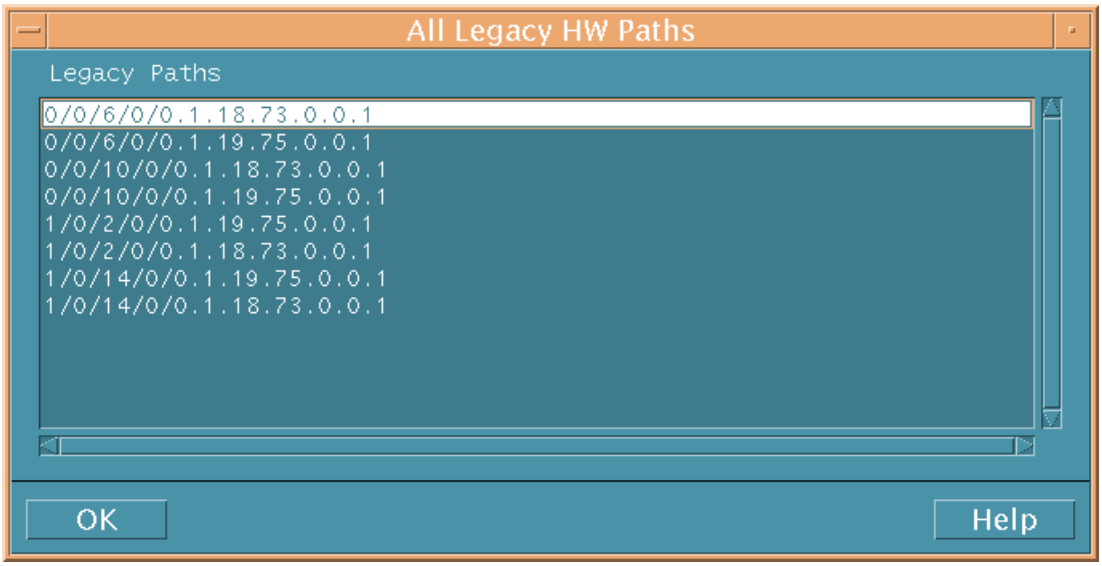
Figure 7-7 More Info Dialog Box



For HP-UX 11i v3 and later, the More Info screen displays all the *lunpath hardware path*s for a device. (Note that the paths can be long - horizontal scrolling may be needed to see the entire path.) The Legacy HW Path displayed depends on the lunpath hardware path currently selected in the selection list window. One *legacy hardware path* is listed. For a concise list of all the legacy hardware paths leading to the device, select the **All Paths...** button (Figure 7-8 (page 87)).

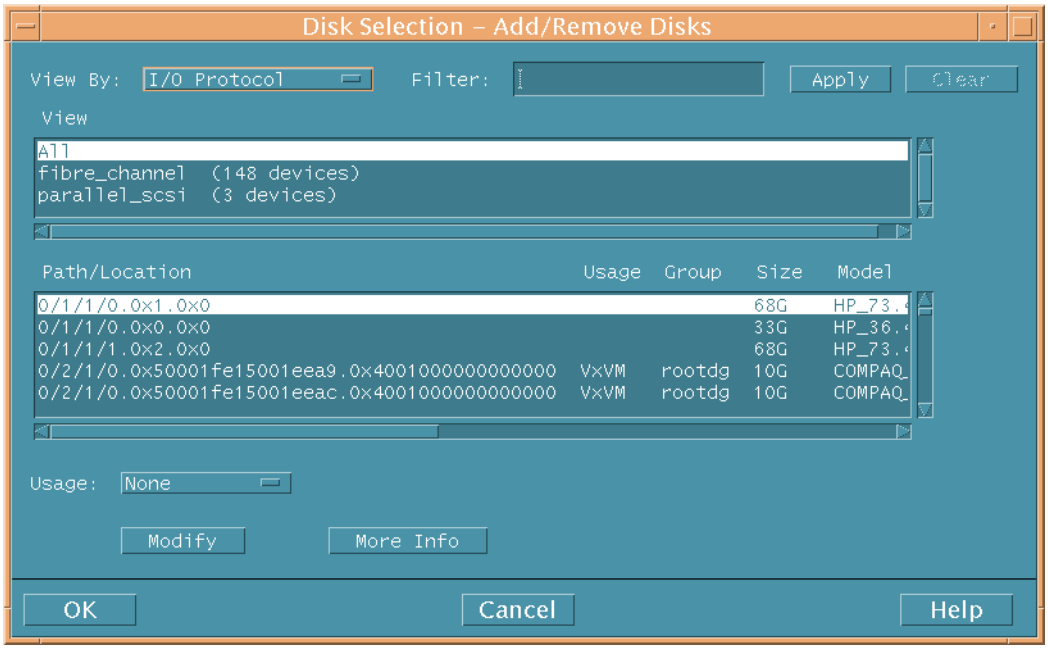
The Legacy HW Path window and the **All Paths...** button are not available on systems running HP-UX 11i v2 and earlier.

Figure 7-8 All Paths Dialog Box



Accessed from the File System tab of the client installation configuration interface, the Disk Selection – Add/Remove Disks dialog box displays the *lunpath hardware path*, the *legacy hardware path*, or the *physical location*. See Figure 7-6 (page 86) for the display format of physical location hardware paths.

Figure 7-9 Disk Selection – Add/Remove Disks Dialog Box



The More Info screen is made available on the Disk Selection – Add/Remove Disks dialog box too, to validate selections and help make the change to the agile naming model.

Identifying Devices for Other Tasks

There are a number of other Ignite tasks that require the identification of I/O devices:

- When building *installation media*, you must use the `ioscan` command to find the tape, CD, or DVD drive. Note that some of the CD/DVD writing SW included with Ignite-UX but not developed by HP only works with legacy device special files.
- When performing a system recovery on Itanium-based systems, you must find the tape device's hardware path to create an *EFI* boot option. (This should be done at the time the tape is created.)
- During *two-step media recovery*, you must select the hardware path of the tape drive to recover from.
- When in *expert recovery* mode, you must have the hardware path of the disk you are attempting to recover.
- When identifying disks in *configuration file*s to define them or combine them in a single volume group, you must supply the hardware paths of the disks.

Important Characteristics of the Agile View

The following points will help you move from the *legacy view* to the *agile view*:

- When using the new *persistent DSF*s, be sure to use the new directory structure:

Table 7-1 Mass Storage DSF Directories

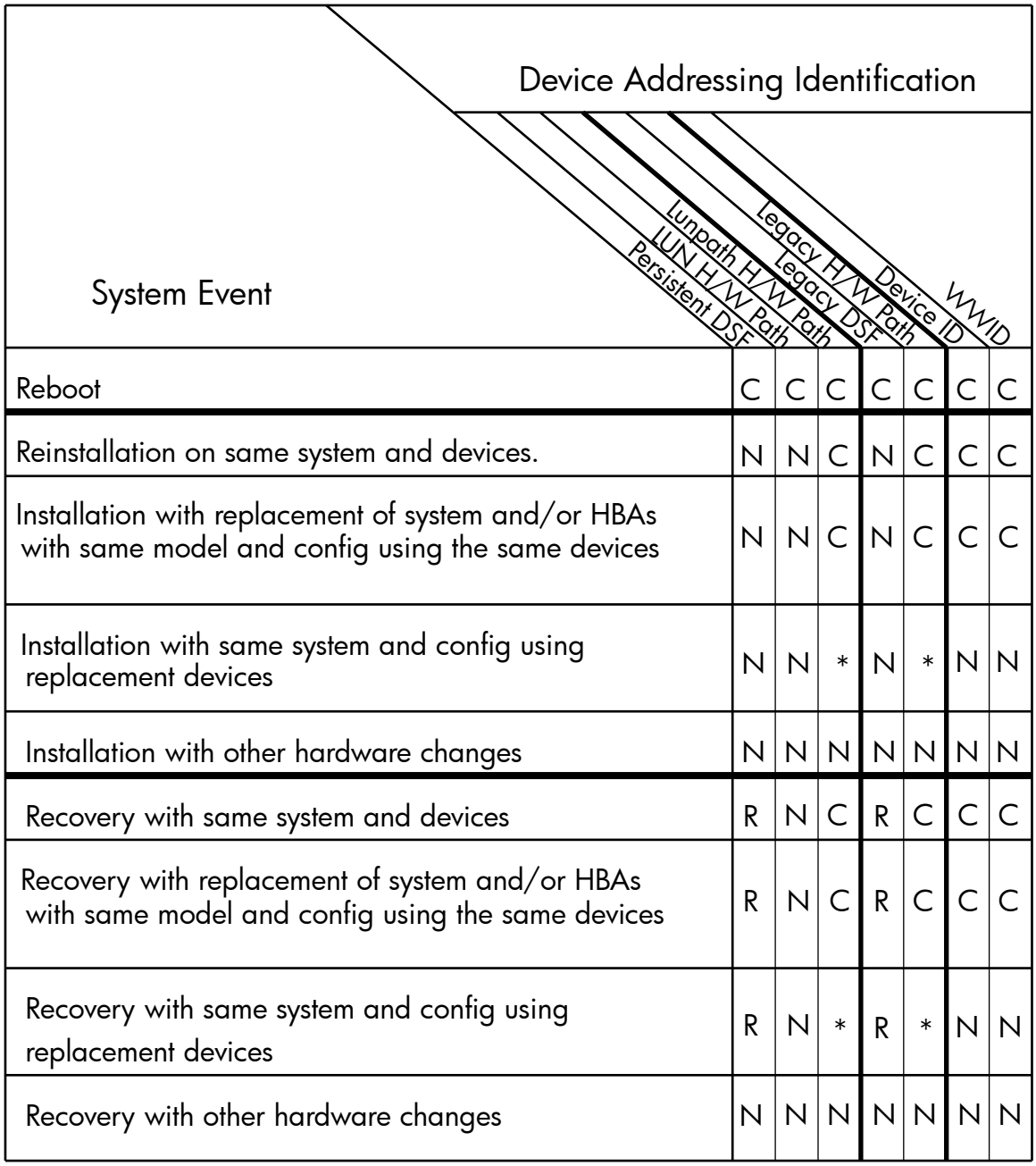
Persistent DSF Directory	Legacy DSF Directory
/dev/disk	/dev/dsk
/dev/rdisk	/dev/rdsk
/dev/rtape	/dev/rmt

- Both legacy and persistent DSFs can be created during installation and recovery of HP-UX 11i v3 depending on your volume manager. When Ignite-UX creates *file system* content for HP-UX 11i v3, persistent DSFs are used for LVM volumes and VxVM 5.0 layout, and *legacy DSF*s are used for VxVM 4.1.

NOTE: The VxVM components in the Ignite-UX *install environment and* the installation *depot* must be version 5.0 in order to use *persistent DSF*s.

- Keep in mind that persistent DSFs endure with recovery if Ignite recovers to the same hardware or can map to replacement hardware, but are not guaranteed to remain the same between installations. See Figure 7-10 for an overview regarding consistency of I/O addressing.
- If hardware is replaced and recovery is not needed, you might want to run `io_redirect_dsf` to correct the system configuration. For more information, see *io_redirect_dsf*(1M).

Figure 7-10 Consistency of I/O Addressing



System Event	Persistent DSF	LUN H/W Path	lunpath H/W Path	Legacy DSF	Legacy H/W Path	Device ID	WWID
Reboot	C	C	C	C	C	C	C
Reinstallation on same system and devices.	N	N	C	N	C	C	C
Installation with replacement of system and/or HBAs with same model and config using the same devices	N	N	C	N	C	C	C
Installation with same system and config using replacement devices	N	N	*	N	*	N	N
Installation with other hardware changes	N	N	N	N	N	N	N
Recovery with same system and devices	R	N	C	R	C	C	C
Recovery with replacement of system and/or HBAs with same model and config using the same devices	R	N	C	R	C	C	C
Recovery with same system and config using replacement devices	R	N	*	R	*	N	N
Recovery with other hardware changes	N	N	N	N	N	N	N

Use the following key for the "Consistency of I/O Addressing" figure above.

C = Consistent — Device addressing id value is saved in HP-UX config content and the correlation between these values and devices does not change as a result of the event.

N = New Value — Device addressing id value is newly created as a result of the event. The device addressing id value for a device might or might not change as a result of the event.

R = Recovered — Ignite-UX will restore the association between device addressing id values and devices when possible. The process used to match previous and current devices is handled by Ignite-UX recovery matching methods. Matching might not be possible if the system configs differ.

* — Might be a consistent value for some I/O protocols such as parallel SCSI, and might be a new value for some I/O protocols such as fibre channel.

Recovery and the Agile View

During recovery, Ignite-UX C.7.x makes changes to the new system I/O configuration to match the original system I/O configuration. This is necessary because some aspects of a system configuration depend on the unpredictable order of system I/O inventory.

The overall goal of this process is to make the system I/O configuration appear as if the system simply rebooted at the time the *recovery archive* was created. This process is complex, and Ignite-UX might be unable to fully restore the I/O configuration. Ignite might not be able to restore aspects of the I/O configuration due to hardware changes, limitations of system I/O software, and limitations of Ignite-UX.

The system I/O configuration should be verified during and after recovery so configuration adjustments can be made if needed.

One part of restoring the I/O configuration is the appropriate matching of device special files (DSFs) and devices. There is one approach used for *legacy DSF*s in HP-UX 11i v3 and previous releases, and another approach used for HP-UX 11i v3 *persistent DSF*s.

Legacy DSFs and Device Matching

Matching *legacy DSF*s and mass storage devices is done based on hardware paths. Generally, legacy DSFs are associated with a particular hardware path. During recovery, a device is associated with its hardware path's *DSF*. (See Figure 7-1 (page 81) for a description of the legacy addressing model.)

Hardware configuration changes are handled by assuming a new device is intended to replace the device originally at that hardware path.

Note that some I/O protocols, such as SAS and USB, associate legacy DSFs with specific devices using *unique LUN ID*s, and so behave like *persistent DSF* matching described below.

SAS devices are a special case, since their legacy DSF/unique LUN ID association can change as a result of I/O configuration changes. If you change a SAS configuration (physically move a SAS device to another bay or remove a SAS device) the hardware path associated with that and other SAS devices can change during an installation or recovery. In such a case, hardware paths are reassigned to SAS devices. Since legacy DSFs are associated with a particular hardware path, a change in a device's hardware path breaks the association between its previous legacy DSF and its unique LUN ID. Note that the way SAS devices are associated during recovery might change in future versions of Ignite-UX to use the *agile addressing* approach described below.

Only certain SAS configuration changes cause remapping of hardware paths. For more information see the white paper, “Ignite-UX and SAS Devices” available at http://www.hp.com/go/ignite-ux-docs.

Persistent DSFs and Device Matching

Matching *persistent DSF*s and mass storage devices is relatively complex due to *agile addressing*. Ignite-UX will attempt to simulate agile addressing during recovery, while also dealing with hardware replacement. This matching is accomplished using the methods described below:

- **WWID —** Matching is done based on the *unique LUN ID* of the device. Most often, this is the device's WWID. This method matches a device's original *persistent DSF* with the same device in the recovered system configuration.
- **Device ID —** (Future) Matching is done based on a user-definable identifier written to the device. This method matches a device's original *persistent DSF* with the device that has the same device ID in the recovered system configuration. This method allows user-provided identification to control device matching. Note that some mass storage devices do not support the device ID feature.
- **Physical Location —** Matching is done based on device *physical location*. This method matches the original *persistent DSF* associated with a particular physical location (e.g. same enclosure

and bay) with the device at that location in the recovered system configuration. This method is intended to handle replacement of devices. Note that not all I/O protocols support physical location addressing.

- **Lunpath —** Matching is done based on *lunpath hardware path*. This method matches the original *persistent DSF* associated with a lunpath hardware path with the device at that lunpath hardware path in the recovered system configuration. This method is intended to handle replacement of devices. Some protocols such as fibre channel have lunpath hardware paths and *legacy hardware path*s that are functionally different (use different hardware attributes).
- **Legacy hardware path —** Matching is done based on the *legacy hardware path*. This method matches the original *persistent DSF* associated with a legacy hardware path with the device at that legacy hardware path in the recovered system. This method matches devices using the same approach used for typical *legacy DSF*s.

Not all methods are appropriate for all protocols. The following are the ordered lists of persistent DSF-to-device matching methods by protocol. The order in which these methods are applied is important. Matching will be done in the order listed.

Table 7-2 Persistent DSF-to-Device Matching Methods by Protocol

Protocol	Ordered List
parallel SCSI	WWID, lunpath
fibre channel	WWID, *physical location*, lunpath
ide	lunpath
SAS	WWID, *physical location*, lunpath
other	no matching will be done

NOTE: There might be devices in the original system configuration that can not be matched with devices in the new system configuration. There might also be devices in the new system configuration that do not match devices in the original configuration. In these cases, the HP-UX operating system I/O drivers will assign legacy and *persistent DSF*s for the non-matching devices.

Controlling the I/O Configuration Process

It is possible to control the I/O configuration process associated with installation and recovery by using variables in the *configuration file*. These variables allow you to select disks that may be replaced with other disks, hide disks from the I/O configuration process, and control *DSF* naming.

By controlling the I/O configuration process, you can make configuration files that are general enough to use with clients having different hardware paths, protect disks from modification, and increase the performance of the I/O inventory process.

This section introduces the variables and value types associated with I/O configuration for use in configuration files. Further details can be found in *instl_adm*(4).

Table 7-3 I/O Configuration Variables

I/O Configuration Variable	Description
`allow_disk_remap`	(boolean) Setting this to true allows Ignite to substitute a disk that was specified in the *configuration file*s but does not exist on the system with a disk that does exist and was not specified to be used, hidden, or blocked. Default for this value is false for a noninteractive installation, and true for an interactive installation. This is useful when creating configuration files for use with multiple clients.
`hide_boot_disk`	(boolean) Setting this to true prevents the installation process from allowing the boot disk to be configured and/or "cleaned". This is useful only when the Ignite *kernel* is booted from a dedicated hard disk you wish to protect from being modified.

Table 7-3 I/O Configuration Variables *(continued)*

I/O Configuration Variable	Description
`_hp_hide_other_disks`	(string) This may be set to one or more space-separated hardware paths of disks that should be "hidden" from being configured or otherwise modified during the installation. This is useful for hiding multiple disks.
`hw_instance_number`	(string) Setting this keyword forces a specific instance number assignment for the specified hardware device. This is useful for producing client configurations consistent with others regardless of variations in hardware configurations.
`inventory_block_path`	(string) This keyword is used to control Ignite inventory functionality by instructing Ignite to not collect inventory information for the devices specified. This is useful when you want devices hidden and not available for selection during installation.
`inventory_block_protocols`	(string) This keyword is used to control Ignite inventory functionality by instructing Ignite to not collect inventory information for the devices of the protocol type specified. This is useful when you want to increase the performance of the I/O inventory process by ignoring all devices of a certain protocol. These devices will not be available for selection during installation.

Below are listed the value types for use with the I/O configuration variables.

Table 7-4 I/O Configuration Value Types

I/O Configuration Value Type	Description
Hardware Path	For keywords that take a hardware path as an index parameter or value, the hardware path may be a series of more than one decimal or hexadecimal numbers separated by the period (.) or the forward slash (/) characters. A complex string or string variable may also be used where a hardware path is expected.
Physical Location	A *physical location* may be a series of alphanumeric values separated by the colon (:) character.
World-Wide Name / WWID	The format of this value varies depending on the protocol and device. This value is often a standard IEEE value in hexadecimal format, however this value may have some other format. This value may not contain white space.
I/O Protocol	The valid protocol values are: • fibre_channel • parallel_scsi • sas • usb

Agile View Questions and Answers

What are the requirements for the use of persistent DSFs? Must I use them exclusively or can I mix them with legacy DSFs?

Ignite-UX will use *persistent DSF*s for installation and recovery on systems running HP-UX 11i v3. Internally, VxVM volume management software controlling installation and recovery uses persistent or *legacy DSF*s as appropriate for the VxVM version used. See "Important Characteristics of the Agile View" (page 89) for more information.

Can the user switch from persistent to legacy DSFs and back if desired?

Switching between persistent and *legacy DSF*s is specific to the volume manager. See your volume manager documentation for more details.

Does the Ignite–UX interface enforce a particular use model with respect to persistent and legacy DSFs?

No, but *persistent DSF*s will be used for HP-UX 11i v3 installation and recovery.

If a persistent DSF is specified, is the equivalent legacy DSF added as well? And vice versa?

Ignite-UX will use *persistent DSF*s for installation and recovery. VxVM installation support software will create VxVM volumes using appropriate DSFs for the VxVM version used. See "Important Characteristics of the Agile View" (page 89) for more information.

8 Security

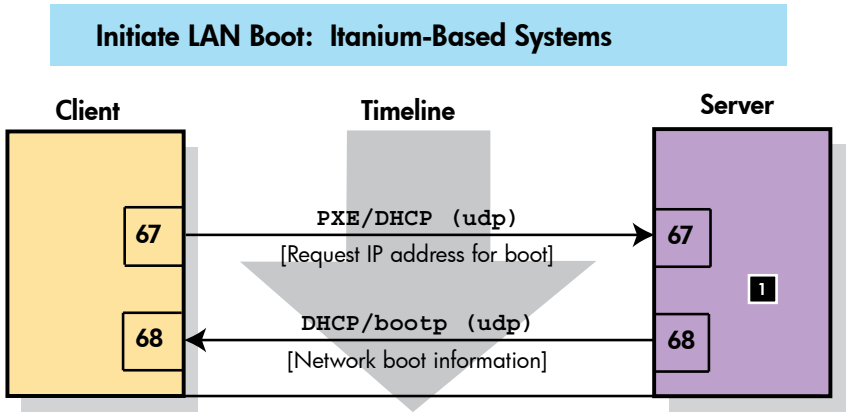
The purpose of this chapter is to assist system and network administrators in understanding the network ports and protocols used by Ignite-UX during its various phases of operation, and to assist in configuring the HP-UX *Bastille* and IPFilter products. HP is not able to provide support for configuring third-party firewalls to work with Ignite-UX.

Ignite-UX Server Ports

Ignite-UX server network port usage is described below by server activity. Diagrams follow that describe the port activity by network communication task and timing. See the product documentation to get the protocol for your system when the protocol is listed as tcp/udp.

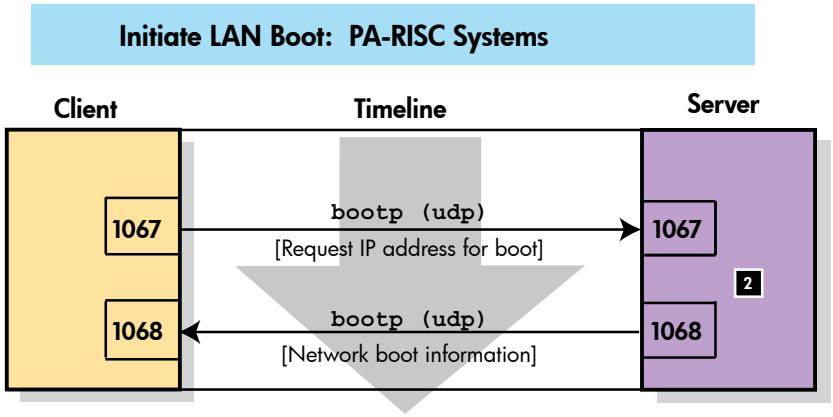
- Initiate LAN Boot for Itanium-Based clients, Figure 8-1 (page 96): ports 67 and 68.
- Initiate LAN Boot for PA-RISC clients, Figure 8-2 (page 96): ports 1067, 1068.
- Cold boot and installation initiated from client, Figure 8-3 (page 97): 69, 2049, 2121, an SD dynamically allocated port.
- Live system reinstall via `bootsys` initiated from the server, Figure 8-4 (page 98): 2049, 69, 2121, an SD dynamically allocated port, and 514 or 22.
- `make_net_recovery` initiated from client, Figure 8-5 (page 99): 69, 2121, an SD dynamically allocated port, 2049.
- `make_net_recovery` initiated from the server, Figure 8-6 (page 100): 69, 2121, an SD dynamically allocated port, 2049, and 514 or 22.
- `make_sys_image` initiated from client, Figure 8-7 (page 100): 514 or 2049.

Figure 8-1 Port Usage: Initiate LAN Boot for Itanium-Based Clients



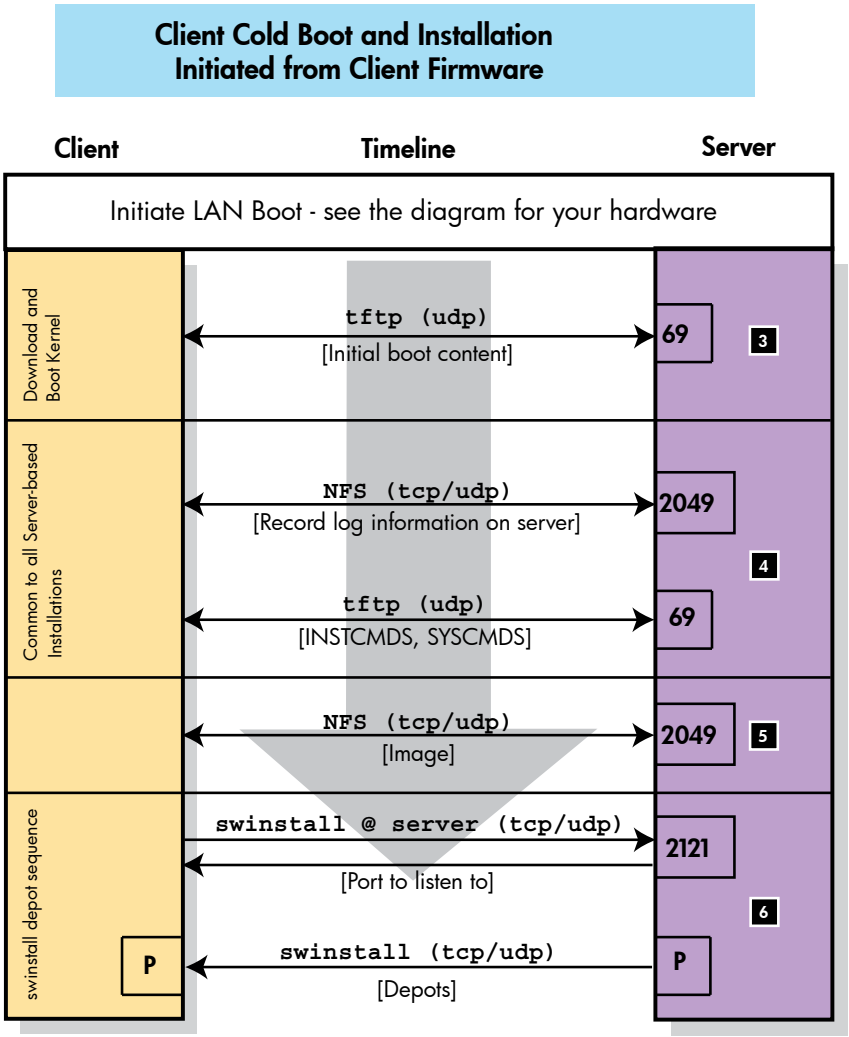
1. The client sends a *boot* request to the server over port 67. The request is handled by the `bootpd` *daemon* on the server. If the client is registered, the `/etc/bootptab` file is referenced for the boot IP address; if the client is anonymous, *DHCP* services are used to assign the boot IP address. The server then sends the networking information to the client on port 68. For more information on booting registered Itanium-based clients, see "Configuring the Ignite-UX Server for Itanium-Based Clients" (page 41). For more information on booting anonymous Itanium-based clients, see "Configuring an Ignite Server to Boot Anonymous Itanium-Based Clients" (page 50). For more information on `bootpd`, see *bootpd*(1M).

Figure 8-2 Port Usage: Initiate LAN Boot for PA-RISC Clients



2. The client sends a *boot* request to the server over port 1067. The request is handled by the `instl_bootd` *daemon* on the server. The `/etc/opt/ignite/instl_boottab` file is referenced whether the client is registered or anonymous. The server then sends the networking information to the client on port 1068. For more information on booting registered PA-RISC clients, see "Configuring the Ignite-UX Server for PA-RISC Clients" (page 37). For more information on booting anonymous PA-RISC clients, see "Configuring an Ignite Server to Boot Anonymous PA-RISC Clients" (page 49). For more information on `instl_bootd`, see *instl_bootd*(1M).

Figure 8-3 Port Usage: Client Cold Boot and Installation

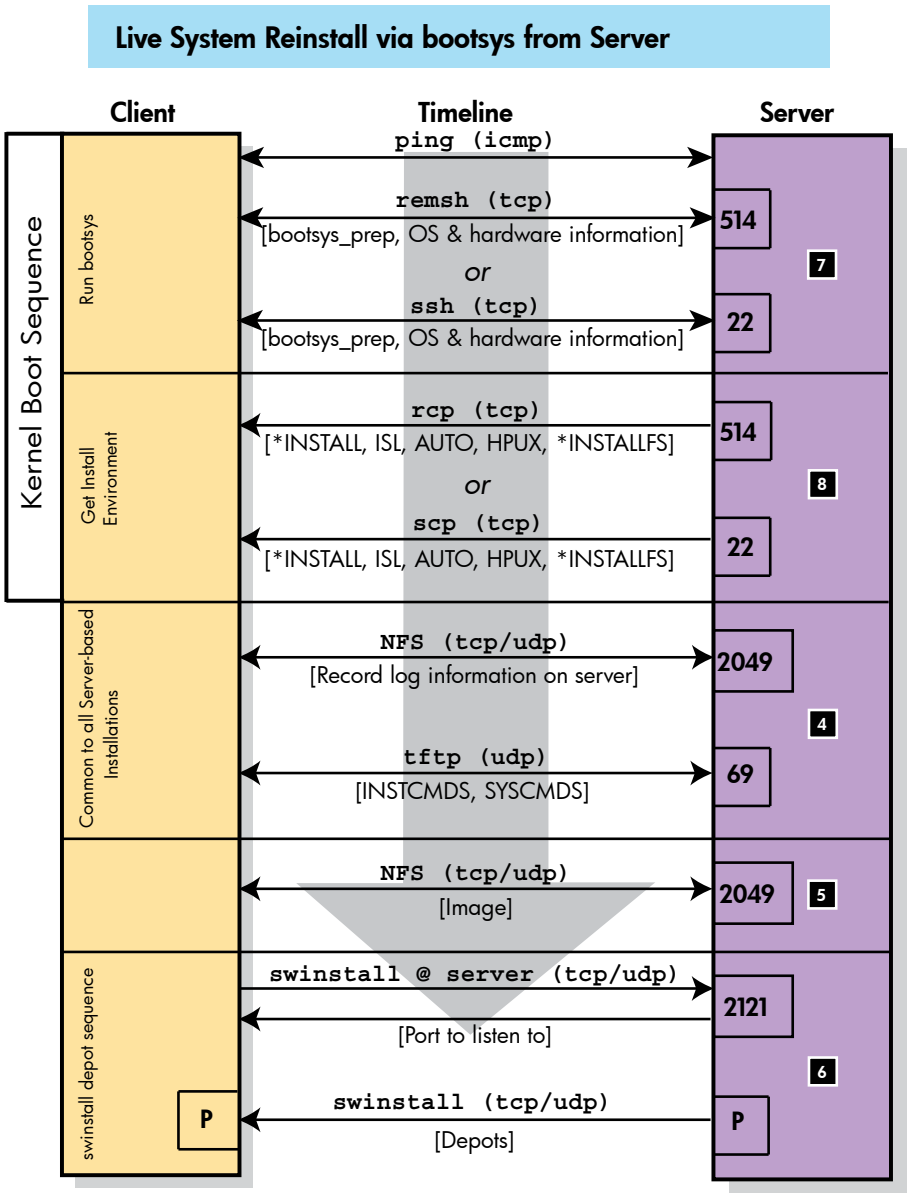


3. The initial *boot content* (*kernel*, *file system*, and required files) is downloaded from the server to the client, then the client is booted. For Itanium-based systems, these files are downloaded in the order listed: `nbp.efi`, *`AUTO`*, `fpswa.efi`, `hpux.efi`, *`IINSTALL`*, and *`IINSTALLFS`*. PA-RISC systems download these files in the order listed: `boot_lif`, `AUTO`, *`WINSTALL`*, and *`WINSTALLFS`*.
4. The file `install.log` is updated on the server in the `/var/opt/ignite/clients/`*`client`* directory. A compressed `tar` *archive* of commands to set up disk volumes and *file system*s is downloaded (*`INSTCMDS`* for PA-RISC, and *`INSTCMDSIA`* for Itanium-based systems). The TUI is run on the client console. The user selects the installation configuration via the TUI and selects **Go!**. A compressed `tar` archive of commands required to complete the installation is downloaded (*`SYSCMDS`* for PA-RISC and *`SYSCMDSIA`* for Itanium-based systems). Ports used by NFS to make RPC (Remote Procedure Call) calls are not discussed here.
5. If the installation is from an image, it is downloaded. Ports used by NFS to make RPC calls are not discussed here.
6. If the installation configuration requires software to be installed from *depot*s on the server, a `swinstall` request is sent to the server's Software Distributor (SD) daemon, `swagentd`, on port 2121. An SD agent, `swagent`, is then spawned on the server that acquires a dynamically allocated communication port for the download. That communication port is then reported to the client on port 2121. The client then spawns a new `swagent` processes that communicates with the server on the acquired communication port *`P`*, where the depot

download takes place. For more information on SD, see the *Software Distributor Administration Guide* available at http://www.hp.com/go/sd-docs.

NOTE: Although `swinstall` is illustrated here, the install could use one or more of `swinstall` with `remsh` (port 514), NFS (ports 49152–65535), `ftp data` (port 20), and `ftp` (port 21).

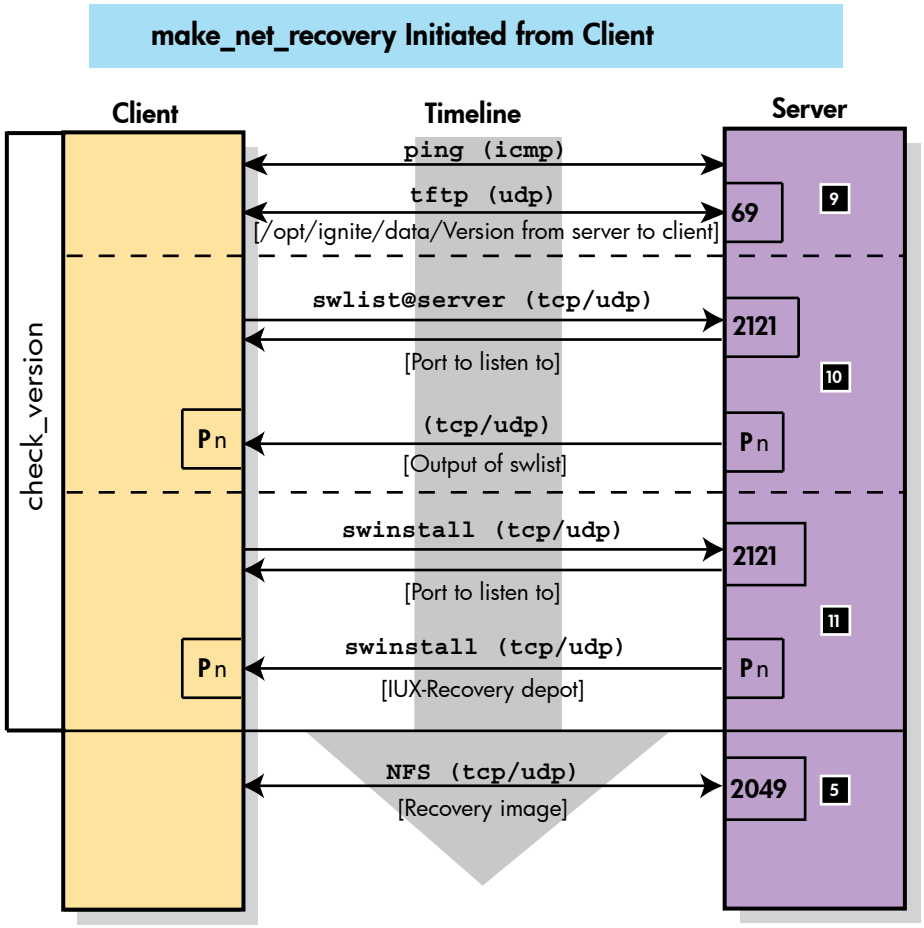
Figure 8-4 Port Usage: Live System Reinstall



7. The server pings the client with an ICMP type 8 echo request. The client answers the ping with an ICMP type 0 echo reply. Files required for `bootsys` are transferred from the server to the client. These files are transferred with `remsh` by default, or by `ssh` if the `bootsys -S` option is used.
8. The *kernel*, *file system*, and required files are downloaded from the server to the client, then the client is booted. These files are transferred with `rcp` by default, or by `scp` if the `bootsys -S` option is used.

NOTE:
The client can specify to use privileged ports (1–1023) or not via the `ssh_config` directive. The default is non-privileged ports. If you want to configure `ssh` to use privileged ports, you have to make the client an `suid` program.

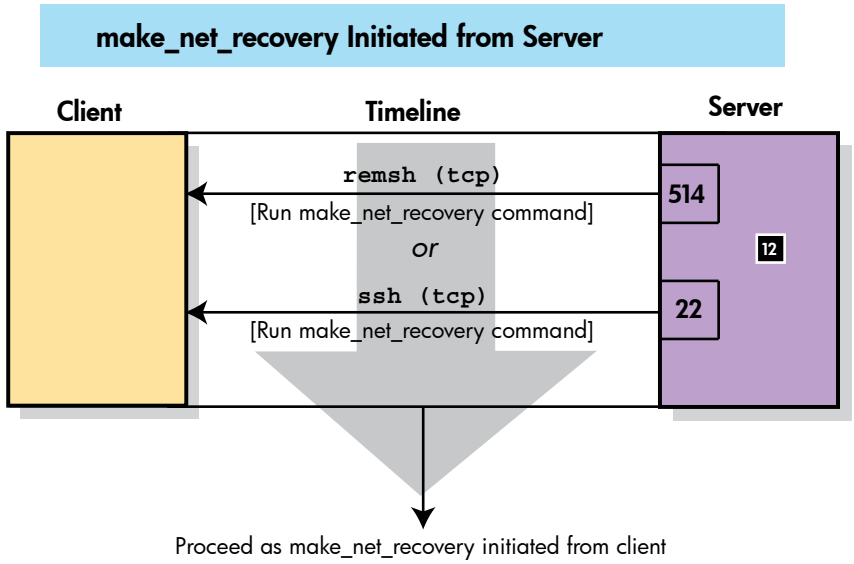
Figure 8-5 Port Usage: `make_net_recovery` Initiated from the Client



9. The server pings the client with an ICMP type 8 echo request. The client answers the ping with an ICMP type 0 echo reply. If `tftp` is enabled, the version check is done with the file `/opt/ignite/data/Version`.
10. If `tftp` is not enabled, the version check is done with `swlist` using the `swinstall` *depot* sequence described in Figure 8-3 (page 97).
11. If the client has a lower version of Ignite than the server, a depot of recovery commands is transferred to the client using the `swinstall` depot sequence described in Figure 8-3 (page 97).

NOTE: Although `swinstall` is illustrated here, the install could use one or more of `swinstall` with `remsh` (port 514), NFS (ports 49152–65535), `ftp data` (port 20), and `ftp` (port 21).

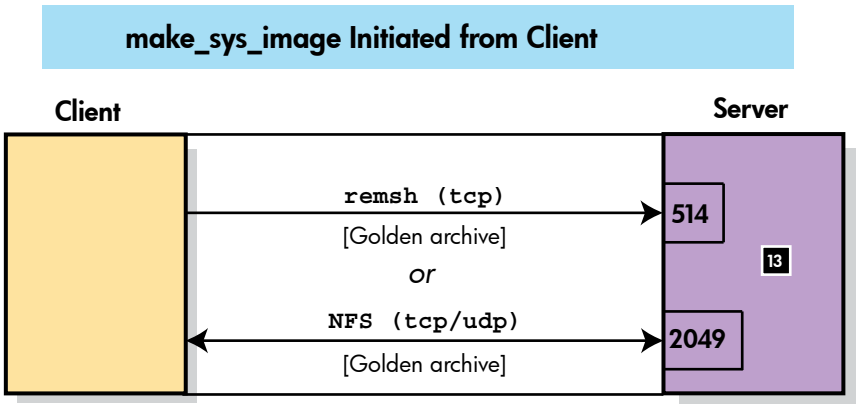
Figure 8-6 Port Usage: `make_net_recovery` Initiated from the Server



12. The server remotely executes `make_net_recovery` from the client. The command is run via `remsh` by default, or by `ssh` if the client was added for recovery on the server with the ssh option.

NOTE:
The client can specify to use privileged ports (1–1023) or not via the `ssh_config` directive. The default is non-privileged ports. If you want to configure `ssh` to use privileged ports, you have to make the client an `suid` program.

Figure 8-7 Port Usage: `make_sys_image` Initiated from the Client



13. The *golden archive* is written to the destination server via `remsh` or NFS. Note that `make_sys_image` does not need networking if the archive is written locally to the client.

Modifying a Bastille-Hardened System to Operate with Ignite-UX

HP-UX *Bastille* is a security hardening/lockdown tool that can be used to enhance the security of the HP-UX operating system. It provides customized lockdown on a system-by-system basis by encoding functionality similar to the Center for Internet Security (CIS) Level 1 Benchmark for HP-UX and other hardening/lockdown checklists. The Bastille technology is available in HP-UX 11i v1 and later versions of HP-UX.

This section describes how to make sure Ignite-UX requirements are enabled on your Bastille system.

For more information on HP-UX Bastille, see *bastille*(1M) , *bastille_drift*(1M), the *HP-UX System Administrator's Guide: Security Management* if you are running HP-UX 11i v3, and *Managing Systems and Workgroups: A Guide for HP-UX System Administrators* for systems running HP-UX 11i v2 and earlier.

CAUTION: The configuration processes in this section change the security properties of your system. When enabling services, protocols, and ports, careful consideration should be given to the impact to your network and system security.

Enabling Ignite-UX Server Requirements

To make sure Ignite-UX requirements are enabled on the server, you must first discover your current lockdown state and then modify that state, if necessary, to allow selected *daemon*s and services to run. You must also allow access to certain ports used by an *Ignite-UX server*.

1. Discover your current lockdown state.
   - If you are using *Bastille* 3.0 or later, create a configuration report. The report will be created in `/var/opt/sec_mgmt/bastille/log/Assessment/assessment-log.config`.

     ```
     # bastille --assessnobrowser
     ```

   - If you are using a version of Bastille earlier than 3.0, get the latest configuration file used by Bastille.

     ```
     # bastille -l
     ```

NOTE: If you get the message

```
NOTE:    The system is in its pre-bastilled state.
```

there is no need to proceed with this configuration, as *daemon*s, services, and ports required by Ignite-UX are not locked-down in the pre-bastille state.

2. Copy the last configuration file used or the assessment report to a place of your choice.
3. Bring up the latest configuration in the *Bastille* GUI.

   ```
   # bastille --os [HP-UX11.00 | HP-UX11.11 | HPUX11.23 | HPUX11.31] -f filename
   ```

4. Make sure the settings in your configuration file for the following *daemon*s and services are set to `No`. Note that if you have to change a setting from `Yes` to `No`, you will likely be required to enable that daemon or service on your system in order to use it. After you have made changes, save the configuration file to a place of your choice.

   ```
   Would you like to deactivate the NFS server on this system
   Would you like to deactivate NIS client programs?
   Should Bastille ensure inetd's bootp service does not run on this system?
   Should Bastille ensure inetd's TFTP service does not run on this system?
   ```

5. To update your firewall or have Bastille create a new one:

a. Backup your `/etc/opt/ipf/ipf.conf` file to a place of your choice.

b. Update the port information for the Bastille-enabled HP-UX IPFilter firewall by editing the file `/etc/opt/sec_mgmt/bastille/ipf.customrules` and making the following changes:

- Add the words **`keep frags`** to the end of the udp outgoing rule line so it looks like

  ```
  pass out quick proto udp all keep state keep frags
  ```

- Remove or comment-out the following line.

  ```
  block in quick proto udp from any to any port = portmap
  ```

- Add the following lines after the `End allow outgoing rules` section.

```
# ports required for Ignite-UX
############################################################
pass in log quick proto udp from any to any port = 69 keep state
pass in log quick proto udp from any port = 68 to any port = 67 keep state
pass in log quick proto udp from any port = 1068 to any port = 1067 keep state
pass in log quick proto tcp/udp from any to any port = 2049 keep frags
pass in log quick proto tcp/udp from any to any port = 2121
pass in log quick proto tcp/udp from any to any port 49152 >< 65535
pass in log quick proto tcp from any to any port = 20
pass in log quick proto tcp from any to any port = 21
pass in log quick proto tcp from any to any port = 22
pass in log quick proto tcp from any to any port = 514
pass in log quick proto icmp from any to any icmp-type 8 keep state
pass in log quick proto tcp from any port = 514 to any keep state
```

c. In the IPFilter Module of Bastille, change the following line to `Yes` if it is not already.

```
Should Bastille setup basic firewall rules with these properties?
```

d. Run Bastille.

```
# bastille -b -f your_configuration_file
```

6. If a Bastille baseline had been created for the system, update that baseline.

```
# bastille_drift --save_baseline baseline
```

Enabling Ignite-UX Client Requirements

To make sure Ignite-UX requirements are enabled on the client, you must first discover your current lockdown state and then modify that state, if necessary, to allow the NFS *daemon* and rtools services to run. You must also allow access to certain ports used by an Ignite-UX client.

1. Discover your current lockdown state.

- If you are using *Bastille* 3.0 or later, create a configuration report. The report will be created in `/var/opt/sec_mgmt/bastille/log/Assessment/assessment-log.config`.

  ```
  # bastille --assessnobrowser
  ```

- If you are using a version of Bastille earlier than 3.0, get the latest configuration file used by Bastille.

  ```
  # bastille -l
  ```

NOTE: If you get the message

```
NOTE:    The system is in its pre-bastilled state.
```

there is no need to proceed with this configuration, as daemons, services, and ports required by Ignite-UX are not locked-down in the pre-bastille state.

---

2. Copy the last configuration file used or the assessment report to a place of your choice.

3. Bring up the latest configuration in the Bastille GUI.

```
# bastille --os [HP-UX11.00 | HP-UX11.11 | HPUX11.23 | HPUX11.31] -f filename
```

4. Make sure the settings in your configuration file for the NFS daemon and rtools service are set to `No`. Note that if you have to change a setting from `Yes` to `No`, you will likely be required to enable that daemon or service on your system in order to use it. After you have made changes, save the configuration file to a place of your choice.

   ```
   Would you like to deactivate the NFS client daemons?
   Should Bastille ensure that the login, shell, and exec services do not run on this system?
   ```

5. To update your firewall or have Bastille create a new one:
   a. Backup your `/etc/opt/ipf/ipf.conf` file to a place of your choice.
   b. Update the port information for the Bastille-enabled HP-UX IPFilter firewall by editing the file `/etc/opt/sec_mgmt/bastille/ipf.customrules` and making the following changes:
      - Add the words **`keep frags`** to the end of the udp outgoing rule line so it looks like

        ```
        pass out quick proto udp all keep state keep frags
        ```

      - Add the following lines after the `End allow outgoing rules` section.

        ```
        # ports required for Ignite-UX
        ##############################################################
        pass in log quick proto icmp from any to any icmp-type 8 keep state
        pass in log quick proto tcp from any to any port = 512
        pass in log quick proto tcp from any to any port = 514
        pass in log quick proto tcp/udp from any port = 2049 to any keep frags
        pass in log quick proto tcp/udp from any to any port 49152 >< 65535
        ```

   c. In the IPFilter Module of Bastille, change the following line to `Yes` if it is not already.

      ```
      Should Bastille setup basic firewall rules with these properties?
      ```

   d. Run Bastille.

      ```
      # bastille -b -f your_configuration_file
      ```

6. If a Bastille baseline had been created for the system, update that baseline.

   ```
   # bastille_drift --save_baseline baseline
   ```

Configuring Ignite to Replace TFTP with NFS

Beginning with Ignite-UX version C.7.9, it is possible to configure the Ignite-UX `loadfile` utility to use NFS instead of TFTP for network access to the Ignite server. This allows users to avoid use of TFTP except during direct network boot of the install kernel. The TFTP protocol can be avoided entirely if the system being installed is booted from media (including vMedia) or via the `bootsys` command.

Overview

In order to use this functionality, minor modifications to Ignite-UX configuration files might have be made to the Ignite-UX server system. These modifications fall into the following categories:

- Add a keyword to the appropriate configuration files instructing Ignite to use NFS instead of TFTP.
- Ensure config files are located in an acceptable directory that is NFS-mounted during the installation. Make sure the `INDEX` file refers to the config files in their new (C.7.9 and later) locations as outlined below.
- Disable the TFTP daemon.

NOTE: Because of changes necessary to replace TFTP with NFS, beginning with C.7.9 the locations of three Ignite product files have moved. Ignite automatically creates symbolic links from the old file to the new file location. These files are:

Table 8-1 Ignite Product Files Moved in Version C.7.9 and Later

Pre-C.7.9 Location	C.7.9 and Later Location
`/opt/ignite/Version`	`/opt/ignite/data/Version`
`/var/opt/ignite/INDEX`	`/var/opt/ignite/data/INDEX`
`/var/opt/ignite/config.local`	`/var/opt/ignite/data/config.local`

Procedure

1. Add the `_hp_loadfile_use_nfs` keyword.

   HP recommends placing this in the config section of the install file system. Use your environment's HP-UX version and install file system in the following commands.

   First, change the working directory to the release-specific boot directory and grab the config content:

   ```
   # cd /opt/ignite/boot/Rel_B.11.31
   # instl_adm -d > /tmp/ifs.cfg
   ```

   Use `vi` to add `_hp_loadfile_use_nfs="true"` to the file:

   ```
   # cat /tmp/ifs.cfg
   # instl_adm defaults:
   # NOTE: Manual additions between the line containing "instl_adm defaults
   #       and "end instl_adm defaults" will not be preserved.
   server="10.1.54.230"
   netmask[]="255.255.248.0"
   route_gateway[0]="10.1.48.1"
   route_destination[0]="default"
   # end instl_adm defaults.
   _hp_loadfile_use_nfs="true"
   ```

   Now use `instl_adm` to update the install file system:

```
# instl_adm -f /tmp/ifs.cfg
```

2. Set up NFS exports and check custom configuration files.

   Because of preexisting Ignite-UX file system layouts, the locations of certain files are automatically moved when Ignite-UX Version C.7.9 or later is installed . These files are `/opt/ignite/Version`, `/var/opt/ignite/INDEX`, and `/var/opt/ignite/config.local`. In addition, certain other customer-created config files might need to be moved or edited.

   First, edit `/etc/exports` or `/etc/dfs/dfstab` as appropriate for the version of HP-UX running on the Ignite-UX server.

   ```
   For Ignite servers running HP-UX 11i v2 or earlier:

   # cat /etc/exports
   /opt/ignite/data -ro
   /var/opt/ignite/config -ro
   /var/opt/ignite/data -ro
   /var/opt/ignite/clients -anon=2

   # exportfs -a


   For Ignite servers running HP-UX 11i v3:

   #  cat /etc/dfs/dfstab
   share -F nfs -o ro /opt/ignite/data
   share -F nfs -o ro /var/opt/ignite/data
   share -F nfs -o anon=2 /var/opt/ignite/clients

   # shareall
   ```

   If there are any customer-created config files that are outside of the directories exported by NFS, they must be moved under `/var/opt/ignite/data`.

3. Edit the `/var/opt/ignite/INDEX` file to refer to `/var/opt/ignite/data`.

   Some customers will need to modify the `config.local` entry in `/var/opt/ignite/data/INDEX`. For example, the INDEX clause

   ```
   cfg "HP-UX B.11.31 Default" {
           description "Example HP-UX 11i v3 (11.31) configuration."
           "/opt/ignite/data/Rel_B.11.31/config"
           "/opt/ignite/data/Rel_B.11.31/hw_patches_cfg"
           "/var/opt/ignite/data/Rel_B.11.31/core_cfg"
           "/var/opt/ignite/config.local"
   }
   ```

   would need to be modified to be

   ```
   cfg "HP-UX B.11.31 Default" {
           description "Example HP-UX 11i v3 (11.31) configuration."
           "/opt/ignite/data/Rel_B.11.31/config"
           "/opt/ignite/data/Rel_B.11.31/hw_patches_cfg"
           "/var/opt/ignite/data/Rel_B.11.31/core_cfg"
           "/var/opt/ignite/data/config.local"
   }
   ```

4. Disable TFTP on the Ignite-UX server (optional).

   Unless you need to initiate installations via network boot, you may now disable TFTP on the Ignite-UX server. You may remove or comment out the "tftp" entry from `/etc/inetd.conf`.

   If the system to be installed is running any version of HP-UX, booting from the network can be avoided by using the `bootsys` command or by booting from media and switching to the Ignite-UX server.

In the boot-from-media case, it will be necessary to either specify the `_hp_loadfile_use_nfs` keyword on the boot loader command line or create custom media with that keyword built into it.

For **PA-RISC** systems, interrupt the autoboot sequence at the prompt

```
Interact with IPL (Y, N, or Cancel)?> y
```

and enter the following (substitute **11.31** for **11.23** depending on the release).

```
ISL> hpux -i_hp_loadfile_use_nfs=\"true\" Rel_B.11.23/WINSTALL
```

For **Integrity** systems, interrupt the autoboot sequence at the prompt

```
Press Any Key to interrupt Autoboot
```

and enter the following (substitute **11.31** for **11.23** depending on the release).

```
HPUX> boot Rel_B.11.23/IINSTALL -i_hp_loadfile_use_nfs=\"true\"
```

If you do need to preserve the ability to perform network boot, but otherwise wish to take advantage of the NFS `loadfile` functionality, you may remove the `/var/opt/ignite` directory from the "tftp" entry in `/etc/inetd.conf`, leaving only `/opt/ignite`.

When Ignite-UX is installed, it automatically enables the TFTP daemon. If you reinstall Ignite-UX, you will need to reapply these changes.

For information on booting from media and then switching to an Ignite-UX server over the network, see "Alternate Boot with Network Server Installation" (page 34). For information about changing configuration content in the install file system, see *instl_adm*(1M) and *instl_adm*(4).

5. Verify that the configuration works.

   Perform an installation, and watch the console output or the `install.log` file for the "Ignite-UX will use NFS for loadfile." message:

```
       * Preparing to execute init...
=======  03/02/09 20:39:20 EST  HP-UX Installation Initialization.
         @(#)Ignite-UX Revision C.7.8.201
         @(#)ignite/launch (opt) Revision:
         /branches/IUX_RA0903/ignite/src@76987 Last Modified: 2009-02-05
         15:45:55 -0700 (Thu, 05 Feb 2009)
       * Configuring RAM filesystems...
NOTE:    Ignite-UX will use NFS for loadfile.
```

9 Booting and Installing HP-UX From the Server Using the Client Console

This chapter discusses booting and installing HP-UX on *client*s from the server using the client console. Ignite-UX can be run in terminal user interface (TUI) mode on the client system.

See the *HP-UX Installation and Update Guide* available from **http://www.hp.com/go/hpux-core-docs** for instructions on how to install HP-UX from the Operating Environment DVD media.

Preparing the Client for Installation

For `bootsys` — The `bootsys` command is used to reboot a client system, already running HP-UX, using a *kernel* and *file system* from a server. The `bootsys` command will copy the *[W|V|I]INSTALL install kernel* and the *[W|V|I]INSTALLFS* file system from the server to the `/stand` directory on the client, and then use them when rebooting.

Make sure there is enough disk space in the `/stand` directory on the client to hold the install kernel and file system before you run `bootsys`.

For V-class PA-RISC clients the files on the server are:

- `/opt/ignite/boot/Rel_`*`release`*`/VINSTALL`
- `/opt/ignite/boot/Rel_`*`release`*`/VINSTALLFS`

For 64-bit PA-RISC clients the files on the server are:

- `/opt/ignite/boot/Rel_`*`release`*`/WINSTALL`
- `/opt/ignite/boot/Rel_`*`release`*`/WINSTALLFS`

For Itanium-based clients the files on the server are:

- `/opt/ignite/boot/Rel_`*`release`*`/IINSTALL`
- `/opt/ignite/boot/Rel_`*`release`*`/IINSTALLFS`

where *`release`* is the release identifier.

For HP-UX 11i v3 — If you are installing HP-UX 11i v3 onto a client, its *boot* disk must be at least 9 GB. HP-UX 11i v3 requires more space on the HP-UX boot disk than prior HP-UX releases.

Minimum Memory Size — During *installation* and *recovery*, Ignite-UX uses system memory to hold a RAM-based *install environment* with a subset of HP-UX. Ignite-UX requires installation and recovery client systems to have at least a minimum amount of RAM to hold this install environment while leaving enough space to run HP-UX. The minimum required RAM size is specific to the HP-UX version to be installed or recovered. See the *Ignite-UX Release Notes* under "Minimum Memory Size" for the current client memory requirements. You can find the *Ignite-UX Release Notes* via the Ignite-UX website, http://www.hp.com/go/ignite-ux, and also at `/opt/ignite/share/doc/release_note` on your system.

If Ignite-UX detects there is not enough RAM on the client system, you will see these errors:

```
ERROR: RAMFS Setup memory issue.
```

```
ERROR: The system does not contain the minimum supported amount of
memory needed to install and run HP-UX. HP-UX requires minimum_amount
of available memory for the "B.xx.xx" release. The system has only
actual_amount of memory available for HP-UX use (this may be less than
physical memory installed due to space reserved by system firmware).
The amount of memory in the system must be increased if this release
is to be installed successfully.
```

```
ERROR: The system install or recovery session cannot continue. The
system will now reboot.
```

If Ignite-UX detects only the minimum amount of RAM on the client system, you will see these messages:

```
WARNING: RAMFS Setup memory issue.

WARNING: The system does not contain the minimal amount of memory needed
for install or recovery. Ignite-UX requires minimum_amount of available
memory for the "B.xx.xx" release. The system has only actual_amount of
memory available for use (this may be less than physical memory installed
due to space reserved by system firmware). The install or recovery may
or may not complete, and it may take an extraordinary amount of time
to complete. It is advised to increase the amount of memory in the
system.

NOTE: The system install or recovery session will now continue.
```

CAUTION: Any data on the client disks that are used for installation, including the operating system, are removed entirely as part of this installation process.

IMPORTANT: During HP-UX 11i v3 installation and recovery, connected Active/Passive devices will cause long delays (one hour or more) or may cause a system to hang. Similarly, connecting an Active/Passive device before installing the Active/Passive Switch (APSW) plug-in can cause some commands to take a long time. Disconnect any Active/Passive devices connected to your system before installing or recovering HP-UX 11i v3. After installation or recovery, it is important that the APSW plug-in be installed before connecting an Active/Passive device to the system.

Making Boot Decisions When Using the Client Console

When deciding which method to use when operating from the *client* console, you should take your server/client configuration into consideration. See Chapter 2 (page 31) for information on configuring your *Ignite-UX server* for your environment.

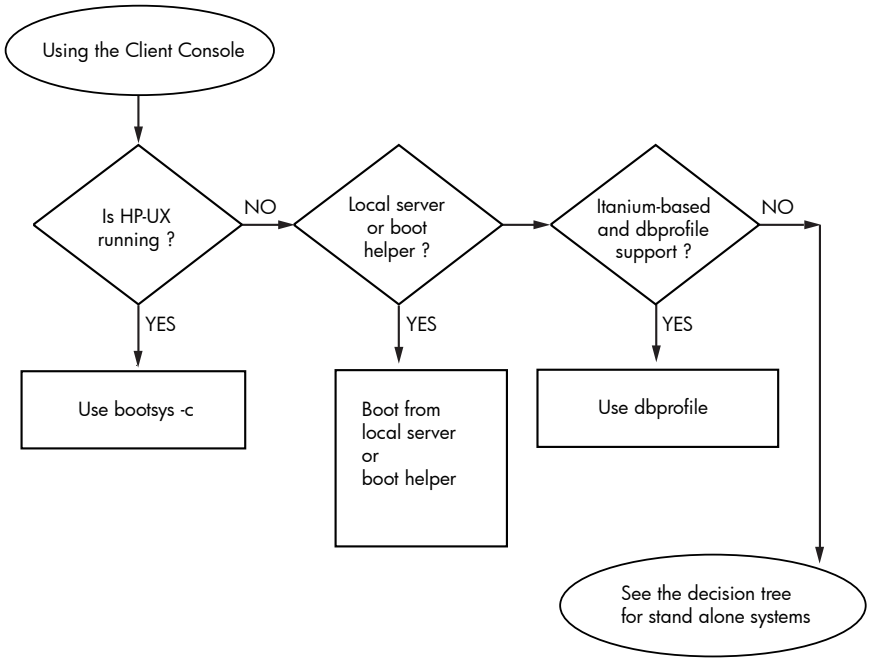
This section provides an overview of options when booting from the client console.

Boot Using the Network

A decision tree for booting and installing HP-UX from the client console using the server is shown in Figure 9-1. If you want to *boot* a client without using an Ignite-UX server, see Figure 9-2 "Decision Tree for Booting From Media and Installing HP-UX From the Server ".

Use the decision tree below when you want to install from the Ignite-UX server and control the installation from the client console.

Figure 9-1 Decision Tree for Booting and Installing HP-UX From the Server Using the Client Console



Use `bootsys -c` - If the client system is currently running HP-UX, you can use `bootsys -c` on the client console to *boot* from the *Ignite-UX server*. See "Using bootsys on the Client Console" (page 110) and the *bootsys*(1M) manpage for more information.

Boot from local server or boot helper - You can boot your client from a server or *boot helper system* using the client console by interrupting the reboot process and invoking the boot from the firmware interface. Details vary depending whether your client is a PA-RISC or Itanium-based. See "Booting PA-RISC Clients from the Console " (page 111), or "Booting Itanium-Based Clients using the Network" (page 112), depending on the hardware of your client.

Use `dbprofile` - All partitionable Itanium-based systems allow the definition of direct boot profiles. This *EFI* functionality is also found in other, non-partitionable systems. With these profiles, you can supply all the networking information needed to contact an Ignite-UX server and perform an install or recovery.

Some systems might require firmware updates to provide support for direct boot profiles. If your system does not provide the `dbprofile` command, check for any firmware updates that might enable it. You can also consult the system's hardware documentation to determine if `dbprofile` is supported.

For more information, see "Direct Boot Profiles for Itanium-Based Systems" (page 115).

See the decision tree for booting *stand alone* systems - This decision tree can be found below in figure Figure 9-2.

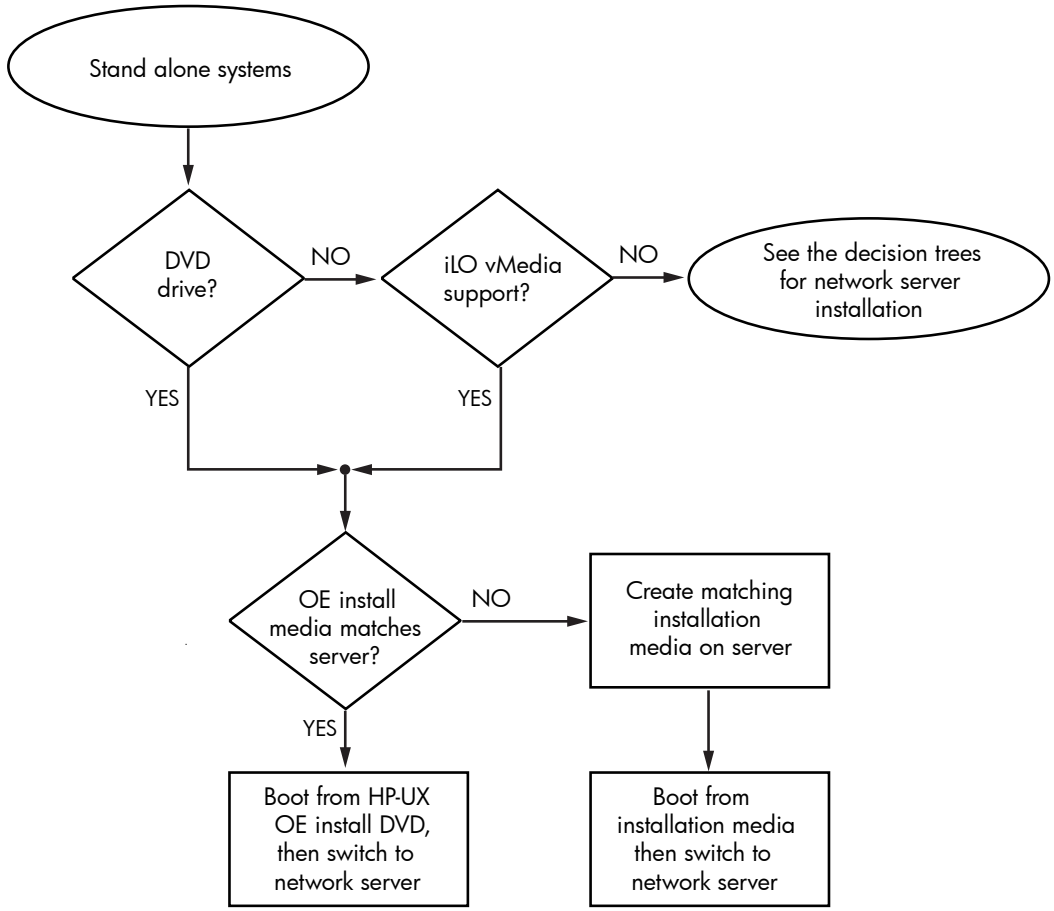
Boot Using Media

Use the following decision tree if you do not have support for network *boot*. The methods described in Figure 9-2 use media content to boot for install.

Once you have booted the system, you will be able to communicate with an *Ignite-UX server* to perform an installation or recovery. Note that if you do not have an active *DHCP* server to provide networking IP address requests, you will need to manually provide networking information before you can communicate with the server.

See "How Ignite Works" (page 25) for more information on the network booting process.

Figure 9-2 Decision Tree for Booting From Media and Installing HP-UX From the Server



Boot from HP-UX OE install DVD, then switch to network server - This option requires the system to have a CD or DVD drive attached, or iLO vMedia support. The version of Ignite-UX on the OE media must match the Ignite-UX version on the server. For more information on iLO vMedia, see Appendix D (page 263) and the *HP Integrity iLO 2 MP Operations Guide* available at http://www.hp.com/bizsupport.

Create matching installation media on server - It is possible to create *installation media* for booting purposes. This type of installation media does not include a full *archive*. See Chapter 14: "Creating Your Own Boot and Installation Media", for more information.

Boot from installation media then switch to network server - This option assumes you have created installation media for booting purposes. See Chapter 14: "Creating Your Own Boot and Installation Media", for more information. This option requires the system to have a CD or DVD drive attached, or iLO vMedia support. For more information on iLO vMedia, see Appendix D (page 263) and the *HP Integrity iLO 2 MP Operations Guide* available at http://www.hp.com/bizsupport.

See the decision trees for network server installation - These decision trees are: Figure 2-1: "Decision Tree When Configuring a Server for Booting PA-RISC Systems", and Figure 2-2, "Decision Tree When Configuring a Server for Booting Itanium-Based Systems."

Using bootsys on the Client Console

The `bootsys` command may be run from the client console if you use the `-c` *`Ignite-UX Server`* option. This option directs `bootsys` to contact the specified *Ignite-UX server*, and then perform a local reboot.

For more information on `bootsys`, see *bootsys*(1M) and the section describing its use from the server: "Installation Using bootsys" (page 126).

After booting the system, see “Installing HP-UX From the Client Console” (page 118) for information on installing HP-UX from the client console.

Booting PA-RISC Clients from the Console

This section describes how to *boot* HP-UX on PA-RISC clients from the client console using an *Ignite-UX server*.

See the "Preparing the Client for Installation " section for important notes.

If you need further help with the boot process, enter:

```
BOOT ADMIN>help boot
```

1. Obtain the IP address of the Ignite-UX server you intend to use.
2. Cycle the power (perform a cold reset) on the client to bring it to a known state.

NOTE: If you have the `AUTOBOOT` flag set, you will have to interrupt the boot sequence by pressing **Esc** when the Processor Dependent Code (PDC) offers the opportunity with this message:

```
To stop selection process, press and hold ESCAPE key.
```

For detailed information regarding the boot sequence, see the *HP-UX System Administrator’s Guide* for HP-UX 11i v3, or *Managing Systems and Workgroups: A Guide for HP-UX System Administrators*.

During the boot sequence, status messages are displayed on the client console. Depending on the type of machine, server or workstation model, a boot administration menu and/or firmware prompt may appear.

3. Boot the client using your Ignite-UX server’s IP address by entering this command at the client console:

   firmware prompt>**boot lan.*n.n.n.n* install**

   where: *n.n.n.n* is the IP address of the Ignite-UX server.

   The client then begins to download the *install kernel* from the network server. This should take approximately 5 minutes.

TIP: To search for Ignite-UX servers, enter the following at the client console (workstations only):

firmware prompt>**search lan install**

The list of servers you can boot the client from is displayed with their corresponding IP addresses, similar to:

```
Searching for potential boot devices(s)...  on Path LAN
This may take several minutes.

To discontinue search, press any key (termination may not be immediate).

Path Number     Device Path               Device Type
-----------     -----------               -----------
P0              LAN.15.1.46.117.3.254      lp2   100/Full Dx
P1              LAN.15.1.41.70.3.254       lp4   100/Full D
```

You may need to run the `nslookup` command on another running system to determine which address corresponds to your Ignite-UX server.

4. When an Ignite-UX server responds, the installation begins with the following query:

```
hpux KernelPrompt "Choose Operating System to Install :"

    1.  target OS is B.11.11
    2.  target OS is B.11.23 PA
    3.  target OS is B.11.31 PA
    4.  Exit

Choose an operating system to install that your hardware supports :
```

Select the operating system version that you want to install on the client by typing the appropriate number, and then press **Enter** to continue the installation.

After booting the system, see "Installing HP-UX From the Client Console" (page 118) for information on configuring the HP-UX installation from the client console.

Booting Itanium-Based Clients using the Network

This section describes how to *boot* HP-UX on Itanium-based clients from an Ignite-UX server using the network.

See the "Preparing the Client for Installation " section for important notes.

1. Cycle the power (perform a cold reset) on the client to bring it to a known state.

NOTE: If you have the AUTOBOOT flag set, you must interrupt the *boot* sequence by pressing **Esc** when the Processor Dependent Code offers the opportunity with this message:

```
Press Any Key to interrupt Autoboot
```

Next, you will need to enter **exit** at the prompt to invoke the *extensible firmware interface (EFI)* Boot Manager menu.

For detailed information regarding the boot sequence, see the *HP-UX System Administrator's Guide* for HP-UX 11i v3, or *Managing Systems and Workgroups: A Guide for HP-UX System Administrators*, and the hardware documentation for your system.

During the boot sequence, status messages are displayed on the client console. Depending on what type of machine, server or workstation model, the *EFI* Boot Manager menu appears and looks similar to:

```
EFI Boot Manager ver 1.10 [14.60]

Please select a boot option

    HP-UX Primary Boot: 0/2/2/0.0.0.0
    EFI Shell [Built-in]
    Boot option maintenance menu
    Security/Password Menu

    Use ^ and v to change option(s). Use Enter to select an option
```

TIP: On some machines, the up-arrow and down-arrow keys may not work. If this is the case, you can use **Shift-6** (**^**) for up and **v** for down.

2. Select **Boot option maintenance menu** using the up and down arrows, which advances you to the EFI Boot Maintenance Manager Main Menu, similar to the following example:

```
EFI Boot Maintenance Manager ver 1.10 [14.60]

Main Menu. Select an Operation

        Boot from a File
        Add a Boot Option
        Delete Boot Option(s)
        Change Boot Order

        Manage BootNext setting
        Set Auto Boot TimeOut

        Select Active Console Output Devices
        Select Active Console Input Devices
        Select Active Standard Error Devices

        Cold Reset
        Exit
```

3. Select **Add a Boot Option**.

```
EFI Boot Maintenance Manager ver 1.10 [14.60]

Add a Boot Option.  Select a Volume

 Removable Media Boot[Acpi(HWP0002,0)/Pci(2|0)/Ata(Primary,Master)]
 Load File [EFI Shell [Built-in]]
 Load File [Acpi(HWP0002,0)/Pci(3|0)/Mac(00306E1E4ED4)]
 Load File [Acpi(HWP0002,100)/Pci(2|0)/Mac(00306E1E3ED6)]
  Exit
```

4. Select the appropriate network interface so that this network boot loads the appropriate file. For example, look for entries identified with a MAC address as in this example.

```
Device Path Acpi(HWP0002,0)/Pci(3|0)/Mac(00306E1E4ED4)
Boot0001: Acpi(HWP0002,0)/Pci(3|0)/Mac(00306E4A134B)
Edit Existing Boot Option or make a new entry [E-Edit N-New]:
```

5. Press **N** to add the new boot option.
6. Enter a brief, descriptive boot option name at the message prompt. In this example, the new boot option is named `LAN1`.

```
Enter New Description:
```

7. Enter a brief description for this boot option.

```
New BootOption Data. ASCII/Unicode strings only, with max of 240 characters

Enter BootOption Data Type [A-Ascii U-Unicode N-No BootOption] :
```

8. Enter the data type of this boot option.

```
Save changes to NVRAM [Y-Yes N-No]:
```

9. Press **Y** to save the new boot option.

```
EFI Boot Maintenance Manager ver 1.10 [14.60]

Add a Boot Option.  Select a Volume

 LAN1 [Acpi(HWP0002,0)/Pci(3|0)/Mac(00306E1E4ED4)]
 Removable Media Boot[Acpi(HWP0002,0)/Pci(2|0)/Ata(Primary,Master)]
 Load File [EFI Shell [Built-in]]
 Load File [Acpi(HWP0002,0)/Pci(3|0)/Mac(00306E1E4ED4)]
 Load File [Acpi(HWP0002,100)/Pci(2|0)/Mac(00306E1E3ED6)]
 Load File [PcieRoot(0x30304352)/Pci(0x7,0x0)/Pci(0x0,0x0)/Fibre(0x500110A0008F063A, 0x0)]
 Exit
```

NOTE: The boot option name, description, and data type information can be modified using the `Edit` operation in a manner similar to the `New` operation as described in this procedure.

10. Exit to the EFI Boot Manager menu taking care not to select a boot option, as you will be forced to reenter the information for the selected option.

    The new boot option should appear in the EFI Boot Manager main menu.

```
EFI Boot Manager ver 1.10 [14.60]

Please select a boot option

   HP-UX Primary Boot: 0/2/2/0.0.0.0
   LAN1
   EFI Shell [Built-in]
   Boot option maintenance menu
   Security/Password Menu

Use ^ and v to change option(s). Use Enter to select an option
```

11. Select the new boot option, `LAN1` in this case, from the list and press **Enter**.

    The following is an example of a successful network boot using the new `LAN1` boot option:

```
Loading.: LAN1
Running LoadFile()

CLIENT IP: 10.1.52.128  MASK: 255.255.248.  DHCP IP: 10.1.53.37
GATEWAY IP: 10.1.48.1
Running LoadFile()

Starting: LAN1

@(#) HP-UX IA64 Network Bootstrap Program Revision 1.0
Downloading HPUX bootloader
Starting HPUX bootloader
Downloading file fpswa.efi  (371200 bytes)

(c) Copyright 1990-2001, Hewlett Packard Company.
All rights reserved

HP-UX Boot Loader for IA64  Revision 1.671

Booting from Lan
Downloading file AUTO   (528 bytes)
Press Any Key to interrupt Autoboot
AUTO ==> boot IINSTALL
Seconds left till autoboot -   0
AUTOBOOTING...
```

After booting the system, see "Installing HP-UX From the Client Console" (page 118) for information on configuring the HP-UX installation from the client console.

Direct Boot Profiles for Itanium-Based Systems

All partitionable Itanium-based systems allow the definition of direct boot profiles. This *EFI* functionality is also found in other, non-partitionable systems.

Some systems might require firmware updates to provide support for direct *boot* profiles. If your system does not provide the `dbprofile` command, check for any firmware updates that might enable it. You can also consult the system's hardware documentation to determine if `dbprofile` is supported.

The `dbprofile` command allows you to create direct boot profiles that can be used in conjunction with the `lanboot` EFI command. A direct boot profile contains information needed to talk over the network to an installation server.

NOTE: This does not affect any *DHCP* support you may need to configure to allow the installation process to talk to an *Ignite-UX server* once the *install kernel* has started. The use of direct *boot* profiles only allows network boot from EFI without needing to setup a `bootp`/DHCP server.

The following `dbprofile` command, when run from *EFI*,

```
dbprofile -dn testprofile -sip 192.1.2.1 -cip \
192.1.1.20 -gip 192.1.1.1 -m 255.255.255.0 -b \
"/opt/ignite/boot/nbp.efi"
```

creates a new direct boot profile (or updates the profile if it exists) called `testprofile`. The profile contains the following information:

Ignite-UX server IP address (`-sip`) 192.1.2.1

Ignite client IP address (`-cip`) 192.1.1.20

Gateway IP address (`-gip`)	192.1.1.1
Network mask (`-m`)	255.255.255.0
Boot file name (`-b`)	`/opt/ignite/boot/nbp.efi`

IMPORTANT: If you do not have the client IP address defined in a direct boot profile, EFI will use DHCP instead of the direct boot profile to start the boot process.

When creating a direct boot profile, always define the boot file using the `-b` option to set the value to `/opt/ignite/boot/nbp.efi`. This is required to install HP-UX.

The `dbprofile` command accepts a *gateway* IP address. This is important because when using a direct boot profile that has a gateway defined, the EFI network stack will have somewhere to route non-local traffic. In this case, the Ignite-UX server does not have be located on the same LAN segment as the client; traffic can be routed to it via the gateway. No boot helper is required for this client on the local LAN segment.

To network boot using a direct boot profile, use the `-dn` option of the EFI `lanboot` command as in the following example:

```
lanboot select -dn testprofile
```

This results in a list of LAN interfaces for you to choose from. However, no DHCP request is sent out when an interface is selected. Instead, the network boot uses information from the direct boot profile `testprofile` and immediately starts booting from the Ignite-UX server.

NOTE: If you have multiple LAN interfaces, you must choose the LAN interface that the direct boot profile is applicable to. If you choose a LAN interface on a network other than (using the example from above) 192.1.1.*x*, the client will fail to contact the Ignite-UX server.

The dbprofile Command

This section provides information for the `dbprofile` command. This command is part of the *EFI* Shell and is accessible from the system console when the system is in an active state but has not booted an operating system. For more information, see the *nPartition Administrator's Guide* or the documentation for your particular system, available at the HP Business Support Center:

http://www.hp.com/bizsupport

Syntax

`dbprofile` [`-dn` *`name`* [`-dhcp`] [`-sip` *`server_ip`*] [`-cip` *`client_ip`*]
[`-gip` *`gateway_ip`*] [`-m` *`subnet_mask`*] [`-b "`*`boot_file`*`"`]
[`-od "`*`optional_data`*`"`]] | [`-h`]
`dbprofile` [`rm` *`name`*] | [`-h`]
`dbprofile` [`cp` *`src_name dst_name`*] | [`-h`]

Options and Operands

`-dn` *`name`*	Display/create/modify a direct boot profile with the name *`name`*.
`-dhcp`	A *DHCP* server will be used to get the client information.
`-sip` *`server_ip`*	Set the static IP address for the boot server to *`server_ip`*.
`-cip` *`client_ip`*	Set the static IP address for the client to *`client_ip`*.
`-gip` *`gateway_ip`*	Set the static IP address for the *gateway* to *`gateway_ip`*.
`-m` *`subnet_mask`*	Set the subnet mask for the local network to *`subnet_mask`*.
`-b "`*`boot_file`*`"`	Load boot file from the boot server. *`boot_file`* is the full path to the boot file on the boot server.
`-od "`*`optional_data`*`"`	Pass optional command line arguments to the boot file.

`-rm` *`name`*	Remove a direct boot profile with the name *`name`*.
`-cp` *`src_name dst_name`*	Copy a db-profile from *`src_name`* to *`dst_name`*.
`-h`	Display help for this command

NOTE: A direct boot profile name can be 12 characters long at most, and spaces are not allowed. If the direct boot profile name already exists, it will be updated with the new values passed as arguments to this command.

If client IP address information is missing in the direct boot profile, *DHCP* is used by default.

Only IPv4 is supported.

Use quotations ("") around the *`boot_file`* and *`optional_data`* fields.

Examples

- To display settings in the direct boot profile:

  `Shell>` **`dbprofile`**

  or

  `Shell>` **`dbprofile -dn test`**

- To create and modify settings in a direct boot profile:

  Shell> **`dbprofile -dn test -dhcp -sip 192.168.0.1 \`**
  `-m 255.255.255.0`

  Shell> **`dbprofile -dn test -b "tmp/nparsys1/loader.efi" \`**
  `-od "optional data"`

  where *`optional data`* is the actual data you want to pass to the boot file.

- To copy a direct boot profile:

  `Shell>` **`dbprofile cp test profile`**

- To remove a direct boot profile:

  `Shell>` **`dbprofile rm test`**

The lanboot Command

This section provides information for the `lanboot` command. This command is part of the *EFI* Shell and is accessible from the system console when the system is in an active state but has not booted an operating system. For more information, see the *nPartition Administrator's Guide* or the documentation for your particular system, available at the HP Business Support Center:

http://www.hp.com/bizsupport

Syntax

`lanboot` [`select`] [`-od "`*`optional_data`*`"`] | [`-dn` *`name`*]

Options and Operands

`select`	Select from a list of available LAN devices.
`-od "`*`optional_data`*`"`	Pass optional command line arguments to the boot file.
`-dn` *`name`*	Specify the direct boot profile name to use for the LAN boot.

NOTE: Use `select` when no default LAN device is provided so a list of available LAN devices is provided for user selection.

Use `-od` to specify optional data to be passed to the boot file.

Use `-dn` to specify a direct boot profile to be used to handle the LAN boot. Use the `dbprofile` command to create and manage direct boot profiles.

Examples

- To LAN *boot* from a boot server that has been previously setup:

  ```
  Shell> lanboot
  ```

  or

  ```
  Shell> lanboot select
  ```

- To specify optional data to be passed to the boot file that will be loaded from the boot server:

  ```
  Shell> lanboot -od "optional data"
  ```

  or

  ```
  Shell> lanboot select -od "optional data"
  ```

  where *optional data* is the actual data you want to pass to the boot file.

- To perform a direct LAN boot using a previously created direct boot profile:

  ```
  Shell> lanboot -dn profile1
  ```

  or

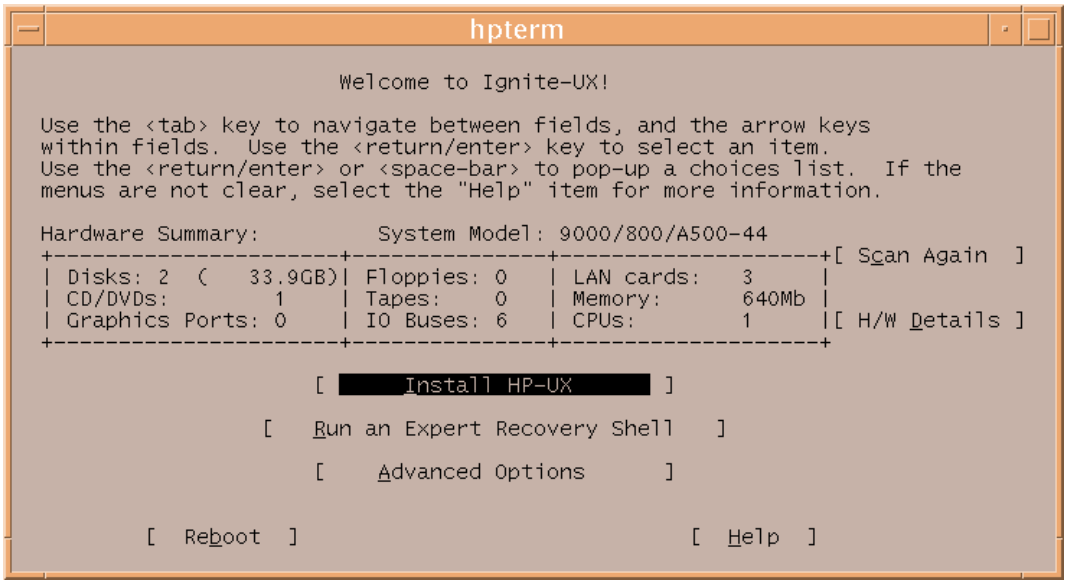
  ```
  Shell> lanboot select -dn profile1
  ```

After booting the system, see "Installing HP-UX From the Client Console" for information on configuring the HP-UX installation from the client console.

Installing HP-UX From the Client Console

Once you have successfully booted the *client* system from the *Ignite-UX server* using one of the previously discussed methods, the following screen will appear on the client console.

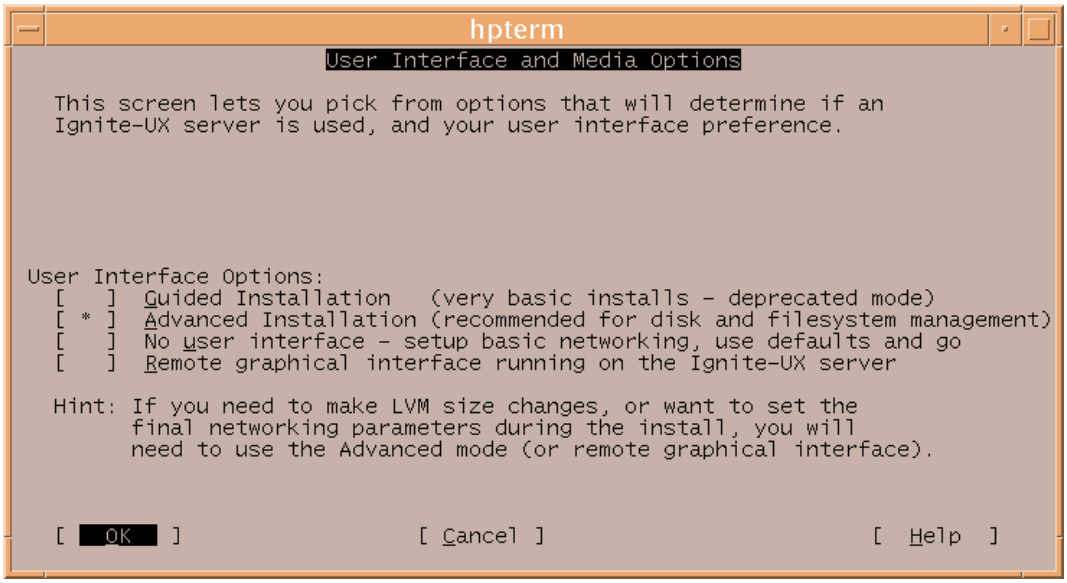
Figure 9-3 Welcome to Ignite-UX!



Select **Install HP-UX**. (Move between menu buttons with the **Tab** key, and select by pressing **Enter**.)

TIP: As an alternative to using the **Tab** key to navigate, simply type the shortcut key (the underlined letter) of the item you wish to select. For instance, press **B** to reboot.

Figure 9-4 User Interface and Media Options



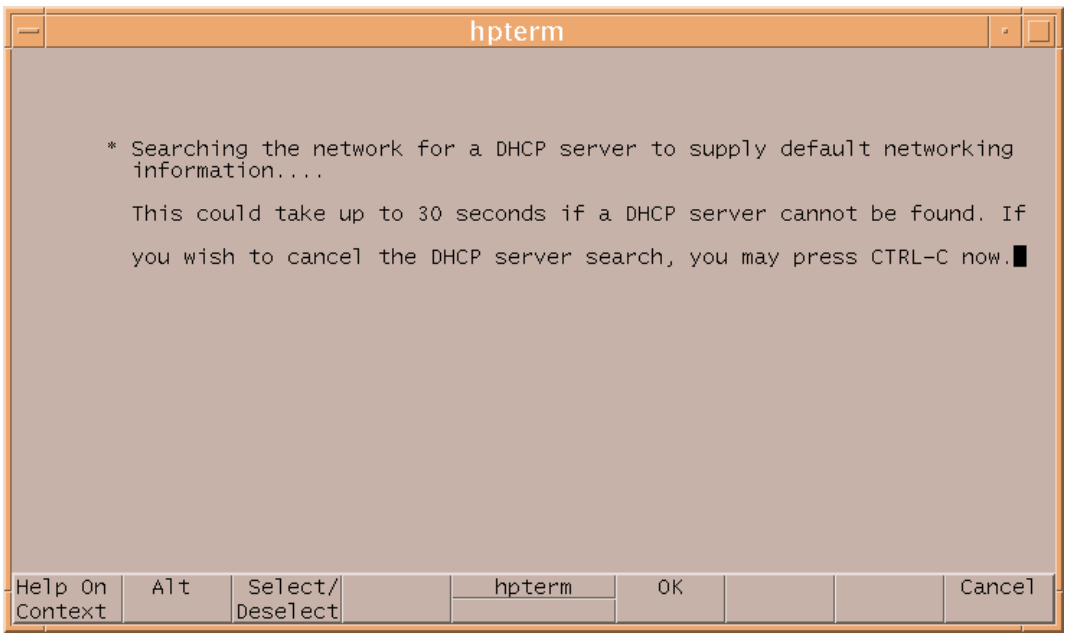
Leave the selection for User Interface Options at the default: **Advanced Installation**.

TIP: Similar to the previous screen, you can use shortcut keys (indicated with an underline) to make a selection.

Selecting **Cancel** will return you to the Welcome to Ignite-UX! screen.

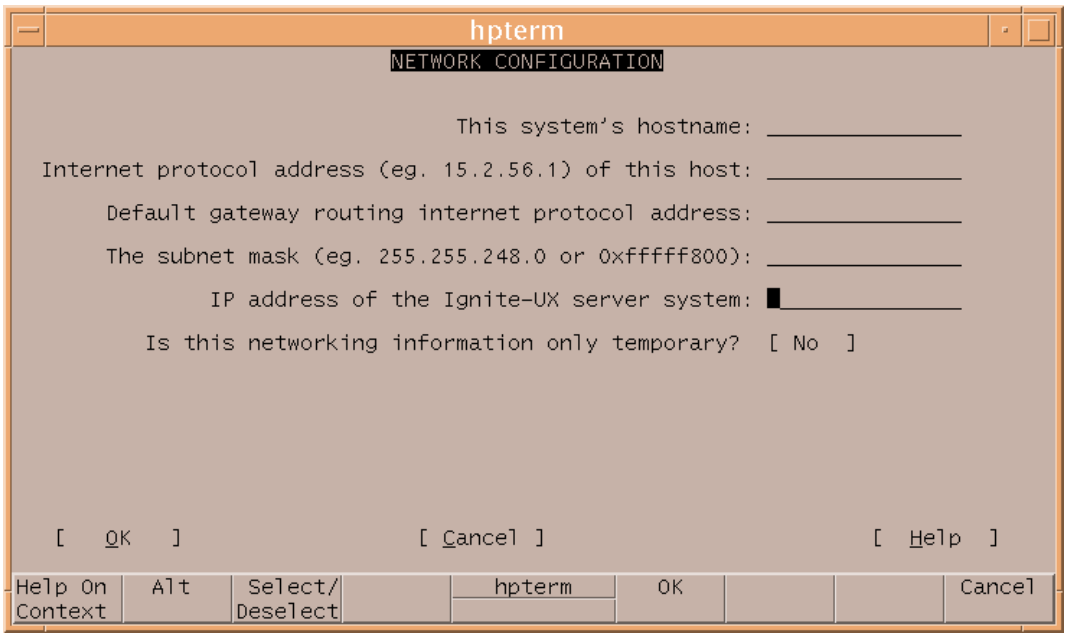
For a network *boot*, Ignite-UX will then look for networking information.

Figure 9-5 Searching for a DHCP Server



If no *DHCP* server is found, the following NETWORK CONFIGURATION screen is displayed.

Figure 9-6 NETWORK CONFIGURATION with no DHCP



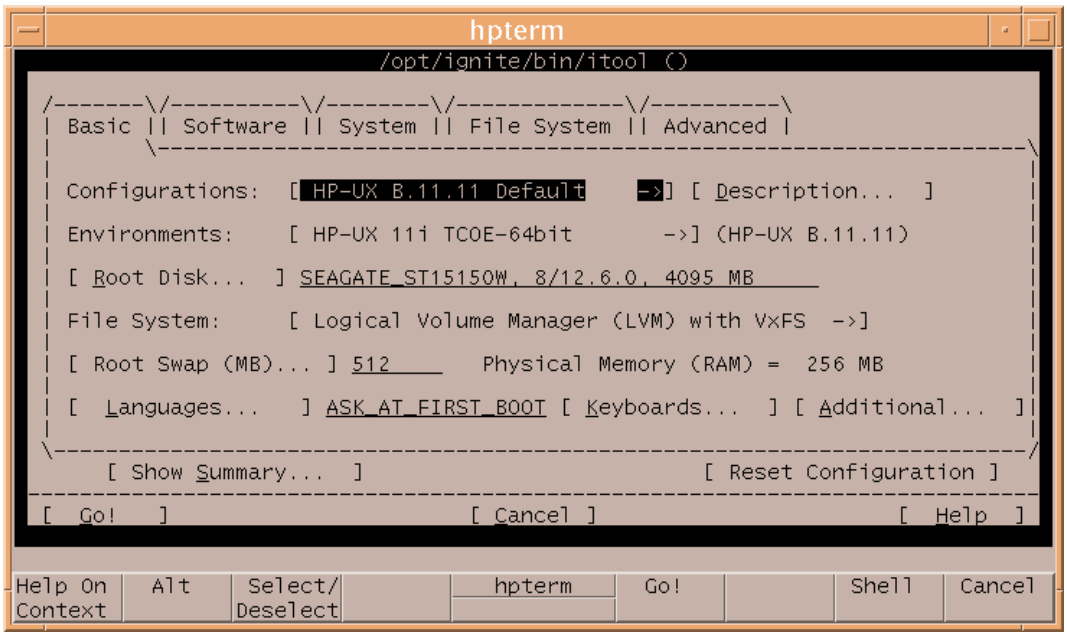
If a DHCP server is found, the NETWORK CONFIGURATION screen will contain the client's hostname and IP address. If there is networking configuration information on the server, the parameters regarding contacting the server will be filled-in as well.

Set the NETWORK CONFIGURATION parameters so the *client* system can contact the *Ignite-UX server*. When the parameters are set, select **OK**.

The server will then be contacted for its installation configuration information for the client. (A dialog box will be displayed on the server with information that this client was found.) The client configuration information is then displayed in the terminal user interface (TUI) version of the client installation configuration interface (`itool`).

NOTE: The NETWORK CONFIGURATION screen has input fields. Typing a shortcut key, such as **O** for **OK**, while the cursor is positioned at an input field will result in an **O** being entered in the field instead of selecting the **OK** button.

Figure 9-7 The TUI Version of `itool`



The TUI `itool` has the same functionality as the GUI `itool` that runs on the Ignite-UX server. Navigate the TUI using the **Tab** to move around and by pressing **Enter** to make selections. See "The Client Installation Configuration Interface" (page 134) for detailed information on using the client installation interface to configure a client's installation.

TIP: Shortcut keys also work in the `itool` TUI, but similar to the NETWORK CONFIGURATION screen, you cannot use them when positioned at an input field. Using shortcuts can significantly reduce the time required to navigate the TUI `itool`.

When the installation has been configured, select **Go!** to start the installation.

Managing Speed and Duplexing of LAN Interfaces Executing Network Boots

The default behavior of the *kernel* at *boot* up is to automatically negotiate the speed and duplex settings of all LAN interfaces that support this feature. This default behavior is later modified in HP-UX by startup scripts that can set any speed and duplex rather than allow the LAN interface to automatically negotiate these settings.

When you use an *install kernel* with Ignite-UX to install or recover a system, there are no startup scripts. If a Fast or Gigabit Ethernet LAN interface fails to autonegotiate, it defaults to 100 Half-Duplex regardless of the speed and duplex settings at the other end. Mismatched speed and duplex settings can cause serious performance and reliability issues when using Ignite-UX.

You can adjust the speed and duplex settings when using Ignite-UX with the `_hp_lanadmin_args` variable. The LAN interface Ignite-UX uses must be brought up before the *Ignite-UX server* can be contacted so this keyword serves no purpose if it is located in the normal *configuration file*s used by Ignite-UX. In view of this, `_hp_lanadmin_args` must be configured in the *install file system* on the server that it boots from. If the client boots using a boot

helper, `_hp_lanadmin_args` must be configured on the *boot helper system* not the Ignite-UX server.

NOTE: In a default setting, modern switches and hubs will autonegotiate the speed and duplex settings with your PA-RISC or Itanium-based system. However, in some cases you may want to set the speed and duplex settings to certain values for your environment. For example, the switch may always fail to autonegotiate with your system. If you do not use autonegotiation on the switch or hub that your system is connected to, you must provide the correct speed and duplex settings for Ignite-UX using `_hp_lanadmin_args`. Mismatched speed and duplex settings may cause significant performance issues or failures during recovery.

Almost all Fast and Gigabit Ethernet interfaces support setting the speed and duplex using the `_hp_lanadmin_args` variable for the `-X` option. Some old Fast Ethernet interfaces might not support full-duplex, so consult the product documentation for any network interfaces you encounter problems with to confirm the speed and duplex settings that each supports.

You can still take advantage of the other options available using the `lanadmin` command. For more information, see the examples in *instl_adm*(4) regarding setting the MTU (`-M`) and Duplex (`-S`) options of `_hp_lanadmin_args`.

IMPORTANT: Due to a problem in Ignite-UX in versions prior to B.5.2.*x*, some parts of the recovery may run with the wrong speed and/or duplex on the LAN interface when the system first reboots and until the final reboot. If this proves to be a problem in your environment, you should ensure that you have installed Ignite-UX version B.5.2.*x* or later.

Examples

Following are two examples of how to set the duplex in differing situations.

Setting 100 Full Duplex

If you want all Fast Ethernet LAN interfaces to run at 100 Full Duplex, this would cause some immediate problems for Ignite-UX because of the duplex mismatches created. To correct the mismatches, you would add the following code to the *install file system* to extract, update, and replace the current configuration in the install file system:

```
instl_adm -d > /tmp/cfg.tmp
```

Edit the file:

```
vi /tmp/cfg.tmp
```

Add the following clause:

```
( lan[].driver == "btlan" )
{
   _hp_lanadmin_args="-X 100FD"
}
```

Add this *configuration clause* to *[W|V|I]INSTALLFS*:

```
instl_adm -f /tmp/cfg.tmp
```

You can then review the current contents of the configuration in the *install file system* with `instl_adm -d`. The output is similar to this example:

```
( lan[].driver == "btlan" )
{
   _hp_lanadmin_args="-X 100FD"
}
env_vars += "TZ=EST-10EDT"
```

Setting Mixed Interface Types

In more complex environments, particularly those with mixed interface types, it is possible to be more complex in setting `_hp_lanadmin_args`. In this example, a default is set for `_hp_lanadmin_args` and a specific setting is defined for Gigabit Ethernet interfaces:

```
( lan[].driver == "btlan" | lan[].driver == "gelan" )
{
   _hp_lanadmin_args="-X 100FD"
} else {
   ( lan.driver == "igelan" )
   {
      _hp_lanadmin_args="-X 1000FD"
   }
}
```

This ensures that if the driver used to control the LAN interface being used is `btlan` or `gelan` it is changed to 100 Full Duplex, and if the driver is `igelan` the interface is set to 1000 Full Duplex.

Alternatively, if you want all Fast and Gigabit Ethernet interfaces to set to 100 Full Duplex you could use the following:

```
( lan[].driver == "btlan" | lan[].driver ~ "gelan" )
{
   _hp_lanadmin_args="-X 100FD"
}
```

The ~ (tilde) operator performs an extended regular expression match. You should exercise care with extended regular expressions to prevent false matches. For more information regarding regular expressions, see *regexp*(5).

The `grep` command can be used from the command line to test extended regular expressions against different input. For more information, see *grep*(1).

10 Booting and Installing HP-UX on Clients Using the Server

This chapter discusses the steps for the *installation* of HP-UX on *client* systems from an *Ignite-UX server*.

See the *HP-UX Installation and Update Guide* available from **http://www.hp.com/go/hpux-core-docs** for instructions on how to install HP-UX from the Operating Environment DVD media.

Methods of Installing Client Systems

Ignite-UX enables you to install client systems from the Ignite-UX GUI, as explained in this chapter, or automatically, as explained in Chapter 13. You can also install clients from a remote system by using `bootsys`, or install the client locally using an operating system obtained from an *Ignite-UX server*, as explained in Chapter 9.

Each installation method requires a configuration (config) file, as explained in Chapter 12. The configuration can include any supported HP-UX 11i operating system, plus any required patches and applications.

This chapter describes installing from the Ignite-UX server, either using the Ignite-UX GUI or remotely using the `bootsys` command.

Begin your installation process with "Preparing the Client for Installation " (page 107).

CAUTION: Any data on the client disks that are used for installation, including the operating system, are removed entirely as part of this installation process.

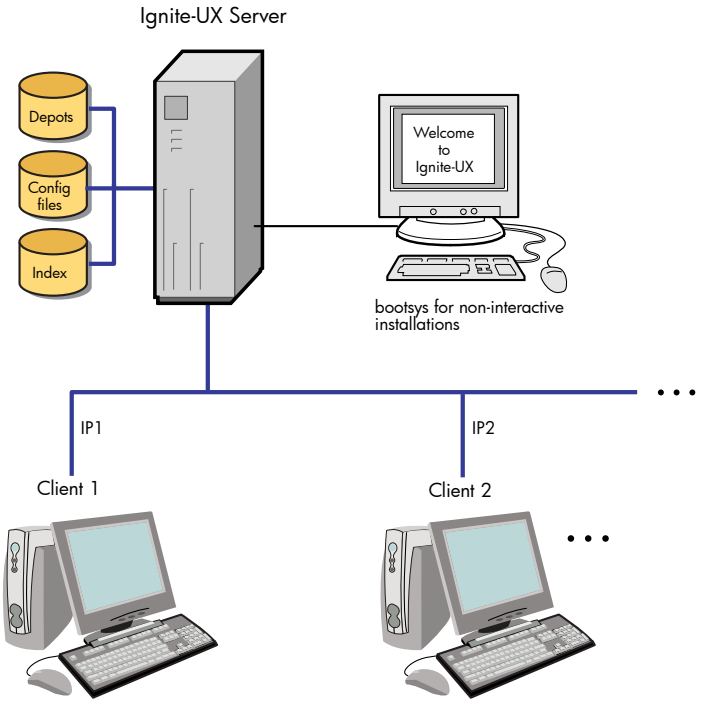
IMPORTANT: During HP-UX 11i v3 installation and recovery, connected Active/Passive devices will cause long delays (one hour or more) or may cause a system to hang. Similarly, connecting an Active/Passive device before installing the Active/Passive Switch (APSW) plug-in can cause some commands to take a long time. Disconnect any Active/Passive devices connected to your system before installing or recovering HP-UX 11i v3. After installation or recovery, it is important that the APSW plug-in be installed before connecting an Active/Passive device to the system.

Installation Using bootsys

Begin your installation process with “Preparing the Client for Installation ” (page 107).

You can use `bootsys` to start a noninteractive system installation on one or more clients without logging onto the client system, as illustrated in Figure 10-1.

Figure 10-1 Noninteractive Installation Using the `bootsys` Command



You can invoke `bootsys` from the command line interface (CLI), or from the Ignite-UX GUI (see “Installation Using the Ignite-UX GUI” (page 128)).

The `bootsys` command copies the Ignite-UX *kernel* and *file system* from the *Ignite-UX server* to each client and then sets the system *AUTO* file in the *LIF* area of the root disk to automatically *boot* from this kernel at the next system reboot.

For more information, see *bootsys*(1M). Common problems using `bootsys` with Ignite-UX are covered in Appendix A (page 247).

IMPORTANT: The `bootsys` command does not support network interfaces that are grouped using HP Auto Port Aggregation (APA). You must not select an APA group as the interface through which Ignite-UX will communicate with the Ignite-UX server. In these cases select an interface outside the APA group, or break the APA group and use a single ungrouped interface with `bootsys` (this may require reconfiguring the switch the APA group is connected to.)

Examples

This sample command boots the client system from the Ignite-UX server, `boot1`, and waits for installation instructions from this server:

`bootsys -R release -w boot1`

where *`release`* is the release that you want to install on the client. For example, `B.11.23`.

If you have already run an installation session from the Ignite-UX server, issuing `bootsys` without the `-w` option results in automatic installation without further intervention.

To automatically install `client1` using a different IP address than what is currently assigned and without waiting for Ignite-UX server interaction, use this command:

```
bootsys -a client1:10.2.3.45
```

TIP: To prevent a critical client from being inadvertently booted using `bootsys`, create the file, `/.bootsys_block`, on the client. For example, you can create this file with:

```
touch /.bootsys_block
```

Installation Using the Ignite-UX GUI

This section describes how to add *client*s to the *Ignite-UX server* using the GUI, and then how to use the GUI to manage Ignite-UX tasks. The Ignite-UX GUI only runs on an Ignite-UX server. Ignite uses a terminal user interface (TUI) with keyboard navigation when run remotely from a client or other system.

This section does not address configuring the Ignite-UX server for network *boot*. To do that, see Chapter 3 and Chapter 4.

Prepare the Client for Installation

Begin your installation process with "Preparing the Client for Installation " (page 107).

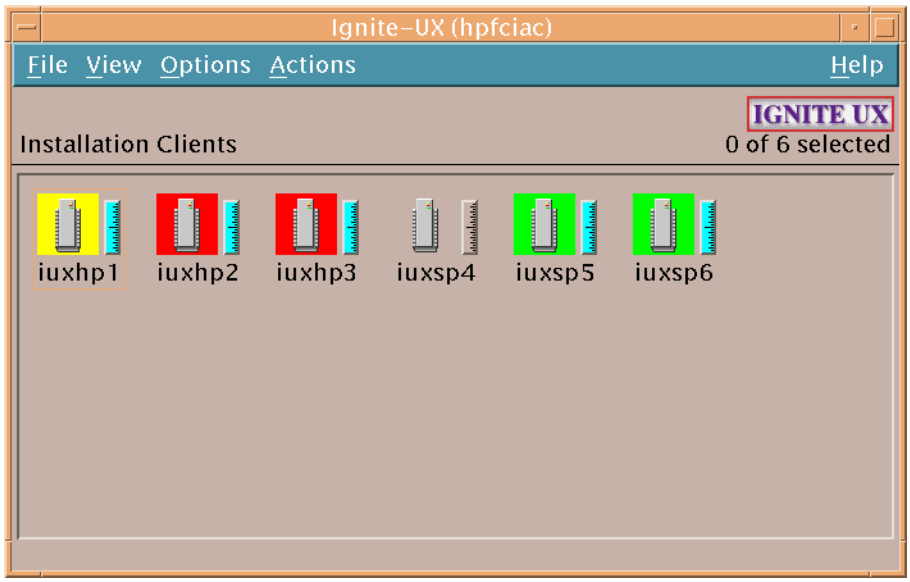
Starting Ignite-UX

Launch Ignite-UX on the server as superuser:

```
ignite
```

The Ignite-UX GUI will be displayed, similar to Figure 10-2.

Figure 10-2 Ignite-UX Graphical User Interface



If this is the first time Ignite-UX has been run on this server, a message will be displayed first, informing you that there are no clients found (Figure 3-1). The Ignite-UX welcome dialog box will also be displayed unless you have previously checked the **Do not show this screen again** checkbox (Figure 3-2).

Adding Clients

A client must already be running HP-UX in order to be added to the *Ignite-UX server*. If the client is not running, see "Booting PA-RISC Clients from the Console " (page 111) or "Booting Itanium-Based Clients using the Network" (page 112).

There are two ways to add a client:

1. *Boot* it from the Ignite-UX server by selecting **Actions->Boot Client**. For detailed information, see "Booting a Client" (page 130).
2. Add it for the purpose of *recovery archive* creation by selecting **Actions->Add New Client for Recovery**. For detailed information, see "Adding Clients for Recovery " (page 227).

After a client is displayed in the GUI workspace, you can:

- Click a client icon to select it for further actions.
- Double-click the client icon to display the Client Status dialog box.
- Right-click to activate the Actions menu for the selected client.

For more about the available Ignite-UX selections, see "Introduction to the Ignite-UX Graphical User Interface" (page 21), or select **Help**.

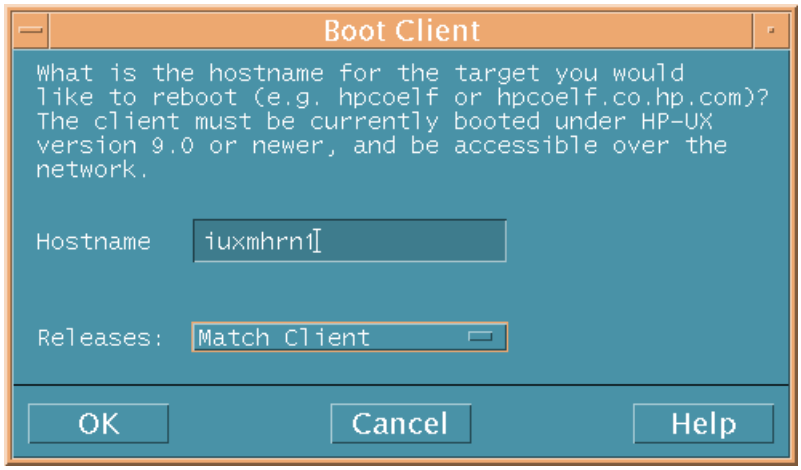
Booting a Client

To add a client to the Ignite-UX server, use the GUI to *boot* a client currently running HP-UX.

To boot a client, follow these steps:

1. From the Ignite-UX workspace, select **Actions->Boot Client**.

Figure 10-3 Boot Client Dialog Box

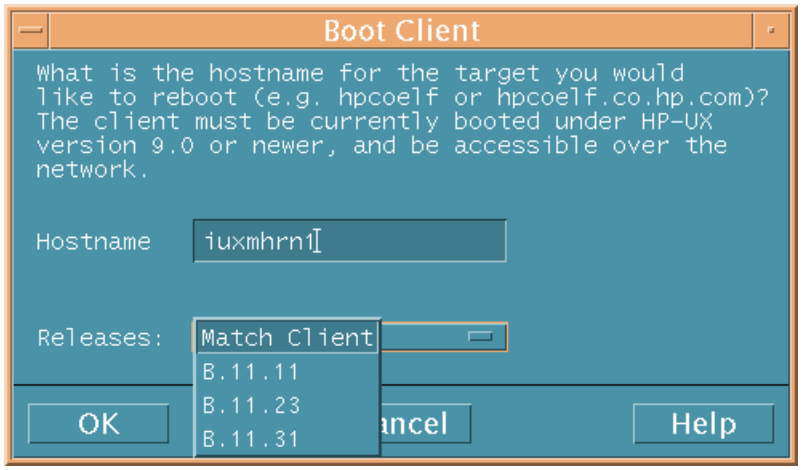


2. Enter the name of the client in the Hostname text box. If a client was selected, its name is in the Hostname box.
3. From the Releases list, select the HP-UX release you want to boot this client with.

   The **Match Client** selection uses the HP-UX version the client was last installed with from the server. If this is the first installation, the server finds out what is currently running on the client, and uses that release.

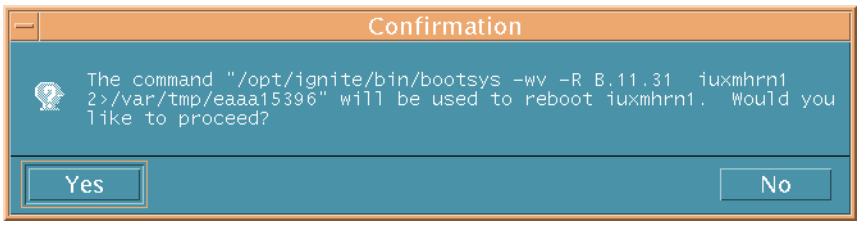
   Make sure you have a configuration on the server for the release you select, otherwise you will not be able to install the client from the server.

Figure 10-4 Select Boot Release



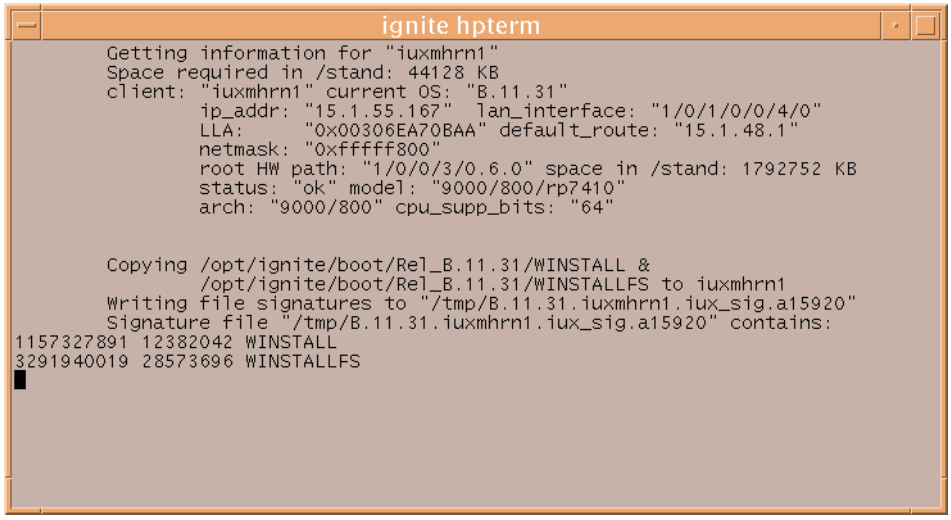
4. Select **OK**.

Figure 10-5 Boot Confirmation Dialog Box



A boot confirmation dialog box is displayed. If you want to boot the client to add it to the Ignite-UX server, select **Yes**; otherwise select **No**.

Figure 10-6 Boot Process Terminal Window



A terminal window is displayed detailing the remote shell (`remsh`) communication and booting progress of the client, including any errors that may occur. If an error occurs, you are advised of the `bootsys` command that can be used from the Ignite-UX server to boot the client manually.

You might have the connection to the client refused. In that case you will have to manually restart the client system. When restarting from the client console, the terminal window might be too small, requiring you to resize the window to make it larger.

Ignite-UX will report a successful reboot and that the client is found and added to the server.

Figure 10-7 Successful Boot

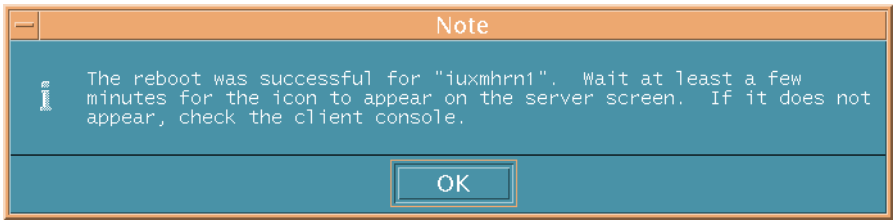
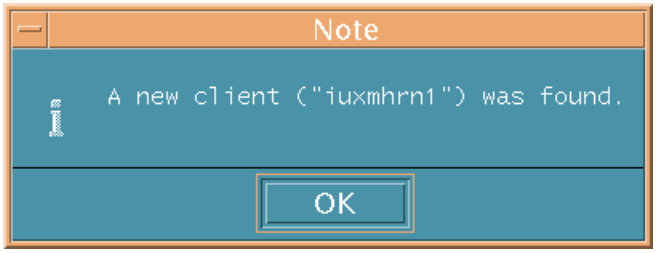
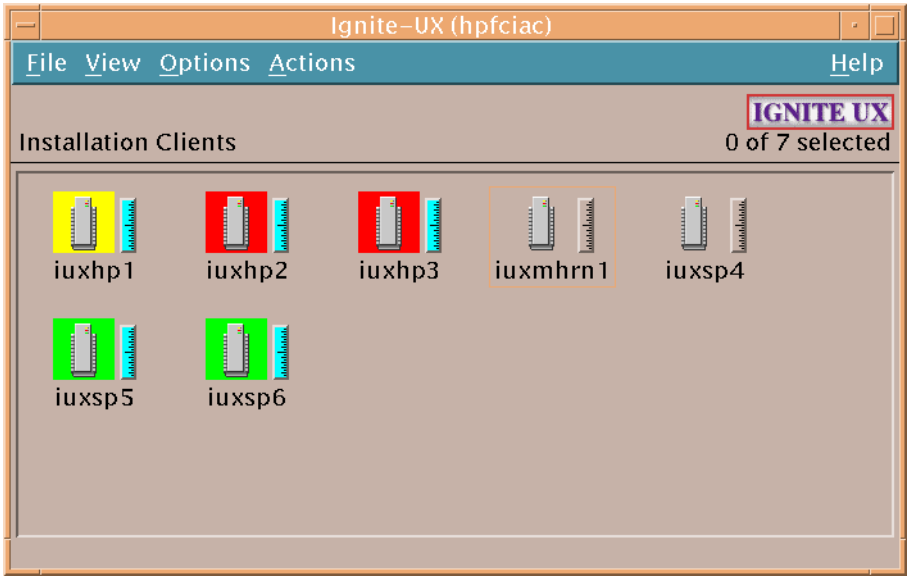


Figure 10-8 New Client Found



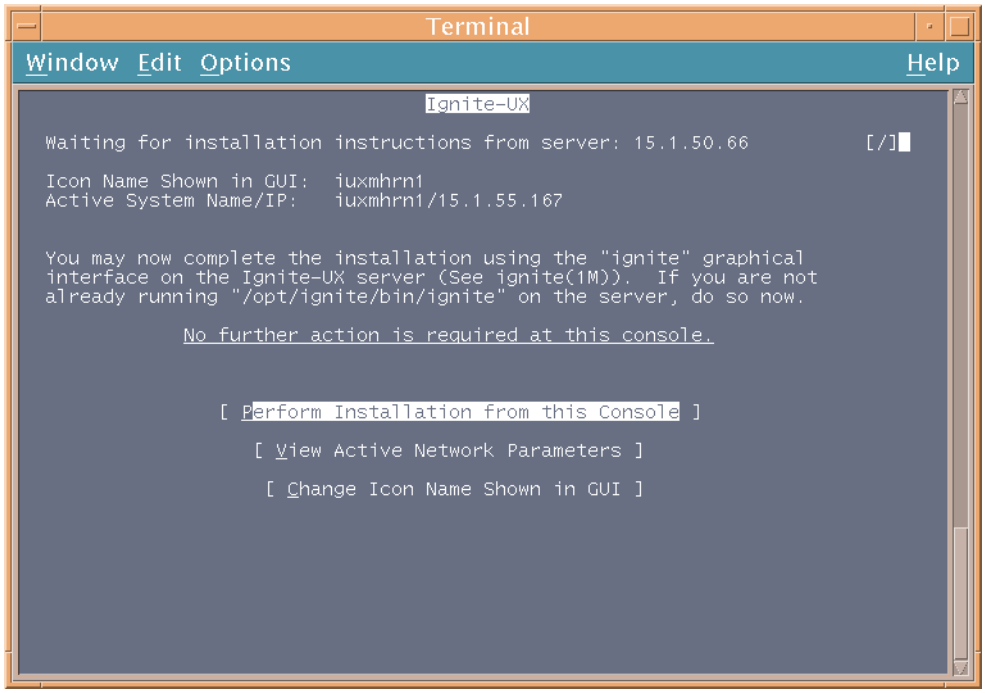
Once the client has successfully booted, its icon is added to the Ignite-UX GUI.

Figure 10-9 New Client Displayed in GUI



On a successful boot, the client console will wait for action from the Ignite-UX server. You can continue with the installation from the Ignite server, or you can install from the client console by selecting **Perform Installation from this Console**, answering **y** to the prompt "Are you sure you want to switch to run the UI locally? [n]," and then installing as described in "Installing HP-UX From the Client Console" (page 118).

Figure 10-10 Client Console Awaiting Server Action



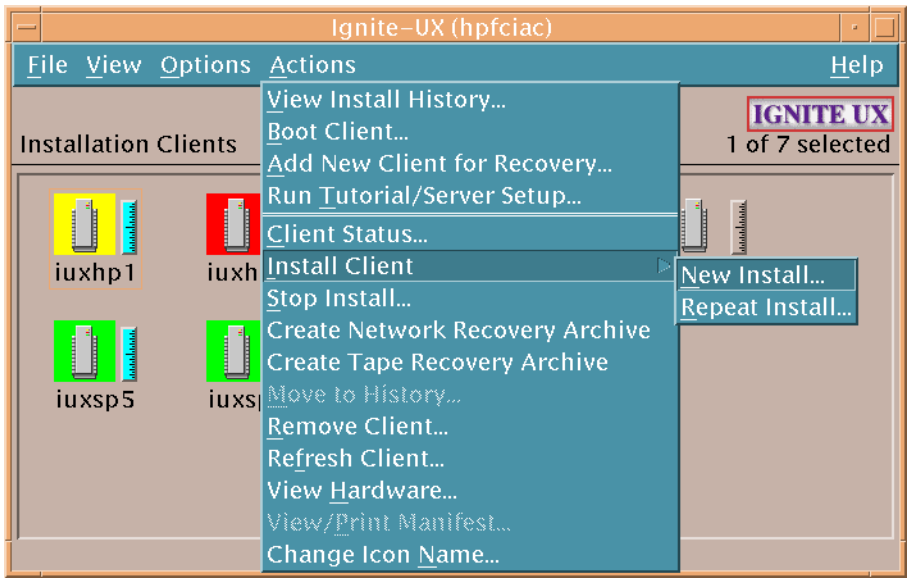
Configuring the Installation

To begin the installation, select a client by clicking on its icon, and then select one of the following from the Actions menu:

— **Install Client->New Install...** to install a new client as described in "New Installation" (page 133).

— **Install Client–> Repeat Install...** to use the configuration of a previously installed client as described in "Repeat an Installation" (page 162).

Figure 10-11 Ignite-UX Actions Menu



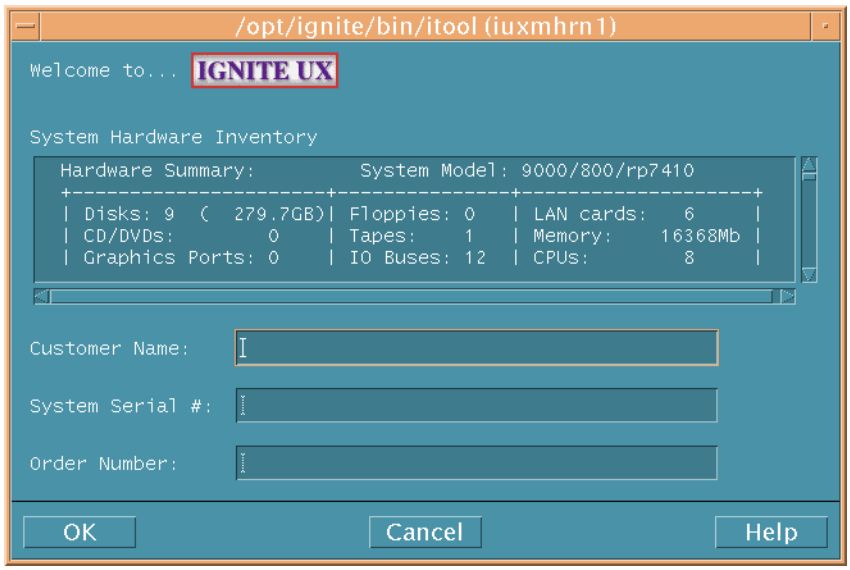
New Installation

The new installation procedure follows a linear path until the client installation configuration interface is displayed . At that point, the amount of configuring done is at your discretion.

Initializing the Installation

The system hardware inventory of the target system is displayed when a new installation is initiated. Check to make sure the hardware connected to the client is present and displayed information is correct. Notice that for Ignite servers running HP-UX 11i v3, all paths to each device and the WWID are displayed. This information is available by scrolling down in the System Hardware Inventory on the GUI, or by clicking the **[H/W Details]** button on the TUI.

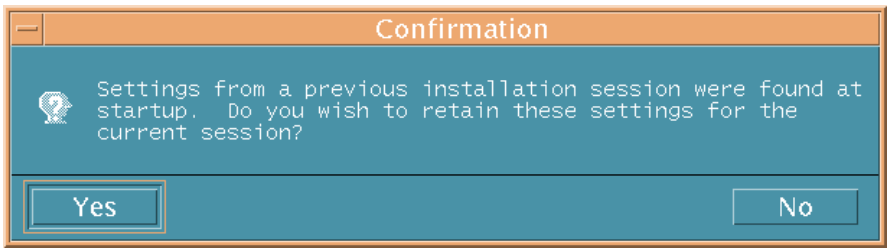
Figure 10-12 System Hardware Inventory



Select **OK** after checking the hardware inventory and optionally entering information.

If you previously installed this client, you are asked if you want to use the last installation configuration used by the Ignite-UX server, as shown in Figure 10-13.

Figure 10-13 Configuration Settings Confirmation Dialog Box



Respond **Yes** to repopulate all GUI fields with the configuration used in the previous session; respond **No** to create an entirely new configuration beginning with the defaults determined by Ignite-UX.

IMPORTANT: To avoid an installation failure, ensure that the `/var/opt/ignite/clients/` directory and its subdirectories are owned by `bin:bin` so the client's configuration information can be written to its *config file*.

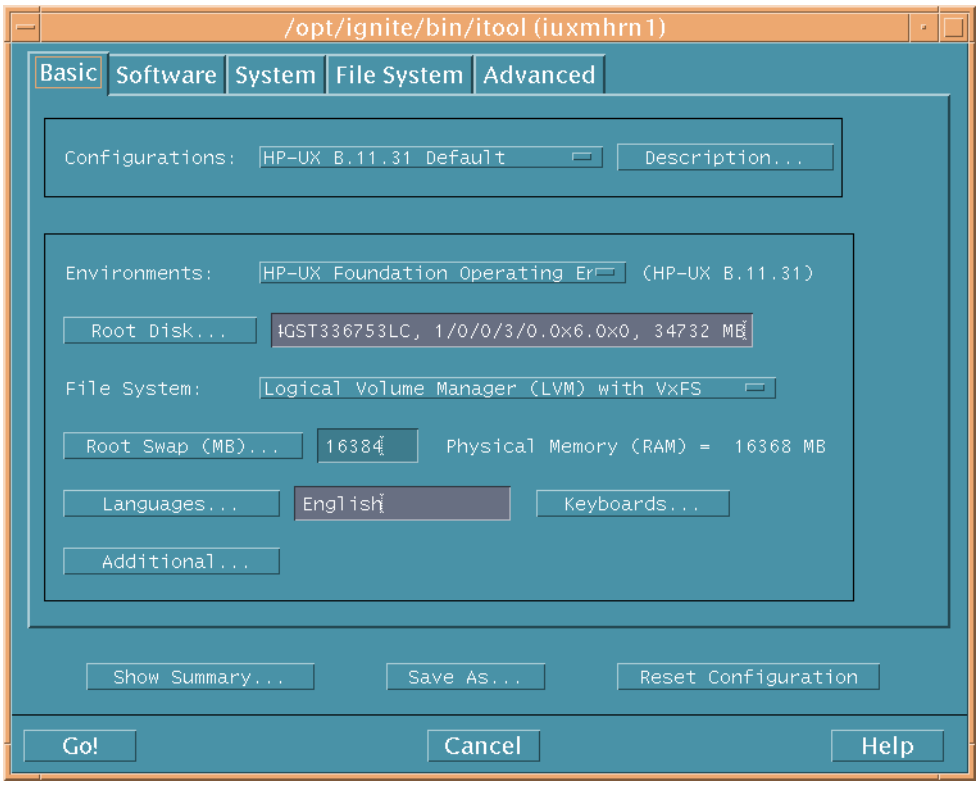
The client installation configuration user interface is then presented (Figure 10-14).

The Client Installation Configuration Interface

Not all fields in the configuration interface require attention. In fact, a quick installation can be launched by selecting the default configuration for your version of HP-UX and then selecting **Go!**.

Modify the fields of the configuration interface at your discretion. Navigate between the tabs by clicking on the tab name. At a minimum, you may want to define the root password by using the **Set Root Password...** button on the System tab, and set the *DNS* information located via the **Network Services...** button on the System tab.

Figure 10-14 Ignite-UX Client Installation Configuration Tabs



The **Show Summary**, **Save As, Reset Configuration, Go!**, **Cancel**, and **Help** buttons are available from all tabs with the same functionality.

Table 10-1 Functions Available From All Tabs

Button	Functionality
Show Summary	Displays the new installation HP-UX release, software, basic *disk layout*, and hardware inventory for the target client.
Save As	Saves the configuration for future use. Not available with the TUI. Load saved configurations from the Configurations... button on the Basic tab.
Reset Configuration	Populates all GUI fields with default settings.
Go!	Starts the installation. You will be asked to confirm before the installation begins.
Cancel	Quits the session.
Help	Help is available for all GUI fields. Also initiated with the **F1** key.

Basic Tab

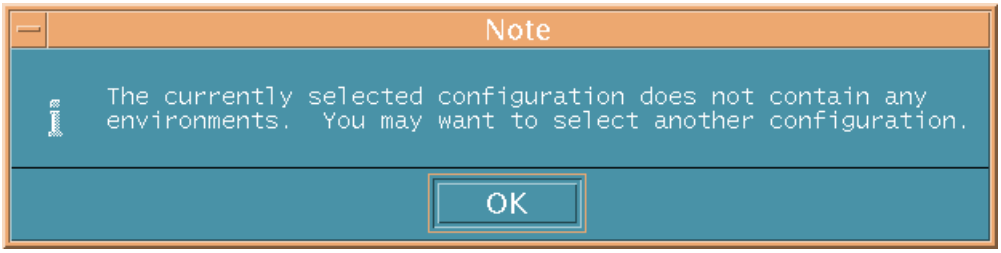
The Basic tab, shown in Figure 10-14, is displayed as default. The elements on this tab are explained in the subsections that follow.

Configurations: List Button

Click the selection list button adjacent to Configurations: and select the configuration you want to use for this installation. The configuration options presented are stored on the Ignite-UX server in the `/var/opt/ignite/data/INDEX` file.

If the selected configuration has no corresponding environments, you will get the message shown below. Select another configuration for installation. If you booted the client with an HP-UX version that is not configured on the server (Figure 10-4), you will not be able to install the client with that version of HP-UX using this Ignite-UX server.

Figure 10-15 No Environments Note



Descriptions... Button

Click this button to view more information about the selected configuration.

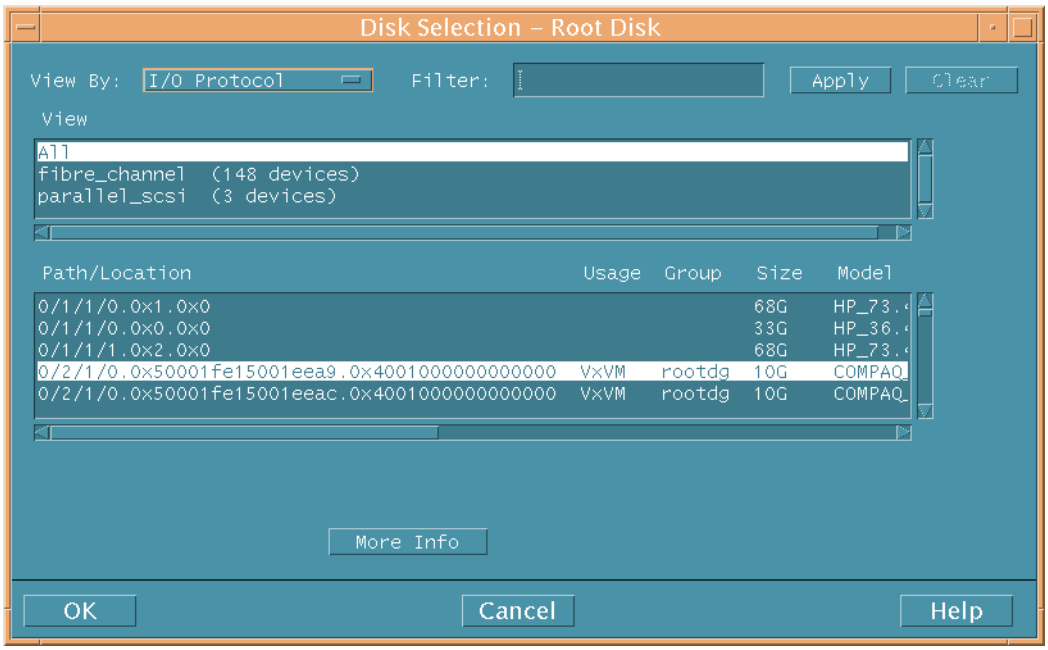
Environments: List Button

Click the selection list button adjacent to Environments: and select the operating system or HP-UX 11i OE that you want to install from the list. This may include 64-bit or 32-bit operating system versions. The choices and defaults depend on the releases available on the Ignite-UX server. *Golden image*s may also be available if any have been created and their configurations have been added to the server's `/var/opt/ignite/data/INDEX` file.

Root disk... Button

Click this button to determine the current root disk or to change the root disk.

Figure 10-16 Disk Selection – Root Disk Dialog Box



Select a disk from the list of available disks, then click **OK**. For information on supported devices, see "Supported Peripherals " (page 30).

The Disk Selection – Root Disk dialog box has a View By: pulldown and a Filter: text box that allows you to control the display of disks.

The selection in the View By: pulldown controls the display in the View window. The selection in the View window controls the display in the Path/Location window. Available selections in the View By: pulldown are:

- **Adaptor**

  Displays the available Host Bus Adaptors (HBAs) in the View window. The Path/Location window displays all the paths to all disks attached to the *HBA* currently selected in the View window.

- **Disks/Paths**

  Displays one path for each disk/LUN in the View window. The Path/Location window displays all the paths to the disk currently selected in the View window.

- **I/O Protocol**

  Displays the available protocols in the View window. The Path/Location window displays all the paths to all disks of the protocol type currently selected in the View window.

- **Size**

  Displays the available disk sizes in the View window. The Path/Location window displays all the paths to all the disks of the size currently selected in the View window.

- **Usage**

  The Path/Location window displays all paths to all disks matching the criterion selected in the View window. Those criteria are:
  - All
  - Disks NOT to be written to in this Ignite session
  - Disks TO be written to in this Ignite session
  - Disks with preexisting data
  - Blank disks and disks with unrecognized data

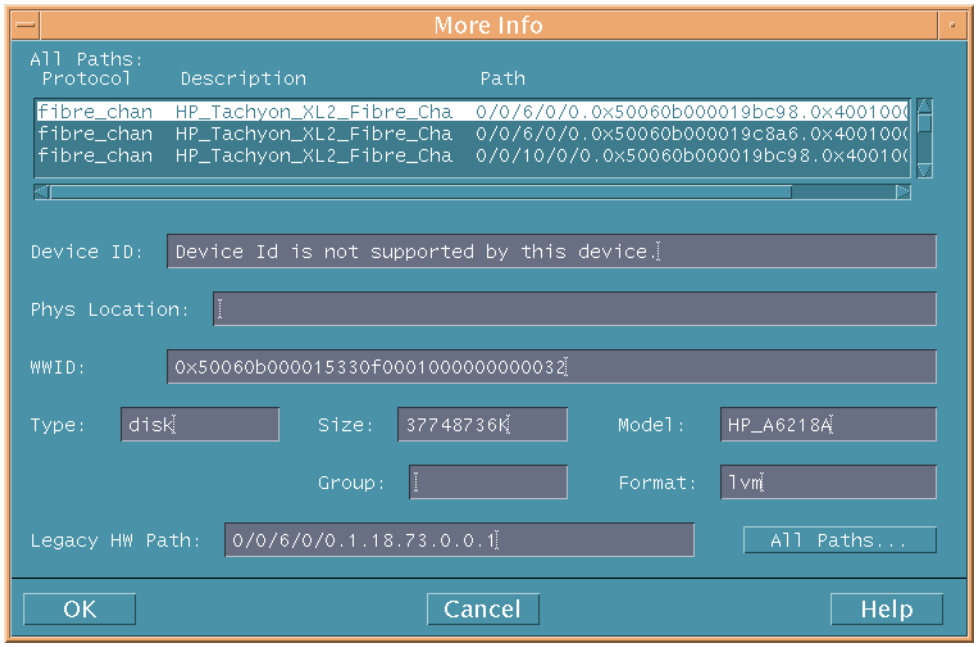
The Filter text box will limit what's displayed in the windows by only displaying results with the filter text in them. Enter the filter text and then select **Apply**. To undo the filter, select **Clear**. The filter acts on the Path/Location window for all Views except Disks/Paths, where it acts on the View window.

The Path/Location window displays a great deal of information, but you have to scroll horizontally to see all the fields. The fields displayed in the Path/Location window are: Path/Location, Usage, Group, Size, Model, WWID, Existing (preexisting data), Legacy Path, and Device ID. The Filter is able to screen all these fields.

The Disk Selection – Root Disk dialog box displays every path for each disk, therefore disks with multiple paths are listed multiple times. To get a concise listing of all the paths for a single device, use the Disks/Paths view or click the **More Info** button. Note that devices blocked from Ignite-UX start-up inventory will not be listed.

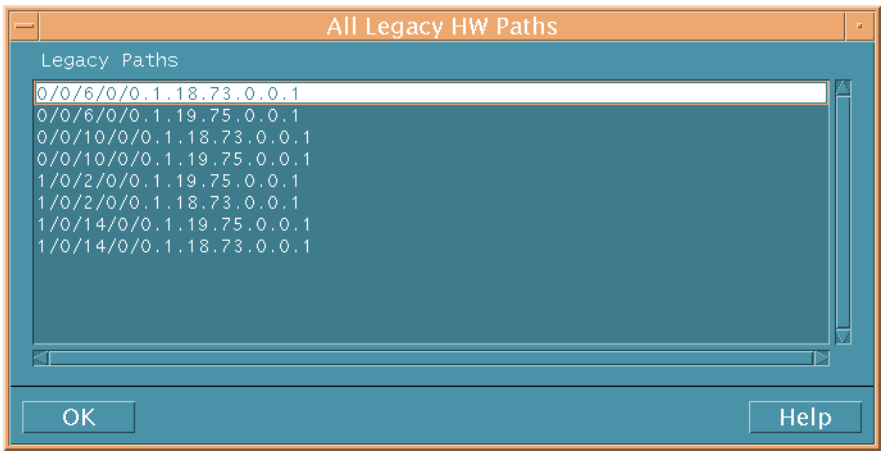
In addition to listing all the paths for a particular device, the **More Info** button provides detailed information, such as Device ID, and WWID.

Figure 10-17 More Info Dialog Box



The Legacy HW Path window is not available on systems running HP-UX 11i v2 and earlier. The Legacy HW Path will update based on the *lunpath hardware path* currently selected in the selection list window. For a concise list of all the *legacy hardware path*s leading to the device, select the **All Paths...** button, only available on systems running HP-UX 11i v3 and later.

Figure 10-18 All Legacy HW Paths Dialog Box



File System: List Button

Click the selection list button adjacent to File System:, and then select from the list of available *file system* architectures.

The list of file system choices that appears is dependent on the hardware architecture of the client. By default the choices are outlined in the following table and descriptions follow.

Table 10-2 File System Default Choices

	PA-RISC	Itanium-Based
Whole Disk with VxFS		Yes
Whole Disk (not LVM) with HFS	Yes	
LVM with HFS	Yes	

Table 10-2 File System Default Choices *(continued)*

	PA-RISC	Itanium-Based
VxVM with VxFS	Yes	Yes
LVM with VxFS	Yes	Yes

For guidelines in planning *file system* layouts and for disk-management strategies, see the *HP-UX System Administrator's Guide* for HP-UX 11i v3, or *Managing Systems and Workgroups: A Guide for HP-UX System Administrators*. Additionally, see the section "File System Tab " (page 154) for detailed information on file system configuration.

- **Whole Disk (Not LVM) with HFS or VxFS -** These selections treat the entire disk as a single unit. This is the option for *hierarchical file system (HFS)*, or if the disk is 2 GB or less on PA-RISC. You can use a 70 GB disk as a single unit on an Itanium-based system.
- **Logical Volume Manager (LVM) with HFS (Hierarchical File System) -** This selection formats single or multiple disk systems to combine the disk space into a single, large disk pool, then allocates volumes as needed. The root volume and the swap must be on the same physical volume and are configured in this manner by Ignite-UX. Use the File System tab (see "File System Tab " (page 154)) to configure the LVM volumes, or use the values that Ignite-UX computes for you. For more information, see *lvm*(7).
- **Veritas Volume Manager (VxVM) with VxFS -** This selection formats single or multiple disk systems to combine the disk space into a single, large disk pool under VxVM, then allocates volumes as needed. The root (/), *boot* (`/stand`), and primary swap volumes must be on the same physical disk and are configured in this manner by Ignite-UX. Use the File System tab (see "File System Tab " (page 154)) to configure the VxVM volumes. VxFS enables you to change file system sizes after installation. With the optional HP OnlineJFS product you can resize, defragment, or make a "snapshot" of a mounted file system.
- **Logical Volume Manager (LVM) with VxFS (Veritas File System) -** This selection formats single or multiple disk systems to combine the disk space into a single, large disk pool, then allocates volumes from this pool as needed. VxFS is the same as the Journaled File System (JFS). With the HP VxFS product, you can resize, defragment, or make a "snapshot" of a mounted file system.

Root Swap (MB)... Button

The amount of root swap space depends on the applications being installed. You can choose to use the default that Ignite-UX computes based on available memory on the client, or you can click **Root Swap (MB)...** and select from the choices that appear in the list. You can also modify the adjacent field directly by typing in the amount you want. The swap is rounded down to a multiple of 4 MB or the LVM extent size.

For information regarding computing swap space, see the *HP-UX System Administrator's Guide* for HP-UX 11i v3, or *Managing Systems and Workgroups: A Guide for HP-UX System Administrators*.

Languages... Button

The languages available with your HP-UX system are listed when you click **Languages...**. The default language, which was set when the GUI was started for the first time, appears in the adjacent field. Select the language you want as the default for the client, if it is other than the current default language. You can select more than one language by double-clicking on each item. To select a range of items in the list, you click and hold the mouse button, drag the pointer down within the dialog box, then click **mark/unmark**.

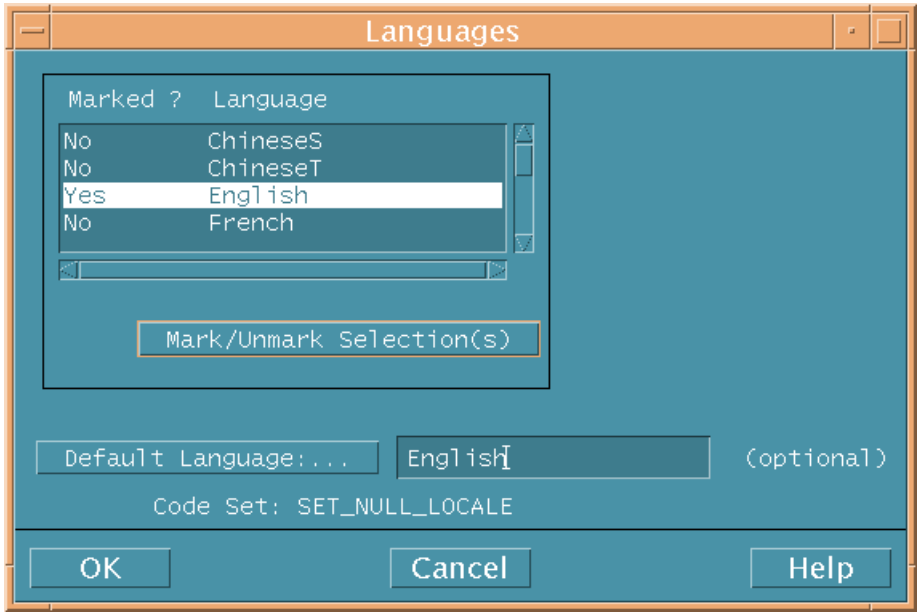
Each language has a corresponding locale (language variant), which describes the system management of a language for doing the following:

- Messaging
- Representing numbers

- Displaying monetary values
- Telling time
- Generating characters
- Sorting text

Click **Default Language...** to see the Default Language Choices (Figure 10-19). They are listed in two columns: Language and Locale. Each language may have more than one way of representing itself on the system. If this is the case, there will be multiple locale entries for the same language.

Figure 10-19 Languages Dialog Box



The locale determines how languages are activated, and are reflected adjacent to Code Set:, as follows:

- **ASK_AT_FIRST_BOOT** enables you to leave the language setting open (unset) until the client system is first booted. At that time, you are prompted. The language setting will be performed as part of the initial system configuration.
- **SET_NULL_LOCALE** creates a `NULL` language environment leaving the locale variables, such as LANG and LC_ALL, unset by default. With no locale set, programs execute without using localized message catalogs, which can increase system performance. All HP-UX programs operate in the `C` locale and messages appear in English.

For information regarding other methods of setting the language environment, see *geocustoms*(1M) and *HP CDE Getting Started Guide*.

Keyboards... Button

Click this button to select the type of keyboard to be used on the client. Otherwise, the default selection is used.

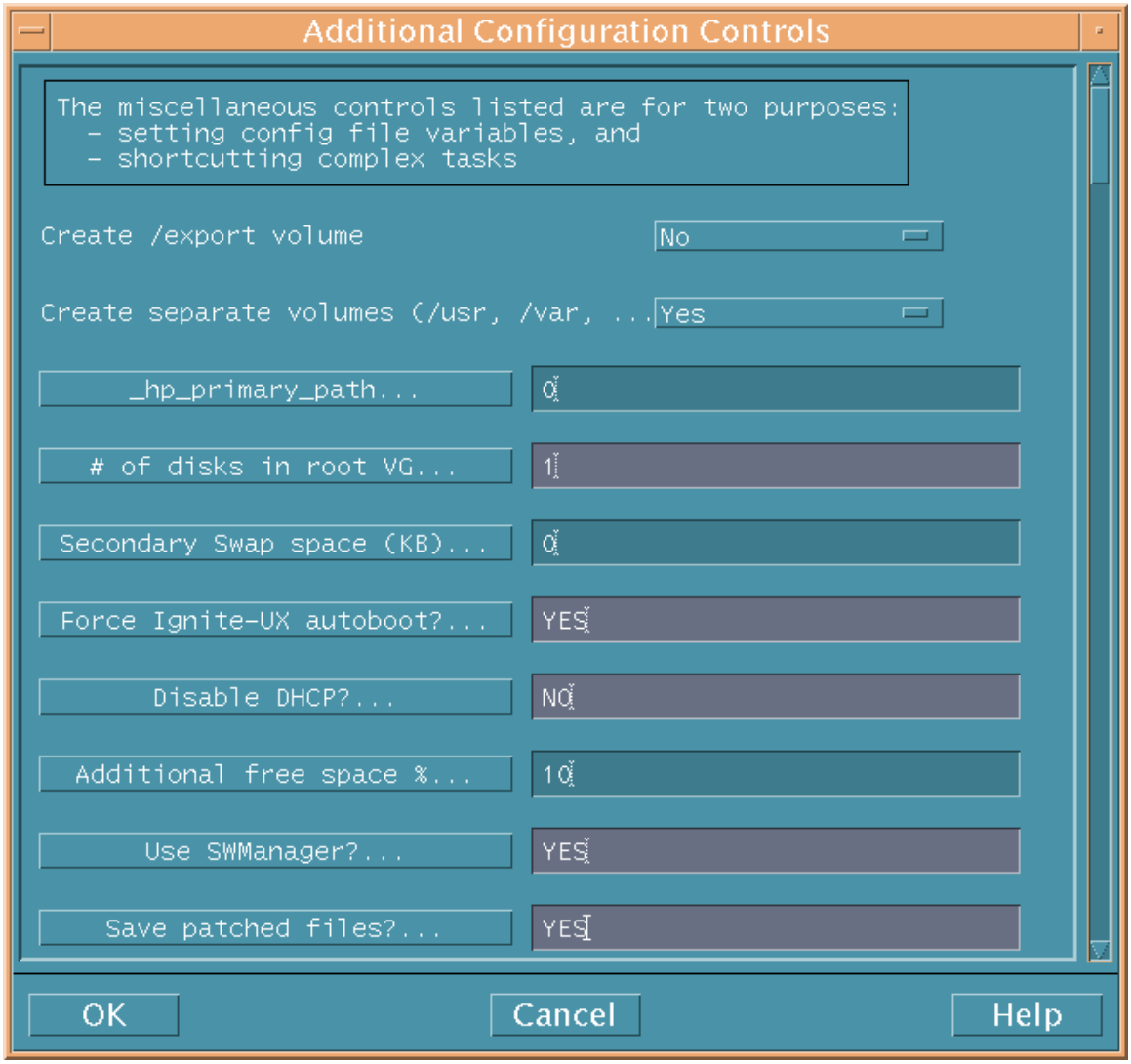
The selected keyboard language name is stored in the `/etc/kbdlang` file. If you have incorrectly set the language, you can quickly recover by removing this file.

Additional... Button

Click **Additional...** to select among certain preconfigured use models and variables from your current *configuration file*s. The buttons available are determined from the variables in your configuration file and enable you to choose a setting for each variable.

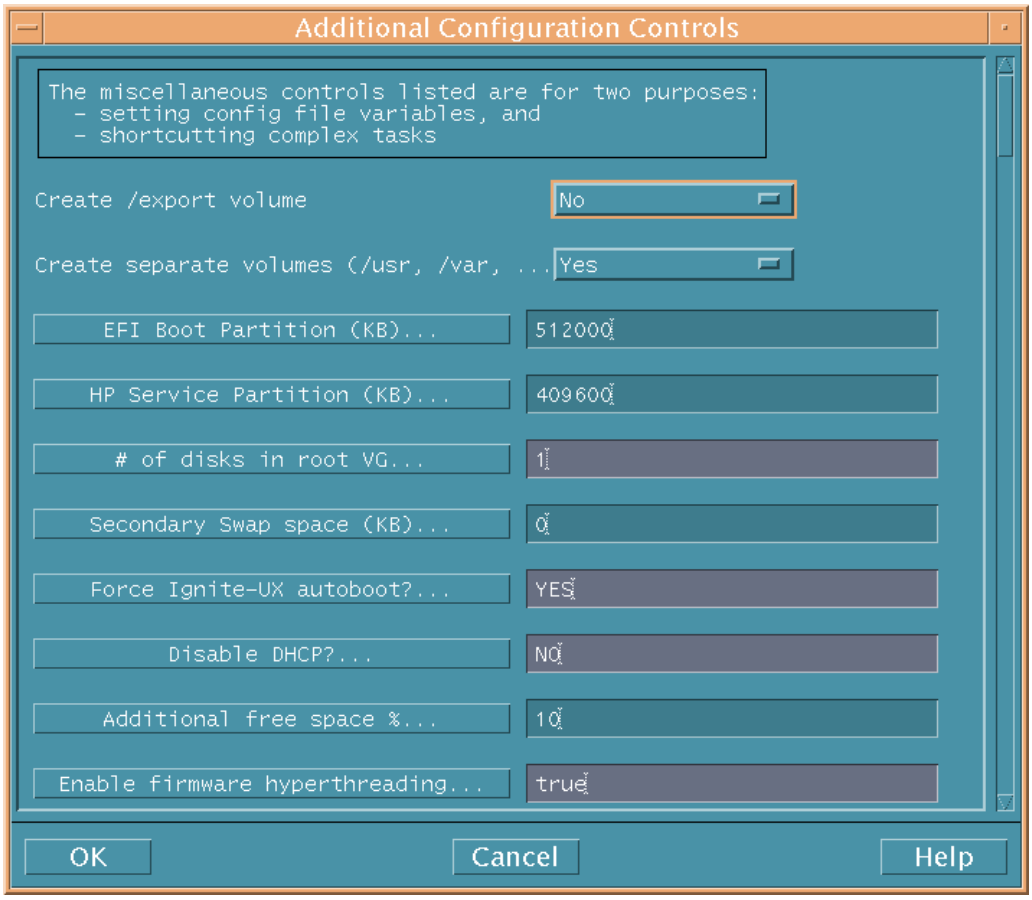
If for example you are using LVM, you will see selections for easily setting up multiple disks, the default LVM version, striping, and *file system* creation. The default value for the **Default root group LVM version...** is 1.0. For more information, see *instl_adm*(4).

Figure 10-20 Additional Configuration Controls Dialog Box



The example Additional Configuration Controls dialog box below illustrates the options for a different system. Systems with firmware hyperthreading have it enabled by default.

Figure 10-21 Additional Configuration Controls Dialog Box 2



Software Tab

Use the Software tab to choose from the applications you configured when you set up your *Ignite-UX server*. You can access a specific *depot* and you can change depot locations. This display does not dynamically update from a newly selected depot. When you choose a new depot, it must be identical in content to the current one. If not, use `make_config` on the Ignite-UX server to configure the new depot prior to invoking Ignite-UX.

An example of what you might see on the Software tab is shown in Figure 10-22 for HP-UX 11i v1 and 11i v2 systems, and Figure 10-23 for HP-UX 11i v3 systems.

Figure 10-22 Software Tab for HP-UX 11i v1 and 11i v2

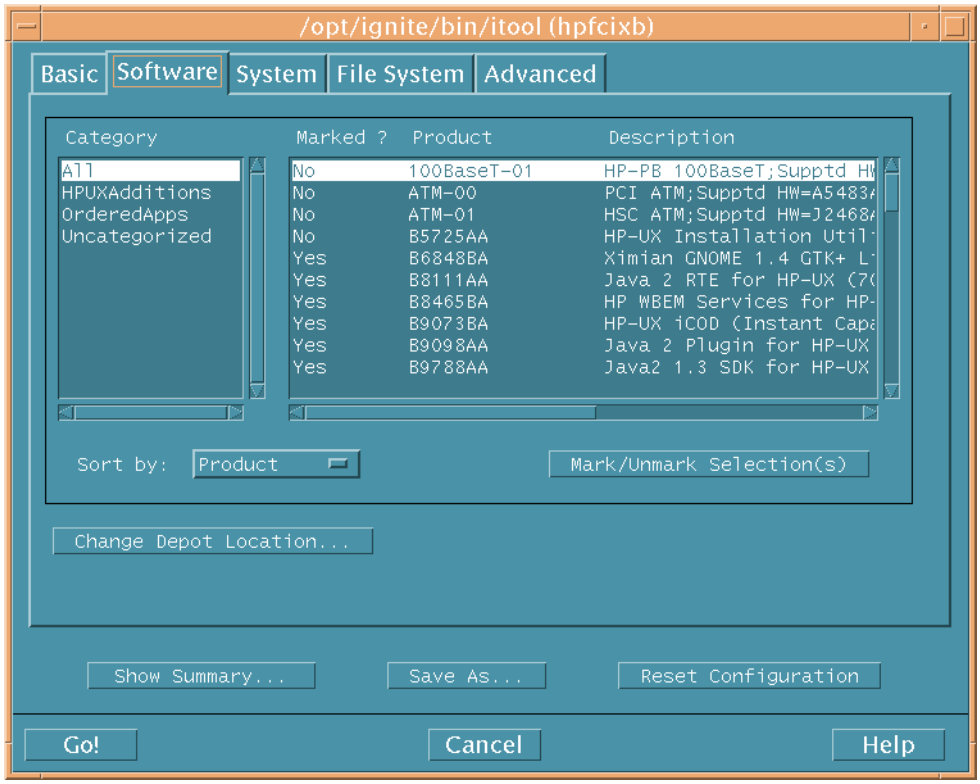
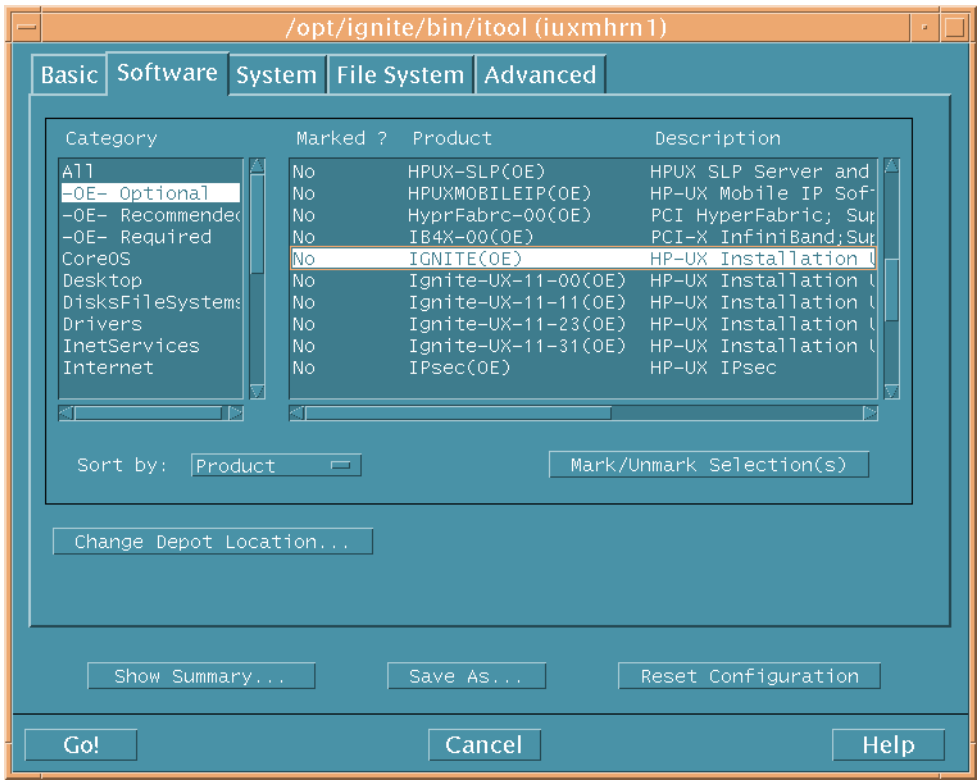


Figure 10-23 Software Tab for HP-UX 11i v3



Category

Select a topical category to display the products that are available. When **All** is selected, all software *bundles* in the *depot* are displayed. Selecting a category other than **All** will shorten the list. Additional categories may be visible if defined by any custom *configuration files*.

Starting with HP-UX 11i v3, software bundles are organized by the categories "optional", "recommended", and "required", instead of "selectable", "*default-installed*", and "*always-installed*". See the *HP-UX Installation and Update Guide* available from **http://www.hp.com/go/hpux-core-docs** for a detailed description of the software bundles and OE structure in your version of HP-UX.

Note that Ignite-UX is in the "selectable" or "optional" category. If you want Ignite installed on the client system, you must select it for installation.

You can select all software in the optional category by marking the "-OE- Optional" product, and all software in the recommended category by marking "-OE- Recommended".

Product

Click on a product in the list to select it. To change the status in the Marked ? column, use the **Mark/Unmark Selection(s)** button, or double-click the product. If you attempt to unmark a product that must be installed, you will get message similar to the following:

Figure 10-24 Software Cannot be Unselected

The Marked ? column status can be:

Required	Selected and cannot be unselected.
Yes	Selected and can be unselected.
No	Unselected and can be selected.
Unavail	Unselected and cannot be selected.

For more information, see *instl_adm*(4).

Your ability to select and deselect installation software is dependent upon what type *bundle* the software is in (for example, "Required") and other software selections you have made. See the *HP-UX Installation and Update Guide* available from **http://www.hp.com/go/hpux-core-docs** for a detailed description of the software bundles and OE structure in your version of HP-UX.

Sort by... List Button

Select a column by which the product listings will be sorted. By default, the listings are sorted by Product. You can also perform a two-level sort. For example, you can sort first by Product and then by Marked to see all the products that are marked Yes listed in alphabetical order, followed by all the products marked No listed in alphabetical order.

Mark/Unmark Selection(s) Button

Select this button to change the Marked ? status of the selected product.

Change Depot Location... Button

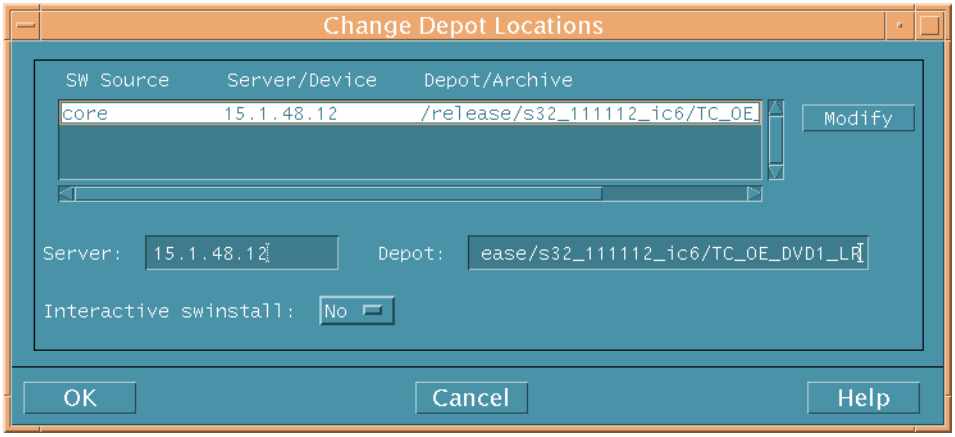
Select this button to change the location of the *depot* from which you are selecting software. Changing the depot location changes the source depot for all products. The following note is displayed when you click this button:

Figure 10-25 Change Depot Location Note

Ignite-UX does not dynamically update from a newly selected depot. When you choose a new depot, it must be identical in content to the current one. If the new depot is not identical to the current one, use `make_config` on the Ignite-UX server to configure the new depot prior to invoking Ignite-UX.

Use the Change Depots Locations dialog box to select a different depot.

Figure 10-26 Change Depots Locations



Use the **Server:** and **Depot:** text boxes to identify the new depot. Click **Modify** to update the depot information within the dialog box. Click **OK** to commit the modifications to the Ignite-UX server.

If you are installing from a *golden image*, the Change Depot Locations dialog box allows you to modify the *archive* server and the archive path.

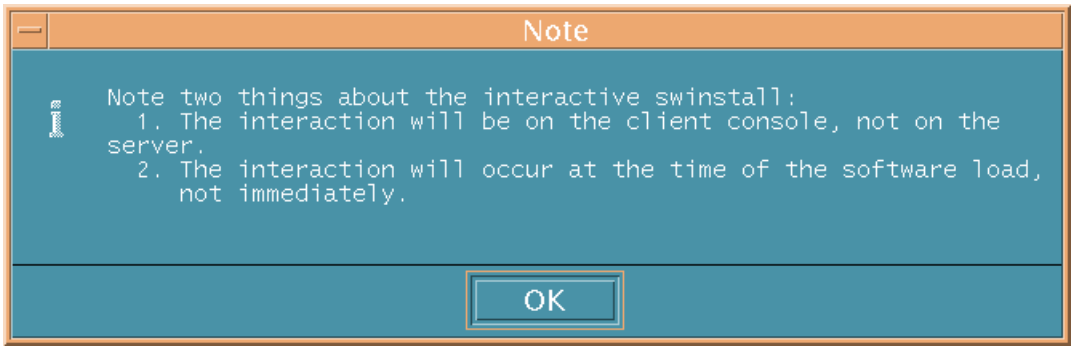
If patches are kept in a separate depot, by default they are installed after the operating system. If there are multiple application depots to be installed, you may need to specify the installation order for the patches in a *configuration file*. For more information see, "Using Configuration Files" (page 177).

IMPORTANT: Inclusion of multiple versions of Veritas Volume Manager from Symantec (VxVM) in the same installation depot, or in separate depots that are used together in a single new installation session, is not supported. Doing so renders it unusable and generates errors when attempting to use the installation depot, or in the case of non-SD depots during the reboot attempt. For more information, see "Considerations When Using Veritas Volume Manager from Symantec" (page 211).

Interactive swinstall An interactive `swinstall` session allows you to select a depot, change installation options, and perform other software installation tasks from the console window

immediately before the software load phase of the installation. After selecting an interactive `swinstall` session, the following message will be displayed.

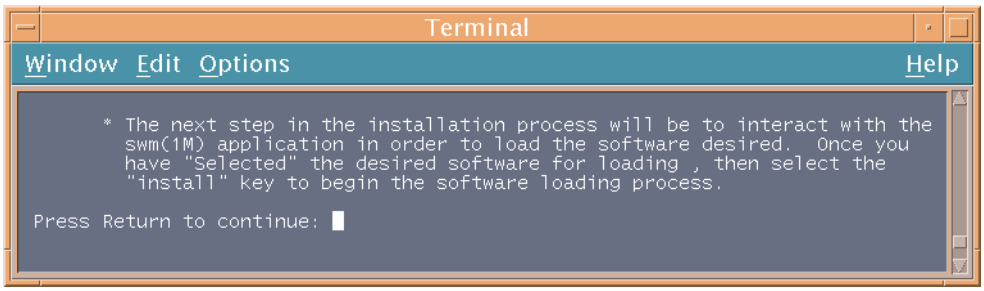
Figure 10-27 Interactive Swinstall Notes



NOTE: For systems running HP-UX 11i v3 and later, `swm` (Software Manager) Will be used instead of `swinstall`. References to `swinstall` by Ignite-UX can mean either `swm` or `swinstall`, depending on the release. On systems running HP-UX 11i v3 and later, see *swm*(1M) for more information.

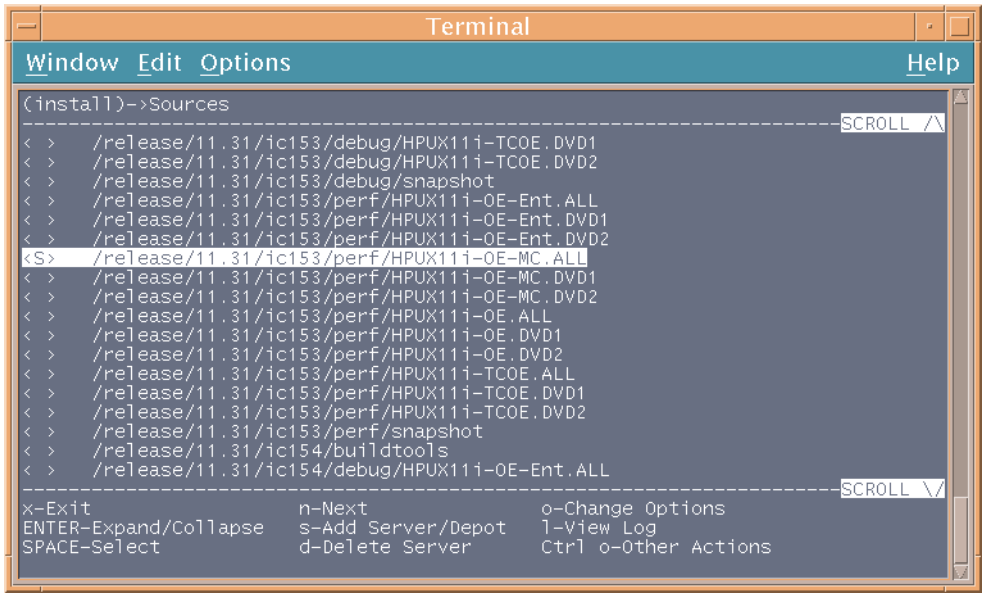
The installation will be interrupted at the start of the software load phase, and the following message will be displayed on the console screen.

Figure 10-28 Entering Interactive Swinstall



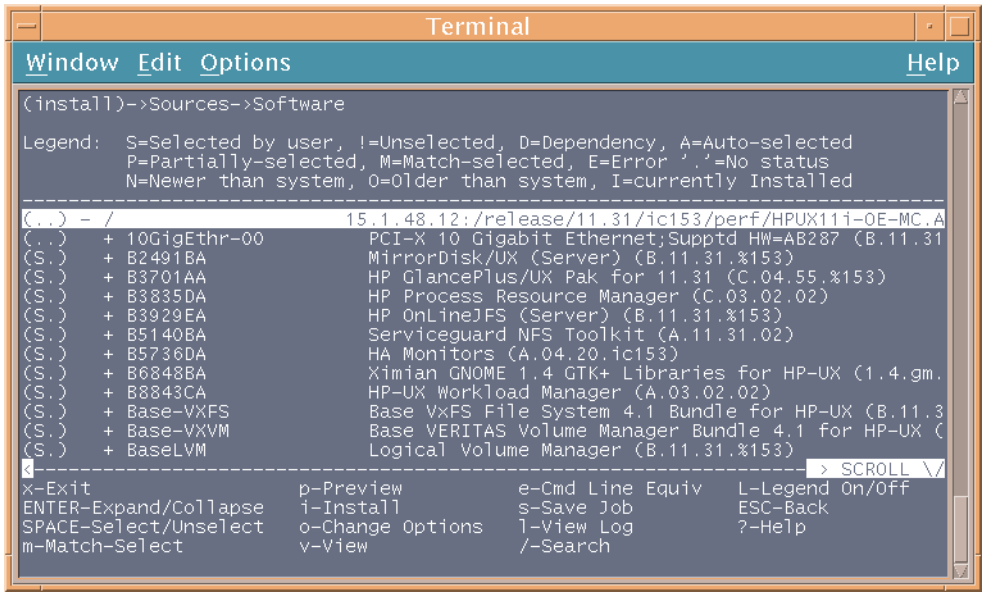
During interactive `swinstall`, you can choose a server, a depot, and individual *bundle*s for installation. Scroll through the list using the arrow keys, and select a new depot for software installation by pressing the space bar. (Navigation help is always displayed at the bottom of the screen.)

Figure 10-29 Interactive Swinstall Depot List



After selecting a depot, press **n** to view the software available within the depot.

Figure 10-30 Interactive Swinstall Bundle List



Select or deselect individual bundles with the space bar. Press **i** to continue the Ignite-UX installation.

If you press **x**, you will receive this message: `Would you like to exit, save or continue the job? [x]/s/c`

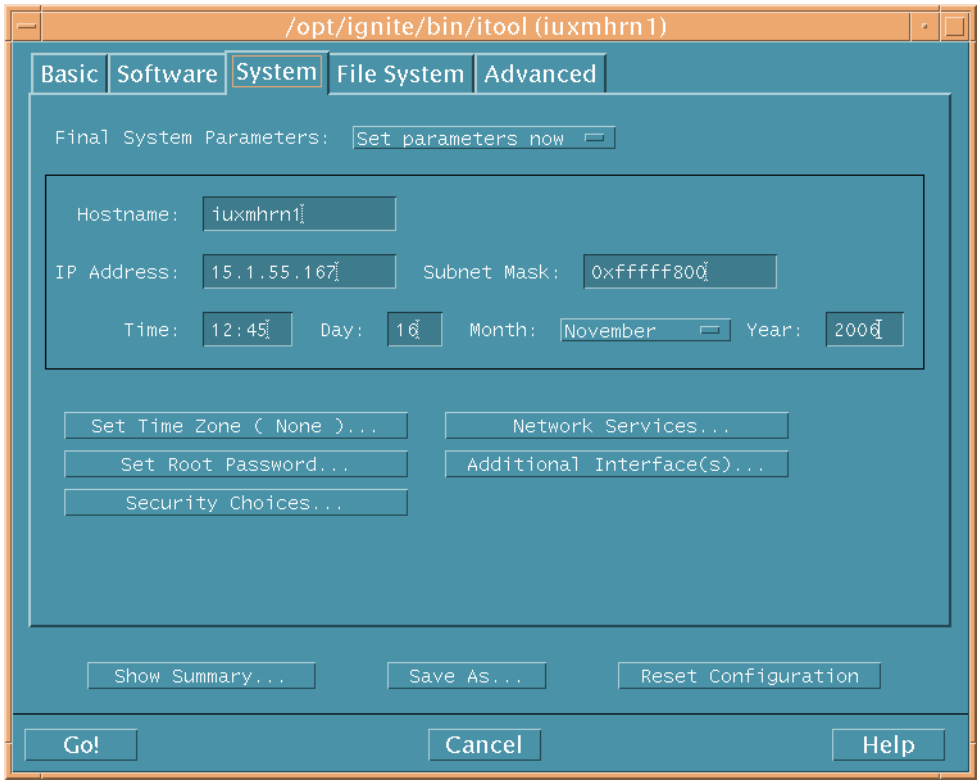
Press **c** to return to the `swinstall` session.

Ignite-UX will check to ensure you have loaded some basic software, but in general, exiting without installing is not recommended.

System Tab

Use the System tab to specify the information unique to this client such as the hostname, IP address, root password, and the time zone.

Figure 10-31 System Tab



Final System Parameters List Button

You can choose to set the System tab parameters now or at the first *boot* of the client. If you select first boot, you do not need to modify any parameter on the System tab now.

DHCP Managed Primary NIC Selection

This check box is used to control whether the primary network interface is configured using information obtained via DHCP (Dynamic Host Configuration Protocol). When this is selected, the IP Address and Subnet Mask fields are not shown.

Hostname Text Box

Your client must have a unique system name. A hostname must fulfill the following conditions:

- It should not contain more than 8 characters, for compatibility with the `uname` command.
- It must contain only letters, numbers, underscore (_), or hyphen (-).
- It must start with a letter. Uppercase letters are not recommended.

For more information, see *hostname*(1).

IP Address Text Box

Use this field to enter the IP address in the form: *n*`.`*n*`.`*n*`.`*n*, where each *n* is a number `0` through `255`.

To determine the IP address of an existing system, use `nslookup`. For example, if the hostname of the client is `test`, enter the following:

```
#nslookup test
Name Server:  server.corp.com
Address:  10.1.48.11

Trying DNS
Non-authoritative answer:
Name:    test.corp.com
```

```
Address:  10.1.50.12
#
```

You can also use `ping` to determine the hostname, the IP address of a system, and if a given IP address is already in use. If you are in doubt, be sure to coordinate IP address allocation with your network administrator.

IMPORTANT: Do not inadvertently use the IP address of the name server. Your network could stop functioning; results are unpredictable.

Subnet Mask: Text Box

This field sets the subnet mask for the client system. The subnet mask is typically provided by your network administrator, and is in the form of an IP address or a corresponding hex number. For example:

`255.255.248.0` *or* `0xfffff800`

or

`255.255.255.0` *or* `0xffffff00`

Time, Day, Month, and Year: Text Boxes

The time and day that is displayed for the client is dependent on the installation method. If you are installing from the server, the time is synchronized with the server; if you are installing from the client, it is set using the client's hardware clock. If necessary, enter information for the Time, Day, and Year fields: For time, use the 24 hour format: *hh*:*mm*.

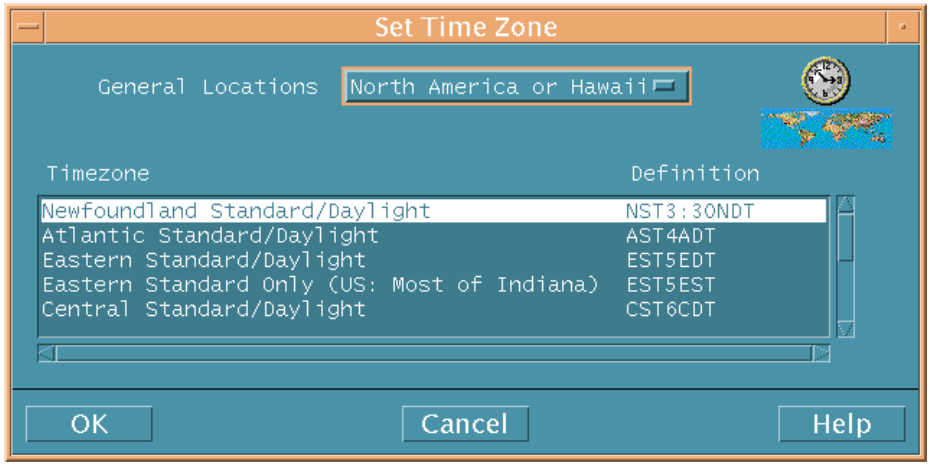
Select the correct month by clicking the adjacent list button, and then selecting from the list. Edit other fields by using the **Backspace** and **Delete** keys.

The time you set is automatically adjusted if you select a different time zone using the **Set Time Zone...** button described below.

Set Time Zone Button

Click this button to display the time zone selections, as illustrated in Figure 10-32. Use the General Locations list button to select the client's location, and then select the time zone. Click **OK** to set the value. The time and date on the System tab are adjusted to reflect the new time zone, and the `TZ` environment variable is set.

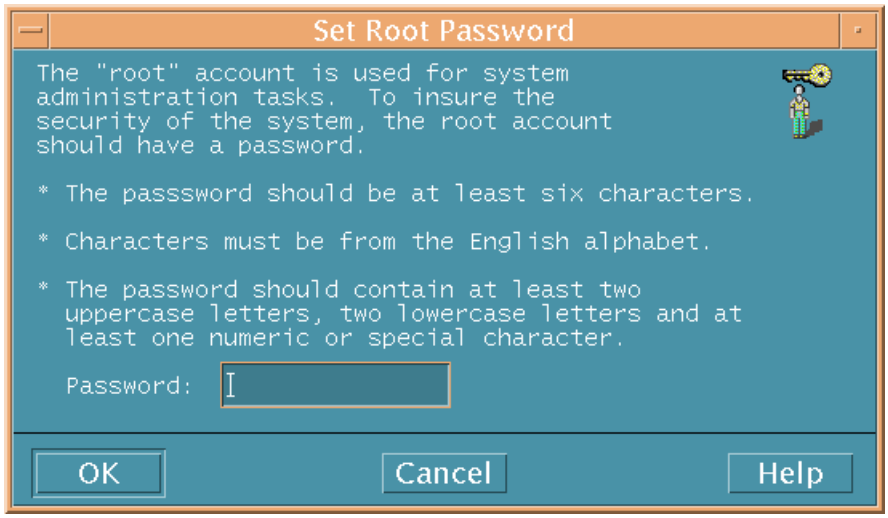
Figure 10-32 Set Time Zone Dialog Box



Set Root Password...

To ensure the security of the client system, create a password for the `root` account by clicking this button. The Set Root Password dialog box is displayed, as shown in Figure 10-33.

Figure 10-33 Set Root Password Dialog Box



To strengthen security, HP recommends that you observe the following requirements when setting a password:

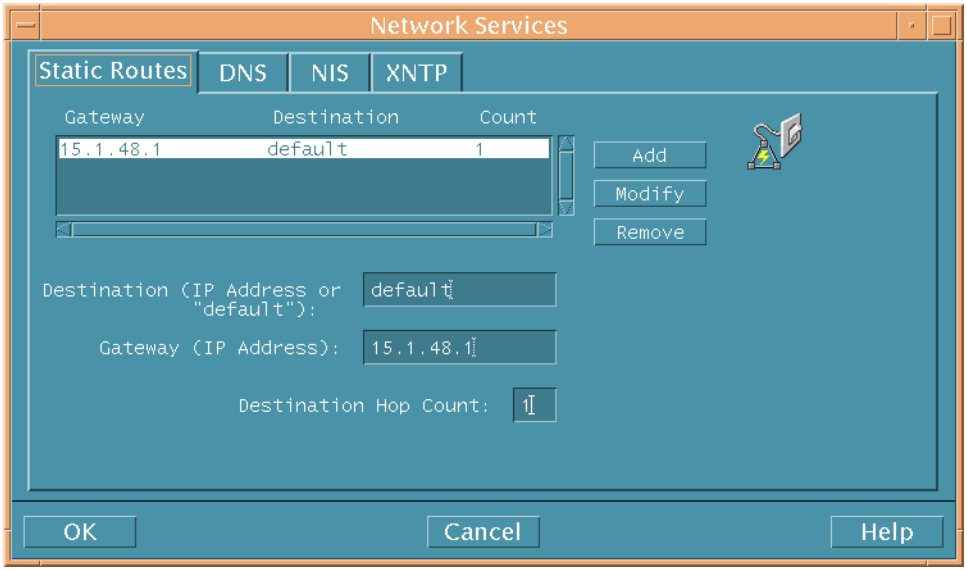
- The password should be eight characters long.
- Characters should be from the English alphabet.
- The password should contain at least two uppercase letters, two lowercase letters, and at least one numeric or special character.
- Avoid the use of the @ special character; if it must be used, you must use the syntax \@.

NOTE: Ignite-UX uses the `crypt` library function to encrypt the entered password. As such, the root password is truncated to eight characters, maximum. See *crypt*(3C) for more information. This limits passwords entered interactively for client recoveries of trusted systems to eight characters, even if a longer one had been in place before the recovery. If the root password is not changed interactively, the old one is retained with its original length.

Network Services... Button

Click this button to configure the network services available to the client.

Figure 10-34 Network Services Tabs

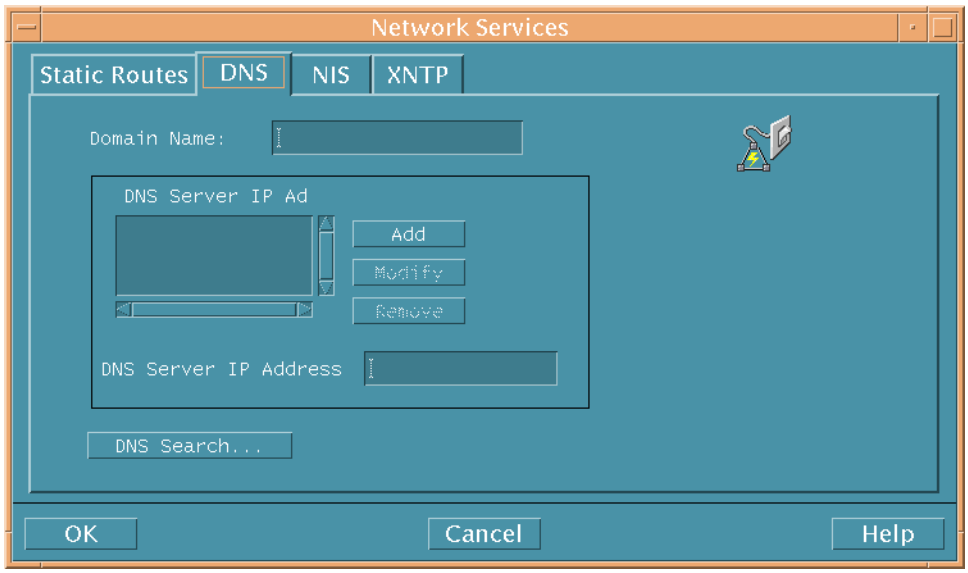


Static Routes. If your network is divided into subnets, you will probably need to specify a *gateway* system to reach other subnets:

- **Destination -** The field has the word "default" or the IP address of the destination network.
- **Gateway -** This is the IP address of the device connecting your network to the remote network, or your own IP, if wildcard routing is used.
- **Destination Hop Count -** If your *gateway* IP address is not the same as the IP address of your clients, this is usually set to `1`. If your gateway IP address is the same as your client IP address, then the destination hop count is `0`.

After you complete the appropriate fields on this tab, click **Add** or **Modify** to save the information. For more information, see *routing*(7).

Figure 10-35 DNS Tab



DNS. On the DNS tab, you can configure the domain name (an extension to the hostname, such as `xx.corp.com`) and the IP address of the domain name server.

Recovery over the network requires the domain name and *DNS* server IP address to be set for the client now, unless the Final System Parameters list button on the System tab is set to **Ask at first boot**.

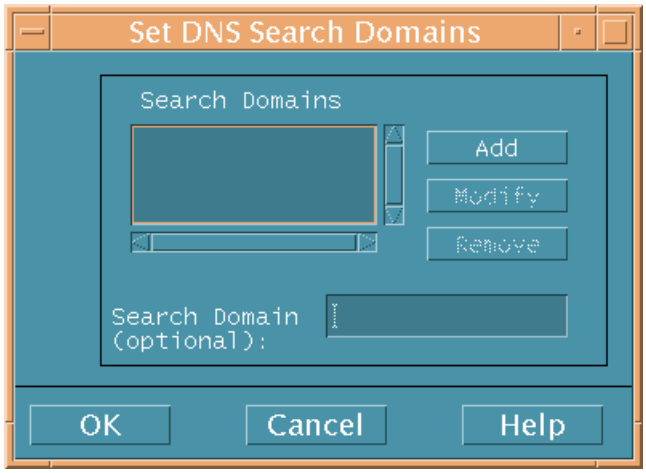
The list of DNS servers by IP address is populated if they are predefined on the Ignite-UX server.

Use the `nslookup` command on a running system to find this information if you need to add a DNS server to this list. Enter the IP address of the name server in the DNS Server IP Address text box.

Use the **Add**, **Modify**, and **Remove** buttons to modify this list directly.

Additionally, you can use the **DNS Search...** button to set optional DNS domains to be searched, as shown in Figure 10-36.

Figure 10-36 Set DNS Search Domains Dialog Box



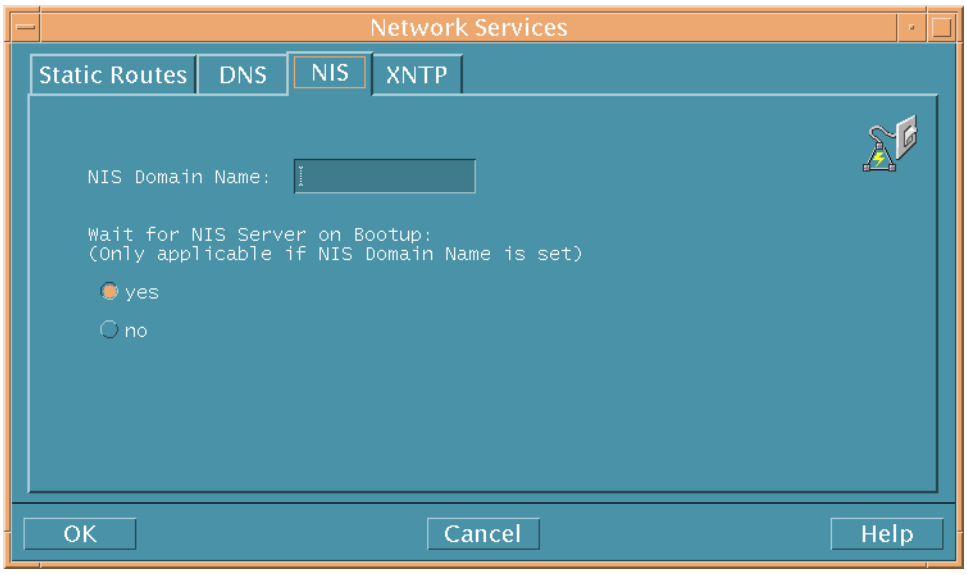
The domains in this search list are used for hostname lookups. Up to six search domain entries are enabled, with a syntax like *xx*.corp.com. For the search to work properly, the first entry in the list should be the local domain for unqualified hostnames.

You can add to or modify the domains to be searched by entering the text in the Search Domain field and clicking the appropriate button, **Add** or **Modify**. To remove a domain, select it from the list and click **Remove**.

NOTE: The domain and search options are mutually exclusive, though both can be specified. If they are both specified, the search option is written out last and takes precedence.

For more information, see *resolver*(4), *Installing and Administering NIS Services* or *HP-UX IP Address and Client Management Administrator's Guide*.

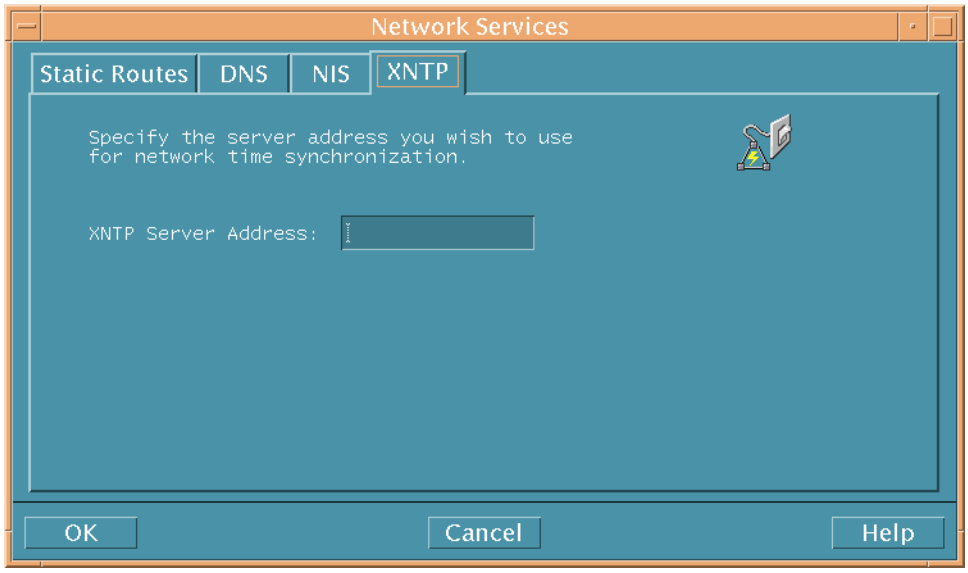
Figure 10-37 NIS Tab



NIS. It is common for the nonserver hosts in a network to be NIS clients. Whenever a process on an NIS client requests configuration information, it calls the NIS server instead of looking in its local *configuration file*s. The set of configuration data maps shared by the servers and clients is called the NIS domain.

For more information, see *domainname*(1), *domainname*(1M), and *Installing and Administering NIS Services*.

Figure 10-38 XNTP Tab

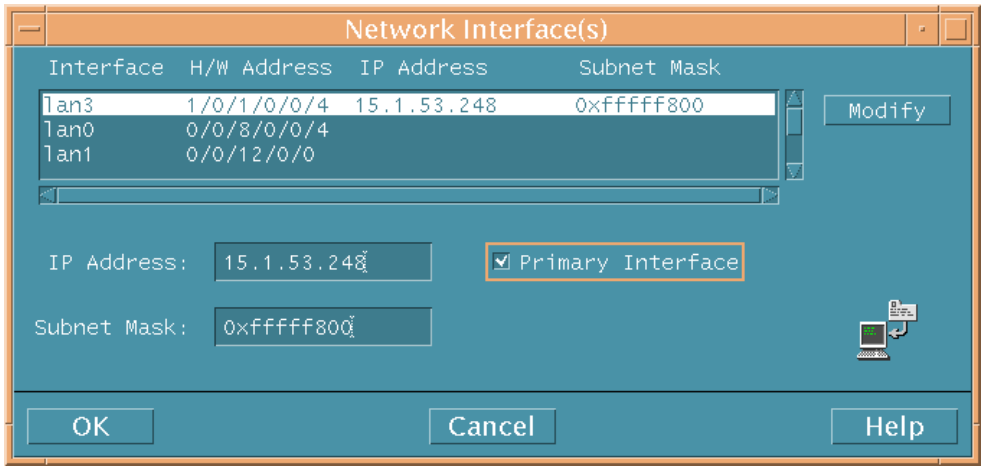


XNTP. The `xntpd` *daemon* maintains system time, in agreement with Internet standard time servers. For more information, see *xntpd*(1M).

Additional Interfaces(s) Button

Click this button on the System tab to identify additional LAN interface cards that exist in the client. You can add or change IP and subnet information as needed, and designate the primary LAN interface to be used.

Figure 10-39 Network Interface(s) Dialog Box

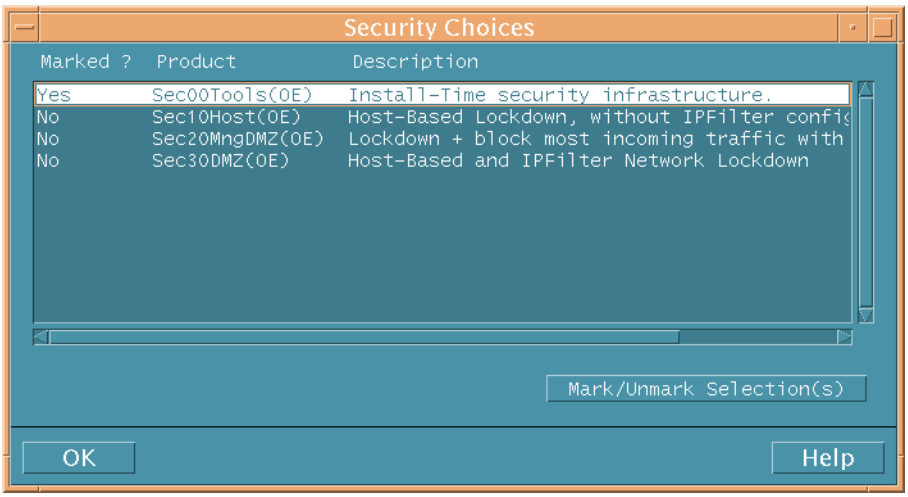


1. Select an Interface card from the selection list.
2. Select **DHCP Managed** if you want the selected interface configured using information obtained via DHCP. When this is selected, the IP Address and Subnet Mask fields are not shown.
3. Enter or modify the IP Address: as needed; this is not required with DHCP Managed.
4. Enter or modify the Subnet Mask: as needed; this is not required with DHCP Managed.
5. Select **Primary Interface**, depending on the status you want for the selected interface. If the client has more than one interface, the Primary Interface will be associated with the hostname of the system in `/etc/hosts`.
6. Click **Modify** when you have finished updating each interface.

Security Choices...

Click this button on the System tab to invoke the Security Choices dialog box. This button is available only if you have install-time security available in your configuration. The install-time security choices allow you to choose from several predefined security levels, thereby hardening the system. If you select a security choice, be sure to set the root password before you click **Go!**, or your system may be vulnerable.

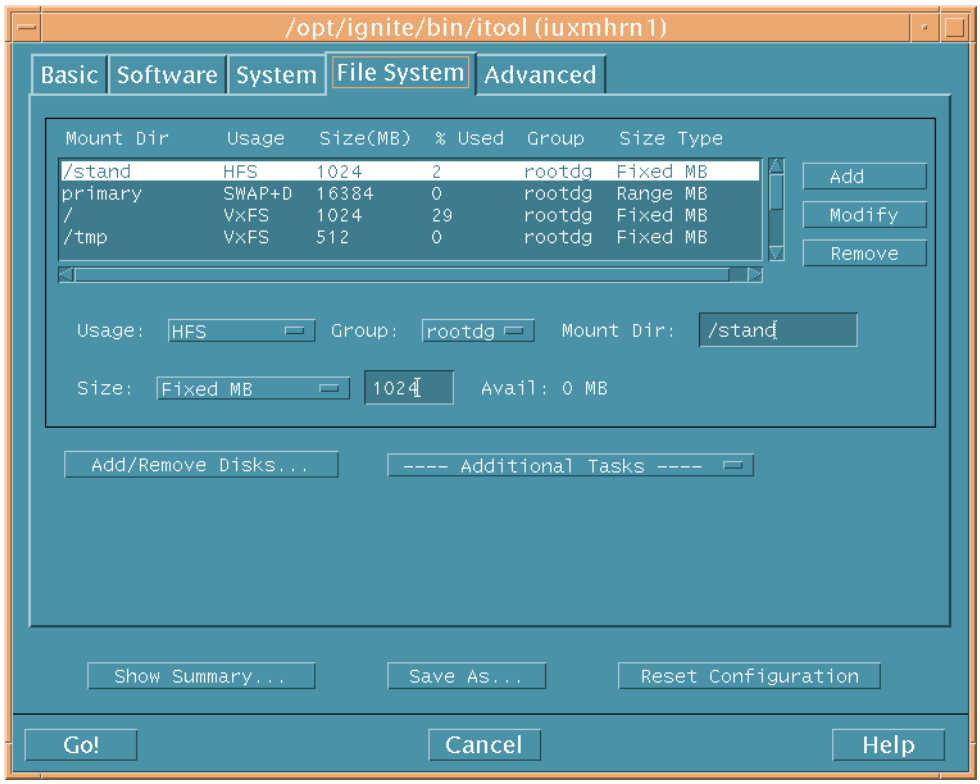
Figure 10-40 Security Choices Dialog Box



File System Tab

The File System tab enables you to do a variety of *file system* and disk configuration tasks. Its appearance depends on your file system selection on the Basic tab. The example in Figure 10-41 illustrates what you would see if you had selected LVM on a PA-RISC client.

Figure 10-41 File System Tab



IMPORTANT: The information appearing on this tab is tailored to the operating system release and hardware architecture. HP recommends that extreme care be exercised when altering these values.

Adding or Changing a File System Configuration

Adding a new *file system* or changing an existing one are similar actions in that the same information is needed for both.

IMPORTANT: Renaming or changing the disk file system structure causes the old file system on that disk to be lost.

To add or change any of the file system configurations, use the following steps:

1. Select a file system from the list. To view all of the available information for the file system, you can use the horizontal and vertical scroll bars, or resize the dialog box.
2. Enter or change information using the following buttons or fields, and then click **Add** or **Modify**. You will lose your changes if you move to another tab without clicking **Add** or **Modify**.

   Usage: Click the selection list button adjacent to Usage: to choose from the list of file system usage types. The available choices are HFS, VxFS, SWAP, Unused, *DUMP*, and SWAP+DUMP. In general, HP recommends the use of VxFS.

   Group: Click the selection list button adjacent to Group: to select the appropriate volume group name for the selected file system.

   To add a new/unused disk and give it a different volume group name or create a new volume group, use the **Add/Remove Disks...** button.

   If you want to reconfigure the volume group in general, including renaming it, click **Additional Tasks** and then select **Group Parameters** to supply a custom group name and to change other disk parameters.

   Mount Dir: Modify this field to set the `root` disk, using the standard HP-UX mount directory designations (`/`, `/usr`, `/stand`, `/var`, `/opt`, etc.). You can also specify your own mount points, such as `/special` or `/apps`.

   Avail: Displays the available disk space on the currently selected volume group.

   Size: The Size: selection list is used to select the type of sizing, and the text field allows you to enter the size value. The Avail: display shows the unallocated disk space in the volume group currently selected. Changes will be reflected in the Size(MB) and Size Type columns of the file system list window after **Add** or **Modify** is selected.

   Note that although you can change the size of file systems, Ignite-UX enforces minimum sizes and modifies the sizes upward if required. These sizes are based on the size of the software to be installed and the size defined for the file system.

   For information on file system size limitations, see the following documents:

   - For HP-UX 11i v3 LVM, see the appendix *LVM Specifications and Limitations* in the document *HP-UX System Administrator's Guide (volume*

3): Logical Volume Management available at http://www.hp.com/go/hpux-core-docs.

- For VxFS on all versions of HP-UX, see the white paper *Supported File and File System Sizes for HFS and JFS* available at http://www.hp.com/go/hpux-core-docs.
- For HP-UX versions earlier than 11i v3, see the section *Planning to Manage File Systems* in the document *Managing Systems and Workgroups: A Guide for HP-UX System Administrators*, available at http://www.hp.com/go/hpux-core-docs.

The types of sizing available are:

Fixed MB - The selected file system is set to the size entered in the text field.

All Remaining - The selected file system automatically takes over all remaining file system space on the disk or volume group.

Free MB - This selection is useful when you know how much free space you want the volume to have after the system is installed. The size of the volume will be the specified Free MB amount plus the amount required for the selected software.

Free % - This category is similar to free size, but is expressed in percent. It is used if you know how full you want the volume to be in percentage of the volume size. If you indicate 20 percent, then the volume would be 80 percent full after the installation of the selected software.

Range MB - Select this category in the list to set a maximum size for the file system (the minimum is determined by the software selected for the file system). The `/usr` file system must have sufficient space to accommodate an operating system update. The absolute minimum is 324 MB for a 64-bit system. See the installation guide supplied with your release of HP-UX.

IMPORTANT: Changes are not saved until you click **Add** or **Modify**. If you make any changes and then select another tab without using one of these buttons, your changes *are not* applied and are lost.

Additional requirements for specific *file system* layouts are as follows:

Table 10-3 Volume Requirements for LVM and VxVM

Volume Manager	Volume Requirements
LVM	• One of the volumes must be `root` (`/`). • A swap volume (primary) is required. • Directory names must have valid HP-UX names (for example, `/usr` or `/database`).
VxVM	• One of the volumes must be `root` (`/`). A root volume group will always be created with VxVM version 3.5. • One of the volumes must be *boot* (`/stand`) with HFS usage. • A swap volume (primary) is required. • Directory names must have valid HP-UX names (for example, `/usr` or `/database`).

Removing a File System

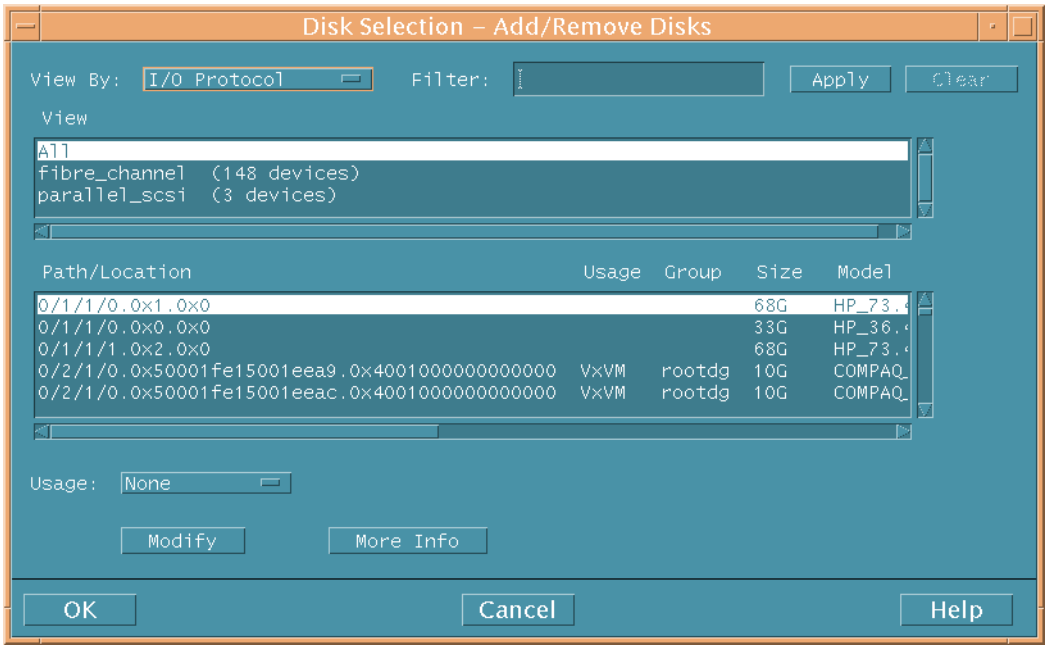
To delete any of the file systems listed on the tab, select the mounted directory and click **Remove**.

Add/Remove Disks... Button

Selecting this button invokes a dialog box that enables you to do the following:

- Determine your current disk usage.
- Change the volume group a disk is assigned to.
- Add a new disk, configure a mount directory, add the disk to a volume group, and assign swap value.
- Remove a disk by designating the usage as “None”. This also protects the disk from being overwritten by this installation.

Figure 10-42 Disk Selection - Add/Remove Disks Dialog Box



To change, add, or remove a disk from the client:

1. Select a disk from the list. The Path/Location window displays a great deal of information, but you have to scroll horizontally to see all the fields. The fields displayed in the Path/Location window are: Path/Location, Usage, Group, Size, Model, WWID, Existing (preexisting data), Legacy Path, and Device ID. See “Root disk... Button” (page 136) for instructions on using the View By: and Filter: items. See “Root disk... Button” (page 136) for instructions on using the More Info dialog box. For information on supported devices, see “Supported Peripherals ” (page 30).
2. Click the Usage: selection list button to select a different use for this disk or **None** to remove it. Depending on your selection, other fields and buttons appear. For example, selecting **VxVM** results in the appearance of the **Disk Group:...** button and associated field.
3. Select **Modify** to execute your changes.
4. Select **OK** when you are finished and to return to the File System tab.

---- Additional Tasks ---- List Button

This button enables you to configure advanced information, if necessary, in the following categories:

- “Disk Parameters”
- “File System Parameters”
- “Volume Parameters”
- “Group Parameters”

The selection of one of these categories results in a change to name of the button and a dialog box that enables you to change the parameters associated with the chosen category appears.

NOTE: Dialog box choices differ depending on the *file system* choices you made on the Basic tab.

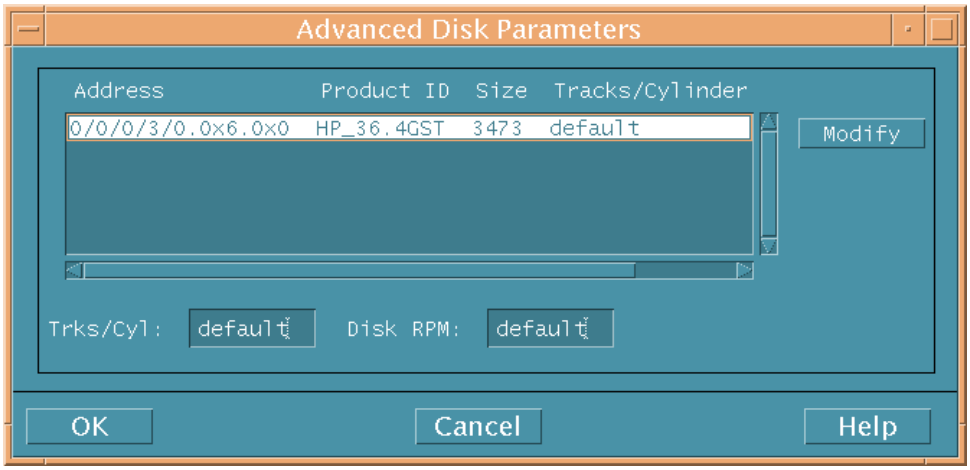
Disk Parameters

Disk Parameters - Follow these steps to further define the tracks per cylinder or revolutions per minute for each disk on the client using the **Disk Parameters** selection of the **Additional Tasks** button. Additionally, you have the option of initializing each disk.

NOTE: HP recommends that you do not modify Trks/Cyl: or Disk RPM: or enable Media Init: to be run. Modern disk drives do not gain any benefit from setting tracks per cylinder or the disk rotations per minute, and they automatically exclude defective sectors from use. If you experience read/write errors on a disk, consider replacement of the disk. In addition, because modern disk drives are so large, running Media Init: to initialize the disk literally takes hours and does nothing useful.

1. Click **----Additional Tasks----** and select **Disk Parameters...**. A dialog box similar to the one in Figure 10-43 appears.

Figure 10-43 Advanced Disk Parameters Dialog Box



2. Highlight a disk in the selection list to select it.
3. Configure the Trks/Cyl: (tracks and cylinder) and Disk RPM: fields by direct editing, as needed. The default values are recommended because these values are predetermined for each supported disk device.
4. Click **Modify** to implement all changes.
5. Click **OK** when you are finished and you are returned to the File System tab.

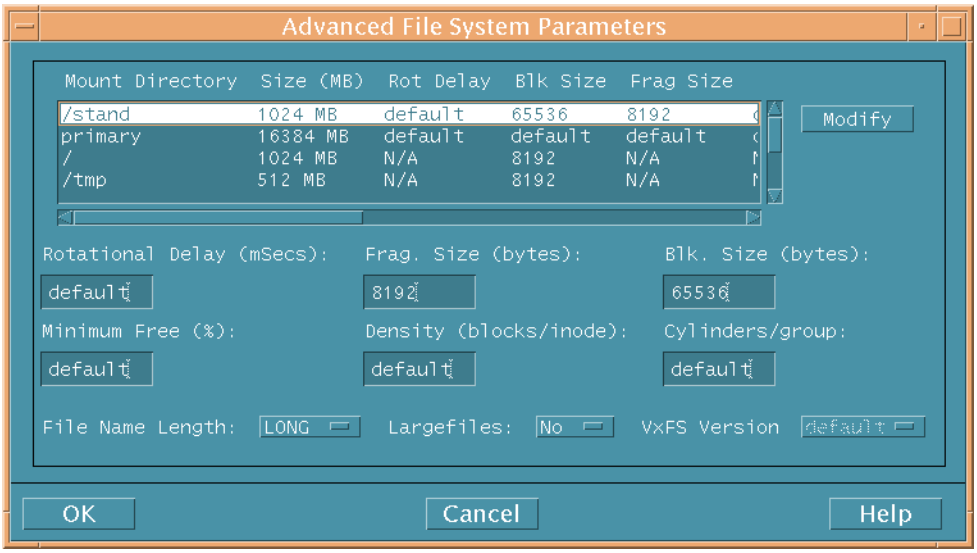
For more detailed information regarding these disk parameters, see *mkfs_vxfs*(1M) and *mkfs_hfs*(1M).

File System Parameters

File System Parameters - Several file systems are already defined by Ignite-UX, but you can modify them as follows to suit your environment.

1. Click **----Additional Tasks----** and select **File System Parameters...**. A dialog box similar to the one in Figure 10-44 appears.

Figure 10-44 Advanced File System Parameters Dialog Box



2. The values that appear are specific to the file system type and can vary. Modify the values as needed.

   The VxFS Version list button is available when a VxFS file system is selected. For client using HP-UX 11i v2 or later, use this option to specify the version of the VxFS file system to create for a mount point. If **Default** is selected, then a value of 5 will be used.

IMPORTANT: Do not enable largefiles on the boot file system. HP does not support doing this, and Ignite-UX will not allow an installation or recovery to proceed while largefiles are enabled on the boot file system.

NOTE: HP does not recommend setting a value of rotational delay for HFS file systems. Setting this value typically does not provide any benefit with modern disk drives.

   For specific information regarding the use of these parameters, see *mkfs*(1M), *newfs_hfs*(1M), *mkfs_hfs*(1M), and *tunefs*(1M).

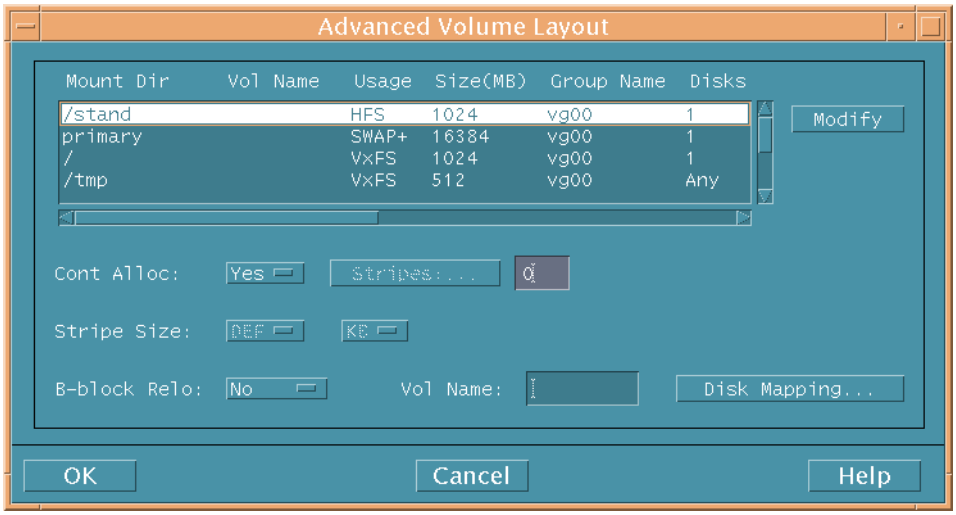
3. Click **Modify** to implement all changes.
4. When you finish modifying all values, click **OK** to return to the File System tab.

Volume Parameters

Volume Parameters - Follow these steps to modify the default volume values and perform detailed volume configurations for LVM and VxVM file systems.

1. Click **----Additional Tasks----** and select **Volume Parameters....** A dialog box similar to the one in Figure 10-45 is displayed, which is an example of an LVM volume.

Figure 10-45 Advanced Volume Layout Dialog Box



2. Modify the following values as needed:

Cont Alloc: This sets the contiguous allocation policy. A contiguous logical volume has these characteristics:

- Physical extents are allocated in ascending order.
- No gap is enabled between physical extents within a mirror copy.
- Physical extents of any mirror copy all reside on a single physical volume.
- The root volume (/), the boot volume (/stand), dump volumes, and primary swap must *always* be created with Cont Alloc set to **Yes.**

Stripes:... If two or more disks are in the volume group, then you may enable data striping over multiple disks for performance purposes.

Stripe Size: Configure this if you have at least two disks in a volume group. The first drop-down allows you to configure the unit, and the second drop-down allows you to configure the size. The default stripe size is equal to the file system block size, which is normally 8 KB.

B-block Relo: Bad-block relocation is a mechanism that, when a disk defect is found, attempts to relocate the failing block to another location on the disk. This is the default behavior when a logical volume is created though it is not supported for `root`, `swap`, or `dump` logical volumes.

Vol Name: Enter the name you want for the selected volume.

Disk Mapping... Displays a dialog box that enables you to restrict the disk drives on which the volume data will reside. Normally, the data will be allocated over these disks sequentially.

For more information, see *lvcreate*(1M) for LVM or *vxassist*(1M) for VxVM.

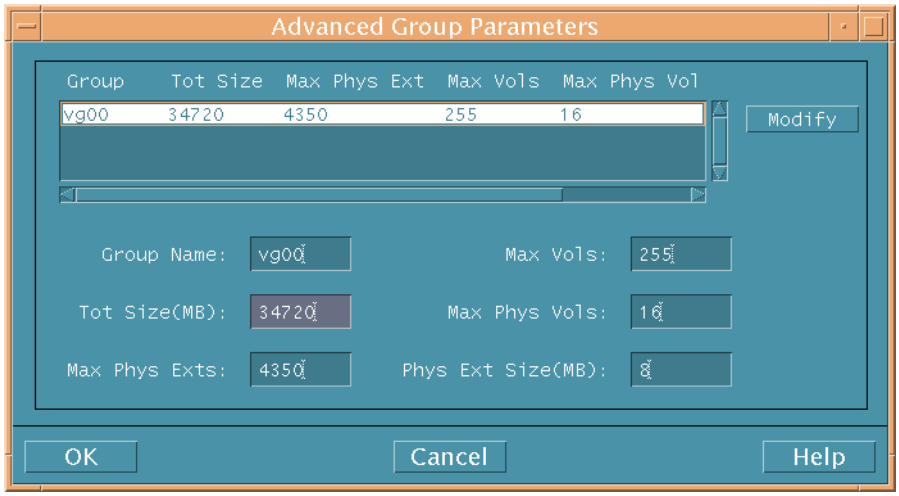
3. Click Modify to implement all changes.
4. When you have finished modifying all values, click **OK** to return to the File System tab.

Group Parameters

Group Parameters - You can set additional volume group parameters as follows to further define volumes.

1. Click **----Additional Tasks----** and select **Group Parameters...**. A dialog box similar to the one in Figure 10-46 appears, which is an example of an LVM group layout.

Figure 10-46 Advanced Group Parameters Dialog Box



2. Modify the following values as needed:
   - LVM Version: — 1.0 or 2.2. The default value can be set via the Basic tab "Additional... Button" (page 140) **Default root group LVM version...** button.
   - Group Name: — Use to rename existing volume groups.
   - Tot Size: — Total size of all volumes in MBs.
   - Phys Ext Size: — Physical extent size in MBs.
   - Max Phys Exts: — Maximum physical extents.
   - Max Vols: — Maximum number of logical volumes.
   - Max Phys Vols: — Maximum number of volumes.

   For more information, see *vgcreate*(1M) for LVM or *vxassist*(1M) for VxVM.
3. Click **Modify** to implement all changes.
4. When you have finished modifying all values, click **OK** to return to the File System tab.

Advanced Tab

You can use the Advanced tab, as shown in Figure 10-47, to activate any HP or custom scripts that you might want to run on the client after the installation has completed. For example, you could include a script to install additional software from other *depot*s, or one that contains specific system configuration changes for the intended use of the client.

Two different lists appear on this tab, Scripts to be Executed: and Available Scripts:.

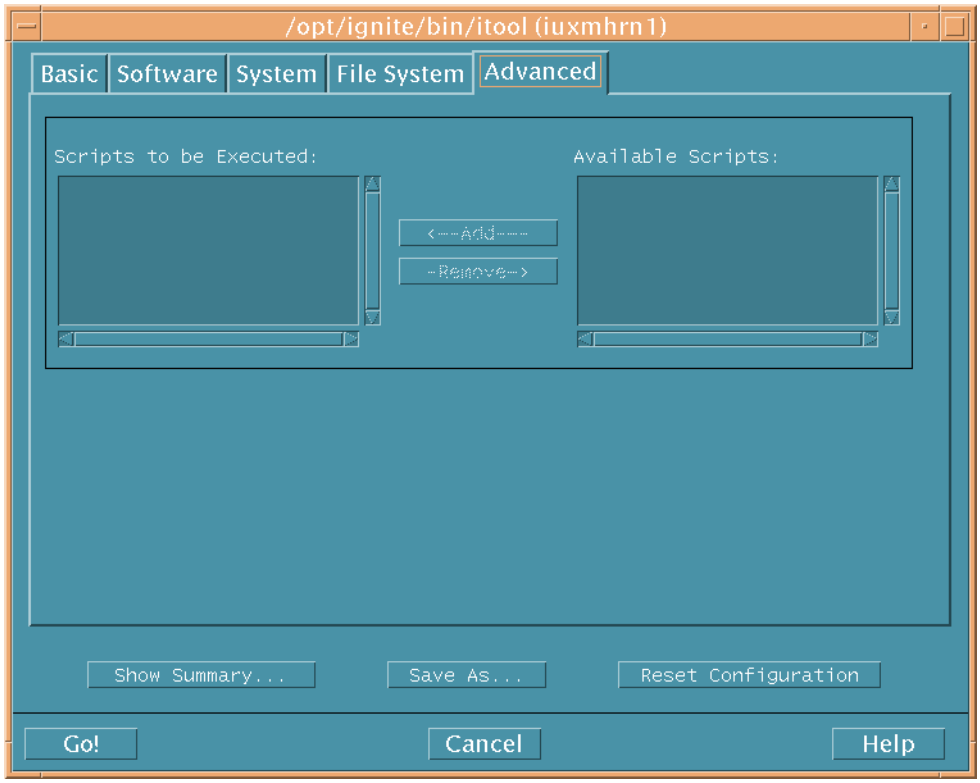
The scripts that populate the Available Scripts list are derived from the `/var/opt/ignite/data/INDEX` file clauses identified by the `scripts` keyword. When you add new scripts to the `/var/opt/ignite/data/INDEX` file using `manage_index`, they are automatically added to this list and are available for use the next time you invoke the Ignite-UX GUI. For example, if you want to add a script called `run_fsck` located in `/var/opt/ignite/scripts`, you would use the following command:

```
manage_index -a -s "/var/opt/ignite/scripts/run_fsck"
```

For more information, see *manage_index*(1M).

Populate the Scripts to be Executed: list by adding or removing scripts from the Available Scripts: list. Descriptions for each action follow.

Figure 10-47 Advanced Tab



Adding a Script

To add a script for Ignite-UX to execute after the installation, select the item from Available Scripts: column, then click **<---Add---**.

For more information regarding the use of scripts to automate Ignite-UX tasks, see Chapter 12: "Customizing Your Installation" (page 177), and see *instl_adm*(4). See also the *Ignite-UX Custom Configuration Files* document, available at

http://www.hp.com/go/ignite-ux-docs.

Removing a Script

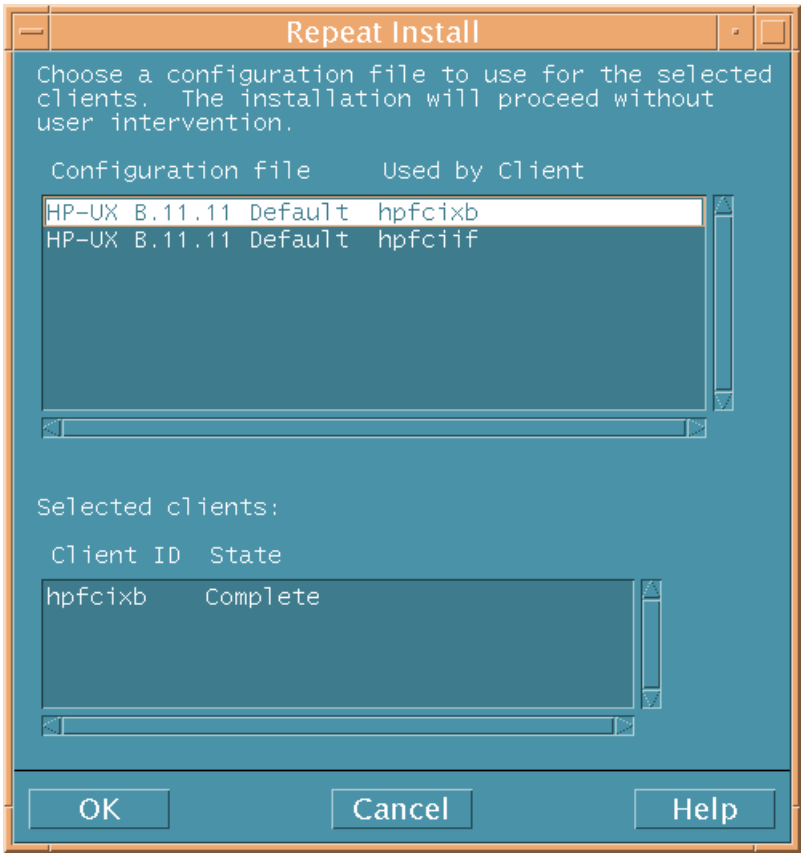
To remove a script, select it in the Scripts to be Executed: list, then click **-Remove->**. The item is removed from the list and is not executed after the installation. The removed script remains in the Available Scripts: list for use at another time.

Repeat an Installation

Once you have successfully installed a client, you have built a repeatable client installation configuration. You can choose to use any past configuration to install or reinstall a client. This method requires no intervention, which enables you to launch multiple installations quickly.

Begin a repeat installation by selecting **Actions→Install Client→Repeat Install** from the Ignite-UX GUI. The Repeat Install dialog box is then displayed.

Figure 10-48 Repeat Install Dialog Box



The Configuration File list (Figure 10-48) is populated by the last installation configuration, and all those that you have created using the **Save As...**button that is on all of the installation configuration tabs. These saved installation configurations are stored in `/var/opt/ignite/saved_cfgs/`.

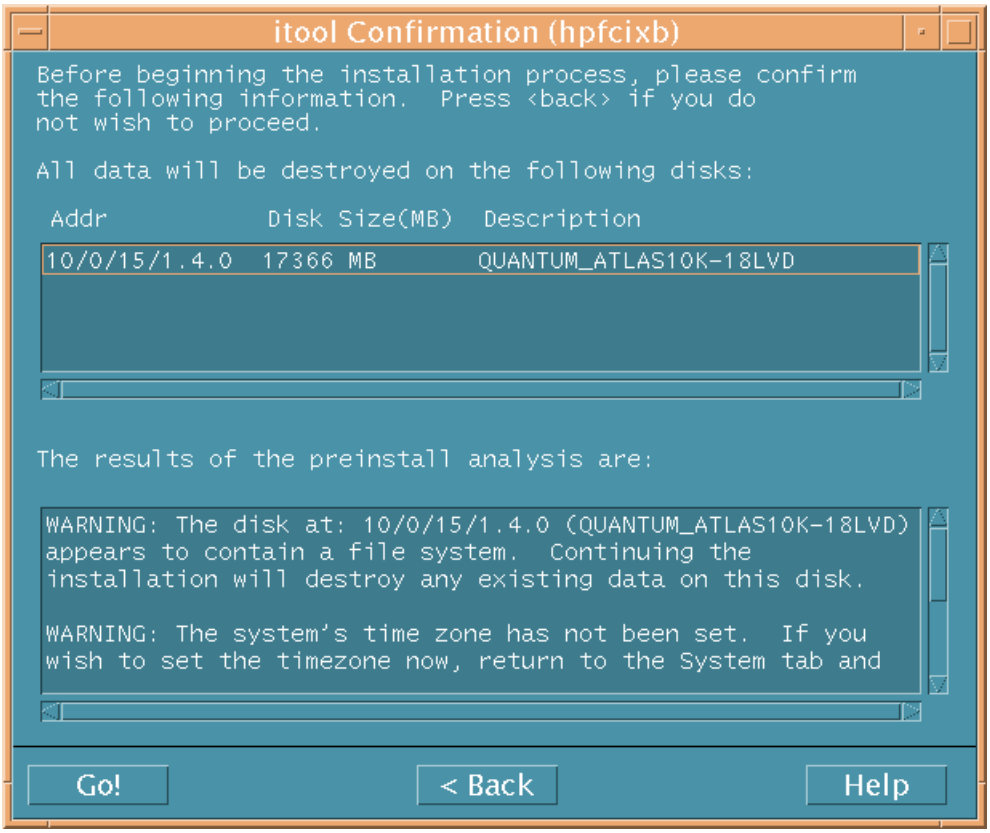
If there are no saved configurations, the **Repeat Install** action is not available and you must begin a new installation as described in "New Installation" (page 133).

Select one of the configurations listed by clicking on it, then click **OK**. Ignite-UX begins the installation process with a confirmation query to alert you that the clients appearing in the Selected Clients list are about be installed. Confirm or cancel the installation of the listed clients using the appropriate button.

Executing the Installation

After selecting **Go!** to initiate a new installation, and also after confirming a repeat installation, Ignite-UX executes a preinstallation consistency check to identify any errors that must be corrected before the installation occurs (Figure 10-49).

Figure 10-49 Installation Confirmation Dialog Box



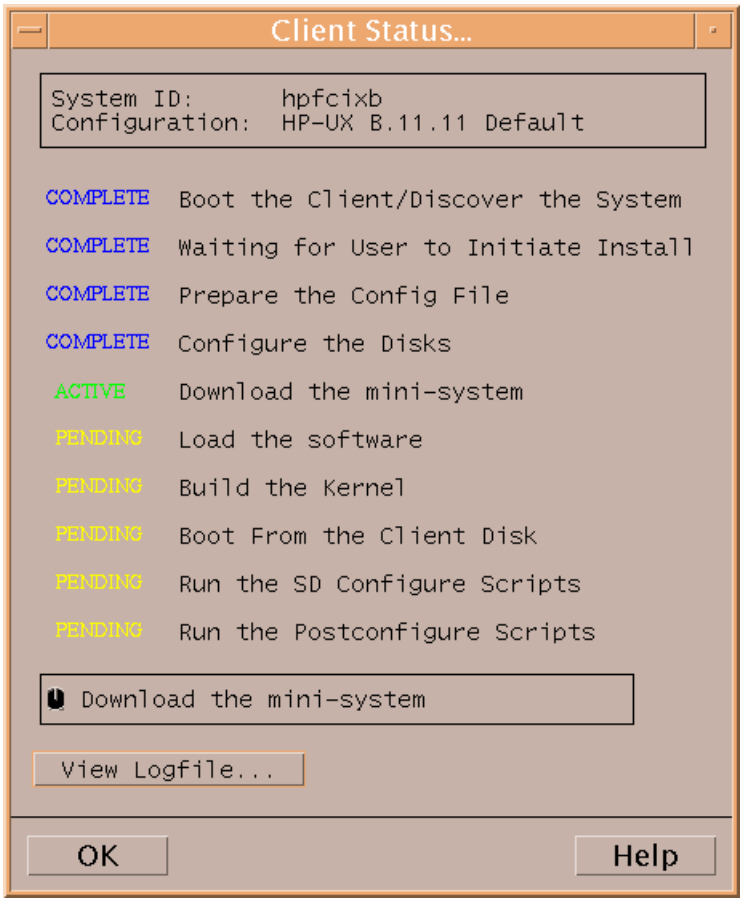
The confirmation dialog box lists disks that are to be written to during the installation process, and a preinstallation analysis log of any notes, warnings, or errors.

- If you do not want to proceed with the installation at this time, click **<Back**.
- View the preinstallation analysis using the scroll bars. Be sure to inspect all of the information and ensure that the correct disk is identified for installation.
- All listed errors *must* be corrected before you can proceed with the installation.

After you have reviewed all of the information and determined whether or not the client installation can continue, click **Go!**.

As the installation proceeds, a Client Status... dialog box is displayed to allow monitoring of the installation progress.

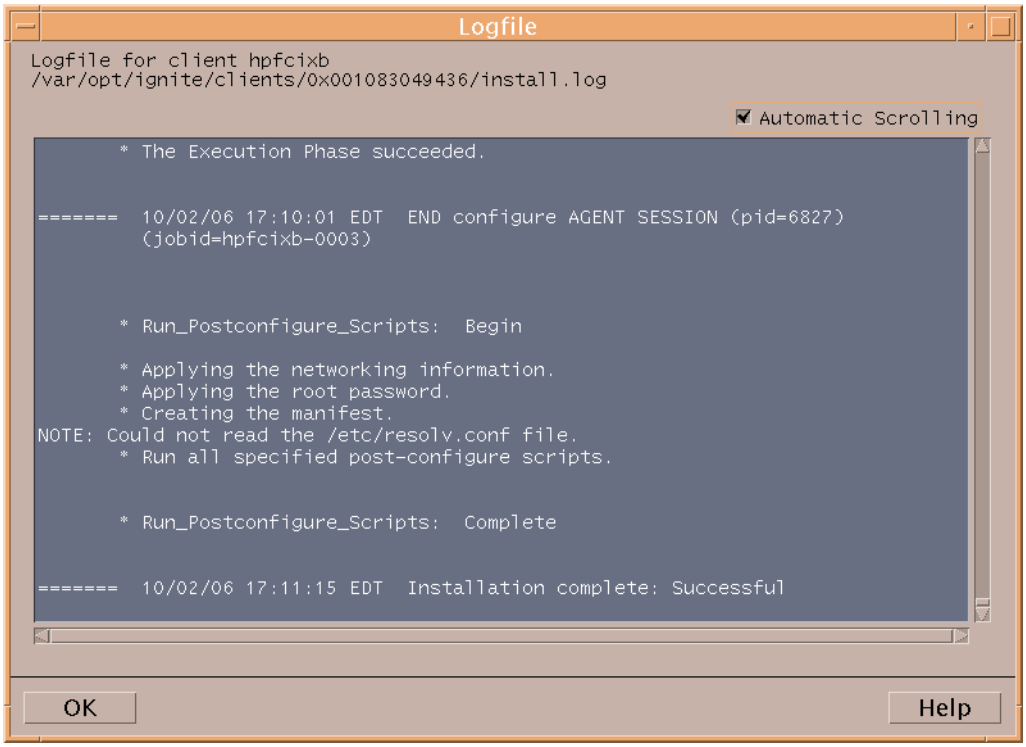
Figure 10-50 Client Status... Dialog Box



The information is updated automatically as the status of the client changes.

You can view specific installation information using the **View Logfile...** button, which invokes a dialog box displaying the log file as it is written to during the installation.

Figure 10-51 Installation Log File



Carefully review the scrolling status information for errors. Continuous scrolling is the default behavior. Toggle scrolling on and off with the **Automatic Scrolling** checkbox.

The complete installation log is available on the *Ignite-UX server* in `/var/opt/ignite/clients/`*`hostname`*`/install.log`.

NOTE: Because errors can occur during the installation that result from other components within HP-UX, all errors may not be written to this log file.

Viewing and Printing a Manifest

Ignite-UX creates a *manifest* for each client upon the successful completion of the HP-UX installation. This manifest details all of the installation information for the selected client. Use it to ensure that the installation executed as expected with all software installed, and to keep a record of each client installation.

The manifest is available for viewing from Ignite-UX, and is saved to files on the server and the client.

The client installation manifest contains the following information:

- System Information
- System Hardware
- Installed Software
- File System Layout
- Disk Layout
- Swap Configuration
- Kernel Configuration

View the manifest from the Ignite GUI by selecting the client, and then clicking **Actions→View/Print Manifest**.

A manifest file for each installed client is saved on the Ignite-UX server in: `/var/opt/ignite/clients/client/manifest/manifest.info`, and on the client in: `/var/opt/ignite/local/manifest/manifest.info`.

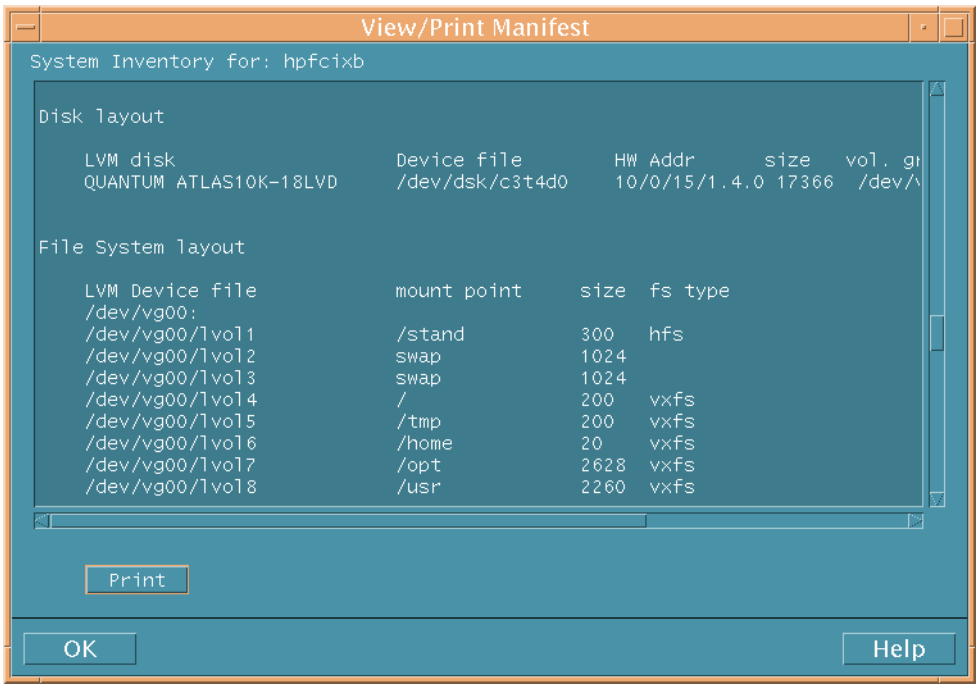
To print the system manifest from the Ignite-UX server command line, enter:

`print_manifest`

The ASCII file is printed to `stdout` using format instructions from the manifest template file. For more information, see *print_manifest*(1M).

An example manifest is shown below.

Figure 10-52 View/Print Manifest Dialog Box



11 Golden Images

A *golden image* is a combination of a *golden archive* (an archive with files set to the *newconfig* state) and a *configuration file* describing a system's *disk layout* and *file system*. A golden image is used as a *common configuration* to install systems. The compressed golden archive format can be `tar`, `cpio`, or `pax`. (To use the `pax` format with 11i v2, you must have the PAX-Enh product installed. The `pax` format is not available for 11i v1.)

Ignite-UX does not require the creation of golden images, but they are a very powerful tool for system administrators.

Golden images allow:

- Increased installation speed versus using `swinstall`, the command used by Ignite-UX when installing software from SD *depot*s.
- Exact replication of a customized system configuration to clients.
- Mass deployment of a customized system configuration to clients.
- Disaster recovery at the *newconfig* level.

This chapter assumes you are creating a golden image to be stored on the *Ignite-UX server* via the network. If you wish to create golden image media, see Chapter 14 (page 197).

Advantages of Golden Images

In addition to supporting the standard software installations from an SD depot, Ignite-UX supports installing from known-good local system images, called golden images. This method recognizes that many clients in a network might be almost identical. You can take advantage of this fact by building an image that contains all of the software you want installed, plus *configuration file*s, and then using Ignite-UX to install it.

This approach can have several advantages:

- Because the compressed system image is unpacked directly to disk over the network, the installation process can be much faster than an equivalent process using SD. The time savings depends on the size of the installation and the capacity of the network, but a typical *golden image* can be extracted in about 20 minutes compared to about an hour for an SD install.
- Instead of troubleshooting a client, it is often more cost-effective to completely reinstall the operating system with a golden image.
- When coupled with dataless nodes (all volatile data is on a separate file server), system replacement or move time is drastically reduced.
- Once a golden image has been created, it is simple to apply it to multiple clients. Very little or no user interaction is required during subsequent installs, reducing the chance of error.

Building a golden image is done by setting up a single system the way you want all of your systems to look, and then creating an image of that system. Follow the instructions below to set up the first system.

Creating a Golden Image

In general, the golden image is created from a golden system configured with all the software and customizations needed to distribute it to a group of clients. The golden image can be saved to media and installed on individual systems, or the golden image can be stored on another system and installed remotely over the network.

If a golden system already exists, proceed to "Configuring the Ignite-UX Server to Recognize the Golden Image" (page 172).

To create a golden system, follow the steps described in this section:

- “Installing the HP-UX Operating System ” (page 170)
- “Installing Critical Patches onto the Operating System” (page 170)
- “Installing Optional Software” (page 171)
- “Customizing the System ” (page 171)
- “Creating the Golden Archive” (page 171)

Once you have a system with the base operating system, you may choose to place patches, applications, *kernel* configurations, etc. on the system, or just include the core operating system. In our example, we only include the HP-UX 11i v1 core operating system.

Ignite-UX is capable of installing software from SD depots. You might want to use this capability when setting up your golden system. See “Setting Up Additional Software on the Server” (page 46) for more information.

Installing the HP-UX Operating System

Although this can be performed without an Ignite-UX server by using `swinstall` from media, this example of an HP-UX 11i v1 installation uses a network depot as the software source.

1. On the Ignite-UX server, set up the 11i v1 core operating system to be distributed. Enter the following commands:

   ```
   # /opt/ignite/bin/make_depots -r B.11.11 \
    -s server:/depots/11.11/B5725AA
   ```

   where *server* is the name of the server on which the depot is located.

   ```
   # /opt/ignite/bin/make_config -r B.11.11
   ```

   The `make_depots` command copies the HP-UX 11i v1 (B.11.11) software from the SD depot pointed to by the `-s` option (the pathname depends on the location of the SD depot you are accessing) to the local Ignite-UX server.

   The `make_config` command then adds this software as a configuration available for Ignite-UX installations. You can run `make_config` and point it to the remote depot directly, if you prefer.

2. Prepare the client system for installation by booting the client:
   - If the client is currently running HP-UX, *boot* the system from the server by entering

     ```
     # bootsys -v -w -f -i "HP-UX B.11.11  Default" \
      client_hostname
     ```

     where *client_hostname* is the name of the client.

   - If the client is not currently running HP-UX, boot the system from the client console. See “Booting Itanium-Based Clients using the Network” (page 112) or“Booting PA-RISC Clients from the Console ” (page 111), depending on the system type.
3. Install the client with the core operating system configuration you have just set up and continue with the next section.

Installing Critical Patches onto the Operating System

At this point, you should have a client installed with the basic HP-UX release – in this example, HP-UX 11i v1. If you have patches to distribute to all users, install them now. This is normally done using the standard SD tools.

For example, to install patch PHCO_34240:

1. Download and use `unshar PHCO_34240` to obtain two files: `PHCO_34240.depot` `PHCO_34240.text`

2. Install the patch non-interactively:

```
# swinstall -x autoreboot=true -x match_client=true \
 -s /PHCO_34240.depot
```

This assumes you have already met the dependencies of PHCO_34240. These instructions can also be found in the `PHCO_34240.text` file.

For more information, see the *Patch Management User Guide for HP-UX 11.x Systems* and the *HP-UX Software Assistant Administration Guide*, available at http://www.hp.com/go/swa-docs.

Installing Optional Software

Load any optional HP and third-party software you want to make available to all clients. Keep in mind that anything you install is distributed to all clients installed using the golden image. Keep in mind any licensing restrictions.

To install HP software from media or a network depot, you normally use Software Distributor (SD) tools. Third-party software installation varies depending on the vendor.

IMPORTANT: Inclusion of multiple versions of Veritas Volume Manager from Symantec (VxVM) in the same installation depot, or in separate depots that are used together in a single cold-installation session, is not supported. Doing so renders it unusable and generates errors when attempting to use the installation depot, or in the case of non-SD depots during the reboot attempt. For more information, see "Considerations When Using Veritas Volume Manager from Symantec" (page 211).

Customizing the System

Perform any customizations you want to distribute to all clients. These might include customized CDE login dialog boxes, base `/etc/passwd` files, manpages, and corporate-wide default *DNS* and NIS setup. Do not include system, workgroup, or site-specific customizations such as *gateway*s, user accounts, and machine-specific networking; these are taken care of by Ignite-UX later.

Creating the Golden Archive

Use the next steps to create the *golden archive* from the golden system, and configure the Ignite-UX server to use it. The `make_sys_image` command creates the golden archive. For more information, see *make_sys_image*(1M).

IMPORTANT: Do not use the golden system while `make_sys_image` is running; the `make_sys_image` command modifies files with host and networking information.

1. Use `ftp` or a similar command to copy the `/opt/ignite/data/scripts/make_sys_image` script from the Ignite server to `/tmp` on the golden system.

NOTE: You don't have to install the full version of Ignite on the golden system to create a golden image; only the `make_sys_image` script is necessary.

2. Set the permissions of `/tmp/make_sys_image` so you can execute it.

```
# chmod +x /tmp/make_sys_image
```

NOTE: By default, `make_sys_image` stores the *archive* in the `/var/tmp` directory. You can have `make_sys_image` save the archive to a remote server. Either method requires sufficient disk space to hold the archive. The amount of disk space is approximately one half the amount of data contained on your system, assuming a 50 percent compression ratio from `gzip`.

3. On the Ignite-UX server, create a directory to store the golden archive:

```
# mkdir -p /var/opt/ignite/archives/Rel_B.11.11
```

The `-p` option creates intermediate directories. It is best to store it in a directory of type `/var/opt/ignite/archives/Rel_rel`. This directory must be NFS-exported if you are using NFS to transfer the archive to the client.

4. If you do not have enough disk space, or you want the *archive* created on a remote server, you may use the following options:

```
# /tmp/make_sys_image -d directory_to_place_archive
  -s destination_system_IP_address
```

5. On the golden system, run:

```
# /tmp/make_sys_image [options]
```

By default, this creates a `gzip`-compressed `tar` archive in `/var/tmp` with the name *`hostname`*`.gz` (where *`hostname`* is the name of the client). All specific host information, device files, log files, and network information are removed.

IMPORTANT: If an error similar to the following appears:`ERROR: Cannot remsh server` *`system_name`* `(check server .rhosts file)`, you must add the name of the client to the `/.rhosts` file on your Ignite-UX server as in the following example entry:

`system_name` `root`

The `make_sys_image` command can build an archive with the `tar`, `cpio`, or `pax` format, compressed with a selectable compression method. HP recommends using `tar` (default) for the archive format and `gzip` (default) for compression. To use the `pax` format with 11i v2, you must have the PAX-Enh product installed. The `pax` format is not available for 11i v1.

IMPORTANT: When using the `pax` command you should consider the size of your files. Files included by the `pax` command with `tar` and `cpio` formats must be less than 8 GB on 11i v2 and v3, and less than 2 GB for v1. Using the `pax` format will allow large files in an archive – up to 2 TB for 11i v2 and up to 16 TB for v3.

Configuring the Ignite-UX Server to Recognize the Golden Image

To create an Ignite-UX *configuration file* for the *golden image*, use the example config file `/opt/ignite/data/examples/B.11.11.golden_image.cfg`. Other example config files are available to match the version of HP-UX your golden systems is running: `B.11.23.golden_image.cfg` and `B.11.31.golden_image.cfg`.

1. Create a copy of the example configuration file:

```
# cp /opt/ignite/data/examples/B.11.11.golden_image.cfg \
/var/opt/ignite/data/Rel_B.11.11/core_archive.cfg
```

Because the configuration file must be accessible by `tftp` or NFS, depending on how you've set up your server, make sure you store it in a directory of type `/var/opt/ignite/data/Rel_rel`.

Ignite-UX manages the names and locations of configurations using the `/var/opt/ignite/data/INDEX` file (see step 3).

2. Modify the `core_archive.cfg` file to set up the golden image for NFS transfer. (See the complete `sw_sel` and `sw_source` examples following these instructions. Use values appropriate for your systems.) Key changes are:
   a. In the `sw_source` clause, change the following line so it has the path to your archive, including the IP address of your Ignite-UX server. This directory will have to be NFS exported later.

```
nfs_source = "10.2.72.150:/var/opt/ignite/archives/Rel_B.11.11"
```

b. Remove the `init sw_sel` clauses you will not be using for this golden image.

c. In the `init sw_sel` clause for your golden image, leave the default description or change it to something representative of your archive.

```
description = "Archive HP-UX 11.11 CDE"
```

d. Modify the following line so `archive_path` resolves to the archive file. This path must be relative to the path specified by `nfs_source` in the `sw_source` clause.

```
archive_path = "hostname.gz"
```

e. Add `impacts` lines in the `init sw_sel` clause by executing the `archive_impact` command and then replacing the example impacts lines with the ones for your archive. It might be helpful to redirect the output of `archive_impact` to a temporary file. See *archive_impact*(1M) for more information.

```
# /opt/ignite/lbin/archive_impact -t -g archive_file
```

f. Review the `init sw_sel "English"` clause and modify it for the locales you want available in the golden image.

The format of an entry in the locale list is

```
"<locale>:<description>"
```

where **<locale>** is the name of a locale shown in the output of the `locale -a` command run on the golden image system (for example: `ja_JP.SJIS`). The **<description>** is a text string labeling that particular locale. The **<description>** text string cannot contain white space; if you need more than one word in the description, use the underscore character. Locale list entries must be enclosed in double quotes.

For example, the following line would make available any of the Japanese or English locales on the system:

```
locale = { "ja_JP.SJIS:Japanese", "ja_JP.eucJP:Japanese",
"ja_JP.utf8:Japanese", "SET_NULL_LOCALE:English", "C:English" }
```

Never add locales that do not exist in the golden image.

Be aware that by listing a locale, it means that it can be selected via *itool*. Any locale present when the golden image is created will still be available on any system installed using this golden image, whether it is listed or not.

g. For HP-UX v2 and later, you need to uncomment the VxVM version of the golden image, if the golden image uses a VxVM disk layout. For HP-UX v3, you must uncomment the version for VxVM or LVM. For example, if the VxVM version on an HP-UX 11i v3 golden image is 5.0, your configuration file would look like:

```
#############################################################################
# VxVM version keyword
#
# If this file is used for VxVM disk layouts, Ignite-UX needs to be
# told which version of VxVM is included in the golden image so that
# the correct version is created. If this is not set, it defaults to
# version 4.1. If VxVM is not used, then this does not need to be set.
# If VxVM is used, uncomment the correct version below.
#############################################################################
# vxvm_version="4.1"
 vxvm_version="5.0"

#############################################################################
# LVM version variable
#
# If this file is used for LVM disk layouts, Ignite-UX needs to be
# told which version of LVM is included in the golden image so that
# the correct choices can be used. If this is not set, it defaults to
# version 1.0. If LVM is not used, then this does not need to be set.
# If LVM 2.2 is in the golden image, uncomment the two lines below.
```

```
    ##########################################################################
    # _hp_provides_LVM_vg_vers="2.2"
    # _hp_provides_LVM_vg_vers visible_if false
```

Here are the HP-UX 11i v1 `sw_source` and `sw_sel` examples:

```
sw_source "core archive" {
  description = "HP-UX Core Operating System Archives"
  load_order = 0
  source_format = archive
  source_type="NET"
  .
  .
  .
  post_load_script = "/opt/ignite/data/scripts/os_arch_post_l"
  post_config_script = "/opt/ignite/data/scripts/os_arch_post_c"

  #If nfs_source is used, be sure to export the source.
  nfs_source = "10.2.72.150:/var/opt/ignite/My_Golden_Images"
  .
  .
  .
}
init sw_sel "golden image" {
  description = "Archive HP-UX 11.11 CDE"
  sw_source = "core archive"
  sw_category = "HPUXEnvironments"
  archive_type = gzip tar
  # For NFS, the path to the archive is relative to the mount
  # point specified in the sw_source:
  archive_path = "hostname.gz"
  # ftp/remsh sources can use a full path:
  # archive_path = "/pub/IUXarchives/B.11.11_CDE.gz"
  impacts = "/" 23Kb
  impacts = "/.dt" 35Kb
  impacts = "/TT_DB" 18Kb
  impacts = "/etc" 1375Kb
  impacts = "/export" 1Kb
  impacts = "/opt" 74079Kb
  impacts = "/sbin" 13449Kb
  impacts = "/stand" 1Kb
  impacts = "/tmp" 1Kb
  impacts = "/usr" 225459Kb
  impacts = "/var" 5736Kb
} = TRUE
```

3. Add the new configuration file (`core_archive_cfg`) to the Ignite-UX `INDEX` file:

   Edit the `/var/opt/ignite/data/INDEX` file to add a new configuration to Ignite-UX. For this example, add a new *cfg clause* as follows:

   ```
   cfg "HP-UX B.11.11 archive" {
     description "some description of this archive..."
     "/opt/ignite/data/Rel_B.11.11/config"
     "/var/opt/ignite/data/Rel_B.11.11/core_archive_cfg"
     "/var/opt/ignite/config.local"
   }
   ```

   The line for the golden image is the one containing `core_archive_cfg`. The `config` and `config.local` files contain default configurations.

   The `/var/opt/ignite/config.local` file should be the last file in the cfg clause. The last configuration file has the highest priority and can override values in the configuration files listed before it.

The file /opt/ignite/data/*Rel_B.11.11*/config supplies the disk and *file system* layout defaults, plus other control information required by Ignite-UX. It must be first in every cfg clause.

Each cfg clause appears as an available configuration to Ignite-UX. Therefore, the string HP-UX B.11.11 archive will now appear as a valid configuration choice.

4. Verify the syntax of the configuration files listed in the INDEX file.

   ```
   # instl_adm -T
   ```

   Fix any errors found by instl_adm and repeat the instl_adm -T to verify any changes.

5. Ensure the NFS file system is exported correctly. In the sw_source clause above, we specified the location of the golden archive to be a file on an NFS server. You must ensure clients have access to this directory on the Ignite-UX server.

   Ignite-UX automatically tries to export /var/opt/ignite/clients for its use. In our example, /var/opt/ignite/archives/Rel_B.11.11 must also be exported because that is where we placed the golden archive.

   On HP-UX 11i v3 Ignite-UX servers:

   - To view the current status, use the following command.

     ```
     # share
     ```

   - Edit the /etc/dfs/dfstab file and add the following lines:

     ```
     share -F nfs -o anon=2 /var/opt/ignite/clients
     share -F nfs -o anon=2,ro /var/opt/ignite/archives/Rel_B.11.11
     ```

   - Run the following command to share all the NFS file systems.

     ```
     # shareall -F nfs
     ```

   On HP-UX 11i v1 and 11i v2 Ignite-UX servers:

   - To view the current status, use the following command.

     ```
     # exportfs -v
     ```

   - Edit the /etc/exports file and add the following lines:

     ```
     /var/opt/ignite/clients -anon=2
     /var/opt/ignite/archives/Rel_B.11.11 -ro,anon=2
     ```

   - Run the following command to export all directories listed in /etc/exports.

     ```
     # exportfs -av
     ```

   See *dfstab*(4) or *exportfs*(4) for more information.

Enabling the Client

Since the Ignite-UX server is now configured with the new golden image, you can use Ignite-UX to install the golden image onto a client. To do this, you need to get the client to inform the Ignite-UX server that it is ready to install a new operating system. There are two methods for doing this.

- If the client is currently running HP-UX, boot the system from the server by entering

  ```
  # bootsys -v -w client_hostname
  ```

  where *client_hostname* is the name of the client.

- If the client is not currently running HP-UX, boot the system from the client console. See "Booting Itanium-Based Clients using the Network" (page 112) or"Booting PA-RISC Clients from the Console " (page 111), depending on the system type.

Installing the Golden Image on the Client

Once a golden image is created, you can install it just like any other Ignite-UX installation configuration. See "Installation Using the Ignite-UX GUI" (page 128). On the Basic tab, select the golden image configuration from Configurations:.

See Chapter 13: "Automating Installations" (page 191) for information on how to execute an installation with no interaction.

12 Customizing Your Installation

This chapter introduces Ignite-UX *configuration file*s, provides examples of how you can use them, and offers debugging tips. Additionally, it describes how to create local customizations for clients using post-installation scripts, and how to exclude software patch files from *archive*s.

Using Configuration Files

Ignite-UX is driven by *configuration file*s that define how clients are installed and configured. A configuration file can be thought of as a set of instructions. Ignite-UX provides a set of default configuration files when you install the product. These default configuration files are used until you change or customize them for use in your environment. By creating your own custom configurations, you can:

- Save time during installation
- Ensure standard configurations for similar clients
- Create configurations specific to operating system version or hardware architecture
- Automate all manner of tasks that would otherwise require manual intervention

The configuration file is expressed in a human- readable language, which is fully defined in *instl_adm*(4). The configuration file language is much like other programming languages in that it supports the use of variables and conditional expressions. You can create configuration files directly or by using the Ignite-UX GUI.

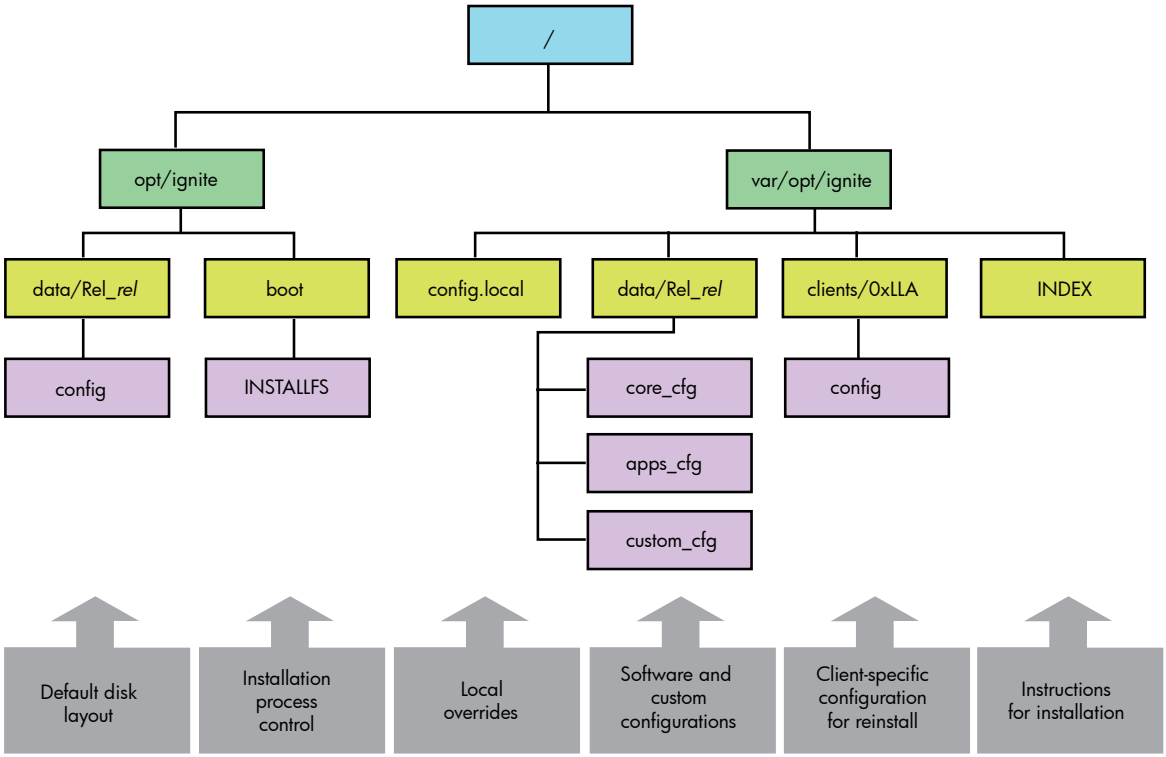
Most of the important elements that make up an installed system are described in the configuration files:

- Identity of the client, presence of network configuration, and *kernel* modifications (additional drivers or tunable parameter settings)
- Disk and *file system* layout
- Software to be installed
- User-defined scripts that run at various points in the installation process to further customize the client

Classes of Configuration Files

The *configuration file*s used by Ignite-UX during the installation process logically group similar information into classes by operating system and functionality. Figure 12-1 illustrates the classes of configuration files and their locations.

Figure 12-1 Configuration File Use and Locations



Ignite-UX processes *config file*s in the order shown below. The following description of each class explains how the various installation parameters can be progressively overridden:

1. **Installation control parameters -** You can define the behavior of the installation process using parameters stored within the segment of the *install file system* that is reserved for configuration parameters (the first 8 KB). This *configuration file* location is special because it is available to Ignite-UX very early in the *boot* process. Some parameters that control installation must be specified here. You can specify defaults for parameters, such as:
   - IP address of the Ignite-UX server
   - Whether to halt the client when the installation is complete
   - Whether to execute installation of new clients from the Ignite-UX server GUI

   Table 12-1 lists the *install kernel* and *install file system* names and supported hardware architecture.

Table 12-1 Install Kernel and File System Names by Hardware Architecture

Hardware Architecture	Kernel Name	File System Name
PA-RISC 32-bit	*INSTALL*	*INSTALLFS*
PA-RISC 64-bit	*WINSTALL*	*WINSTALLFS*
Itanium-based	*IINSTALL*	*IINSTALLFS*
V-Class	*VINSTALL*	*VINSTALLFS*

These install kernels and install file systems are located in the `/opt/ignite/boot/Rel_`*`release`* directory. Install kernels are normally hard linked, such that `INSTALLFS`, `WINSTALLFS`, `IINSTALLFS`, & `VINSTALLFS` files are one and the same. Ignite-UX uses the `INSTALLFS` file system as a default unless an alternate is specified using the `-F` option of the `instl_adm` command. For more information, see *instl_adm*(1M).

Control parameters, such as `run_ui`, `control_from_server`, and `disable_dhcp`, can only be specified in the install file system configuration area and are accessed early in the

installation process when this area is available. Boot control parameters are detailed in the `Control Parameters` section of *instl_adm*(4).

You must use *instl_adm*(1M) to add, change, or delete these boot control and network definitions.

NOTE: Before upgrading to a new version of Ignite-UX, consider retaining the current control parameters, located in the first 8 KB of your install file system, so that you can reapply them after you have successfully updated your Ignite-UX server.

Extract the current parameters into a file, with the following command:

```
instl_adm -d -F [W|V|I]INSTALLFS > first8k_param_file
```

Edit the `first8K_param_file` to define your control parameters. Check your syntax with the following command:

```
instl_adm -T -f first8k_param_file
```

If you want to reapply these control parameters to all install file systems on your Ignite-UX, use the following command:

```
instl_adm -f first8k_param_file
```

If you want these control parameters applied to only one specific install file system, use the `-F` option. For more information, see *instl_adm*(1M).

---

2. **Default disk and file system layout -** The capabilities of each operating system release differ somewhat so HP supplies a different set of disk and file system layout configuration defaults for each release. These configuration files are located in:

   `/opt/ignite/data/Rel_`*`release`*`/config`

   Enter **`uname -r`** on the command line to determine the *`release`*. For example, the file that contains the default *disk layout* for HP-UX 11.11 would be in:

   `/opt/ignite/data/Rel_B.11.11/config`

   as revealed by the `uname -r` command.

3. **Software description of a single SD depot -** Configuration files that describe software available from SD depots can be automatically generated using the `make_config` tool within Ignite-UX. This tool produces one configuration file per SD depot. Software description configuration files are located in:

   `/var/opt/ignite/data/Rel_`*`release`*`/*`

4. **Software description of an archive** — You can create configuration files to enable access to *archive*s (templates are provided with Ignite-UX in `/opt/ignite/data/examples/` to give you a good starting point). Archive software description configuration files are also located in: `/var/opt/ignite/data/Rel_`*`release`*`/`

5. **Local configuration overrides that apply to all clients -** It is often convenient to specify defaults to be applied to every client, in addition to the necessary operating system configuration installed from a particular Ignite-UX server. For example, you might want to specify the same NIS domain for all systems. You must include this type of configuration override information in:

   `/var/opt/ignite/config.local`

   This file is not overwritten when the operating system is updated.

6. **Client-specific configuration file -** This file contains specific directives appropriate for a specific system to override what may have been defined as general defaults for all systems in earlier configuration files. For example, you might want to customize the *disk layout* beyond what the operating system release defaults allow in:

   `/opt/ignite/data/Rel_`*`release`*`/config`

The unique customizations appear in the directory dedicated to the client by MAC address, which is linked to a directory containing the client name:

`/var/opt/ignite/clients/client/config`

This file is created when you use the Ignite-UX GUI to specify the client configuration.

7. **Creating and saving custom configuration choices -** You can create your own custom configurations using the Ignite-UX GUI, save them for repeated use, and easily select them when installing clients. For example, you might have a large number of users with similar systems who all run Computer Aided Design (CAD) tools. You could build a configuration that defines all necessary parameters and save it in a configuration called `CAD System`. When you want to install a new system for a CAD user, you can select **CAD System** from the GUI and you are done (or you could customize it further using `CAD System` as the template). Saved configurations are located in: `/var/opt/ignite/saved_cfgs/`

NOTE: Configuration files are often referred to as config files because the word configuration is shortened to create file and directory names. For example, a client's local configuration file is `config.local`.

You can build your own configuration files that specify the various installation parameters you are interested in, and then combine them in arbitrary ways into any number of different custom configurations using the `/var/opt/ignite/data/INDEX` file. Place these custom configuration files in one of the HP-UX release-specific operating system directories:

`/var/opt/ignite/data/Rel_release/*`

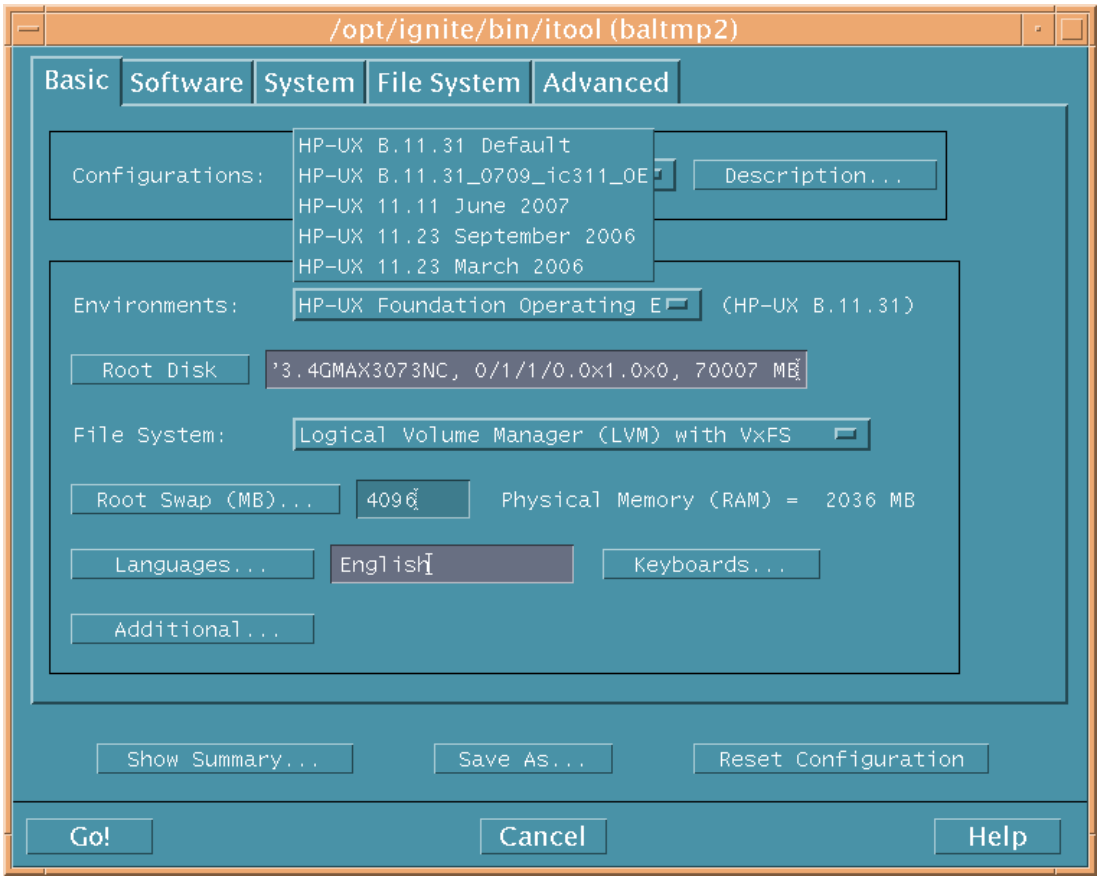
The next section describes how to combine multiple configuration files (default or customized) to define a single configuration.

Combining Configuration Files Using INDEX Entries

Grouping *configuration file*s into useful configurations is accomplished in `/var/opt/ignite/data/INDEX`. This file contains a list of configurations in separate clauses; each comprising one or more configuration files that define an installation. Each *configuration clause* begins with `cfg` and a name by which the configuration is known.

You can view these configuration names using the `instl_adm` command. When installing a new client from the Ignite-UX GUI, you can view these configurations by clicking the button adjacent to Configurations... on the Basic tab by as shown in Figure 12-2.

Figure 12-2 Configuration Choices Dialog Box



A typical `/var/opt/ignite/data/INDEX` file might contain clauses similar to the following excerpt:

```
.
.
.
cfg "HP-UX B.11.23 Default" {
       description "Default B.11.23 release configuration."
        "/opt/ignite/data/Rel_B.11.23/config"
        "/opt/ignite/data/Rel_B.11.23/core_cfg"
        "/opt/ignite/data/Rel_B.11.23/hw_patches_cfg"
        "/var/opt/ignite/config.local"
} = TRUE
.
.
.
cfg "CAD System-11.23" {
        description "Supplies the CAD System configuration."
        "/opt/ignite/data/Rel_B.11.23/CAD_config"
        "/opt/ignite/data/Rel_B.11.23/CAD_core_cfg"
        "/opt/ignite/data/Rel_B.11.23/hw_patches_cfg"
        "/opt/ignite/data/Rel_B.11.23/CAD_sw_sels_cfg"
        "/var/opt/ignite/config.local"
}
.
.
.
```

With this `/var/opt/ignite/data/INDEX` file, the Ignite-UX GUI would present two configurations: **HP-UX B.11.23 Default** and **CAD System-11.23**. The **HP-UX B.11.23 Default** configuration is the default because that *cfg clause* is set to `TRUE`. After choosing a configuration,

you can further customize the configuration using the GUI, or accept the configuration defaults to begin the installation immediately.

The order of the configuration files within a cfg clause is significant; attributes specified in a later configuration file can override the same attributes specified in an earlier configuration file. Two configuration files are used implicitly every time:

- Any information stored in the first 8 KB of `/opt/ignite/boot/Rel_`*`release`*`/[W|V|I]INSTALLFS` is implicitly prepended to each configuration list and is the first configuration data processed.
- The client-specific configuration file `/var/opt/ignite/clients/`*`client`*`/config`, if it exists, is implicitly added as the last configuration file for each configuration.

A default cfg clause for each release is shipped as part of Ignite-UX. Additional cfg clauses are added when you:

- Save a named configuration from the GUI with the **Save As** button.
- Create a configuration by modifying the `/var/opt/ignite/data/INDEX` file directly.
- Use the `manage_index` command to automate `/var/opt/ignite/data/INDEX` file modifications.

NOTE: To facilitate client recovery configurations, a *`CINDEX`* configuration file, similar to an installation `INDEX` file, is created. For more information, see Chapter 15: "Recovery" (page 209) or see *manage_index*(1M) and *make_net_recovery*(1M).

Additionally, you can specify how installation software is handled by Ignite-UX using the following three constructs:

- A `sw_source` clause specifies an SD depot or an access method to a server containing software depots.
- The `sw_sel` clause specifies the software contained in the SD depot or specifies the path to a depot on the server or media. Typically there is one `sw_sel` definition per software bundle or depot.
- The `sw_category` clause is simply a mechanism for grouping `sw_sel` definitions.

See the clauses in Defining an Installation Depot for example usage of the above constructs. For more information, see *instl_adm*(1M).

Be sure to pass all user-generated *configuration file*s through the following command to check for syntax errors:

`instl_adm -T -f `*`cfg_file`*

Example Configuration Files

This section shows a few example configuration files to give you an idea of their look and capabilities. For a complete description of Ignite-UX configuration files, see *instl_adm*(4).

For additional examples of configuration files, see the document, *Ignite-UX Custom Configuration Files* available at

`http://www.hp.com/go/ignite-ux-docs`

Defining Disks

This example shows how a disk might be defined. Here, the disk is located at hardware address `2/0/1.6.0` and does not use Logical Volume Manager (LVM) or Veritas Volume Manager by Symantec (VxVM). The disk contains the root (/) *file system* and a swap area. The swap area takes up 512 MB and the root file system assumes the remainder:

```
partitioned_disk
{
    physical_volume disk[2/0/1.6.0]
```

```
    fs_partition {
        usage = HFS
        size = remaining
        mount_point = "/"
    }
    swap_partition {
        usage = SWAP
        mount_point = "primary"
        size = 512
    }
}
```

Combining Disks to Form a Single Volume Group

You can put two disks together to form a single volume group. Two file systems are defined; both are striped across both disks. The following example illustrates this concept:

```
volume_group "appsvol" {
    usage=LVM
    physical_volume disk[2/0/1.5.0] {
    }
    physical_volume disk[2/0/1.4.0] {
    }
    logical_volume "apps1" {
        mount_point = "/apps1"
        usage = VxFS
        size=30% free
        minfree = 5
        stripes = 2
    }
    logical_volume "apps2" {
        mount_point = "/apps2"
        usage = VxFS
        size = remaining
        minfree = 5
        stripes = 2
    }
}
```

The preceding example uses LVM as the volume manager. However, it is also applicable to VxVM if `usage=LVM` is changed to `usage=VxVM`.

The first file system, `/apps1`, is sized by calculating the amount of space required by the software that is to be installed, and then adding 30 percent for free space. The second file system, `/apps2`, uses the remaining space on the disks.

NOTE: You can modify the file system volume sizes in the *recovery image* when the image is installed. By default, Ignite-UX ensures that there is 10 percent free space for each volume and modifies the file system volume size accordingly. If you do not want Ignite-UX to modify the file system volume sizes automatically, add `init _hp_ignore_sw_impact=1` to your `/var/opt/ignite/recovery/latest/system_cfg` file, or to the `/var/opt/ignite/clients/`*`client`*`/recovery/latest/system_cfg` file.

Defining Networking Parameters

The following example lines define a few of the network parameters that are assigned to the system after it has been installed:

```
final system_name = "acorn1"
final ip_addr["lan0"] = "10.99.45.123"
final netmask["lan0"] = "255.255.255.0"
final nis_domain = "nis1"
final route_gateway[0] = "10.99.45.1"
```

Defining an Installation Depot

The next example defines a single SD *depot* from which software can be installed. Two different pieces of software are defined for the SD depot. Each can be selected independently for installation. The `impacts` lines tell Ignite-UX how much space this software requires in a given directory. This information is used to size the file systems correctly. The `sw_category` construct enables you to group the software so the GUI can present it in chunks that make sense to you. Because this example references an SD depot, it could have been created by `make_config`:

```
sw_source "ee_apps_depot" {
   description = "Electrical Engineering Application Depot"
   source_format = SD
   source_type = "NET"
   sd_server = "10.23.45.6"
   sd_depot_dir = "/var/opt/ignite/depots/Rel_B.11.11/ee_apps"
}
sw_category "Applications" {
   description = "User Applications"
}
sw_sel "EE CAD Package" {
   sw_source = "ee_apps_depot"
   sw_category = "Applications"
   sd_software_list = "EECad,r=1.2,a=HP-UX_B.11.11"
   impacts = "/var" 90524Kb
   impacts = "/sbin" 1248Kb
}
sw_sel "EE Routing Package" {
   sw_source = "ee_apps_depot"
   sw_category = "Applications"
   sd_software_list = "EERoute,r=2.4,a=HP-UX_B.11.11"
   impacts = "/usr" 12568Kb
   impacts = "/var" 26788Kb
}
```

Customizations Based on the Client Hardware

The *configuration file* syntax provides a large number of system attribute keywords that describe the *client*. Some examples are:

`disk[`*`hw_path`*`].size`	size of the disk at the specified *`hw_path`*
`memory`	amount of memory present on the client
`hardware_model`	string returned from **`uname -m`**
`,lla`	MAC address of the client

Using the logical expressions provided by *instl_adm*(4), you can use system attribute keywords to construct expressions in *configuration file*s so that a particular clause is only included in specific client situations. The basic format of these clauses is:

`(`*`x`*`){y}`

which translates roughly to "if the expression *x* is true, then do *y*."

For example, this clause sets the size of two *kernel* tunable parameters if the client has more than 4096 MB of memory:

```
(memory > 4096MB) {
    mod_kernel += "nproc (20+100*MAXUSERS)"
    mod_kernel += "maxuprc 1000"
}
```

As another example, use this if you want to run a script to do some particular graphics customizations, but you only want to do so when the client has the appropriate hardware:

```
(graphics[0].planes > 0) {
    post_config_script +=
```

```
        "/var/opt/ignite/scripts/multi_plane_graphics"
}
```

You can also specify multiple conditions. The following example installs a particular piece of previously defined application software if the client is a supported PA-RISC or Itanium-based server or workstation having at least two disks. A message lets you know why it is happening:

```
( (HARDWARE_MODEL ~ "9000/7.*" | MODEL ~ "ia64 .* workstation .*") & (num_disks >= 2) ) {
    note += "Installed application software contained in apps1."
    init sw_sel "apps1" = TRUE
```

You must use both `HARDWARE_MODEL` and `MODEL` because of the differences in the way the `uname` and `model` commands work on Itanium-based systems. For example on an Itanium-based client you can use the following commands to find this information:

`uname -m`

```
# ia64
```

`model`

```
# ia64 hp workstation zx2000
```

Notice that the response from the `uname` command is truncated so it is not possible to determine if the client is a server or a workstation, whereas on a PA-RISC client, the same command results in the following:

`uname -m`

```
# 9000/785
```

`model`

```
# 9000/785/J6000
```

Additionally, you can add an `else` clause so that a choice can be executed automatically. The following example uses a generic variable capability and mathematical expressions to set the primary swap size based on the amount of memory in the client:

```
(memory > 512Mb) {
    init _hp_pri_swap = 512Mb
}
else {
    init _hp_pri_swap = memory * 2
}
```

The preceding examples represent a few of the numerous ways that system attribute keywords can be used in client configurations and should not be considered an exhaustive list.

Customizations Based on User Selection

One of the ways you can use Ignite-UX to your advantage is to create a customized configuration independent of the client's hardware setup that can be selected for use repeatedly. For example, you might have some clients that you intend to use as NFS file servers and you would like to be able to quickly install these clients by selecting the same configuration from the GUI.

Let's assume that you have found NFS file servers to be more efficient if two of their *kernel* parameters are modified. NFS file servers also require some changes to the `/etc/rc.config.d/nfsconf` file using the `ch_rc` command.

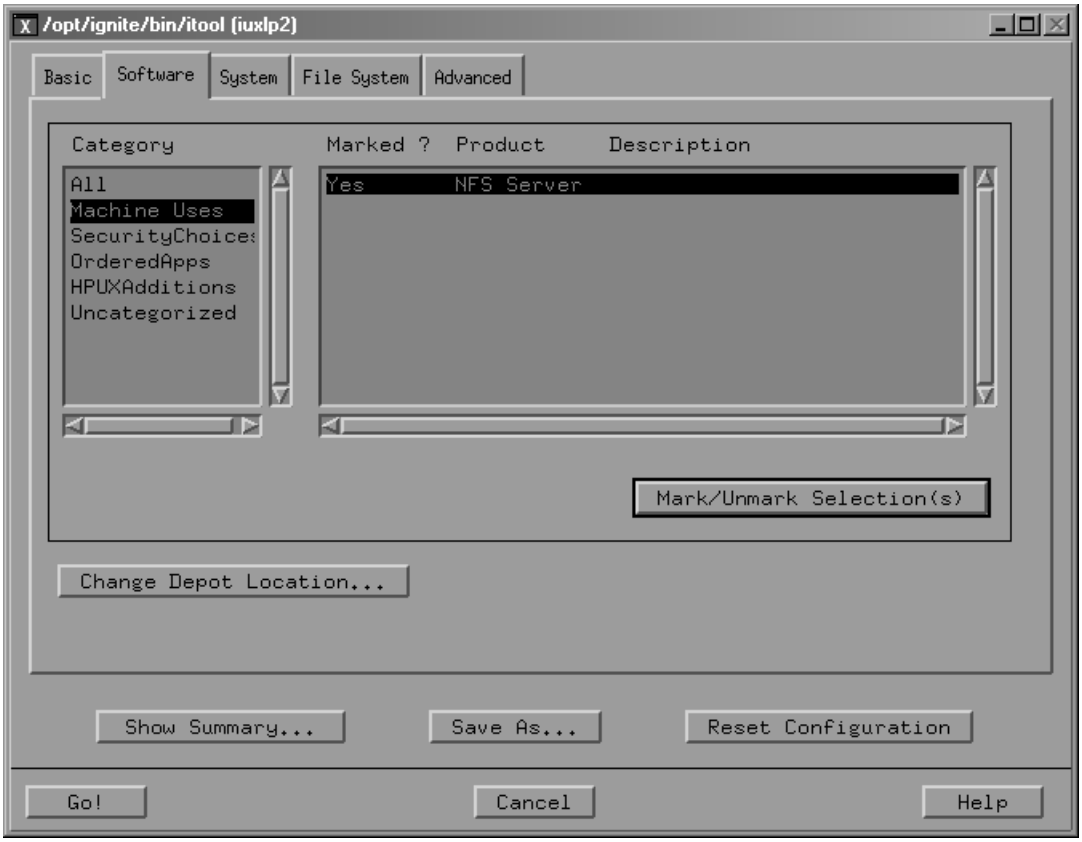
One alternative to effecting these changes manually is to define a custom software selection in `/var/opt/ignite/config.local` with a `sw_sel` clause, which then becomes a selection on the Software tab when you are configuring a new client installation. For example, the following clauses would automatically configure your NFS file servers:

```
sw_source "special configs" {
    source_format = cmd
}
sw_sel "NFS Server" {
    sw_category = "Machine Uses"
    sw_source = "special configs"
```

```
    mod_kernel += "dbc_min_pct 35"
    mod_kernel += "dbc_max_pct 65"
    post_config_cmd += "
        /usr/sbin/ch_rc -a -p NFS_SERVER=1
        /usr/sbin/ch_rc -a -p NFS_CLIENT=1
        /usr/sbin/ch_rc -a -p NUM_NFSD=8"
}
```

Figure 12-3 shows the Software tab when the NFS server configuration file is used. As shown, the selected category is Machine Uses as defined in the configuration file using the `sw_category` clause as in the previous example. This selection causes the kernel modifications and the `ch_rc` commands to be applied during the installation in addition to the other software categories you select.

Figure 12-3 Software Tab



Using the installation tabs to configure client installations is explained in Chapter 10: "Booting and Installing HP-UX on Clients Using the Server".

Avoid Archiving Patch Files

When you install HP-UX patches from SD *depot*s, the files that are patched are normally saved, in case you want to remove the patch at a later date. However, doing this takes up additional space in the `/var` directory, so you might want to turn this feature off.

In general, you can control this feature by using the following option of the `swinstall` command:

`-xpatch_save_files=[false|true]`

You can specify `swinstall` options in a *configuration file* using the `sd_command_line` keyword either at the global level or within individual `sw_source` clauses, depending on whether you want it specified for all installations or just for certain ones.

This option is specified in the `/opt/ignite/data/Rel_`*`release`*`/hw_patches_cfg` file. It is controlled by the configuration file variable `hp_patch_save_files` and can be modified

from the Ignite-UX GUI using the Additional Configuration Controls dialog box. For more information see the section "Additional... Button" (page 140).

To specify this option at the global level (for example in the `/var/opt/ignite/config.local` file), you can add the following line:

```
sd_command_line += " -xpatch_save_files=false "
```

To default the variable controlling the operating system patches to "no", add the following to `config.local`, which must be listed after `hw_patches_cfg` in the `/var/opt/ignite/data/INDEX` file:

```
init _hp_patch_save_files = "NO"
```

For more details, see the "Managing Patches" chapter in the *Software Distributor Administration Guide* and the *Patch Management User Guide for HP-UX 11.x Systems*.

Debugging Configuration Files

Designing a configuration file to meet your needs can be a very tedious task. It usually requires a lot of trial and error. You will need to install systems to test your configuration files. The `instl_dbg` command is designed to help you with configuration file design and avoid wasting time attempting to install with invalid configuration files. With the `instl_dbg` command you can:

- Parse a client's configuration files for syntax errors.
- Display and set variables, software selections, and use models.
- Detect errors that may occur during a client installation due to faulty configuration files, such as missing software depots/archives.

After you have developed a new configuration file, run the `instl_dbg` command from the Ignite-UX server to ensure that there are no errors in your configuration files. The `instl_dbg` command first scans for any syntax errors. Then, `instl_dbg` substitutes variables, use models, and software selections (`sw_sel`) with real values, and writes a single, unified configuration file if the `-f` option is specified. Now, you can use this file to install the client. Other options are available for more thorough checking or to provide more details. For more information, see *instl_dbg*(1M).

Examples

Enter the following command to debug a client configuration file named `system1`, print the debugged configuration file to `stdout`, and then save the debugged configuration file to a new file, `system1_cfg.out`:

```
instl_dbg -D /var/opt/ignite/clients/system1 -d \
-f system1_cfg.out
```

To debug this same client configuration file, `system1`, to show the effects upon the *disk layout* when the `_hp_disk_layout` and `_hp_pri_swap` values are changed, and then print the "very, very verbose" (`-vvv`) output to the console and to the `system1_cfg.out` file, you would enter the following command:

```
instl_dbg -D /var/opt/ignite/clients/system1 -d \
-V _hp_disk_layout="Whole disk (not LVM) with HFS" \
-V _hp_pri_swap=500MB -vvv -f system1_cfg.out
```

For more information, see *instl_dbg*(1M).

Using Post-Installation Scripts

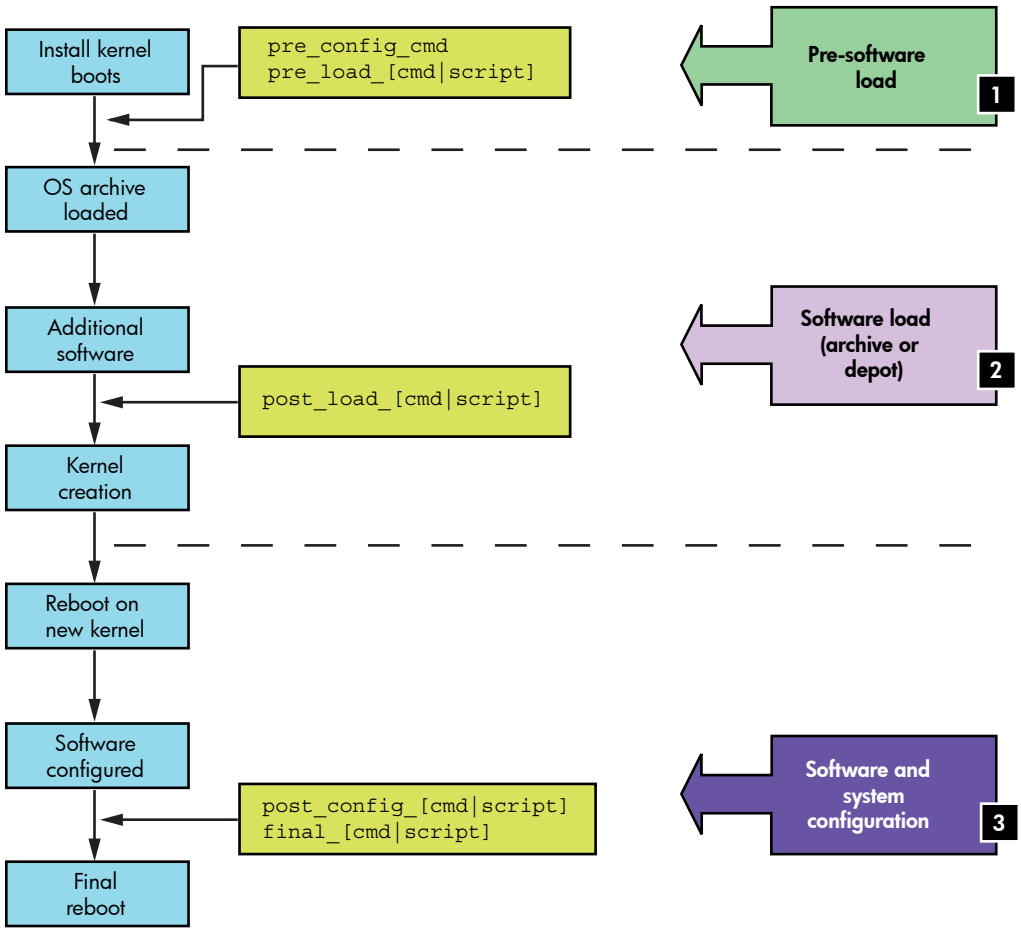
You can perform any number of tasks on a client after installing the operating system by providing a script to be executed. This section provides some common examples to assist you. Additionally,

you can write your own scripts to mount additional disk drives, add additional software, or modify configurations based on system use.

How the Installation Functions

It is important to understand how an installation functions when using Ignite-UX. The diagram in Figure 12-4 illustrates when the various commands and scripts are executed during the installation process.

Figure 12-4 Installation Process



1 The commands available are very limited and execute before any volume groups or *disk group*s are created.

2 The available preload commands are limited and the script executes before any *archive*s or *depot*-based software are installed.

3 The `install.log` file is closed, then the final scripts execute; any messages that are generated appear only on the console.

There are a number of points in the installation process in which you can force scripts or commands to be run. See the "Command and Script Execution Hooks" section of *instl_adm*(4) for specifics.

Adding a Post-Installation Script

Example

One task you generally need to perform after a new operating system installation is setting up printers. To automate this process, you can write a script that performs the HP-UX commands for adding a printer.

Following is a script for adding a remote printer named `remoteprinter`, making it the default printer, and then enabling the printer so that it is ready the next time that the `lp` scheduler is started:

```
#!/usr/bin/sh
# Post process IUX script to add a local default printer
# Performing task "Add Remote Printer": Adding "remoteprinter"
#
/usr/sbin/lpadmin -premoteprinter -ormserver.com -orptsslj \
 -mrmodel -v/dev/null -orc -ocmrcmodel -osmrsmodel
/usr/sbin/lpadmin -dremoteprinter
/usr/sbin/accept remoteprinter
/usr/bin/enable remoteprinter
```

NOTE: Make sure that the directory the post-installation script resides in is available to `tftp` by examining and changing the `/etc/inetd.conf` file.

1. Create a script to set up access to a default network printer.
2. Ignite-UX post-installation scripts are defined using the `post_config_script` variable. To add the script you created in the previous step, edit your `core_cfg` configuration file and add the following line:

   ```
   post_config_script += \
   "/var/opt/ignite/scripts/install_default_printer"
   ```

   This sets the `install_default_printer` script to be run as a post-installation process on the client. This line should stand alone, and be placed *outside* of any clauses in the file (such as a `sw_sel` clause). The script set to execute with the `post_config_script` variable executes after all software has been installed and the system has been booted with its final kernel, but *before* any of the normal `/etc/rc` startup scripts have been run.

   You can change the default installation script behavior from the Actions menu, by choosing **Install Client**, and then choosing **New install**. Then click on the Advanced tab and select the post-installation scripts that you want to be executed.

TIP: If you want to make a script available under all configurations, add it to the `/var/opt/ignite/data/INDEX` file. For example, to execute the `install_default_printer` script following every client installation, add the following line to the end of this file:

```
scripts {"/var/opt/ignite/scripts/install_default_printer"}
```

This script is made available for selection on the Advanced tab for all configurations. For more information, see "Advanced Tab" (page 161).

13 Automating Installations

This chapter shows how to use the `bootsys` command and *configuration file*s to automate the Ignite-UX *installation* process.

Starting a Noninteractive Installation with bootsys

Ignite-UX can install HP-UX on a client noninteractively using the default configuration specified in the `/var/opt/ignite/data/INDEX` file for the given operating system. The `/var/opt/ignite/data/INDEX` file can have a unique clause for each operating system you want to support. Alternatively, you can choose from other configurations that exist as unique clauses in the `INDEX` file to use with *bootsys*(1M) from the command line.

To start an automatic, noninteractive installation, execute the `bootsys` command from the server as follows:

`bootsys -a -v [-i configuration][-f]client_hostname`

`-a` Specifies an automatic installation.

`-v` Specifies verbose mode.

`-i` Specifies a default configuration for installing the client. If not set, the default specified on the Ignite-UX server is used. If an incorrect configuration or one that does not match the last configuration saved for the client is selected and the `-f` option is not used, an error results.

`-f` Forces Ignite-UX to disregard and delete the previous configuration information that was saved for a client during the last installation.

You can define the default configuration from the Ignite-UX GUI Server Options menu (see "Configuring Server Options" (page 43)) or you can set it manually with a `=TRUE` statement immediately following the closing of a *cfg clause* in the `/var/opt/ignite/data/INDEX` file.

After you have initiated the installation, Ignite-UX contacts the client and extracts its hostname, IP address, and default *gateway*. The configuration you specified is installed. After the installation completes, Ignite-UX resets the hostname, IP address and gateway to their original values. This installation configuration is then automatically saved, or overwritten if one exists, for this client and can be reused the next time an installation is needed. For `bootsys` to function, access to the client using `remsh` is required; if a remote shell is not available, the `bootsys` command prompts you for the `root` password of the client.

A noninteractive installation using `bootsys` is the quickest way to install a client. You can install a client with a custom configuration using `bootsys` with the following simple four-step process:

1. Create the *configuration file*s that describe how you want the client to be installed.
2. Save the configuration files that you created.
3. Add a cfg clause to the `/var/opt/ignite/data/INDEX` file that describes your custom configuration.
4. Execute the `bootsys` command, specifying the custom configuration.

For more information regarding configuration files, see "Using Configuration Files" (page 177) and the white paper, *Ignite-UX Custom Configuration Files*, available at

`http://www.hp.com/go/ignite-ux-docs`

For details on how to select a configuration, force its use, and use the many other installation options that are available, see *bootsys*(1M).

Using a Saved Configuration

After you specify a configuration for a particular client during an installation, you can save the result as a named configuration. This saves any changes that you made during the installation

session for use in subsequent sessions. For more information regarding configuration files, see "Using Configuration Files" (page 177).

To specify a saved configuration as the default, define it in your server's `/var/opt/ignite/data/INDEX` file, or use the `bootsys` command with the `-i` option and name you have given it.

Specifying Defaults in the `config.local` File

The `/var/opt/ignite/config.local` file is normally included in every configuration (`cfg`) clause in the `/var/opt/ignite/data/INDEX` file. It provides a convenient location, in which to store default parameters that are the same for all configurations. Typically these are networking parameters, default software selections and *kernel* modifications.

The following is an example of what a `config.local` file could look like. The `sw_sel` definitions depend on what you have defined in other configuration files on the Ignite-UX server.

```
dns_domain="corp.com"
dns_nameserver[0] = "10.2.72.2"
nis_domain="corpdomain"
wait_for_nis_server=FALSE
root_password="JPDw02THwAhiQ"
timezone="MST7MDT"
ntpdate_server="10.1.48.1"
init sw_sel "Misc_Patches"=TRUE
init sw_sel "B3919DA_AGP"=TRUE
mod_kernel += "maxuprc 100"
set_kernel += "dbc_max_pct 20"
```

NOTE: After making manual edits to `config.local`, always use the following command to verify that the syntax is correct and to avoid installation errors:

```
instl_adm -T -f config.local
```

Setting Defaults with instl_adm

Some network parameters must be known by the client when it first *boot*s. The netmask and *gateway* can be supplied in the installation or RAM *file system* (*[W|V|I]INSTALLFS*). You can do this by using the `instl_adm` command, which has options to set the client's netmask, gateway, Ignite-UX `/tftp` server, and so on.

For example, you might want to set the keyboard language so you are never prompted for it when booting from Ignite-UX. To do this, first display the current configuration information and redirect it to a temporary file for editing. Then, edit the file to include the new configuration information. Finally, apply the new configuration information from the temporary file.

In the following example, the keyboard language is set to `PS2_DIN_US_English`.

```
# instl_adm -d > /tmp/installfs.out
# echo '_hp_keyboard="PS2_DIN_US_English"' >> /tmp/installfs.out
# instl_adm -f /tmp/installfs.out
```

For more information, see "Using Configuration Files" (page 177) , *instl_adm*(1M), and *instl_adm*(4).

Using the Per-Client Configuration File

When Ignite-UX begins an installation session, it scans the directory `/var/opt/ignite/clients` for a directory matching the MAC address of the client. For example, if the MAC address of the client is `0x08000992E346`, Ignite-UX looks for a file named `config` in `/var/opt/ignite/clients/0x08000992E346`. However, to enable you to find clients easily Ignite-UX also creates a symbolic link between the client's name and the MAC address. If the client has more than one LAN interface, the MAC address of each interface is used until a matching directory is found.

Ignite-UX overwrites this client-specific config file at the end of the installation, creating a record of the parameters defined for this client's most recent installation.

The configuration data in this file is used to override the default installation parameters. For more information regarding configuration files, see "Using Configuration Files" (page 177).

The easiest way to create a client-specific custom configuration file is to start with one of the default files provided by Ignite-UX. However, if you have previously installed a client, you can start with a configuration file in the `/var/opt/ignite/clients/client` directories. It is best to use one from a client similar to the one you want to install. You could use this as the basis for your new configuration file.

The following is an example configuration file:

```
cfg "HP-UX B.11.11 archive"=TRUE
#
# setting _hp_cfg_detail_level to ipvs indicates that the file
# contains information about which cfg INDEX selection to be used,
# the variable settings, software selection settings, and system
# parameters.
_hp_cfg_detail_level="ipvs"
#
# Variable assignments
init _hp_pri_swap=68304K
init _hp_root_disk="2/0/1.5.0"
init _hp_sec_swap=0K
init _hp_root_grp_disks=1
init _hp_root_grp_striped="NO"
init_hp_locale="SET_NULL_LOCALE"
init_hp_keyboard="PS2_DIN_US_English"
init _hp_default_final_lan_dev="lan0"
init _hp_boot_dev_path="2/0/1.6.0"
#
# Software Selections
init sw_sel "English"=TRUE
#
# System/Networking Parameters
init _hp_custom_sys="Current System Parameters"
_hp_custom_sys help_text "Final System/Networking Parameters"
{
final system_name="server"
final ip_addr["lan0"]="10.2.75.14"
final netmask["lan0"]="255.255.248.0"
final dns_domain="xx.corp.com"
final dns_nameserver[0]="10.2.72.254"
TIMEZONE="MST7MDT"
is_net_info_temporary=TRUE
}
```

To customize this configuration file, change the final networking parameters in the example file to the correct values for your client. For example:

```
final system_name="system11"
final ip_addr["lan0"]="10.2.75.193"
```

It is also possible to add *kernel* parameters to this file. See "Setting Installation Parameters Dynamically" (page 194).

IMPORTANT: You should update the variable `_hp_cfg_detail_level` to reflect the parameters that you have modified to indicate that they should be used by Ignite-UX; by default this variable is set to `ipvs`. This variable is fully described in *instl_adm*(4).

To perform an automatic installation of a client using a custom configuration file, execute the following steps on your Ignite-UX server:

1. Determine the MAC address of the client, either through the `boot_admin` commands at boot time or by using *lanscan*(1M) after the client is booted.

   For more information, see *boot*(1M), *efi*(4), *isl*(1M), and *pdc*(1M).

2. Create the following directory, assuming the MAC address is `0x08000992E346`:

   `mkdir /var/opt/ignite/clients/0x08000992E346`

3. Copy an example configuration file to be used as a starting point:

   `cp config /var/opt/ignite/clients/0x08000992E346/config`

4. These files are accessed using NFS so you must ensure that the correct ownership and directory access permissions are assigned.

   Set the client directory to 644 and change the ownership to ensure that `bin` can read and write to the directory:

   ```
   chmod 644 /var/opt/ignite/clients/0x08000992E346
   chown bin:bin /var/opt/ignite/clients/0x08000992E346
   chown bin:bin /var/opt/ignite/clients/0x08000992E346/config
   ```

5. Edit your new configuration file to define your client-specific information.
6. Run `bootsys` on the Ignite-UX server to begin the installation:

   `bootsys -a -v client_hostname`

   The client is booted and the operating system is installed without interaction using the client-specific configuration file you created. Errors are reported in `/var/opt/ignite/clients/0x08000992E346/install.log` and in the client status dialog box (see "Client Status... Dialog Box" (page 165)) in the Ignite-UX GUI on the Ignite-UX server.

Scheduling Installations

Client installations can also be automated using the `cron` *daemon*. To perform repeated installations for a client, you could use `crontab` entries. For more information, see *cron*(1M) and *crontab*(1).

For single installations, use the `at` command. For example, to perform an installation of a client at 8:00 PM using the `at` command, as `root` enter:

```
at 8:00pm
bootsys -a -v client_hostname
```

Press **Ctrl-D**

NOTE: This example requires `remsh` access from the Ignite-UX server on which you execute the `bootsys` command to the client because the server cannot obtain the `root` password by prompting you.

Setting Installation Parameters Dynamically

Ignite-UX can make intelligent decisions about installation parameters when it runs, based on information it reads from the client. Instead of forcing static values (for example, swap size or kernel parameters), Ignite-UX can determine the best values for these based on the characteristics of the client.

This feature can make the configurations that you set up more general purpose and limit the need for multiple, custom configurations to handle minor system differences.

These decisions are specified in a C-like language and grammar unique to Ignite-UX. For more information, see *instl_adm*(1M).

The following examples show how Ignite-UX set installation parameters dynamically.

Example 1

In this example, if the client's root disk is smaller than 8.5 GB and has less than 1 GB (1024 MB) of random access memory (RAM), then the primary swap partition size (_hp_pri_swap) is set to 512 MB; otherwise, it is set to 1 GB. If the disk is larger than 8.5 GB, then _hp_pri_swap is set to 1024 MB or 4096 MB, depending on the memory size.

Add the following lines to the end of the file /var/opt/ignite/config.local as this file is typically referenced in all configurations defined in the /var/opt/ignite/data/INDEX file:

```
# default to very minimal swap of 8500MB
# unless the disk is larger than 8.5 GB
# and we have more than 512MB RAM

(disk[_hp_root_disk].size < 8500MB)
{
    (memory < 1024Mb) {
        init _hp_pri_swap=512MB
    } else {
        init _hp_pri_swap=1024MB
    }
} else {
    (memory < 2048Mb) {
        init _hp_pri_swap=1024MB
    } else {
        init _hp_pri_swap=4096MB
    }
}
```

You could also put this in a separate file, for example, /var/opt/ignite/data/Rel_B.11.11/custom_cfg, and then add that file name to the desired *cfg clause* in the /var/opt/ignite/data/INDEX file.

Additionally, you could add this to the configuration file created for automatic installations. If the _hp_pri_swap parameter is set later in the order of files searched in the cfg clause, this setting is overwritten. The configuration file parsing order and precedence is documented in *instl_adm*(4) and in "Using Configuration Files" (page 177).

CAUTION: The per-client configuration file in /var/opt/ignite/clients/*client* used for noninteractive installations is overwritten as part of the installation process.

Example 2

To force the installation of a patch bundle if the client matches a regular expression, such as a hardware architecture type like J6700 or J6750 PA-RISC workstations, add the following lines to the end of the /var/opt/ignite/data/Rel_B.11.11/custom_cfg configuration file:

```
# check for H/W model J6700 or J6750
# and add the Misc_Patches bundle if true

(hardware_model ~ "9000/785/J67[05]0") {init sw_sel "Misc_Patches" = true}
```

Example 3

In this example, Ignite-UX runs a previously created post-installation script and increases a tunable kernel parameter if it determines the client is a C3600, C3650, C3700, or C3750 workstation. If not, it sets a default value for the kernel parameter:

```
post_config_script += "/var/opt/ignite/scripts/new_C_series_special"
(HARDWARE_MODEL == "9000/785/C3[67][05]0") { mod_kernel += "maxuprc 300"
} else {mod_kernel += "maxuprc 100"}
```

Example 4

You can specify an entirely different custom configuration based on the size of the client's RAM and disk size. To set the default configuration dynamically to put into effect, it must go into the *[W|V|I]INSTALLFS* file using *instl_adm*(1M) as in the following script excerpt:

```
# For a system with only one disk and small memory, select
# the "small system configuration"
(num_disks == 1 & memory < 256MB )
{cfg "small system configuration" = true}
```

Checking Modified Files for Errors

You can validate the syntax of a file that is *not yet* in the `/var/opt/ignite/data/INDEX` file to avoid installation problems later, by entering the following:

`instl_adm -T -f file`

After you modify configuration files, *always* validate the syntax of all of the files that are referenced in the cfg clauses in the `/var/opt/ignite/data/INDEX` file, by entering the following:

`instl_adm -T`

14 Creating Your Own Boot and Installation Media

This chapter explains how to create custom HP-UX *boot* and *installation media*.

Installation media can be a tape or DVD containing:

- A *golden archive*
- A *recovery archive*
- A software *depot*
- A golden or recovery archive, plus a software depot

NOTE: It is possible to create a bootable USB flash drive that works with Integrity systems. See the *Ignite-UX USB Memory Stick Boot* white paper, available at http://www.hp.com/go/ignite-ux-docs.

All installation media are bootable. To create installation media, you need a basic knowledge of Ignite-UX functionality.

Why Use Custom Boot and Installation Media?

You might want to build boot or *custom installation* media if:

- You have to recover systems that cannot boot from a recovery tape or the network, so boot media is required for the *two-step media recovery* process.
- You have a large number of systems that are basically identical, and the networking does not allow easy or fast access to an Ignite-UX server, perhaps due to dispersed geography or for security reasons. For *common configuration* installation solutions, use a *golden image*. Golden image creation is described in Chapter 11 (page 169).
- You want a disaster *recovery image* of a system. For system-specific recovery solutions, you should use a recovery image. Recovery image creation is described in Chapter 15: “Recovery” (page 209).

Building PA-RISC Boot and Installation Tape

This section describes building a tape for just booting, or for boot and installation on PA-RISC systems. This functionality is not supported on Itanium-based systems, although you can utilize “Tape Recovery With No Tape Boot Support — Two-Step Media Recovery” (page 234).

IMPORTANT: The media and data format (density and compression) of the installation tape you create must be compatible with the tape devices of client systems on which it will be read. You should consider writing the tape using a device special file (*DSF*) that selects the most compatible data format settings.

Possible Tape Contents

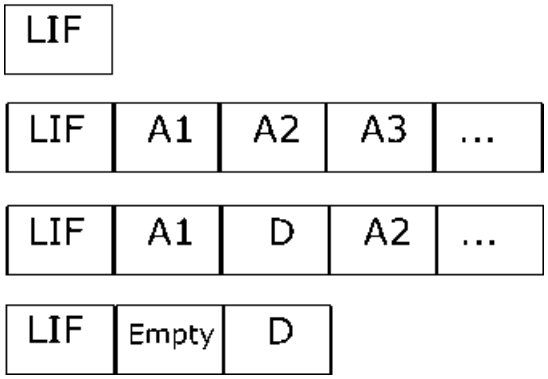
A PA-RISC tape may consist of

- Just a *LIF volume*.
- A LIF volume followed by one or more *archive*s (A1, A2, A3, ...).
- A LIF volume followed by an optional archive (A1, or Empty), a serial *depot* (D), and zero or more additional archives (A2, ...).

By far, the most common configuration is a LIF volume followed by one archive. The archive can be a *golden archive* or a *recovery archive* depending on the purpose of the media.

The high-level structures of possible PA-RISC installation tape layouts are shown below.

Figure 14-1 Possible PA-RISC Installation Tape Layouts



Logical Interchange Format

The first file on a PA-RISC bootable tape is a *logical interchange format* (LIF) volume containing all the components required to *boot* from the tape. It also includes the Ignite-UX toolset and configuration information that controls how Ignite-UX will operate. Additionally, this tape file will include *configuration file*s that describe installation functionality. See Appendix C (page 259) for information regarding the contents of a *LIF volume*.

For PA-RISC systems you can create a boot tape that is made up of the LIF volume only. This boot tape could then be used to boot a system, and then the software could be downloaded over the network.

The `make_medialif` command creates the LIF volume. A typical LIF volume would have the following contents:

- *ISL*
- *AUTO*
- *INDEX*
- *CONFIG*
- *HPUX*
- *[W|V|I]INSTALL*
- *[W|V|I]INSTALLFS*
- *INSTCMDS*
- *INSTCMDSIA*
- *SYSCMDS*
- *SYSCMDSIA*
- *RECCMDS*
- *RECCMDSIA*
- *SCRIPTS*

See Appendix C (page 259) for a full description of LIF contents.

IMPORTANT: The first 8 KB of the *install file system* ([W|V|I]INSTALLFS) can contain Ignite-UX configuration content. If it does, this information is used first and can override configuration statements in the LIF `INDEX` and `CONFIG` files. If the tape contains a golden archive, you will want to consider whether the tape should contain special config content to control system setup in some custom way. If the golden archive will be used with different system configs, conditional expressions can be used to adapt the config appropriately, such as `run_ui`, which controls whether the installation is interactive or not, and `server` and `sd_server`, which can be used to control network installation. See *instl_adm*(4) for more information.

Archives and Depots

The LIF volume is followed by the first archive (a *golden archive* or a *recovery archive* depending on the purpose of the media) or is empty if the installation is solely from a software depot.

The first archive is then followed by either a serial *depot* or another archive. There can only be one serial depot on a tape, and it must be the third file on the tape. If the tape includes a serial depot, the `make_config` command must be used to create the *config file* for the depot content. This configuration file is then modified to reflect the final destination of the depot and the depot is written to the tape.

The previous files can be followed by one or more additional *archive*s, limited only by the capacity of the tape. These archives are optional.

TIP: If a tape will contain multiple archives, you might want to place the most commonly used archive first so installations take less time.

Creating and Modifying an Archive Configuration File for Tape

Create the archive configuration file — Proceed as described in Chapter 11 (page 169) or Chapter 15 (page 209), depending on the usage of the installation tape.

Modify the archive access attributes — Change the following archive access attributes in the `sw_source core` clause:

- Find the `source_type` attribute and change the value from `NET` to `MT` (for example, `source_type=MT`).
- Find the `change_media=FALSE` attribute line and remove the comment hash mark (`#`).
- Find the `nfs_source` attribute line and place a comment hash mark before it (for example, `#nfs_source=`).

  The above edits change the `source type` from network (`NET`) access (which is `NFS`, `ftp` or `remsh`) to magnetic tape (`MT`). Since the archive is going to reside on the same media as the configuration file, `change_media` is set to `FALSE` by uncommenting that attribute. To stop the NFS mount attempt, the `nfs_source` attribute is commented out.

Modify the archive path and impacts — In the `HARDWARE_MODEL ~ 9000.*` clause:

- Find the `archive_path` attribute and set the value to the number of EOF markers that have to be skipped to get to the archive (`archive_path=`*n*). There is an EOF marker after the *LIF volume*, archives, depots, and empties. For example, to reach the first archive on the tape, the `archive_path` would have to be set to "1", indicating the LIF volume EOF would be skipped and the archive begins thereafter.
- Find the `impacts` attribute with the value `/ 27KB` and change the value to what is reported by `archive_impact`.
- Optionally, you can find the `sw_sel` and `description` attributes and change their values to something more descriptive and applicable for your situation. These modified values are visible on Basic tab of the Ignite-UX GUI.

Other edits —

- The `archive_type` must match what was done by `make_sys_image`. For more information about `archive_type`, see *instl_adm*(4).
- If you have only one archive, as in this example, you can delete the entire clause called `golden image2`. It is included in the template file in case you have two different types of archives, such as one for 32-bit and one for 64-bit architecture.
- If more than one archive per architecture is on the media, use an `exrequisite` attribute between them so only one archive can be selected at one time.

IMPORTANT: Do not change anything else in the file, unless you are very sure of what you are doing. In particular, it is potentially dangerous to change the `sw_category` and other `sw_source` and `sw_sel` attributes not mentioned above.

Creating and Modifying a Serial Depot and its Configuration File for Tape

A depot put on a tape is called a serial depot. It can exist as a regular file, but it cannot be registered, which means it cannot be accessed remotely.

Create the serial depot configuration file – Issue the following command:

```
# make_config -s /var/tmp/depot -c /var/tmp/depot_cfg
```

Modify the serial depot config file – The depot must always be the third file on the tape, so there is no need to specify a path to the depot. Remove the following lines:

- `sd_server = IP_address`
- `sd_depot_dir = /var/tmp/depot`

Change the `source_type` attribute from `NET` to `MT`:

- `source_type=MT`

Create the serial depot – To create a serial depot from `/var/tmp/depot` and store it in `/var/tmp/serialdepot`, enter:

```
# swpackage -s /var/tmp/depot \
  -x media_type=tape @ /var/tmp/serialdepot
```

PA-RISC Installation Tape Creation Example

Assumptions

The following example assumes you have created a golden archive and a depot containing all the applications you want to install. Including a serial depot is optional and is not needed if the tape will only contain a golden archive, which is normally the case.

The golden archive in this example is assumed to have been created using `make_sys_image` on a system running HP-UX 11i v1 (B.11.11). See Chapter 11 (page 169) for information regarding golden archive creation.

The archive is in `gzip` compressed `tar` format and is named `/var/tmp/myOSarchive.gz`.

The `archive_impact` command should be used to obtain disk-space usage information for this archive so that disk space impacts can be included in the config content. The `make_config` command should be used to create a *config file* that describes the archive content. It's assumed that a config file named `myOSarchive_cfg` has been created for this archive.

The optional depot is named `/var/tmp/serialdepot` and its config file is named `/var/tmp/depot_cfg`.

Example PA-RISC Installation Tape Creation

1. **Create the *LIF volume*.**

   The LIF will be created in `/var/tmp/lifvol` by entering the following `make_medialif` command:

   ```
   # make_medialif -f /opt/ignite/data/Rel_B.11.11/config \
     -f /var/tmp/myOSarchive_cfg \
     -f /var/tmp/depot_cfg \
     -l /var/tmp/lifvol -r B.11.11
   ```

   The LIF volume now contains the default settings Ignite-UX provides, plus information on the archives and depot.

2. **Modify the install file system configuration.**

   You can control the environment in which Ignite-UX operates at client installation time by changing the `run_ui` and `control_from_server` configuration parameters in the *install file system* (*[W|V|I]INSTALLFS*). When `control_from_server` is true, an attempt is made to contact the Ignite-UX server as defined in the configuration information. For more information, see *instl_adm*(4).

   To set `run_ui` and `control_from_server` using `instl_adm`, execute the following steps:

   - Create a temporary configuration file:

     ```
     instl_adm -d -F /var/tmp/lifvol > /var/tmp/cfg
     ```

   - Open `/var/tmp/cfg` for editing.
   - Add or change the `run_ui` variable.

     For an interactive installation, `run_ui=TRUE`.

     For a noninteractive installation, set `run_ui=FALSE` — no interaction occurs and the installation proceeds according to the configuration information provided to Ignite-UX.

     NOTE: To get a fully-automated installation, you usually need to set `control_from_server=FALSE` and `env_vars+="INST_ALLOW_WARNINGS=10"`, otherwise the warning about an operating system already on the disk will cause the installation to become interactive.

   - Add or change the `control_from_server` variable.

     To control the installation from the Ignite-UX server, set `control_from_server=TRUE`. An attempt will be made to contact the server.

   - Check the syntax of your changes to avoid writing errors to the install file system:

     ```
     instl_adm -T -f /var/tmp/cfg
     ```

   - Write the modified temporary configuration file into the LIF volume:

     ```
     instl_adm -F /var/tmp/lifvol -f /var/tmp/cfg
     ```

   - Verify your changes:

     ```
     instl_adm -d -f /var/tmp/lifvol
     ```

3. **Select the appropriate tape device file.**

   Create the tape using the proper density, no compression, and a no-rewind device file. See *insf*(1M), *tape*(7), and *stape*(7) for more information about tape device special file creation and selection.

   Note that a no-rewind tape device special file must be used. Normally, a default AT&T semantic tape device special file is used, for example, `/dev/rmt/0mn`.

4. **Create the installation tape.**

   This series of commands creates a tape with DDS-1 characteristics and copies the files in the correct tape layout as described in "Possible Tape Contents" (page 197):

```
# mt -t /dev/rmt/0mn rew
# dd if=/var/tmp/lifvol of=/dev/rmt/0mn obs=2k
# dd if=/var/tmp/myOSarchive.gz of=/dev/rmt/0mn obs=10k
# dd if=/var/tmp/serialdepot of=/dev/rmt/0mn obs=10k
# mt -t /dev/rmt/0mn rew
```

NOTE: You must ensure you use a no-rewind device file, such as `/dev/rmt/0mn`. This prevents the tape device from rewinding between the successive `dd` commands in this step and overwriting files that were previously written to the tape.

The tape is now ready to use for the installation of PA-RISC systems.

Creating a Boot CD/DVD or an Installation DVD

You can use scripts supplied by Ignite-UX to create boot and archive-based CD/DVDs as outlined in "Boot and Archive-Based CD/DVDs" (page 203).

To create depot-based installation DVDs, follow the procedures provided below in "Depot-Based DVDs" (page 206).

Assumptions

This section assumes the system creating the media has a DVD writer.

If your system does not have a DVD writer, you can create the ISO image and then copy it to another system equipped with a DVD writer to burn the media.

If your system is enabled with the HP Integrity Integrated Lights-Out Management Processor (iLO MP) virtual media (vMedia) feature, you can generate an ISO image that can be used with iLO vMedia virtual devices. For more information, see Appendix D (page 263) and the *HP Integrity iLO 2 MP Operations Guide* available at http://www.hp.com/bizsupport.

File and ISO Image Size Considerations

The ISO image must fit on the DVD media (4.7 GB).

Some implementations of ISO9660 limit the maximum file size to 2 GB.

Prior to Ignite-UX version C.6.5 for HP-UX 11i v2, and C.6.8 for HP-UX 11i v1, the total size of the ISO image cannot be larger than 2 GB.

For more information, see the sections FILE SIZES and IMAGE SIZE at the start of the `make_media_install` script.

Boot and Archive-Based CD/DVDs

With one simple command, a CD/DVD can be created that supports *two-step media recovery*. Additionally, `make_media_install` can write *archive*s to a bootable DVD.

All HP-UX 11i v2 and 11i v3 bootable CD/DVDs created with `make_media_install` support both Itanium-based and PA-RISC *boot*.

The `make_opticaldisc_recovery` script can be used to create a *recovery image* and then write it to a DVD. The script creates the recovery image with `make_net_recovery`, and then calls `make_media_install` to write the recovery image to DVD. This script is to be run from the client system and requires one argument — the hostname or IP address of the Ignite server. The script first creates a recovery image of the client and stores it on the Ignite server. Then, the recovery image is written to the *Ignite server's* DVD writer. The `make_opticaldisc_recovery` script was introduced in Ignite-UX version C.7.2.

You can find both scripts in the directory

```
/opt/ignite/data/scripts/examples
```

Detailed usage information is available via

```
# make_media_install -?
```

and summary information is available via

```
# make_opticaldisc_recovery -?
```

Additional information is available at the start of the scripts.

Boot CD/DVD Examples

Two-step media recovery using a bootable CD/DVD is necessary to restore from tapes created with the Ignite-UX command `make_tape_recovery` on certain Itanium-based systems that do not support direct tape *boot*. See "Tape Recovery With No Tape Boot Support — Two-Step Media Recovery" (page 234) for more information on two-step media recovery, and the *Ignite-UX*

Installation Booting white paper available at http://www.hp.com/go/ignite-ux-docs for information on configurations supporting HP Integrity native tape boot.

Two-step media recovery requires the version of Ignite-UX on the tape, on the bootable CD/DVD, and on the system creating the CD/DVD media to be identical. If you do not have a matching Operating Environment DVD, the `make_media_install` script can be used to easily create a bootable CD/DVD with the correct Ignite-UX version.

Create HP-UX 11i v3 bootable CD/DVD media for two-step media recovery

On a system with a CD/DVD writer and the same version of Ignite-UX as on your tape, execute the following command (*`iso_directory`* is the target directory for the `.iso` media file and the `-r` option specifies the release, such as `B.11.23` or `B.11.31`):

```
# make_media_install -r B.11.31 -m iso_directory
```

Create HP-UX 11i v2 bootable media on USB DVD drive for two-step media recovery

In order to write to a USB DVD drive on an HP-UX 11i v2 system, you must install the patch PHKL_37814 and use the `-c` option with `make_media_install`. In the following example, the *`iso_directory`* is the target directory for the `.iso` media file, the `-r` option specifies the release, and the *`device_file`* is the DVD device file such as `/dev/rdsk/c0t0d0`.

```
# make_media_install -r B.11.23 -m iso_directory -c device_file
```

Installation Archive-Based DVD Examples

Installation media can be built for PA-RISC systems, Itanium-based systems, or both since the bootable DVD created with `make_media_install` on HP-UX 11i v2 or 11i v3 will support *boot* for either architecture. The *archive*, however, can only be installed on a system matching the architecture type it was created from.

The `make_media_install` script will generate impact statements for you, using either the `-l` command line option or the default `$ImpactLevelDefault`. See the start of the script for more information.

NOTE: Building installation DVDs requires the `mkisofs` and `growisofs` commands. Beginning with Ignite-UX version C.6.2.*x*, these commands are delivered in the `/opt/ignite/lbin/` directory.

Put an Itanium-based HP-UX 11i v3 golden archive on a DVD

On a system with a DVD writer, execute the following commands. (The *`iso_directory`* is the target directory for the `.iso` media file. The `-r` option specifies the release — valid values for Itanium-based systems are `B.11.23` and `B.11.31`. The `-o` option specifies the hardware architecture — the only valid value for an Itanium-based system is IA.)

```
# mkdir /var/opt/ignite/media/pseudo_root/archive
# cp goldenarchive.gz /var/opt/ignite/media/pseudo_root/archive
# make_media_install -r B.11.31 -o IA -a goldenarchive.gz -m iso_directory
```

Put a PA-RISC HP-UX 11i v2 golden archive on a DVD

On a system with a DVD writer, execute the following commands. (The *`iso_directory`* is the target directory for the `.iso` media file. The `-r` option specifies the release, such as `B.11.11`, `B.11.23`, or `B.11.31`. The `-o` option specifies the hardware architecture — valid choices on a PA-RISC system are: `32`, `64v`, or `64w`.)

```
# mkdir /var/opt/ignite/media/pseudo_root/archive
# cp goldenarchive.gz /var/opt/ignite/media/pseudo_root/archive
# make_media_install -r B.11.23 -o 64w -a goldenarchive.gz -m iso_directory
```

Put two HP-UX 11i v2 golden archives, one Itanium-based and one PA-RISC, on a DVD

The Itanium-based *golden archive* can only be installed on another Itanium-based system. Similarly, the PA-RISC golden archive can only be installed on another PA-RISC system. The DVD will be able to *boot* either type of architecture.

On a system with a DVD writer, execute the following commands. (The file suffixes `.pa.gz` and `.ia.gz` replace the `make_media_install -o` option. The *`iso_directory`* is the target directory for the `.iso` media file. The `-r` option specifies the release, such as `B.11.23`, or `B.11.31`.)

```
# mkdir /var/opt/ignite/media/pseudo_root/archive
# cp PAgoldenarchive.pa.gz /var/opt/ignite/media/pseudo_root/archive
# cp IAgoldenarchive.ia.gz /var/opt/ignite/media/pseudo_root/archive
# make_media_install -r B.11.23 -a PAgoldenarchive.pa.gz \
  -a IAgoldenarchive.ia.gz -m iso_directory
```

Create a recovery DVD

Create a *recovery image* of the current system on the named Ignite-UX server, and write that recovery image to the DVD writer on the server system by issuing the following command on the client system.

```
# make_opticaldisc_recovery Ignite_server_name
```

Create an HP-UX 11i v2 Itanium-based recovery DVD using an existing network recovery image

Run the following command from the Ignite-UX server on which the *recovery image* exists. (The `-r` option specifies the HP-UX release, and the `-o` option specifies the hardware architecture.)

```
# make_media_install -r B.11.23 -o IA \
  -f /var/opt/ignite/clients/client/recovery/archive/archive_cfg \
  -f /var/opt/ignite/clients/client/recovery/archive/control_cfg \
  -f /var/opt/ignite/clients/client/recovery/archive/system_cfg \
  -a archive.gz \
  -p /var/opt/ignite/recovery/archives/client \
  -m /var/tmp
```

Error messages

No DVD available

The `make_media_install` script will automatically write the `.iso` image (default file name is `image.iso`) to the DVD. If there is no writable DVD available, the script will halt with a message similar to

```
:-[ LOAD TRAY failed with SK=5h/ASC=24h/ACQ=00h]: I/O error
```

The `.iso` file will remain in the `iso_directory` specified with the `-m` option. Make sure you have enough disk space in the `iso_directory` to hold the `.iso` image.

No DVD special files

Prior to the first time you run `make_media_install`, you must create DVD special files. If the DVD special files are missing when you run `make_media_install`, you will receive an error message similar to

```
:-( unable to open("/dev/rscsi/c0t0l0".O_RDONLY): No such file or directory
:-! consider 'mknod /dev/rscsi/c0t0l0 c 203 0x000002; chmod 0600 /dev/rscsi/cotol0'
+ Error 83 /opt/ignite/lbin/growisofs failed.
/opt/ignite/data/scripts/examples/make_media_install: ERROR: /opt/ignite/lbin/growisofs failed.
```

To add DVD special files:

- Run the command `mkdir /dev/rscsi`
- Then run the suggested `mknod` and `chmod` commands.

NOTE: If the system with the DVD writer is running HP-UX 11i v3, you must use a legacy device special file to identify the DVD.

Missing `-c` argument on HP-UX 11i v2 USB DVD drive

On an HP-UX 11i v2 system with a USB DVD drive, you must use the `-c` argument to `make_media_install` and install the patch PHKL_37814. For more information, see "Create HP-UX 11i v2 bootable media on USB DVD drive for two-step media recovery" (page 204). If the `-c` option was not used, an error message similar to the following will be displayed:

```
:-( unable to open ("/dev/rdsk/c0t0d0",O_RDONLY: No such device
```

Depot-Based DVDs

The procedure to create depot-based installation DVDs is dependent on the version of HP-UX you will be installing; use "HP-UX 11i v2 Depot-Based Installation DVDs" (page 206) or "HP-UX 11i v3 Depot-Based Installation DVDs" (page 207).

HP-UX 11i v2 Depot-Based Installation DVDs

This process is largely documented in the *make_medialif*(1M) and *instl_combine*(1M) manpages. The approach is to run `make_config` against each depot separately, then use `make_medialif` to form a single config file for the depots, the default file system layout file, and any other config files you wish.

Preparation

1. Copy the installation content to your Ignite-UX server.

   You will need a great deal of disk space for this. Estimate 4.7 GB per DVD, which is the space you have for the depot and the *ISO image*. Use multiple depots and make sure each depot is small enough to fit on one DVD.

   For example, to create two depots on the Ignite server from OE media, use the following commands. Each DVD will have to be mounted before the `swcopy` and unmounted afterwards.

   ```
   # swcopy -x enforce_dependencies=false -s <mount point of OE_DVD1> \* \
     @ <path_to_depot1>
   # swcopy -x enforce_dependencies=false -s <mount point of OE_DVD2> \* \
     @ <path_to_depot2>
   ```

2. Create a configuration file for each depot using `make_config`.

   For example, create a configuration file for the first depot with the following command:

   ```
   # make_config -x no_sd_server -s <path_to_depot1> \
     -c <path_to_cfg_file1>
   ```

   The `no_sd_server` option instructs `make_config` to omit the network attributes for that depot.

   When dealing with more than one depot, use the `load_order` option to control the exact sequencing of media installations. It is helpful to use the `media_title` option so Ignite-UX can prompt for media using meaningful text.

   To create the configuration file for the second depot, you could use the following command:

   ```
   # make_config -x no_sd_server -s <path_to_depot2> \
     -c <path_to_cfg_file2> -x load_order=2 \
     -x media_title="Installation DVD Number 2"
   ```

Continue to create configuration files for all your depots.

3. Create the LIF header

   Use the `make_medialif` command to deposit the LIF file into the **first DVD** depot pseudo-root. This is the norm for installation media.

   The following `make_medialif` command uses two standard configuration files and two depot configuration files, but you can specify as many as you wish.

```
# make_medialif -f /opt/ignite/data/Rel_B.11.23/config \
  -f /opt/ignite/data/Rel_B.11.23/hw_patches_cfg \
  -f <path_to_cfg_file1> \
  -f <path_to_cfg_file2> \
  -r B.11.23 -a -V -v -R -l <path_to_depot1>/media_lif

# instl_adm -b -F <path_to_depot1>/media_lif
```

   The commands above create a LIF configuration file that contains the default file system layout configuration file for 11i v2, the patch handling configuration file for v2, and two depot configuration files created with `make_config`. Other options are to include the `Version` file, `RECCMDS`, and support for Integrity and PA-RISC systems.

   If you need to include configuration information in the *install file system*, such as settings of control keywords, it must be set now using the `instl_adm` command. For example, if you wanted a common root password, you could place that information in a file using the `root_password` keyword, and then run the following command:

```
# instl_adm -f <path_to_special_cfg_settings> \
  -F <path_to_depot1>/media_lif
```

   For more information, see *instl_adm*(1M) and *instl_adm*(4).

Create the first DVD

1. Copy the EFI partition into the first DVD pseudo-root:

```
# cp /opt/ignite/boot/Rel_B.11.23/EFI_CD_image <path_to_depot1>
```

2. Run `mkisofs` to create the first DVD image (*`path_to_dvd1image`*):

```
# /opt/ignite/lbin/mkisofs -D -r -U -v -max-iso9660-filename \
  -no-emul-boot -b EFI_CD_image -hide EFI_CD_image \
  -eltorito-alt-boot -no-emul-boot \
  -b media_lif -hide media_lif \
  -o <path_to_dvd1image> <path_to_depot1>
```

3. Run `instl_combine` to position the LIF header at the beginning of the image:

```
# /opt/ignite/lbin/instl_combine -C <path_to_dvd1image>
```

Create the second DVD

1. Run `mkisofs` to create the second DVD image (*`path_to_dvd2image`*):

```
# /opt/ignite/lbin/mkisofs -D -r -U -v -max-iso9660-filename \
  -o <path_to_dvd2image> <path_to_depot2>
```

Burn the DVD images and test them

1. Write the ISO images to media as a raw file using open source software such as `cdrecord`, `growisofs`, or software included with CD/DVD hardware.
2. Test the DVD images to verify you can boot from the first one, and test that you can use them to install systems successfully.

HP-UX 11i v3 Depot-Based Installation DVDs

This section describes the differences in the 11i v3 process from the v2 process. (The differences are due to the requirements of the `swm` (software load) command.)

In order to support the 11i v3 process, the `createMediaCatalogs` script was added to Ignite-UX starting with the C.7.10 release. The `createMediaCatalogs` script is found in `/opt/ignite/data/scripts`. Detailed usage information is available via

```
# /opt/ignite/data/scripts/createMediaCatalogs -?
```

See *createMediaCatalogs*(1M) and *swm*(1M) for more information.

See "HP-UX 11i v2 Depot-Based Installation DVDs" (page 206) for detailed information on this procedure, including command syntax.

To create 11i v3 depot-based DVD media, follow these steps:

1. Use `swcopy` to create one depot with all the DVD content in it. If you have additional software from other sources, use `swcopy` to add that to the depot too.
2. Run `make_config` against the combined depot using the `-x no_sd_server` option.
3. Use `swcopy` to create a separate depot for each DVD.
4. Run the `createMediaCatalogs` script, supplying all the depots as arguments. It's important to list the depot for the first DVD as the first argument.
5. Use `make_medialif` to create the *LIF volume* for the first DVD. This step will use the config file you created in step 2.
6. Use `mkisofs` to create the ISO image for the first DVD.
7. Run `instl_combine` to position the LIF volume in the install content.
8. For each subsequent DVD, use `mkisofs` to create the ISO image for each depot.
9. The ISO images may now be burned to DVD and tested.
10. Once the DVD has been verified, you can remove the combined depot created for `make_config` in step 2.

15 Recovery

Overview

System recovery restores a *recovery image* previously created for a specific system, and is useful in the case of hardware or software failure. See the System Recovery section below for more information.

System Recovery

Ignite-UX system recovery allows quick recovery from a failed disk. The failure can be either a hardware failure or a catastrophic software failure.

This section assumes you are creating a *recovery image* to be stored on the *Ignite-UX server* via the network, or on tape. If you wish to create recovery image media, see Chapter 14 (page 197)

System recovery requires some work before a problem occurs. On a regular basis, you need to run the appropriate tool on each of your systems: `make_net_recovery` or `make_tape_recovery`. Use the `make_net_recovery` command to create a recovery image on another system, or the `make_tape_recovery` to create a recovery image on tape.

The `make_tape_recovery` and `make_net_recovery` commands each create a bootable, installation recovery image that is customized for your machine. Recovery images contain your system's configuration information (*disk layout*, etc.) and files from one or more disks. You can exert some control over which files are saved as part of the image - see "Recovery Image Contents" (page 211) for more information.

The `make_net_recovery` command and the `make_tape_recovery` command are collectively referred to as: *`make_[tape|net]_recovery`*.

You can use the `make_[tape|net]_recovery` commands on a command line, the Ignite-UX GUI from the server, or the Ignite-UX TUI from the client to create a recovery image.

Once you have a *recovery image* on tape or Ignite-UX server, recovering a failed system is easy:

1. If a disk failed, replace it.
2. Boot from your recovery tape or system.
3. Wait for the recovery to complete.
4. Once the system comes back up, verify the system configuration and recover the latest copies of files from the last system backup. Ensure that you do not recover operating system files as this can create unexpected results.

If you have SAS devices connected to the recovery client, be aware that as of Ignite version C.7.5, Ignite will recover to the original disk based on WWID, even if it has been moved. However, moving SAS devices can result in a changed device file name. For more information, see the *Ignite-UX and SAS Devices* white paper, available at http://www.hp.com/go/ignite-ux-docs.

IMPORTANT: The offline diagnostic environment (ODE) command `copyutil` is a diagnostic tool for HP-UX 11i and should not be used for system recovery. Instead, use `make_[tape|net]_recovery`.

IMPORTANT: During HP-UX 11i v3 installation and recovery, connected Active/Passive devices will cause long delays (one hour or more) or may cause a system to hang. Similarly, connecting an Active/Passive device before installing the Active/Passive Switch (APSW) plug-in can cause some commands to take a long time. Disconnect any Active/Passive devices connected to your system before installing or recovering HP-UX 11i v3. After installation or recovery, it is important that the APSW plug-in be installed before connecting an Active/Passive device to the system.

System Recovery Tools

The *`make_[tape|net]_recovery`* tools have few differences aside from using different media. Both system recovery tools share the same basic recovery image creation options, data structures, recovery image file content, and installation dialog boxes. The main differences are that `make_tape_recovery` does not require an Ignite server and `make_net_recovery` can be run from the client with a small subset of the Ignite product.

The `make_[tape|net]_recovery` tools are not intended for backup of all your system data. Use a restore tool such as `fbackup` in conjunction with your recovery image. See *fbackup*(1M) for more information.

Recovery Tool Comparison

To determine which system recovery tool is best suited for your needs, consider the following:

Use `make_tape_recovery` to:

- Manage single or a limited number of systems locally.
- Manage systems that are not networked.
- Create tape media for an off-site recovery system.
- Create recovery images for clients on a different subnet than the Ignite-UX server without using a *boot helper system*.

Use `make_net_recovery` to:

- Centrally manage networked systems.
- Avoid tape issues (handling, multi-tape images, etc.)
- Use disk space for image storage.
- Perform unattended creation of recovery images without tape handling.
- Create recovery images for clients on a different subnet than the Ignite-UX server without using a boot helper system. (A boot helper system or a similar solution must be used when installing a recovery image across subnets.)

The following table summarizes and compares some of the features of the `make_[tape|net]_recovery` tools:

Table 15-1 Comparing System Recovery Tool Features

	make_tape_recovery	make_net_recovery
Minimum hardware configuration	• *Stand alone* system • Local tape drive	• Two networked systems • Sufficient disk space to hold image
Creation interfaces	• Client command line • Ignite-UX server GUI • Client TUI	• Client command line • Ignite-UX server GUI • Client TUI
Recovery image personality	• Self contained • Written to the client's tape drive	• Requires an Ignite-UX server to install • Written to NFS mounted *file system*

Considerations When Using Veritas Volume Manager from Symantec

If you intend to use or are using VxVM, consider the following issues that impact the `make_[tape|net]_recovery` tools:

— Ignite-UX only supports *cold install*ing Symantec products that are in an OE *depot* as shipped by HP.

— HP only supports an environment where the versions of VxFS and VxVM match. For example, if you install VxFS 4.1, you must also install VxVM 4.1.

— Systems with mixed versions of VxVM *disk group*s are recovered with all disk groups specified in the *recovery image* converted to the higher VxVM version. For example, when your recovery image contains both VxVM 3.5 and 4.1 disk groups, all disk groups are created as VxVM 4.1 disk groups.

— If your system has different versions of VxFS and VxVM, such as VxFS 4.1 and VxVM 3.5, you will still have mixed versions after the system is recovered.

— Before an installation is allowed to proceed, Ignite-UX verifies that the correct version of VxVM is being installed, if the installation is coming from an SD depot. If the installation is coming from a *recovery image* or *golden image*, Ignite-UX will log a note stating it is assumed the correct version is included in the image. For example, if your *disk layout* is VxVM 4.1, your image must contain the VxVM 4.1 software.

Recovery Image Contents

The *make_[tape|net]_recovery* commands enable you to view and control *recovery image* contents.

- Ignite-UX creates a symbolic link between directories named after the client's name and its MAC address. For example, if you had a system with a client name of "longs_peak" and a MAC address of "0x00306E4C9B54", the directories `/var/opt/ignite/clients/0x00306E4C9B54` and `/var/opt/ignite/clients/longs_peak` would be symbolically linked. This chapter uses the client name to reference the directory, but either will work.
- The list of *essential* files to be included in the *recovery image* is available as a simple text file: `/opt/ignite/recovery/mnr_essentials`. This file allows you to see what files and directories are included in the recovery image by default.
- You can specify what additional volume groups, directories, and files you want included, and what directories and files you want excluded. This is done using simple syntax in the client-specific content file, `/var/opt/ignite/clients/client/recovery/archive_content`, or by using command line options. You are not restricted to one or two volume groups. You can create a complete multivolume group recovery image.
- You can use the user interface (launched with the `-i`option to `make_[tape|net]_recovery`) to find out which volume groups and/or disks will be untouched, which will be partially restored, and which will be restored in full if the recovery

image is used, based on the specifications in the `mnr_essentials` file and the `archive_content` file.

- You can also use the user interface to edit the `archive_content` file and dynamically see the changes in the volume groups and disks that are affected.
- The policies for user-specified content are documented in the Recovery Image Configuration Policies section below.

The `make_tape_recovery` tool creates a bootable tape that can be used to restore a system using the system's tape drive. Remember that `make_tape_recovery` is subject to the requirements and limitations inherent with tape media:

- A tape drive must be available on each system to be archived.
- If you want to save previous recovery images, remove the tapes containing the existing recovery images from the tape drives before creating new ones.
- If a recovery image exceeds the capacity of a tape, you need to swap tapes for both creation and extraction.
- If you want to make sure that the newly created tapes are good, you must check the log files on the system.
- Tape drives are more error prone than a local network.

Recovery Image Configuration Policies

When specifying *recovery image* content for *`make_[tape|net]_recovery`*, the following rules apply:

- No *essential* file or directory can be excluded.
- Files and directories inside an included directory will be included recursively.
- If an essential file or directory exists outside the root disk or volume group, the disk or volume group it resides in is included in the recovery image. If you want to include all the files within that disk or volume group in the recovery image, use the `make_[tape|net]_recovery -A` or `-x` options.
- If a symbolic link to a file or directory is included, only the link will be included in the recovery image. The actual file or directory is *not* included unless it is specified or the symbolic link is essential. A warning will be given when an item is only a symbolic link.
- If a directory is included that contains symbolic links to other files or directories, the symbolic links will be included but not the referenced files or directories, unless they too are included. No warnings are given regarding these links.
- If a directory contains local mount points, the files and directories under the local mount points are not included, by default. This policy can be waived by specifying the option `inc_cross` (include directories and cross-mount points) in the selection interface or command line.
- In case of conflicting entries in the selections, exclusions take precedence over inclusions.
- *File system* volume size must provide 10 percent free space for each volume - Ignite-UX automatically modifies the file system volume size accordingly. For more information, see the description of the `_hp_addnl_fs_free_pct` variable in *instl_adm*(4).

Reconciling Client and Server Ignite-UX Versions for Recovery

If you initiate a recovery **from the server GUI** and the client system has a lower version of Ignite-UX than the server does, Ignite uses `swinstall` to update the existing Ignite-UX software on the client. If the client system does not have Ignite installed, a small subset of Ignite-UX software will be installed. (The small subset of Ignite-UX software is not a full Ignite-UX server installation, and does not provide Ignite-UX server capability to the client.)

If you initiate a recovery **from the client** with `make_tape_recovery -s` or `make_net_recovery -s`, and the client system has a lower version of Ignite-UX than the server

does, behavior depends on the degree of version mismatch. If the version *letters* don't match, such as C.*x.x* and B.*x.x*, Ignite will display an error and the process will stop. If the version *numbers* do not match, Ignite will display a warning and the process will continue.

In any case, if the server has a lower version of Ignite-UX than the client, a message to this effect will be displayed and the process will stop.

As of Ignite version C.7.7, *make_[tape|net]_recovery* has a `-u` option that will update the client Ignite software to the version on the server specified by the `-s` option. For more information, see *make_net_recovery*(1M) or *make_tape_recovery*(1M).

Recovery Image Creation Process

The process of creating a recovery image using Ignite-UX is described as follows:

1. Prepare the client.

   The `make_net_recovery` command and the `make_tape_recovery -s` command first check that the recovery tools installed on the client are compatible with the version on the Ignite-UX server as described in "Reconciling Client and Server Ignite-UX Versions for Recovery" (page 212).

2. Create files and directories for the recovery image.

   The `make_net_recovery` and `make_tape_recovery -s` commands create a new directory for the client on the server in `/var/opt/ignite/clients` if needed. For `make_tape_recovery` run on the client without the `-s` option, the *config file*s and log files are created in the `/var/opt/ignite/recovery` directory on the client.

   The commands generate a timestamp for naming the recovery archive, the configuration, and the configuration directory. The directory containing the configuration files for the recovery image is similar to the following:

   `/var/opt/ignite/clients/`*`client`*`/recovery/2005-03-17,11:19`

   The corresponding recovery archive is named `2005-03-17,11:19`, and is in the `/var/opt/ignite/recovery/archives/`*`client`* directory.

   The timestamp is important for coordinating configuration files and recovery archives, and for ongoing recovery image management.

   An overview of the files is as follows:

```
/var/opt/ignite/clients/client
        CINDEX
        client_name
        client_status
        config.sys
        host.info
        hw.info
        install.log
        recovery/
                client_status
                defaults
                latest -> 2005-03-17,11:19

                2005-03-17,11:19/
                        archive_content
                        system_cfg
                        archive_cfg
                        control_cfg
                        recovery.log
                        flist
                        manifest
```

   The `archive_content` file contains keyword and volume/disk/directory pairs that are used to generate the `flist` file, which defines the contents of the recovery image. See

make_net_recovery(1M) and *make_tape_recovery*(1M) for more information on inclusion and exclusion of files in the recovery image.

3. Run the recovery interface.

   If the `-i` option is specified on the command line, the recovery user interface is executed next. This interface enables users to set or change the following default values for the image:

   - Long description of the recovery image. This description adds identifying information that can help distinguish between recovery images when the timestamp is not sufficient. This information is shown by clicking **Description** on the **Basic** tab during installation configuration.
   - Maximum number of recovery images to keep. When the number of recovery images in the destination directory reaches this maximum, `make_[tape|net]_recovery` removes the oldest one. It uses the timestamp in the name to determine which to remove.
   - Destination host for the recovery image.
   - Destination directory for the recovery image.

     The user interface also gives you the opportunity to review and edit the `archive_content` file as mentioned in the previous step. When you exit the recovery user interface, the default values you entered are written to:

     `/var/opt/ignite/clients/`*`client`*`/recovery/defaults`.

     The list of files included in the recovery image is written to `archive_content` in the `/var/opt/ignite/clients/`*`client`*`/recovery` directory.

4. Save the system configuration.

   For all disk and volume groups, even those not included in the recovery image, `make_[tape|net]_recovery` backs up disk and volume group configuration information, and then stores it in the `system_cfg` file. For LVM, it also obtains map files for volume groups that are not part of the recovery image. The volume group configuration files and the map files generated at this stage are stored in `/etc/lvmconf`. This directory is included in the list of *essential* files, so the LVM files are included in the recovery image. For VxVM, commands are included in `control_cfg` that restore *disk group*s.

   After the volume group information is saved, `make_[tape|net]_recovery` creates the `control_cfg` file. This file includes the `post_config_cmds` to import all volume or disk groups that were not included in the recovery image, and to activate all volume groups that were imported. It also includes control flags, such as `recovery_mode=true`, to guide the behavior of Ignite-UX during recovery.

5. Build the recovery archive.

   Next, `make_[tape|net]_recovery` calls `make_sys_image` to create the recovery archive. Then `make_sys_image` passes a prebuilt `flist` to calculate the total disk space currently used by all the files to be included in the archive. It uses this information with a compression ratio to estimate the final size of the archive. If the destination directory has sufficient free disk space for the archive, `make_sys_image` creates the archive using the `pax` command. For more information, see *pax*(1) and *make_sys_image*(1M).

6. Prepare the configuration file.

   Once the recovery archive is created, `make_[tape|net]_recovery` calls `make_arch_config` to create the `archive_cfg` file to reference it. Then

`make_arch_config` uses `archive_impact` to calculate the *file system* impacts for the recovery archive, and includes these in the `sw_sel` clause it writes.

7. Update the *`CINDEX`* file.

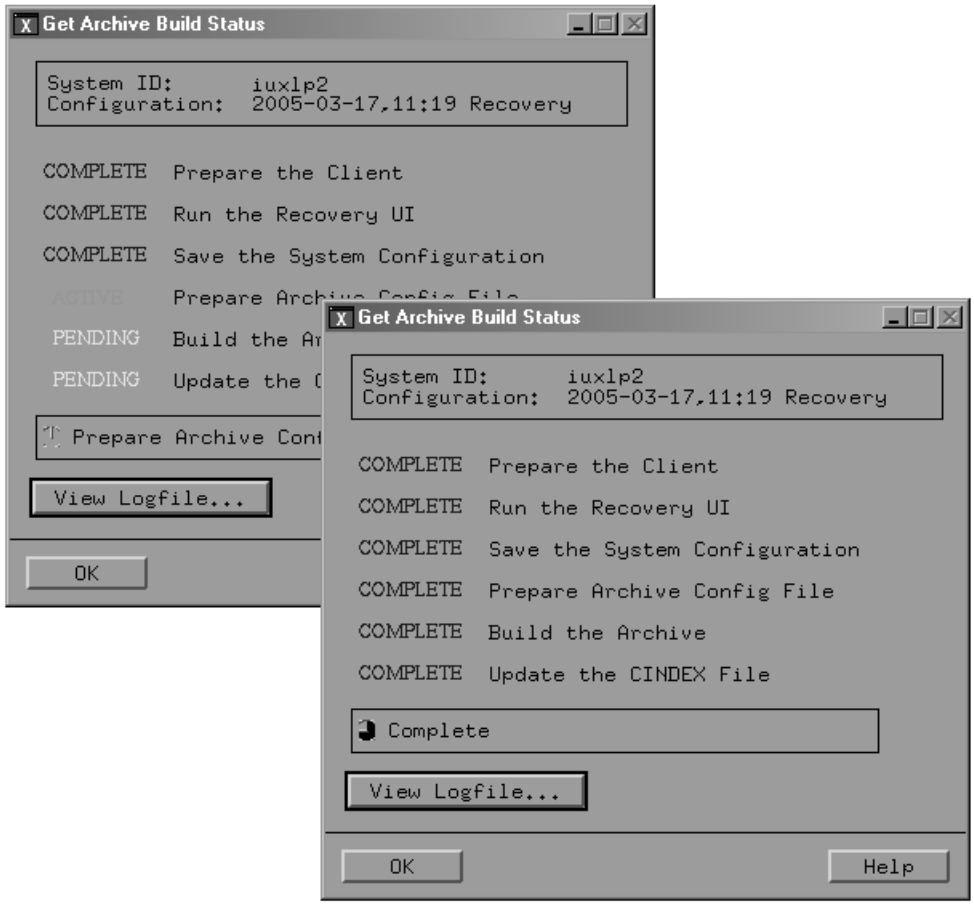
   Lastly, `make_[tape|net]_recovery` uses `manage_index` to update the `/var/opt/ignite/clients/`*`client`*`/CINDEX` file for the client. This file contains a list of all the recovery configurations available for the client. The configuration clause for the most recently created recovery archive is similar to the following excerpt:

```
cfg "2005-03-17,11:19 Recovery Archive" {
        description "Recovery Archive"
        "recovery/2005-03-17,11:19/system_cfg"
        "recovery/2005-03-17,11:19/control_cfg"
        "recovery/2005-03-17,11:19/archive_cfg"
}=TRUE
```

Recovery Image Creation Status

You can monitor the status of the recovery image creation process by right-clicking on the client icon or clicking the **Actions** menu, then selecting **Client Status....** The resulting dialog box details the progress as the recovery image is created with `make_net_recovery` as shown in Figure 15-1.

Figure 15-1 Get Archive Build Status Dialog Box



Examining Recovery Image Contents

The commands `make_[tape|net]_recovery` call `/opt/ignite/lbin/list_expander` as part of the process of determining what to include in a recovery image. You can use the `list_expander` command independently to determine for yourself what will be in the recovery image.

To list the files and directories included in a recovery image, use the `list_expander` command in the following way:

```
/opt/ignite/lbin/list_expander -f archive_content
```

where `archive_content` is the file that identifies keywords specifying inclusions and exclusions for the recovery image. It is the same `archive_content` file discussed in the "Recovery Image Creation Process" section above.

NOTE: The `/var/opt/ignite/clients/client/recovery/archive_content` file is overwritten whether a recovery image is successfully produced or not. Be sure the `archive_content` file matches the recovery image you are exploring.

Running `list_expander` without specifying `-f archive_content` causes a list of the *essential* recovery image files and directories to be listed.

You can also use `list_expander` to list disks and volume groups included in a recovery image by using the `-d` option:

```
/opt/ignite/lbin/list_expander -d -f archive_content
```

Omitting the `-f archive_content` will cause the essential list to be displayed.

The following is example `list_expander -d` output:

```
In?     dsk/vg    name               minor#   Associated disks
0         d       /dev/dsk/c0t3d0
1         v       /dev/vg00          0x00     /dev/dsk/c0t6d0
                                              /dev/dsk/c0t4d0
0         v       /dev/vg01          0x01     /dev/dsk/c0t1d0
0         v       /dev/vg02          0x02     /dev/dsk/c0t2d0
```

The `In?` column shows, for each disk or volume group, if it will be:

`2` = included in full (`inc_entire` specifies entire disk/volume group), or `1` = included in part (some files are included, some not), or `0` = not included at all (no files from this disk/volume group are included.)

The `0` means the disk or volume group will not be touched. The `1` or `2` means that the disk or volume group will be recreated and files from the recovery image will be restored during a recovery operation.

The `dsk/vg` column shows that the system has one whole disk (`d`) and three volume groups (`v`). The next column gives the names of the disks and volume groups.

NOTE: The following circumstance may cause `list_expander` to halt:

When processing information, `list_expander` uses the `libc` function `ftw`, see *ftw*(3C), to recursively descend the directory hierarchy.

When `ftw` encounters a directory containing a large number of files, all of the descendent files are processed recursively. This can cause stack size problems. For example, if the directory contained 400,000 files, the stack size must be at least 51.2 MB to support the number of recursive function calls (400,000 calls x 128 bytes/stack frame), since each `ftw` call allocates a stack frame of 128 bytes. In this case it is wise to allocate additional space - 64 MB is a better stack size choice.

If the stack size is not large enough, `list_expander` is killed due to a stack growth failure. To avoid this situation, you should configure the *kernel* tunable `maxssiz` accordingly. See the *kctune*(1M) manpage for more information on tuning kernel parameters.

The above information is based on the HP-UX 11i v1 `ftw() libc` function. The behavior of `ftw()` between releases or patch versions of HP-UX may change. Consider this information indicative of the potential setting required; it should not be considered authoritative.

The file system volume sizes in the recovery image can be modified when the recovery image is installed. By default, Ignite-UX ensures that there is 10 percent free space for each volume, and

modifies the file system volume size accordingly. If you do not want Ignite-UX to modify the file system volume sizes automatically, add

```
init _hp_ignore_sw_impact=1
```

to your `/var/opt/ignite/recovery/latest/system_cfg` file, or to the `/var/opt/ignite/clients/client/recovery/latest/system_cfg` file.

Verifying Recovery Image Results

During a system recovery, Ignite-UX by default restores the system to the state it was in when the recovery image was created. Ignite-UX is a general-purpose installation tool. It modifies many system configuration files if changes from the recovery configuration are required, such as increasing volume sizes.

When you run `make_[tape|net]_recovery`, system configuration information is gathered and saved in configuration files that are used later when the system is recovered. During the system recovery you are allowed to make changes to this information, and Ignite-UX makes the appropriate changes to the system configuration. If you do not make any changes, Ignite-UX simply reapplies the same information, and there should be no change to the system after recovery.

Most of the system configuration files that Ignite-UX will modify are listed in the script, `/opt/ignite/data/scripts/os_arch_post_1`. The `os_arch_post_1` script checks for the system recovery case by checking the `$RECOVERY_MODE` variable. When this variable is `TRUE`, the `os_arch_post_1` script causes some configuration files to be protected from modification by using the "save_file" function. The `os_arch_post_1` script uses the "merge_file" function on files that Ignite-UX knows how to merge information into.

The files operated on by "merge_file", as well as those that have a commented out "save_file" line, are likely to be modified by Ignite-UX. Comments in the file explain any exceptions.

Because the list of files modified by Ignite-UX may change from release to release, it is best to look at the `os_arch_post_1` file on your system to see which files are saved as-is and which are merged with information from the Ignite-UX configuration files.

Creating and Using Recovery Tapes

The Ignite-UX `make_tape_recovery` command creates a system recovery tape that can be used to boot and recover a system that is not bootable due to corruption of the root disk or root volume group. A system can be booted and installed from the tape without user intervention, including configuration, customization, software selection, hostname, and networking information.

A bootable recovery tape can be created from the Ignite-UX server, however the client must have a local tape drive.

It is preferable to use the Ignite-UX GUI on the Ignite-UX server when running an interactive `make_tape_recovery` session. Executing it from the Ignite-UX GUI causes any additional server configuration of NFS mounts to be performed. Additionally, more informative progress reporting is provided, and it is easier to use that interface.

IMPORTANT: The media and data format (density and compression) of the installation tape you create must be compatible with the clients on which it will be read. For example, if you have a mixture of DDS4 and DDS5 tape drives on your systems and you wish to be able to read recovery tapes on any of them, you should only use DDS4 media, as DDS5 media will not work in DDS4 drives.

The contents of the system recovery image will always include all files and directories that are considered essential to bringing up a functional system. This essential list is predefined by `make_tape_recovery` and is located in the following file:

```
/opt/ignite/recovery/mnr_essentials
```

In addition to the essential list, data can be included in the recovery image on a disk/volume group, file, or directory basis. Nonessential files and directories can also be excluded.

The tape created by `make_tape_recovery` is completely self-contained and does not require an Ignite-UX server to install the recovery image. The `make_tape_recovery` recovery image contains a specially prepared *LIF volume*. The configuration file in the LIF volume is the configuration file for the recovery archive. The `/var/opt/ignite/data/INDEX` file in the LIF volume specifies the recovery configuration as the default for the system. The recovery tape contains additional configuration information so no user interaction is required.

Additional files needed for booting and installing are copied from `/opt/ignite/boot/Rel_`*`release`* and `/opt/ignite/data` to the LIF volume on the tape, so everything the system needs to recover is there.

NOTE: During the recovery process, when the file system is set up and the I/O tree is initialized, tape device files may be mapped differently from when the original recovery tape was made. Therefore, it is possible for a recovery tape to be created with one tape device file, for instance `/dev/rmt/0m`, and recovered from a different device file, such as `/dev/rmt/2m`, though the physical device is the same.

You can also replicate a system and create a recovery image that can be used for installing clients. The section, "Notes on Cloning Systems" (page 236) describes how to make use of this process. For additional information regarding system cloning, see the *Successful System Cloning using Ignite-UX* white paper at

`http://www.hp.com/go/ignite-ux-docs`.

IMPORTANT: If you use `make_tape_recovery` for recovery, your tapes should be clearly labeled with the Ignite-UX version used to create them to avoid mixing Ignite-UX versions when *two-step media recovery* is used. See "Tape Recovery With No Tape Boot Support — Two-Step Media Recovery" (page 234) for more information.

Recovery Tape Creation Examples

The following examples are intended to assist you in using the `make_tape_recovery` tool.

Recovering a Minimal Operating System

To create a minimal operating system recovery tape at `/dev/rmt/0mn` containing only the operating system elements required to boot the system, perform the following steps:

1. Load a writable tape in the default tape drive for your system.
2. As `superuser`, enter `make_tape_recovery`.

   A tape will be created without further interaction.

System recovery from this tape involves booting from the tape to recover the minimum core operating system. Then you would follow up with data recovery of all user files newer than those restored from the recovery tape.

NOTE: If you are creating a recovery tape for an Itanium®-based system, you can choose to use the `-D` option of `make_tape_recovery` to specify the name of the ANSI tape volume.

Creating a System Recovery Tape of the Entire Root Disk Volume

To create a system recovery tape at the default device, `/dev/rmt/0m`, that includes the entire root disk in the recovery image, perform the following steps:

1. Load a writable tape in the default tape device for your system.

2. Enter the command:

   ```
   make_tape_recovery -x inc_entire=vg00
   ```

   A tape will be created without further interaction.

Creating a System Recovery Tape of the Root Disk Volume with /usr on a Different Volume Group

You can easily create a system recovery tape of the entire root disk, even if the `/usr` file system resides on a different volume group, by using the `-A` option of `make_tape_recovery`. This option has `make_tape_recovery` determine which disks and volume groups the specified files reside on, and then include all files from those disks and volume groups in the recovery image.

1. Load a writable tape in the default tape device for your system.
2. Create a system recovery image with all the disks and volume groups containing the files specified by the default *essential*s file list `/opt/ignite/recovery/mnr_essentials`, or a user-defined version that replaces it, `/var/opt/ignite/recovery/mnr_essentials`, by entering:

   ```
   make_tape_recovery -A -s myserver -a /dev/rmt/0m
   ```

   A tape is created on the default device, `/dev/rmt/0m`, without further interaction. You can boot this tape on your new system.

TIP: The use of the `-p` option can be particularly helpful, as it allows you to preview the processing that would take place without actually creating the tape.

Tape Recovery for PA-RISC Systems

To install a system recovery image from a tape on a PA-RISC system, use the following procedure:

1. Load the system recovery tape in the tape drive.
2. Boot the system.
3. Interrupt the boot sequence by pressing **Esc**.
4. Select the tape drive you want to use, and then boot from it.
5. Allow the installation process to complete.

For more information on creating recovery tapes, see *make_tape_recovery*(1M).

Tape Recovery for Itanium-Based Systems

To boot from tape on an Itanium-based system you must first create a tape boot option on the *EFI* Boot Manager menu. Verify that your Itanium-based system has firmware support for tape boot. If there is firmware that supports tape boot available for your system, you may first need to upgrade your firmware to make this functionality available. A set of tables showing minimum firmware revisions and SCSI HBAs that support tape boot is available in the *Ignite-UX Installation Booting* white paper available at

http://www.hp.com/go/ignite-ux-docs.

The first version of Ignite-UX to support native tape boot for Itanium-based systems is C.6.8. Recovery tapes created before that version of Ignite-UX can only be used with two-step recovery. See "Tape Recovery With No Tape Boot Support — Two-Step Media Recovery" (page 234) for more information on two-step recovery.

The screens shown in this example are from an HP Integrity rx1620 system. Other systems may vary in method and screen format. For information on how to configure boot devices for your system, consult your system's hardware documentation.

IMPORTANT: Configuring an EFI menu option for tape boot requires downtime since it can only be done from the EFI Boot Manager. If you are going to use tape recovery on your Itanium-based system, consider adding the tape boot option at your next planned maintenance window.

TIP: An ideal time to test tape recovery on your unique combination of system, tape drive, and *HBA*, is after you have configured an EFI Boot Manager menu option for tape boot. You do not need to recover the system. If you create a recovery tape with the **`-I`** option, you will enter an interactive recovery. When you get to the interactive screens, reset the system instead of performing a recovery.

Determining the Tape Drive's EFI Path

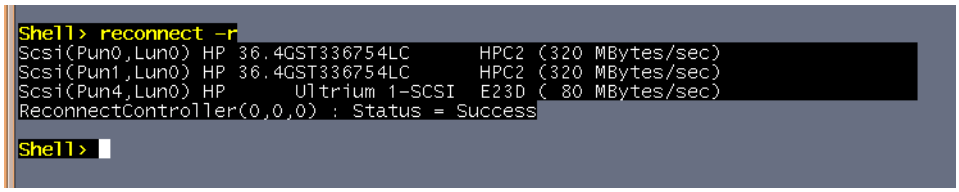
When adding a tape boot option to the firmware, you must identify the tape drive you will use for booting. The EFI menus will display device paths to choose from. Before beginning the tape boot configuration process at the EFI level, you must determine the device path to your tape drive so you can select the correct one to use for booting.

The `ioscan -e` command does not report EFI device paths for tape drives. Alternative methods must be used to determine the correct path.

The EFI device path for our example is `Acpi(HWP0002,100)/Pci(1|1)/Scsi(Pun4,Lun0)`

One way to identify the tape drive's path is to use the `reconnect -r` EFI command to get its SCSI Physical and Logical unit numbers (Pun and Lun). The Pun and Lun numbers can be mapped to the last part of the EFI device path. Below is the output of `reconnect -r` for our example.

Figure 15-2 Output From `reconnect -r`



Finding the Ultrium tape drive's Pun and Lun numbers in this example is simple because not many devices are listed.

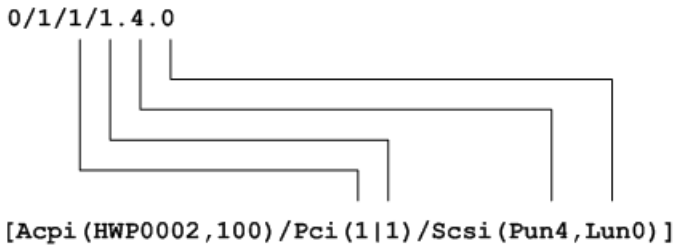
If your system is partitionable, EFI will not automatically enumerate all connected devices. (This allows for a speedier boot.) For this reason the tape drive you want to use may not be listed. If this is the case, you will need to use the `search` command to list the devices on the HBA the tape drive is connected to. See your system's Operations Guide for details on the `search` command.

A third way to find the EFI device path is to use the tape drive's hardware path as a map to it. The `ioscan -fkeCtape` command will list the hardware path of the tape drive.

For our example, the hardware path is `0/1/1/1.4.0`

Use the following diagram to map the hardware path to the EFI device path:

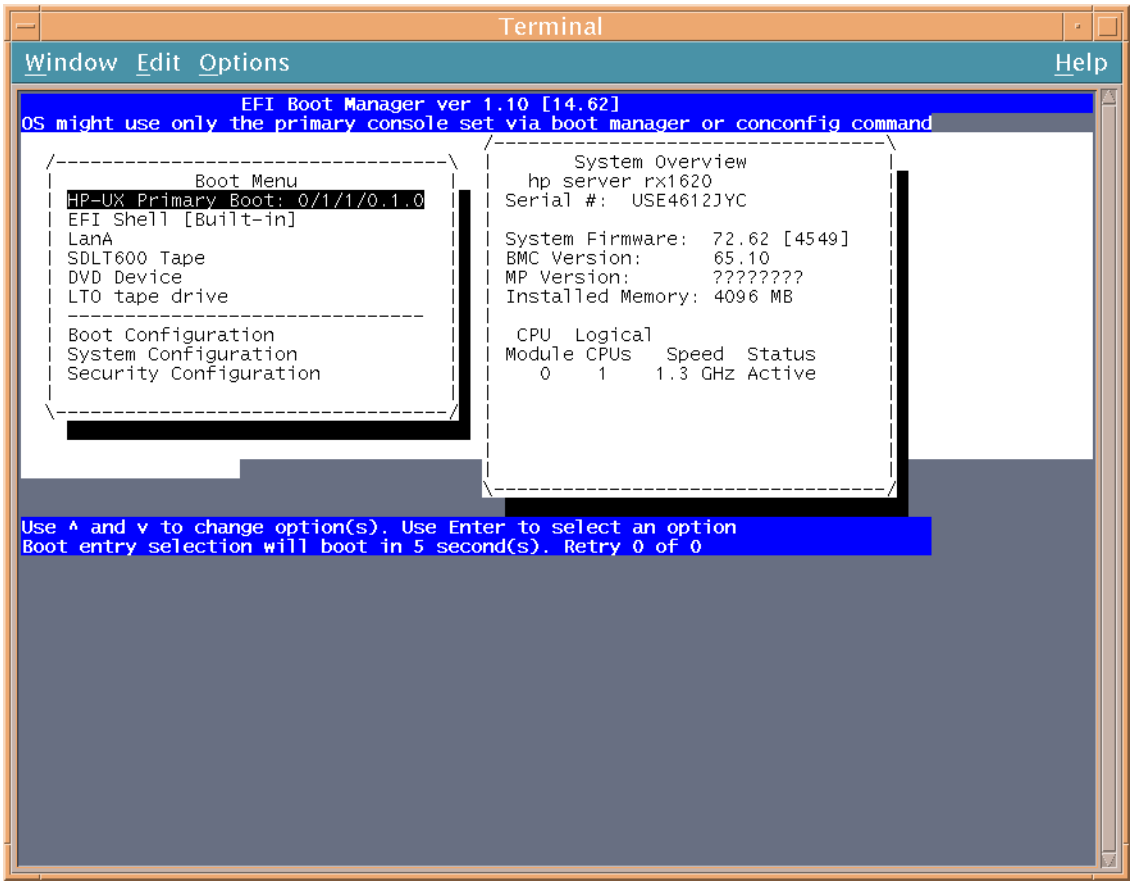
Figure 15-3 Mapping the Hardware Path to the EFI Device Path



Configuring the Tape Boot Option

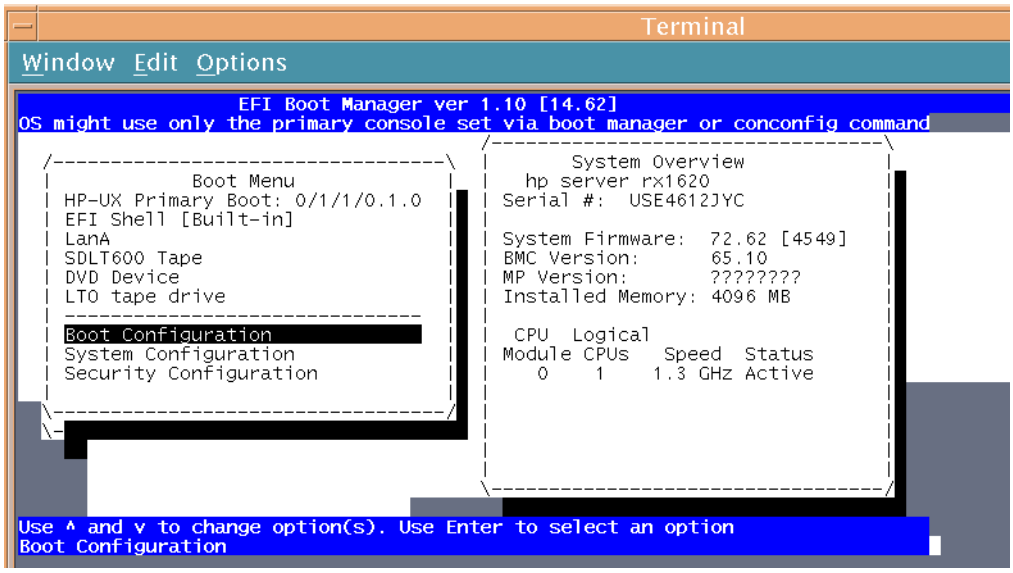
Reboot your system and stop the process at the EFI menu before it times-out, as shown in the figure below. Notice the last line warns that reboot will occur after the remaining seconds expire.

Figure 15-4 EFI Menu With Timer



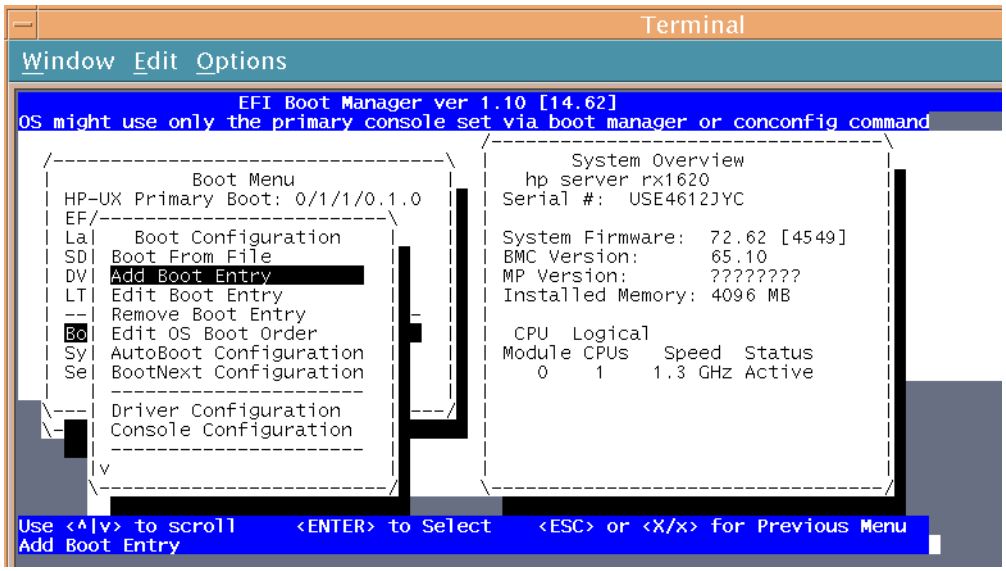
Select **Boot Configuration** from the Boot Menu.

Figure 15-5 Boot Configuration



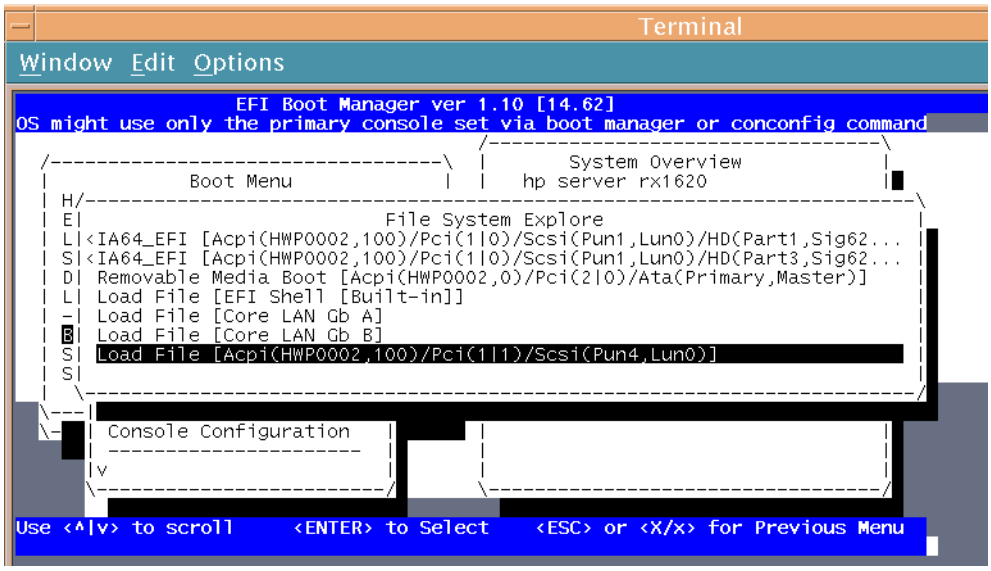
Select **Add Boot Entry** from the Boot Configuration menu.

Figure 15-6 Add Boot Entry



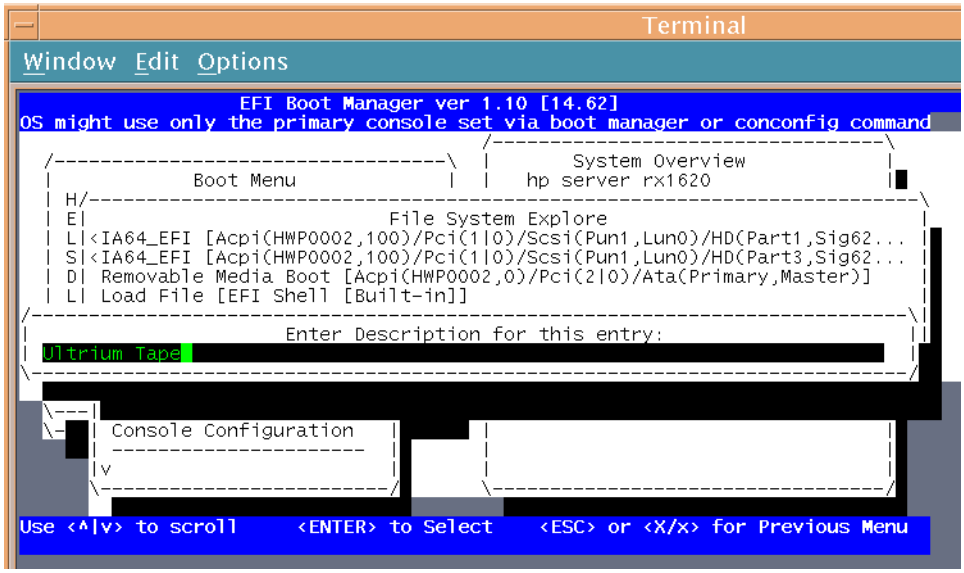
The EFI Boot Manager will then display a menu listing the available devices to choose from. Select the tape drive you wish to boot from. See Determining the Tape Drive's EFI Path above for how to select the correct device.

Figure 15-7 List of Selectable Boot Devices



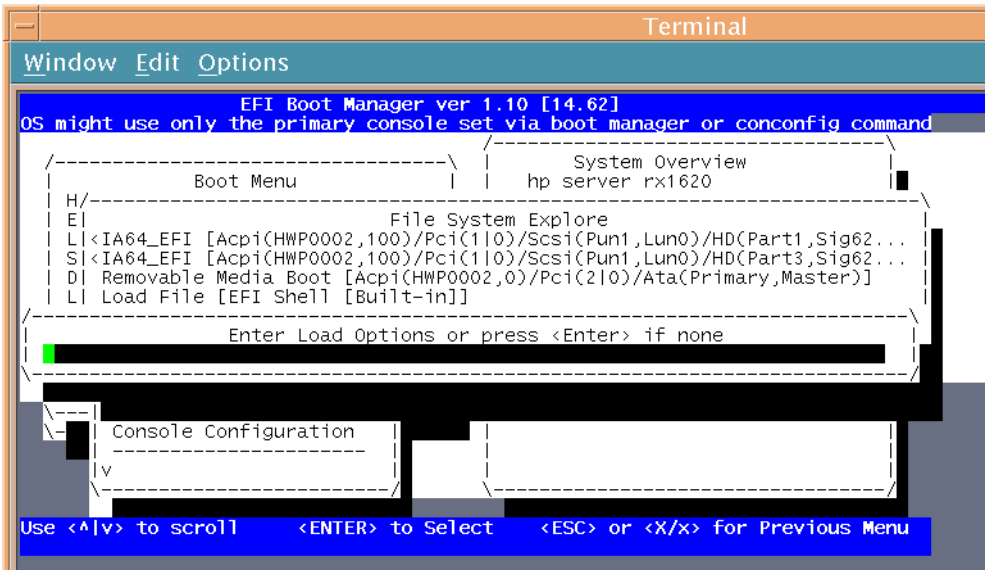
Enter a description in the next dialog box. This is the text that will appear in the Boot Menu listing. For this example, the new boot option will be called "Ultrium Tape."

Figure 15-8 Enter a Description for the Boot Option



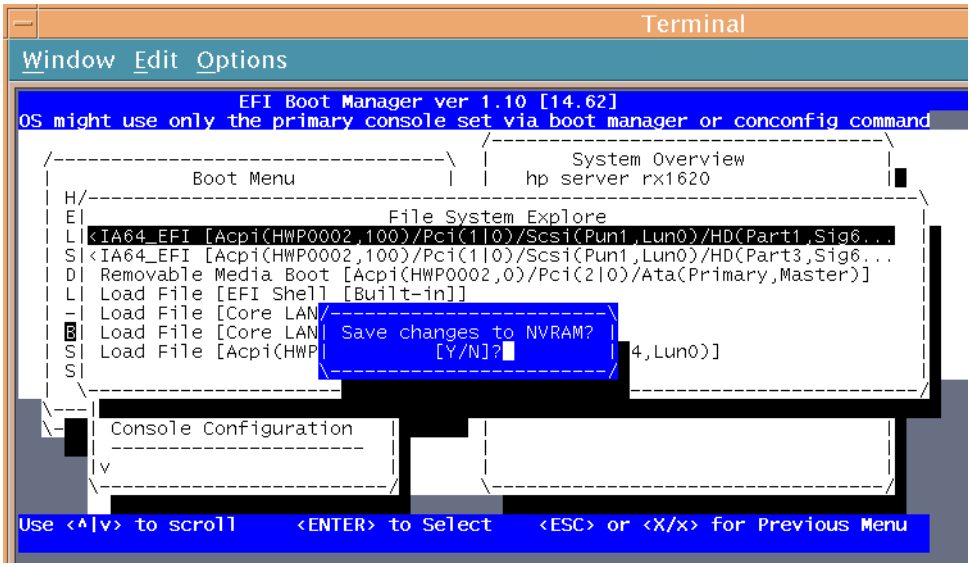
Next, you will be prompted for load options. Press **Enter** at this point without entering anything.

Figure 15-9 Enter Load Options



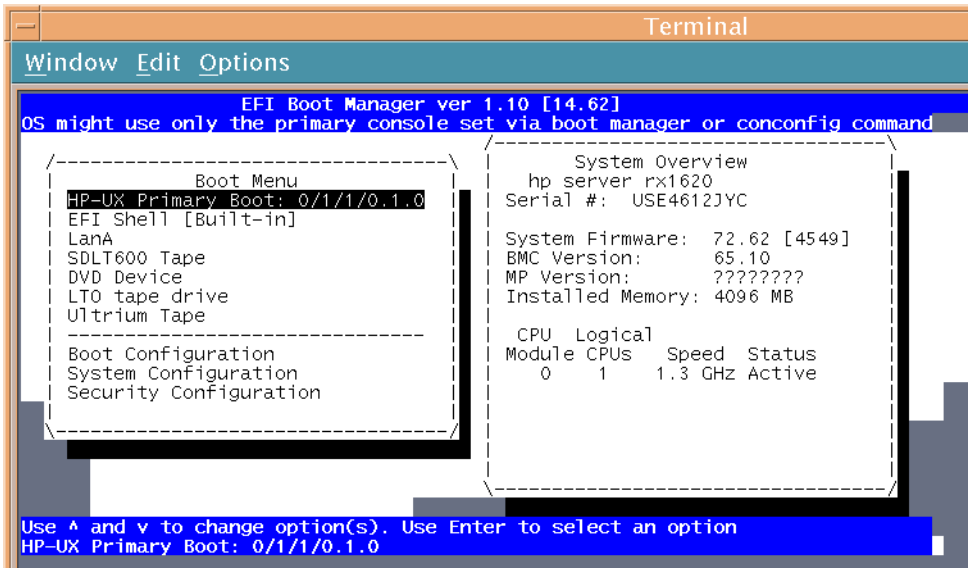
The last step is to save your edits to NVRAM. If you have made a mistake, press **n**, otherwise press **y** and the changes will be saved to NVRAM.

Figure 15-10 Save Changes to NVRAM



You will be returned to the main EFI Boot Manager menu. If you answered **y** to the Save changes to NVRAM question, your new boot option will appear listed with the description text you entered in Figure 15-8.

Figure 15-11 The Boot Manager Menu with the New Option



At this point you have successfully configured a tape boot option, and it may be selected from the EFI Boot Menu. For more information on creating recovery tapes, see *make_tape_recovery*(1M).

NOTE: When executing any Itanium-based boot using the *install kernel*s and *install file system*s, the following errors will appear in the output:

```
execve("/sbin/sh") failed, errno 0xffffffff
execve("/bin/sh") failed, errno 0xffffffff
```

These errors are not indicative of any Ignite-UX problem and can be safely ignored. The failures occur because `/sbin/sh` and `/bin/sh` are not present on the system when the *kernel* is starting; Ignite-UX does not need them at this point. On a non-installation boot, the kernel would be attempting to run `/sbin/pre_init_rc`, a script.

Tape Recovery for Integrity Blade Systems

To boot from tape on an Integrity blade system you must first create a tape boot option on the *EFI* Boot Manager menu. Verify that your Integrity blade system has firmware support for tape boot. If there is firmware that supports tape boot available for your system, you may first need to upgrade your firmware to make this functionality available.

The first version of Ignite-UX to support native tape boot for Integrity blade systems is C.7.10.474. Recovery tapes created before that version of Ignite-UX can only be used with two-step recovery. See "Tape Recovery With No Tape Boot Support — Two-Step Media Recovery" (page 234) for more information on two-step recovery.

The screens shown in this example are from an `ia64 hp Integrity BL890c i2` system. Other systems may vary in method and screen format. For information on how to configure boot devices for your system, consult your system's hardware documentation.

Determining the Tape Drive's EFI Path

When adding a tape boot option to the firmware, you must identify the tape drive you will use for booting. The EFI menus will display device paths to choose from. Before beginning the tape boot configuration process at the EFI level, you must determine the device path to your tape drive so you can select the correct one to use for booting.

One way to identify the tape drive's path is to use the `tapeboot select` command at the EFI shell prompt.

Determine Tape Drive EFI Path Using Command Line Interface

To get the tape driver EFI path, run the following command at the EFI shell:

```
Shell>> tapeboot select Fibre-Channel
```

Figure 15-12 Output From `tapeboot select`

```
Shell> tapeboot select
tapeboot: Device Path 01  PcieRoot(0x30304352)/Pci(0x7,0x0)/Pci(0x0,0x0)/Fibre(0
x500110A0008F063A,0x0)
Select Desired Tape:
```

Another way to identify the tape driver's path is to use the **EFI Boot Manager** menu. The **File Explorer** window of the **EFI Boot Manager** menu displays tape devices connected through Fiber Channel path.

Figure 15-13 The File Explorer Window

```
/------------------------------------------------------------------------------\
|                                File Explorer                                 |
\------------------------------------------------------------------------------/

Load File
[PcieRoot(0x30304352)/Pci(0x7,0x0)/Pci(0x0,0x0)/Fibr
e(0x500110A0008F063A,0x0)]
Load File
[PcieRoot(0x30304352)/Pci(0x8,0x0)/Pci(0x0,0x0)/Pci(
0x2,0x0)/Pci(0x0,0x0)/MAC(001E0B59DF2A,0x0)]
Load File
[PcieRoot(0x30304352)/Pci(0x1C,0x0)/Pci(0x0,0x0)/MAC
(00237D441EA4,0x0)]
Load File
[PcieRoot(0x30304352)/Pci(0x1C,0x0)/Pci(0x0,0x1)/MAC
(00237D441EA5,0x0)]
Load File
[PcieRoot(0x30304352)/Pci(0x1C,0x2)/Pci(0x0,0x0)/MAC
(00237D442E68,0x0)]

/------------------------------------------------------------------------------\
|                                                                              |
| ^v=Move Highlight       <Enter>=Select Entry       X/x=Exit this Menu        |
\------------------------------------------------------------------------------/
```

After identifying the EFI device path, enter the device path to start the boot recovery.

Figure 15-14 Tape Boot Device Path

```
Shell> tapeboot select
tapeboot: Device Path 01  PcieRoot(0x30304352)/Pci(0x7,0x0)/Pci(0x0,0x0)/Fibre(0
x500110A0008F063A,0x0)
Select Desired Tape: 01
```

For more information about booting from the tape device, see “Booting Itanium-Based Clients using the Network”.

Creating and Using Network Recovery Images

Ignite-UX enables you to create recovery images using the network and store them onto the Ignite-UX server system or any other specified system. Systems can be recovered across subnets after booting. See “Making Boot Decisions When Using the Client Console” (page 108) and the

sections in Chapter 10 on "Installation Using bootsys" and "Installation Using the Ignite-UX GUI" for booting options.

The `make_net_recovery` tool creates a system recovery image and stores it on a system that may be accessed using the network. The recovery image created by `make_net_recovery` is specific to the system it was created for and its identity includes hostname, IP address, networking information, etc. In the event of a root disk failure, the recovery image can be installed using Ignite-UX to recover the system.

The contents of the system recovery image will always include predefined files and directories that are considered *essential* to bringing up a functional system. By running `make_net_recovery` in interactive mode (with the `-i` option), the directories and files that make up the essential list can be displayed. In addition to the essential list, data can be included in the recovery image on a disk/volume group, file, or directory basis. Nonessential files and directories can also be included. See "Recovery Image Contents" (page 211) for more information.

Network Recovery Server Dependency

The recovery images created by `make_net_recovery` are designed to work with an Ignite-UX server; you cannot remove your Ignite-UX server and still use your recovery image.

Networking Features

Two NFS mount points are established on the client by `make_net_recovery`. The `/var/opt/ignite/clients` directory on the Ignite-UX server is mounted to the client system to store *configuration file*s that describe the client configuration and location of the *recovery image*. The second mount point is made to the *`archive_server:archive_dir`* (see the `-a` option) and is used to store the recovery image of the client system. The default storage location on the Ignite-UX server is `/var/opt/ignite/recovery/archives`. After successful or unsuccessful creation of the system recovery image, the NFS mount points are unmounted.

The NFS mount for the recovery image directory may be exported on a per-client basis. A separate recovery image directory is used for each client. This enables you to NFS export each directory only to the individual client owning the recovery image, which provides security.

NOTE: If clients obtain temporary IP addresses from *DHCP* that differ from the IP address that they use during normal operation, you must allow the client access to all of the possible IP addresses to ensure access to the recovery image. If you do not, the client may fail to mount the recovery image directory from the NFS server and the recovery will fail.

Log Files

On an Ignite-UX server, progress and errors are logged to:

`/var/opt/ignite/clients/`*`client`*`/recovery/`*`datetime`*`/recovery.log`

On a local system, progress and errors are logged to:

`/var/opt/ignite/recovery/`*`datetime`*`/recovery.log`

Adding Clients for Recovery

You can add a new client to your Ignite-UX server for the purpose of creating recovery images if the client is already running HP-UX. Unlike installation, adding a client for recovery does not require you to reboot the client. This is useful when you have installed the operating system, customized it, and now want to be able to recover it in the event of a problem or for disaster recovery purposes.

To add a new client to your Ignite-UX server, and then create a system recovery image, use the following steps:

TIP: You can execute the Ignite-UX GUI from a different system as if you were on the Ignite-UX server by using the following commands:

- On your host system, allow the Ignite-UX server to access your display by adding the Ignite-UX server hostname to your `xhost` list:

  `xhost +`*`Ignite-UX_server_hostname`*

- Set the `DISPLAY` variable to your local host system, if necessary. For example:

  `export DISPLAY=`*`your_host_system`*`:0`

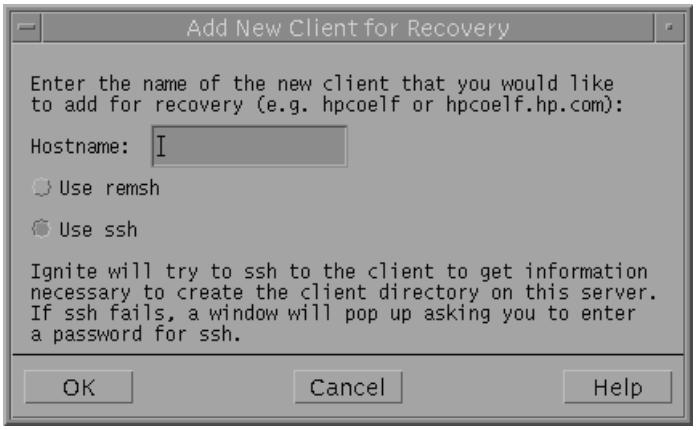
  where *`your_host_system`* is the `hostname` of your system.

1. On the Ignite-UX server, as `superuser` enter

   `/opt/ignite/bin/ignite`

2. Select **Add New Client for Recovery** from the **Actions** menu.

Figure 15-15 Add New Client for Recovery Dialog Box



3. In the Hostname box, enter the name of the client for which you want to create a recovery image.
4. Select how you want to communicate with the client, `ssh` or `remsh`, and then click **OK**. Use the default, `ssh`, for secure encrypted communications, or the unsecured `remsh`.
5. If you choose `ssh`, you are asked if you want to use this communication method for all subsequent recovery, as well as any installation sessions run from the Ignite-UX server. Click **Yes** to set `ssh` as the default client communication, or **No** if you only want to use `ssh` for this recovery session.

   The Ignite-UX server then attempts to contact the client to begin the recovery initialization process and create a directory to contain the client's information. In the event access to the client is denied, as in Figure 15-16, you are asked if you want to provide the `root` password.

Figure 15-16 Confirmation Dialog Box

Clicking **Yes** produces a terminal window allowing you to enter the `root` password, clicking **No** halts the addition of this client and returns you to the Ignite-UX GUI.

Once communication with the client is established and the client directory is constructed, a client icon appears in the Ignite-UX GUI.

6. Ensure that the client icon for which you want to create a recovery image is selected, and then select **Create Network Recovery Archive** from the **Actions** menu. You may be prompted for the `root` password for the client.

The network recovery tools needed on the client are automatically installed.

After some informative dialog boxes, an Include/Exclude Selection dialog box appears. To view the *essential* files, click **Show**. Essential files cannot be excluded, but you can customize the image by specifying additional volumes, directories, or files. When an item is identified as both Include and Exclude, the Exclude category takes precedence.

Examples of Network Recovery Image Creation

Create a Recovery Image from the Client

This command creates a *recovery image* from the client, using settings from the last invocation of Ignite-UX, and using the options file on the Ignite-UX server (*`myserver`*) in the default location, `/var/opt/ignite/clients/`*`client`*`/recovery/`:

```
make_net_recovery -s myserver
```

Create a Recovery Tape on a Client that Includes the Volume Group, `vg00`

To create a recovery image from the client that includes files from all *file system*s in the `vg00` volume group, enter:

```
make_net_recovery -s myserver -x inc_entire=vg00
```

Preview System Recovery

To preview the processing that would take place without actually creating the recovery image, enter:

```
make_net_recovery -s myserver -p
```

Recovering using the Network for PA-RISC Clients

To recover a failed disk or volume group using the recovery image:

1. Boot the failed system using one of these methods (see "Booting PA-RISC Clients from the Console " (page 111)):
   - Use Ignite-UX after `reboot` with `boot lan install`.
   - Boot from an Ignite-UX server using `bootsys` if the client operating system is running.
   - Boot the failed client locally by using a boot tape previously created with `make_boot_tape`.
2. Do not interact with *ISL*.

   If your Ignite-UX server supports installing more than one version of HP-UX, a target operating system menu will appear:

   ```
   ISL booting hpux KernelPrompt "Choose Operating System to Install:"

   1. target OS is B.11.11
   2. target OS is B.11.23 PA
   3. target OS is B.11.31 PA
   4. Exit

   Choose an operating system to install that your hardware supports :
   ```
3. At the client, from the main menu, select **Install HP-UX**.

a. Respond to the Network Configuration dialog box.
b. Respond to the UI Display Options dialog box (run at the Ignite-UX server or at a client.)
c. If working from the Ignite-UX server, select the client for the system to be recovered.

4. Select **Install/New Install**.
5. Select the recovery configuration to use, and then allow the recovery to continue.

Recovering using the Network for Itanium-Based Clients

To recover a failed disk or volume group using the system *recovery image*:

1. From the EFI Boot Manager menu, you will see a prompt to select a boot option. Select **Boot Configuration**.

```
EFI Boot Manager ver 1.10 [14.62]

Please select a boot option

    HP-UX Primary Boot: 0/1/1/1.2.0
    EFI Shell [Built-in]
    --------------------------------
    Boot Configuration
    System Configuration

 Use ^ and v to change option(s). Use Enter to select an option
```

2. The Main Menu appears and prompts you to choose an operation. Select **Add a Boot Option**.

```
EFI Boot Maintenance Manager ver 1.10 [14.62]

Main Menu. Select an Operation

        Boot from a File
        Add a Boot Option
        Delete Boot Option(s)
        Change Boot Order

        Manage BootNext setting
        Set Auto Boot TimeOut

        Select Active Console Output Devices
        Select Active Console Input Devices
        Select Active Standard Error Devices

        Cold Reset
        Exit
```

3. Select the appropriate network interface so that this network boot option loads the appropriate file from the following menu. For example, look for entries identified with a MAC address as in this example.

```
EFI Boot Maintenance Manager ver 1.10 [14.61]

Add a Boot Option.  Select a Volume

    IA64_EFI [Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun0,Lun0)/HD(Part1,Si
    IA64_EFI [Acpi(HWP0002,100)/Pci(1|0)/Scsi(Pun0,Lun0)/HD(Part3,Si
    IA64_EFI [Acpi(HWP0002,100)/Pci(1|1)/Scsi(Pun2,Lun0)/HD(Part1,Si
    Removable Media Boot [Acpi(HWP0002,0)/Pci(2|0)/Ata(Primary,Master)
    Load File [EFI Shell [Built-in]]
    Load File [Acpi(HWP0002,0)/Pci(3|0)/Mac(123456789000)]
    Load File [Acpi(HWP0002,100)/Pci(2|0)/Mac(987654321000)]
    Exit
```

4. Enter an appropriate boot option name at the message prompt. For this example, the new boot option is named `lan0`.

5. Exit to the main menu by pressing **Esc**. The new boot option now appears in the EFI Boot Manager main menu.

```
EFI Boot Manager ver 1.10 [14.62]

Please select a boot option

    HP-UX Primary Boot: 0/1/1/1.2.0
    EFI Shell [Built-in]
    Boot from lan0
    ---------------------------------
    Boot Configuration
    System Configuration

 Use ^ and v to change option(s). Use Enter to select an option
```

6. Select the new boot option you created. The following is an example of a successful boot using the new boot option.

```
Starting: Boot from lan0

@(#) HP-UX IA64 Network Bootstrap Program Revision 1.0
Downloading HPUX bootloader
Starting HPUX bootloader
Downloading file fpswa.efi   (328192 bytes)

(C) Copyright 2004 Hewlett-Packard Development Company,
    L.P.All rights reserved

HP-UX Boot Loader for IPF  --  Revision 2.018

Booting from Lan
Downloading file AUTO   (26 bytes)
Press Any Key to interrupt Autoboot
AUTO ==> boot Rel_B.11.23/IINSTALL
Seconds left till autoboot -   0
AUTOBOOTING...
AUTO BOOT> boot Rel_B.11.23/IINSTALL
Downloading file Rel_B.11.23/IINSTALL
```

7. At the client, from the main menu, select **Install HP-UX**.
   a. Respond to the Network Configuration dialog box.
   b. Respond to the UI Display Options dialog box (run at Ignite-UX server or at a client).
   c. If working from the Ignite-UX server, select the client for recovery.
8. Select **Install/New Install**.
9. Select the recovery configuration to use, and then allow the recovery to continue.

Retaining Recovery Images

The `-n` option of the `make_net_recovery` command allows you to retain a fixed number of *recovery image*s on your system, the default being two images. The oldest recovery image is removed when a new recovery image is created and the specified limit is exceeded. For more information, see *make_net_recovery*(1M).

You might want to prevent a specific recovery image from being deleted from your system. To do this, you must rename the recovery image and the image directory, and use `manage_index` to reflect the new names in the *`CINDEX`* file.

The following example renames a recovery archive *`yyyy-mm-dd,hh:mm`* to `Recovery_Archive.sav`:

1. Log in to the system where recovery images are stored. This could be a system other then the Ignite-UX server.

2. Rename the recovery archive. (The path to your recovery archive might be different from the example.) The name of the saved recovery archive can be anything unique, but it should be outside the naming convention *yyyy-mm-dd,hh:mm*.

   ```
   # cd /var/opt/ignite/recovery/archives/client
   # mv yyyy-mm-dd,hh:mm Recovery_Archive.sav
   ```

3. If the system where recovery images are stored is different from your Ignite-UX server, log in to the Ignite-UX server.
4. Rename the recovery archive directory.

   ```
   # cd /var/opt/ignite/clients/client/recovery
   # mv yyyy-mm-dd,hh:mm Recovery_Archive.sav
   ```

5. If the renamed recovery archive directory is the target of the symbolic link `latest`, then link `latest` to the new directory.

   ```
   # rm latest
   # ln -s Recovery_Archive.sav latest
   ```

6. Edit the `archive_cfg` file in the `Recovery_Archive.sav` directory to reference the new recovery archive name.

   Change the `archive_path` variable inside the `(source_type == "NET")` clause to the name of the saved recovery image.

   ```
   (source_type == "NET") {
     archive_path = "Recovery_Archive.sav"
   }else {
     archive_path = "1"
   }
   ```

7. Use the `manage_index` command to update the *configuration clause* name and description, and to change the directory to the archive's *configuration file*s.

   Rename the configuration clause:

   ```
   # manage_index -m 'yyyy-mm-dd,hh:mm Recovery Archive' \
       -c 'Your configuration name' \
       -i /var/opt/ignite/clients/client/CINDEX
   ```

   Update the configuration clause description:

   ```
   # manage_index -y 'Your configuration description' \
       -c 'Your configuration name' \
       -i /var/opt/ignite/clients/client/CINDEX
   ```

   Update the archive's configuration files to the new directory.

   Use `manage_index` to get a list of all the files associated with the *cfg clause*.

   ```
   # manage_index -w -c 'Your configuration name'\
       -i /var/opt/ignite/clients/client/CINDEX
   ```

   For each configuration file to rename, remove the references to the old directory. This example renames `archive_cfg`. The other two configuration files to move are `control_cfg` and `system_cfg`.

   ```
   # manage_index -t -c 'Your configuration name' \
       -f recovery/yyyy-mm-dd,hh:mm/archive_cfg \
       -i /var/opt/ignite/clients/client/CINDEX
   ```

   For each configuration file to rename, add the references to the new directory:

   ```
   # manage_index -a -c 'Your configuration name' \
       -f recovery/Recovery_Archive.sav/archive_cfg \
       -i /var/opt/ignite/clients/client/CINDEX
   ```

Making Recovery Configuration File Additions

Using the recovery config.local file

To have a *configuration file* automatically added to all new recovery *configuration clause*s for a given client, create a new Ignite-UX configuration file called:

/var/opt/ignite/clients/*client*/recovery/config.local

For local tapes the file is located in:

/var/opt/ignite/recovery/config.local

This config.local file will automatically be included in your recovery configuration for this client each time you run make_net_recovery. (The make_net_recovery command is run for you when you create a *recovery image* using the Ignite-UX TUI or GUI.)

If you already have recovery configurations for this client and would like them to include the recovery config.local file, use the manage_index command to include a reference to recovery/config.local in all of the configuration clauses.

The following example adds the recovery config.local file to all the 11i v2 cfg clauses in the *CINDEX* file.

```
# manage_index -a -r B.11.23 -f recovery/config.local \
    -i /var/opt/ignite/clients/client/CINDEX
```

For more information, see *manage_index*(1M).

Adding a depot

If you want to install a *recovery image* on a different hardware platform or HP-UX Virtual Partitions (vPars) software, you might have to add software to the recovery configuration in the CINDEX file.

To add software to a recovery configuration, first create a configuration file for the software depot with make_config. Then, add the configuration file to the recovery configuration clause in the client's CINDEX file.

The following example creates a configuration file sw_cfg from the depot sw_depot and adds the configuration file to all the configuration clauses for the release specified in the configuration file name (Rel_*release*).

```
# make_config -s sw_depot -c /var/opt/ignite/data/Rel_release/sw_cfg
# manage_index -a -f /var/opt/ignite/data/Rel_release/sw_cfg \
    -i /var/opt/ignite/clients/client/CINDEX
```

During recovery, the software bundles available in sw_depot will be available for selection from the user interface software tab.

For more information, see *manage_index*(1M).

If you want the sw_cfg configuration file to be added to all new recovery configurations created for the client, add the sw_cfg file to the config.local file. For more information, see "Using the recovery config.local file" (page 233).

Selecting File Systems During Recovery

It is possible to change the way your disks are configured when you recover using a *recovery image* created by make_net_recovery. If you want to use a standard HP *file system* layout, you can specify the disk configuration using Ignite-UX. For more information, see "Basic Tab" (page 135).

If you do not want to use a standard HP file system layout, you can modify the /var/opt/ignite/clients/*client*/CINDEX file for the client you are recovering. The CINDEX file contains one or more configuration clauses that refer to the recovery images you have previously created with make_net_recovery. Add a new configuration file entry to the

clause you intend to recover from. If you want to add the standard HP file system choices, add the file

`/opt/ignite/data/Rel_release/config`,

where *release* is the operating system release on the client you intend to recover. For example:

`/opt/ignite/data/Rel_B.11.11/config`

would be added for a client with the HP-UX 11.11 operating system. This new configuration file entry should be the first entry in the clause you are modifying.

When you use the Ignite-UX GUI during recovery, select the File System type you want to use on the **Basic** tab.

Tape Recovery With No Tape Boot Support — Two-Step Media Recovery

You can use the Ignite-UX tape recovery tool to recover your system even if there is no tape boot support on the system.

Certain configurations, which are on most HP Integrity servers, allow you to directly boot a recovery tape. For information about what configurations and minimum firmware revisions support native tape boot on HP Integrity servers, refer to the *Ignite-UX Installation Booting* white paper, available at http://www.hp.com/go/ignite-ux-docs.

IMPORTANT: HP recommends you *boot* from the Operating Environment (OE) media that corresponds to the version of Ignite-UX you used to create your recovery tape. HP supports *only* matched OE media and recovery tape version combinations.

Using unmatched versions might produce unpredictable results, including: missing files, error messages, system hangs, and panics.

If you have misplaced the matching OE, or are using an Ignite-UX version with no matching OE, you can make your own minimal bootable CD/DVD to support *two-step media recovery*. See "Creating a Boot CD/DVD or an Installation DVD" (page 203), or see the section *How do I create the CD equivalent of a tape created by make_boot_tape?* in the *Ignite-UX Custom Configuration Files* document found on **http://www.docs.hp.com/en/oshpux11iv3.html#Ignite-UX**.

1. Insert the HP-UX OE media or the bootable custom media into the appropriate drive, and boot from it. Either media must match the HP-UX version used when creating the recovery tape.

   The User Interface and Media Options screen appears. Select Media only installation and then **OK**.

```
                    User Interface and Media Options

      This screen lets you pick from options that will determine if an
      Ignite-UX server is used, and your user interface preference.

   Source Location Options:
      [ * ]  Media only installation
      [   ]  Media with Network enabled (allows use of SD depots)
      [   ]  Ignite-UX server based installation

    User Interface Options:
      [ * ]  Advanced Installation (recommended for disk and file
             system management)
      [   ]  No user interface - setup basic networking, use defaults
             and go
      [   ]  Remote graphical interface running on the Ignite-UX server

      Hint: If you need to make LVM size changes, or want to set the
            final networking parameters during the install, you will
            need to use the Advanced mode (or remote graphical
            interface).

      [   OK   ]              [ Cancel ]                 [  Help  ]
```

2.
```
                Media Installation

This screen provides an option to switch the install source
from the default CD/DVD to a recovery tape.  This is helpful
for those systems and for tape devices which do not support
booting from a tape.

  [   ] CD/DVD Installation
  [ * ] Boot from CD/DVD, Recover from Tape

  [ OK ]      [ cancel ]    [ Help ]
```

3. Select **Boot from CD/DVD, Recover from Tape** and select **OK** to advance to the Tape Drive Selection screen:

```
                Tape Drive Selection

There are one or more tape drives detected on the system.
Insert your recovery tape into one of the drives and then
select that drive from the list below.

Use the <tab> and/or arrow keys to move to the desired TAPE device,
then press <Return/Enter> to select.

    HW Path              Device File            Description
    -------------------------------------------------------

   [ 0/4/1/0.0x6.0x0    /dev/rmt/c6t6dOBEST      HP_SDLT600 ]
```

4. Select the tape drive that contains the *recovery image* tape, then press **Enter** to start the installation of the recovery image from the chosen tape drive.

Notes on Cloning Systems

Ignite-UX offers two main options for replicating (cloning) systems. The more flexible and complex *golden image* method makes use of `make_sys_image` to create an *archive* of the source system, followed by manually modifying *configuration file*s to meet your needs. A much simpler (but less flexible approach) uses *`make_[tape|net]_recovery`*. The pros and cons of each are described here.

In each case, the source system that is used must contain software that is compatible with all clients. This means that the version of HP-UX, patches, drivers, etc., must be sufficient for all systems involved. This often requires installing a superset of software and drivers onto the source system that will be used on all potential clients.

Using `the make_sys_image` method

Using the golden image method of creating an archive with `make_sys_image` and then modifying Ignite-UX configuration files to reference the archive is very flexible, but somewhat time consuming. The end result gives you:

- The ability to install systems from network or media from either an Ignite-UX server or local clients.
- The ability to customize the process and tune it to accommodate many different situations.
- A "clean" system: log files and most remnants specific to the source system are removed.
- A rebuilt *kernel* containing just the drivers needed by the client's hardware.
- The ability to install additional software or patches on top of the system *archive* from an SD *depot*. This reduces the need to recreate the archive and enables you to add support for new hardware that requires new patches or drivers without making a new archive.

See Chapter 11: "Golden Images", for more information.

Using the make_[*tape*|*net*]_recovery method

The make_[tape|net]_recovery tools are designed to reproduce a system exactly the way it was at the time the snapshot was taken. These tools try to accommodate cloning in various ways:

- You can change hostname and networking information.
- You can make changes to disks and *file system*s during the recovery.
- You can detect hardware model changes and rebuild the *kernel*.

However, their attempt to reproduce a system exactly may be undesirable:

- The *disk layout* is saved "as-is" from the original system and does not have flexible logic to accommodate disks of varying sizes or locations.
- Hardware instance numbers for devices that exist at the same paths between systems have the instance numbers preserved from the original system. This can cause non-contiguous assignments in instance numbers, which is usually only a cosmetic problem.
- Many files that are specific to the system the recovery image was taken from are preserved. This includes many log files, etc.
- When the kernel is rebuilt (in the "cloning" situation), drivers may be added as needed by the hardware, but unused drivers will not be removed.

Cloning a System Using make_net_recovery

The recovery configurations and archives created by make_net_recovery are stored in a separate directory on the Ignite-UX server for each client. Using the configuration and archive created by make_net_recovery on one system to install a different system involves manually copying some configuration files and allowing NFS access to the source system's archive.

A system recovery tape created using make_tape_recovery can also be used to clone systems. The system you are installing by cloning must have a local tape drive so you can boot from the system recovery tape.

The following example illustrates how to clone a system:

1. Use make_net_recovery or Ignite-UX to create a system recovery image of the source system.
2. On the Ignite-UX server, if the client to be installed does not currently have a directory in /var/opt/ignite/clients but is up and running, use the Ignite-UX GUI to create that directory using **Add New Client for Recovery** from the **Actions** menu. For more information, see "Adding Clients for Recovery " (page 227).

   If the client is not running, you will either need to boot it from the Ignite-UX server or from media in order for this directory to be created.

3. Copy the *CINDEX* and recovery directory from a source client to the target client directory. If the target client has previously used make_net_recovery, it will already have a CINDEX file. If the CINDEX file for the client exists, you might want to save a copy and then edit the file to add the desired entries from the source client. The following commands copy the required files. You may specify *src_client* and *target_client* using either the MAC address or the client's hostname, which is a symbolic link to the MAC address:

   ```
   # cd /var/opt/ignite/clients/src_client
   # find CINDEX recovery | cpio -pdvma ../target_client
   ```

4. Give the target client NFS access to the recovery image of the source system. Typically each target client has its own directory on the source system for storing the recovery images and

the directory is exported only to the individual client. To do this, log in to the system that holds the recovery image (normally the Ignite-UX server).

For HP-UX 11i v3 systems:

- Edit the `/etc/dfs/dfstab` file on the source client.
- Append **`,ro=`*`target_client`** to the -o argument of the source client's line, where *`target_client`* is a fully qualified client name.
- Run # **`shareall -F nfs`**

For HP-UX 11i v1 and 11i v2 systems:

- Edit the `/etc/exports` file on the source client.
- Append :***`target_client`*** to the end of the source client's line, where *`target_client`* is the hostname of the target system.
- Run # **`exportfs -av`**

See *dfstab*(4) or *exportfs*(4) for more information.

5. Boot the target client from the Ignite-UX server using any method you prefer. When you install the system, you can select from the recovery configurations of the source system.
6. Change the system networking parameters for the client during the installation.

For additional information regarding system cloning, see the *Successful System Cloning using Ignite-UX* white paper at

`http://www.hp.com/go/ignite-ux-docs`.

System Recovery Questions and Answers

Question:

Can I use a network recovery image if my system is not on the same subnet as the Ignite-UX server?

Yes, there are the commands `make_boot_tape`, *`make_ipf_tape`*, and `make_media_install` that create minimal boot media for use by any client. The media contain just enough information to boot a client and then connect to the Ignite-UX server where the tape, CD, or DVD was created. If that is the server where the client's recovery *configuration file*s are stored, then the client can be recovered.

It is not possible to boot all systems from a tape device. See "Tape Recovery With No Tape Boot Support — Two-Step Media Recovery" (page 234).

If you initiate recovery tape creation from the Ignite-UX server, the server will warn you if the client requires boot media. If you ignore this warning, misplace your boot media, or find that your media are for the wrong Ignite-UX server, you can always create new boot media on the server you want to use. There is no client-specific information on the media.

Notice that media created by `make_boot_tape`, `make_ipf_tape`, and `make_media_install` are useful not only for recovery situations, but also for ordinary installations. If you do not want to set up a boot helper for systems on a separate subnet than the Ignite-UX server, you can simply create bootable media.

For more information, see Chapter 14 (page 197), *make_boot_tape*(1M), and *make_ipf_tape*(1M).

Other options include direct boot profiles (see "Direct Boot Profiles for Itanium-Based Systems" (page 115)) and boot helpers (see "Ignite-UX `bootp` Boot Helper" (page 63)).

Question:

How can I change my setup so a network recovery image is available not only on the system for which it was created, but also on other systems with very similar hardware?

Because networking information can be changed using the interface and will not be overwritten by files extracted from the image, it is natural to think about sharing *recovery image*s for systems with identical or nearly identical hardware. But unlike shared configurations that appear in the configuration list for all clients, network recovery configurations only appear in the configuration list of the client for which they were created.

The source for shared configurations is the `/var/opt/ignite/data/INDEX` file that is created when Ignite-UX is installed, and the source for client-specific configurations is the *`CINDEX`* file that is created by `make_net_recovery` in the `/var/opt/ignite/clients/`*`client`* directory. One simple way to share a recovery configuration among two systems with similar hardware is to copy the `CINDEX` file and the recovery directory of the client with the image to the directory of the client without the image. The fact that the entries in `CINDEX` use relative paths means you do not have to change the `CINDEX` file when you copy it. You will need to NFS export the directory containing the image to the sharing client. For detailed information on this process, see "Cloning a System Using `make_net_recovery`" (page 237).

Question:

I do not want to interact with the user interface after I reboot the client. How can I have my latest network recovery image chosen automatically?

As long as the client is currently booted, use `bootsys -a` to start the installation process on the client without the need to interact with the user interface.

Ignite-UX chooses a configuration to use based on these guidelines:

- If `/var/opt/ignite/clients/`*`client`*`/config` exists, use the *cfg clause* specified there.
- If `/var/opt/ignite/clients/`*`client`*`/config` does not exist, use the default cfg clause for the client.

  The default cfg clause for the client is the last applicable entry set to true in the `CINDEX` file if it exists. Otherwise, the default cfg clause is the last applicable entry set to true in the `INDEX` file. Because `make_net_recovery` sets the most recently created recovery cfg clause to true in `CINDEX` whenever it creates a new image, that will be the default unless it is manually changed.

To set Ignite-UX to choose the latest network *recovery image* automatically:

1. Rename or remove the `config` *configuration file* currently in the client's directory, or use the `bootsys -f` option.
2. Run this from the Ignite-UX server:

   `bootsys -a` *`client`*

For information on automating an installation, see the descriptions of `run_ui`, `control_from_server`, and `INST_ALLOW_WARNINGS` in *instl_adm*(4).

Question:

What causes `tftp` errors when recovering or installing a system?

- Only `/opt/ignite` and `/var/opt/ignite` should be needed for `tftp` access.
- Check `/etc/inetd.conf`
- Files in `INDEX` should not be in directories outside `/opt/ignite` and `/var/opt/ignite`.

Question:

What can I do when problems occur from hot-swapping disks during recovery?

Ignite-UX supports only hot-swappable disks that are completely in place and not removed when creating a *recovery image*. Proper software and hardware procedures must be used for hot-swap disk removal or replacement before or after recovery, but not during. The LVM command,

`lvlnboot`, used by `save_config` does not work when a disk is removed and the system is in this intermediate state. If this command is not working, a recovery cannot succeed.

Question:

Why is the EFI volume not restored during a recovery?

Ignite-UX destroys the old *EFI* volume on the boot disk and creates a new EFI volume every time the system is installed. At no point during the installation is the old EFI volume copied and restored to the disk.

To restore the EFI volume to the disk, reinstall the application or look at the SD configure scripts for the application and then rerun the commands that put the EFI volume in place on the disk.

Question:

Why does `make_net_recovery` fail when the image is 2 GB or more?

The `make_net_recovery` command uses NFS to write/read the system image from the client to/from the server. To manage images greater than 2 GB requires that both the client and server use NFS protocol Version 3 (PV3). NFS PV3 is standard on all HP-UX 11i releases.

If you know you have NFS PV3 and are having problems, check the `/etc/rc.config.d/nfsconf` file for the configured parameter, `MOUNTD_VER` that defines the default mount to be PV2 or PV3; it must be set to 3.

Question:

Why is the LAN address different after replacing a client system?

Ignite-UX uses a separate directory for each client under `/var/opt/ignite/clients`. Each subdirectory is named based on the client's LAN address. If you replace the client hardware or even the LAN interface that the old LAN address was based on, it will no longer access the same directory on the server.

The simplest solution is to obtain the new LAN address with the BCH command `LanAddress` or the *EFI* command `lanaddress`. Once you have the new address, manually rename the directory. You may just remove the hostname symlink (it will be recreated automatically). Note that the LAN address must be in all uppercase, and begin with `0x`.

If you already booted from the client and caused the server to create a new directory, you can just remove that directory before renaming the old directory. To avoid losing the recovery information, be careful not to remove the original directory. For example:

```
# cd /var/opt/ignite/clients
# mv 0x00108300041F 0x00108300042A
# rm old_hostname
```

Question:

When recovering a system across multiple disks, how are the volumes assigned to disks?

Ignite-UX will do all it can to find a solution to refitting the volumes back to disks. If Ignite cannot find a solution, it will automatically turn off the mapping by setting the Disk Mapping value from **Assigned Disk** to **Any**. For information regarding how to set the Disk Mapping value, see "Volume Parameters" and the File System/Swap Attributes section in *instl_adm*(4).

Question:

Why is the tape device different between making the recovery image and using the recovery image?

During the recovery process, when the *file system* is set up and the I/O tree is initialized, tape device files might be mapped differently from when the original recovery tape was made.

Therefore, it is possible for a recovery tape to be created with one tape device file, for instance `/dev/rmt/0m`, and recovered from a different device file, such as `/dev/rmt/2m`, even though the physical device is the same.

16 Support and Other Resources

Contacting HP

Before you contact HP

Be sure to have the following information available before you contact HP:

- Technical support registration number (if applicable)
- Product serial number
- Product model name and number
- Product identification number
- Applicable error message
- Add-on boards or hardware
- Third-party hardware or software
- Operating system type and revision level

HP contact information

For the name of the nearest HP authorized reseller:

- See the Contact HP worldwide (in English) webpage (http://welcome.hp.com/country/us/en/wwcontact_us.html).

For HP technical support:

- In the United States, for contact options see the Contact HP United States webpage (http://welcome.hp.com/country/us/en/contact_us.html). To contact HP by phone:
  - Call 1-800-HP-INVENT (1-800-474-6836). This service is available 24 hours a day, 7 days a week. For continuous quality improvement, calls may be recorded or monitored.
  - If you have purchased a Care Pack (service upgrade), call 1-800-633-3600. For more information about Care Packs, refer to the HP website (http://www.hp.com/hps).
- In other locations, see the Contact HP worldwide (in English) webpage (http://welcome.hp.com/country/us/en/wwcontact_us.html).

Documentation feedback

HP welcomes your feedback. To make comments and suggestions about product documentation, send a message to docsfeedback@hp.com.

Include the document title and manufacturing part number. All submissions become the property of HP.

Related information

Documents

The following documents are available on the HP Business Support Center (BSC) www.hp.com/bizsupport/ and most are available on the HP-UX core documents page on the BSC www.hp.com/go/hpux-core-docs.

The Ignite-UX documents page on the BSC is www.hp.com/go/ignite-ux-docs and the HP-UX Software Deployment documents page is www.hp.com/go/sw-deployment-docs.

- *Ignite-UX Quick Start Guide*
- *Ignite-UX Reference*
- *Ignite-UX Release Notes*
- *Ignite-UX Supported Versions*

- *Ignite-UX Frequently Asked Questions*
- *Ignite-UX Custom Configuration Files*
- *Ignite-UX USB Memory Stick Boot* White Paper
- *Ignite-UX and SAS Devices* White Paper
- *Ignite-UX and MirrorDisk/UX* White Paper
- *Successful System Cloning using Ignite-UX* White Paper
- *Successful System Recovery using Ignite-UX* White Paper
- *Installing and Updating Ignite-UX* White Paper
- *Ignite-UX Installation Booting* White Paper
- *Read Before Installing or Updating to HP-UX*
- *HP-UX Installation and Update Guide*
- *HP-UX Reference*
- *HP-UX System Administrator's Guide*
- *Managing Systems and Workgroups: A Guide for HP-UX System Administrators*
- *Software Distributor Administration Guide*
- *Patch Management User Guide for HP-UX 11i Systems*
- *nPartition Administrator's Guide*
- *Veritas File System 4.1 Administrator's Guide*
- *Veritas File System 5.0 Administrator's Guide*
- *Veritas Volume Manager 4.1 Administrator's Guide*
- *Veritas Volume Manager 5.0 Administrator's Guide*
- *HP Integrity iLO 2 MP Operations Guide*

Websites

- The Ignite-UX Product Website, http://www.hp.com/go/ignite-ux, contains:
  - Product announcements
  - Access to downloads of the latest version of Ignite-UX
- Ignite-UX Software Depot webpage http://www.hp.com/go/ignite-ux-download
- Links to documentation can be found on the Business Support Center http://www.hp.com/go/ignite-ux-docs
- HP_UX_Docs Twitter account http://www.twitter.com/HP_UX_Docs
- HP-UX 11i Knowledge-on-Demand: Best practices from our lab to yours http://h71028.www7.hp.com/enterprise/us/en/os/hpux11i-kod-overview.html?jumpid=reg_R1002_USEN
- Ignite-UX Training http://www.hp.com/education/courses/h1978s.html
- HP-UX training http://www.hp.com/education/sections/hpux.html
- HP IT Resource Center (ITRC) http://www.itrc.hp.com
- ITRC Ignite-UX Forum http://forums1.itrc.hp.com/service/forums/categoryhome.do?categoryId=209
- HP-UX Technical Documentation www.hp.com/go/hpux-core-docs
- HP Integrity Systems Family Overview http://www.hp.com/go/integrity
- Software Depot http://www.hp.com/go/softwaredepot
- HP Software Releases and Media http://www.hp.com/softwarereleases/releases-media2

For information about the firmware versions required to support direct boot profiles on various Itanium systems, see the following links:

- http://docs.hp.com/en/FCN-SM/FCN-SM.htm
- http://h20000.www2.hp.com/bizsupport/TechSupport/ProductRoot.jsp?lang=en&cc=us&taskId=135

Typographic Conventions

The following conventions are used in this document:

Table 16-1 Typographic Conventions

Typeface	Usage	Examples
Italics	Emphasis	The VxVM components in the Ignite-UX install environment *and* the installation must be version 5.0.
Book Title	Book titles	*Ignite-UX Reference*
glossary term	Glossary term	*configuration clause*
Key	A keyboard key (**Return** and **Enter** refer to the same key)	**Esc**
Bold	A selectable GUI or TUI item.	**Go!**
`Command`	Commands entered via the keyboard	`bootsys -R`
`File name`	Files and directories	`/dev/dsk/c0t0d0`
`Computer output`	Text a program displays	`Please select a boot option`
`User input`	Text you type	**`15.1.54.117`**
`Variable`	Variables to be replaced by a name or value	*`IP Address`*
`Listing`	File contents	`cfg "HP-UX b.11.23 Default" {` `}`
`Screen`	An example display	`Seconds left until autoboot - 0` `AUTOBOOTING...`
`[ ]`	The contents are command options. If the contents are a list separated by `|`, choose one of the items.	`ls [ -a ]` `mount [suid | nosuid ]`

.
.
.
	Extensive computer output or an excerpt	`source_type="NET"`
		.
		.
		.
		}

CAUTION	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.	**CAUTION:** Any data on the client disks that are used for installation, including the operating system, are removed entirely as part of this installation process.
IMPORTANT	An alert that calls attention to essential information.	**IMPORTANT:** You must select **Save** to enable the new boot menu option before selecting **OK** to exit.
NOTE	An alert that contains additional or supplementary information.	**NOTE:** Depending on your server, this screen might look slightly different.

A Troubleshooting

For information regarding system recovery troubleshooting, see "System Recovery Questions and Answers" (page 238).

Also, see the *Ignite-UX Frequently Asked Questions* manual, available at http://www.hp.com/go/ignite-ux-docs.

Errors and Warnings

During an HP-UX installation, messages indicating the status of tasks are written to the *client* log file and either the log viewer (*Ignite-UX server* GUI-based installation) or the console (client UI-based installation). Some of these messages may be `ERROR` or `WARNING` messages, which have the following significance:

`ERROR` This indicates a serious problem, usually requiring action from you in order to proceed with an installation.

`WARNING` This indicates something out of the ordinary, but *not* fatal. The warning *may* require action.

If you encounter a message or experience unusual behavior, you can use the following sections as prioritized lists of likely problems and their solutions.

Ignite-UX Server Problems

Mixed Versions of Ignite-UX

Cannot find `/d_cfg_mnt_sb61/monitor_bpr` after updating the server.

This is caused by having a mix of Ignite-UX *fileset* versions on your server. In most cases it happens when you update only one release *bundle* even though you install other releases from that server.

An easy way to check for this case is to look at the output from the command:

```
swlist Ignite-UX
```

All the filesets should have the same version; if not, then you need to install all consistent versions. If you have *boot helper system*s (see "Ignite-UX `bootp` Boot Helper" (page 63)), they also need to have the Ignite-UX product updated to match the same version as the server that they reference.

Installing Systems with Ignite-UX

Numerous `samreg` errors

Installing from an image returns numerous `samreg` errors.

The problem is that the SAM *fileset*s have not been configured when certain products are trying to register themselves with SAM.

The work-around is to place the following *configuration clause* in `/var/opt/ignite/config.local` or in the *configuration file* that contains the `sw_source` clause of the core operating system.

```
sw_source "core"
{
post_load_cmd += "swconfig -xautoselect_dependencies=false
-xenforce_dependencies=false SystemAdmin.SAM -xreconfigure=true"
}
```

NOTE: Due to formatting limitations, unintended line wraps exist in the previous example; the `post_load_cmd` line should not wrap in your *configuration file*. You verify the syntax with the `instl_adm -T` command.

Problem Installing Clients on Multiple Subnets

Problems occur when installing clients on multiple subnets.

Executing a LAN *boot* of clients on multiple subnets to a single, multi-homed Ignite-UX server has the following limitations:

- The `instl_bootd` *daemon* allocates IP addresses from the `instl_boottab` file and matches the IP addresses with the subnet from which the request came. If the `instl_boottab` file does not contain an IP address that is valid for the client's subnet, the client will not be able to boot from the server. Due to this lack of information, it can allocate an IP address that is not valid for the client's subnet, and thus the client will not be able to boot from the server.

  The workarounds for this problem are:

  — For every subnet from which you may want to boot clients, you must have at least one IP address that is valid for that subnet in `/etc/opt/ignite/instl_boottab`. This ensures that `instl_bootd` can allocate an appropriate address.
  — For every possible client that you may want to boot, assign "reserved" IP addresses in `/etc/opt/ignite/instl_boottab` that are tied to the client's MAC address. This ensures that `instl_bootd` allocates an appropriate address (see the comments in the `instl_boottab` file on how to reserve addresses).
  — Alternatively, you can set up entries in `/etc/bootptab` for every client that you want to boot from the Ignite-UX server.
  — Configure a *boot helper system* on each subnet that the client can boot from before contacting your central Ignite-UX server. See "Ignite-UX `bootp` Boot Helper" (page 63).

- The `server` keyword that specifies the IP address for your Ignite-UX server can only correspond to one of the LAN interfaces. If each subnet is routed such that all clients can use the one IP address to contact their server, then the installation will work. However, it is more efficient for the client to use the server's IP address that is connected directly to the client's own subnet. If a client is on a subnet that does not have a route to the IP address specified by `server`, it will not be able to contact the server after it boots.

  Workarounds for this problem are:

  — Manually correct the server's IP address on the networking dialog box that appears on the client console when you boot the client.
  — Use a boot helper on each subnet. When using a boot helper, the server's IP address can be specified correctly on each helper system.

Too Much File Space Needed

Ignite-UX requests more file system space than expected.

Ignite-UX adds the value of `_hp_addnl_fs_free_pct`(normally 10 percent) to the amount required by the software impact. The configuration variable, `_hp_addnl_fs_free_pct` can be set in any *configuration file*. It is set to a default value of 10 percent in each default release configuration file. You can set this value using the **Additional...** button on the Basic tab in the Ignite-UX GUI during an interactive installation. For more information, see "Basic Tab" (page 135).

You may have software *bundle*s that have overlapping contents (*fileset*s or files or both). The `make_config` command makes `sw_impact` statements for each bundle without doing anything special to guard against over-counting when the bundles overlap. For example the `Ignite-UX-11-`*xx* bundles all overlap quite a bit so when you install all of them using Ignite-UX, it estimates too much space. To find the space needed, add the `sw_impact` of all the `sw_sel` software you are installing.

Debugging SD During Cold-Installation

How do I monitor Software Distributor operations during cold-installations from the Ignite-UX server?

Software Distributor debugging can be enabled on a per-session basis without modifying the Ignite-UX server configuration files. From the initial Ignite-UX menu on the client, select **Advanced Options** then **Edit config file**. This invokes `vi` and you could add, for example:

```
env_vars += "SDU_DEBUG_RPC=1"
sd_command_line += "-x logdetail=true -x loglevel=2"
```

Additionally, these configuration statements can be added to the Ignite-UX server configuration file `/var/opt/ignite/config.local` if debug output is wanted for multiple clients or for multiple installation sessions and avoids adding them interactively each time.

Booting Errors on PA-RISC Systems

Error:`IPL error: bad LIF magic`

Possible problems are:

- The `tftp` service does not have access to `/opt/ignite` and `/var/opt/ignite`. The `/etc/inetd.conf` file on the server should have an entry similar to:

  ```
  tftp dgram  udp wait   root /usr/lbin/tftpd    tftpd\
  /opt/ignite\
  /var/opt/ignite
  ```

  If not, correct the `inetd.conf` file and run **`inetd -c`**. Kill any `tftpd` processes that may be running. Installing Ignite-UX should set up `inetd.conf`. The `tftp` service can also be configured using SMH or SAM.
- Using a `bootptab` entry for the client that is referencing a nonexistent boot file (bf).
- A corrupted `/opt/ignite/boot/boot_lif` file.

Booting Error on Itanium-Based Systems when using `/etc/bootptab`

Error:`PXE-E16: Valid PXE offer not received.`

`Exit status code: Invalid Parameter`

When using `/etc/bootptab` to define Ignite-UX *boot* services, a number of problems can be introduced resulting in this error. The following checklist can be used to isolate the problem:

1. Check `inetd`:
   - Check `/etc/inetd.conf` to make certain `bootps` and `tftp` entries have been uncommented. Make certain the `tftp` line contains `/opt/ignite` and `/var/opt/ignite` paths.
   - Signal `inetd` to reread the *configuration file*s (`inetd -c`) after the files are edited. If the `inetd` process is not running, start it using:

     `/sbin/init.d/inetd start`

   - Check `/var/adm/syslog/syslog.log` to make certain `inetd` was restarted and no bad messages are found. Specifically check for messages from `bootpd` and `tftpd`.
   - Check for entries in `/var/adm/inetd.sec` that may cause `inetd` to deny service to certain clients.
2. Check `bootpd`:
   - Check the `/etc/bootptab` entry. Make certain the MAC address matches the client MAC address. Use `dhcptools -v` to validate the format of the `/etc/bootptab` file.
   - Check for entries in `/etc/dhcpdeny` to ensure that `bootpd` is not set to deny service for particular clients.

— Check `/var/adm/syslog/syslog.log` for a message from `bootpd` that indicates it was started when a `bootpd` packet was received.

If packets were not received, use a tool such as `tcpdump` to check for network packets. Verify that `bootp` packets are being seen by the system. If you do not have `tcpdump` on your system, you can download it for either HP-UX 11i v1 or HP-UX 11i v2 from the HP Internet Express Website at

https://h20293.www2.hp.com/portal/swdepot/try.do?productNumber=HPUXIEXP1111

or

http://h20293.www2.hp.com/portal/swdepot/-displayProductInfo.do?productNumber=HPUXIEXP1123

— Check to see if there are other systems on the network that may also be replying to the booting client system.

— Check to see if the system booting is on a different subnet to the `bootp` server to ensure that any router between the two allows the forwarding of `bootp` requests. The configuration is router specific.

3. Check `tftpd`:

— Check the `tftp` line in `/etc/inetd.conf` to make certain that the `/opt/ignite` and `/var/opt/ignite` directories are listed.

— Check the `tftpd` connection manually by using the `tftp` command, for example:

a. `$ tftp [server-name]`

b. `tftp> get /opt/ignite/boot/nbp.efi /tmp/nbp.efi`

Received [*n*] bytes in [*s*] seconds

c. `tftp> quit`

Problems Pointing to Client Over Network

The `control_from_server=true` and `run_ui=false` variables are in [W|V|I]INSTALLFS, but I still get prompted for information on the client.

Possible problems are:

- If the dialog box is showing the client name in an editable field and a **Cancel** button at the bottom of the dialog box, then all is well and there should be an icon waiting for you on the Ignite-UX GUI. The text box enables you to change the icon name or switch to a client side install.
- If the dialog box is showing two or more LAN interfaces to choose from, then there was not enough information in the *configuration file*s to tell it which LAN to use. Once you select a LAN and select **Install HP-UX**, you should be set.
- If the dialog box is prompting you for networking information, then either *DHCP* did not respond or there is no entry in `/etc/bootptab` for the client. Enter the network information, select **Install HP-UX** and continue the install.

Applications Hang After Igniting

Some applications and shells hang over NFS after igniting.

The reason for the hang is most likely due to a problem with the NFS file locking *daemon*s, `rpc.statd` and `rpc.lockd`, caused by the action of reinstalling the system. Many applications use file locking and can hang in this situation. Most common are user home directories that are NFS mounted, in which case `sh` and `ksh` will attempt to lock the `.sh_history` file and hang before giving you a prompt.

When a system is running and has an active NFS mount with a server in which files have been previously locked, both the client and server cache information about each other. Part of the information that is cached is what RPC port number to use to contact the `rpc.lockd` daemon on the server and client.

This RPC port information is cached in memory of the running `rpc.statd/rpc.lockd` process on both the server and client sides. The `rpc.statd` process keeps a file in the directory `/var/statmon/sm` for each system that it knows it should contact in the event that the system reboots (or `rpc.statd/rpc.lockd` restarts). During a normal reboot or crash, `rpc.statd` will contact all systems in `/var/statmon/sm` and inform them to flush their cache regarding this client.

When you reinstall a system, the `/var/statmon/sm` directory is wiped out. In this case, if the reinstalled system tries to recontact a server that has cached information, the server will try to communicate over an old RPC port. The communication will fail for `rpc.lockd` and any file locking done by an application over that NFS mount will hang.

There are a several ways to avoid and/or fix the problem if it happens:

- If you are using `bootsys` to install clients, use the `-s` option to allow the client to shutdown normally and inform servers that it is going down.
- If you experience a hang, you can reboot the client or kill/restart `rpc.lockd` and `rpc.statd` on the client. At the point of the hang, the `/var/statmon/sm` directory will contain the name of the server, thus rebooting or restarting the *daemon*s will tell the server to flush its cache. If more than one server is involved, you may end up doing this multiple times until all servers are notified.
- As part of the installation, create a file for each server in `/var/statmon/sm` that contains the server's name. This will cause the first *boot* to generate a crash recovery notification message to each server, causing it to purge the stale port information. Following is an example `post_config_cmd` that could be placed in your `/var/opt/ignite/config.local` file. Replace `sys*` with your NFS server names.

```
post_config_cmd += "
   mkdir -p /var/statmon/sm
   for server in sys1 sys2 sys3
   do
      echo $server > /var/statmon/sm/$server
      chmod 0200 /var/statmon/sm/$server
   done
"
```

The `bootsys` Command Seems to Work in Reverse

With `bootsys -w client`, the client does not wait for the server. With `bootsys client`, the client waits for the server.

This was probably due to running through the GUI once on the server prior to running `bootsys`. The server drops the instruction for the client to start installing, and the next time the client *boot*s it picks that instruction up and goes. Ignite-UX tells you that the installation will happen the next time `bootsys -w` is used, but does not say it happens automatically. Then, the next time you run `bootsys`, you did not use the GUI without the client being booted from the server.

Server Not Listed

The `search lan install` command does not list the server.

Check these items on the *Ignite-UX server* from which you are trying to *boot*:

- Messages from `instl_bootd` `/var/adm/syslog/syslog.log`. If you need to add more IP addresses to `/etc/opt/ignite/instl_boottab`, you will see messages in `syslog.log` such as the following:

  ```
  instl_bootd: Denying boot request for host: 080009F252B3 to
  avoid IP address collision. Try booting again in 214 seconds, or
  add more IP addresses to /etc/opt/ignite/instl_boottab.
  ```

- A message in `syslog.log` that indicates that you have no IP addresses in `/etc/opt/ignite/instl_boottab` is:

```
instl_bootd: No available IP address found in:
 /etc/opt/ignite/instl_boottab
```

- If the client is an older system that does not use the `BOOTP` protocol (like 720s, 710s, 735s, 750s), then also look in the log file `/var/adm/rbootd.log`, and check to make sure the `rbootd` *daemon* is running. The `rbootd` daemon always runs, whereas `instl_bootd` is started using `inetd` and only runs when needed.

  Also, for these older clients, there is an intentional delay built into the `rbootd` process when a client wants to do an installation *boot* (as opposed to a diskless boot). This prevents the server from showing up during the first search. Retrying the search two or three times may be necessary.

The `bootsys` Command Fails with Insufficient Space

The `bootsys` command fails due to insufficient space in the `/stand` volume.

The `bootsys` command must copy the two files, `/opt/ignite/boot/Rel_`*`release`*`/[W|V|I]INSTALL` and `/opt/ignite/boot/Rel_`*`release`*`/[W|V|I]INSTALLFS` from the server into the client's `/stand` directory. This error indicates that there is not enough space in `/stand`. To correct this situation, you may need to remove any backup *kernel*s. Also, check for kernels in the `/stand/build/` directory (like `vmunix_test`).

TUI Does Not Accept User Input

The text fields in the TUI do not accept keyboard input during client installations.

The text fields within the TUI do not recognize keyboard input, causing dialogs to reopen or loop. This occurs when the **Insert** key is active so you must ensure that the **Insert** key is deactivated by pressing it to enter data in the TUI.

Installing from Media

DCE/RPC Errors

DCE / RPC errors (RPC exceptions) occur during the configuration stage, and a failure message is printed at the end of the installation.

There is an apparent problem with certain SD operations (for example, `swacl`) when only loopback networking is enabled. This would occur if the "media only" installation option is selected. The work-around is to install using the "media with networking enabled" option and set up (perhaps temporary) networking parameters: hostname, IP address, netmask, routing, etc. SD operations will complete normally.

Installing from Golden Images

Cannot Find Specified Archive

```
Errors: gunzip: stdin: unexpected end of file

  pax_iux: The archive is empty.

ERROR: Cannot load OS archive (HP-UX Core Operating System
        Archives)
```

The NFS mount probably succeeded, but the file was not accessible from the client. Check these possibilities:

- File has a different name (check your *configuration file*s).
- File has the wrong permissions such that it is not readable. Check `/etc/exports` for HP-UX 11i v1 and 11i v2 systems, and `/etc/dfs/dfstab` for HP-UX 11i v3 systems.

Missing `.conf` Files

The `/etc/nsswitch.conf` and `/etc/resolv.conf` files from the archive do not end up on the installation client.

Ignite-UX changes some files during the configuration process, including `resolv.conf` and `nsswitch.conf`. The Ignite-UX `os_arch_post_l` and `os_arch_post_c` scripts place these files on the client after the install.

These scripts are delivered in `/opt/ignite/data/scripts/`. You will probably only need to modify `os_arch_post_l`. Search on `resolv.conf` and `nsswitch.conf` for directions on what to change. After the script has been changed, modify your *configuration file*, which describes the *archive* to point to the new script.

`pax_iux` Errors

Errors resulting from `pax_iux` similar to the following:

```
pax_iux: X: Cross-device link
pax_iux: X: File exists
```

Both of these errors may occur when installing a system from an *archive* that does not have the same *file system* partitioning as the system from which the archive was created.

The `Cross-device link` error is caused when two files exist as hard links in the archive, and when the two files would end up in separate file systems. For example, if you created an archive on a system that did not use LVM, the root file system is all one file system. If you have two files, `/usr/local/bin/f1` and `/opt/myprod/bin/f2` as hard links, this error occurs if you make an archive of this system and try to apply it to a system that uses LVM and has `/usr` and `/opt` as separate file systems.

The `File exists` error may occur when the *archive* has a symlink or regular file that is named the same as a directory or mount point that exists when the archive is installed. This may happen, for example, if the original system that the archive was made from has a symlink like `/opt/myprod -> /extra/space`; then, when you are installing a system from the archive, you decide to create a mounted *file system* as `/opt/myprod`. The `pax` command will fail to create the symbolic link because a directory exists in its place.

When the error happens, you will be asked if you want to push a shell (on the client's console). Answer **`yes`,** and from the shell enter **`exit 2`** to ignore the error, and it will continue. Once the system is up, you can more-easily determine what should be done with the paths it complained about.

To avoid the error, the system that the *archive* is created from should not contain hard links between directories that are likely to be created as separate *file system*s.

Common Network Booting Errors

Insufficient Response to PXE Boot Request

```
PXE-E16: Valid PXE offer not received
```

When the above error is seen during a PXE *boot*, one of the following may be true (this is not an exhaustive list) :

- The response from the *DHCP* server did not contain enough information to continue the PXE boot.
- No PXE boot response was received.

Additional errors may be printed after this PXE error to show what the problem is related to.

File Size Miscalculated

```
PXE-E05: Download buffer is smaller than requested file.
```

This problem is related to firmware in *EFI* on the HP Integrity rx1600/rx1620 systems. Upgrade the firmware on the system to a version that addresses the above message (system firmware 02.11 or later.)

When using a tftp server that does not support the `tsize` option, a file must be completely downloaded to determine its size. The file is then downloaded again into memory allocated based on the determined size. The EFI code calculates the wrong size, hence the allocated buffer is too small to hold the file.

Apart from updating the firmware, using a tftp server that supports the `tsize` option will also solve this problem. The `tsize` option allows a tftp client to ask the tftp server for the size of the file. This saves the client from having to download the file to determine its size.

The HP-UX 11i v2 operating system supports the `tsize` option from first release; HP-UX 11i v1 requires the patch PHNE_32825.

Using a tftp server that supports the `tsize`option makes network booting faster, since every file to be downloaded from the Ignite server via tftp does not have to be downloaded twice (once to determine its size, and again to download it into memory.)

File Size Miscalculated on HP Integrity Virtual Machines

```
AUTOBOOTING...Obtaining size of Rel_B.11.23/IINSTALL (63450640 bytes)
Downloading file Rel_B.11.23/IINSTALL /
TFTP session failed. (reason:Buffer Too Small)
Could not open Rel_B.11.23/IINSTALL
```

The error `TFTP session failed. (reason:Buffer Too Small)` is applicable only to HP Integrity Virtual Machines. The reason is the same as described under the heading: File Size Miscalculated. No version of HP Integrity VM contains fixed firmware at this time. To resolve this issue, install the patches listed in the File Size Miscalculated section so the `tftpd` *daemon* supports the `tsize` option.

B Configuring DHCP Services

HP-UX and Ignite-UX support retrieving network information using the Dynamic Host Configuration Protocol (*DHCP*). This appendix describes setting up DHCP.

Ignite-UX allows for setting up *DHCP* for use during system installation. This appendix is for those who want to use DHCP for ongoing IP address management, as well as for system installation.

All partitionable Itanium-based systems provide an alternative to using *DHCP* for network booting. This *EFI* functionality is also found in other, non-partitionable systems. With these profiles, you can supply all the networking information needed to contact an *Ignite-UX server* and perform an install or recovery. For more information see "Direct Boot Profiles for Itanium-Based Systems" (page 115).

Overview of DHCP Services

DHCP provides these features:

- Enables dynamic allocation of IP addresses and hostnames.
- Automatically supplies most of the networking defaults that are requested during a system installation or first-time *boot*.
- Provides for ongoing IP address maintenance using a concept of an "IP address lease." Having a lease on an IP address means that if the system "goes away" for a specified period of time without renewing the lease, then that IP address can be given to a different system that requests a new IP address lease.
- Assists in reestablishing valid network parameters when a system has been moved from one DHCP-managed network to another.

DHCP works best under these conditions and restrictions:

- When a range of currently unused IP addresses can be allocated for use during new system installation.
- When the IP address-to-hostname mapping can be made ahead of time (before the system to use it is installed), and this mapping can be configured in the name services database before installing a system.
- When the IP address and hostname that get assigned to a system are not important. A system will keep the same IP address and hostname for as long as it renews the lease. However, the original assignment is arbitrary.
- When you do not want to choose a hostname for the system and will accept the hostname already registered for the IP address supplied by DHCP. This ensures that the system will be recognized immediately by its hostname.
- When existing systems that did not use DHCP before are willing to accept an arbitrary hostname and IP address.

An alternative to using DHCP is to create `/etc/bootptab` entries for each specific client on the network. This allows for specific IP address mappings and greater control. For more detail, see "Using bootptab as an Alternative to DHCP " (page 257).

For additional information regarding the use of DHCP with Itanium-based systems, see "Configuring an Ignite Server to Boot Anonymous Itanium-Based Clients" (page 50) and see the *HP-UX IP Address and Client Management Administrator's Guide*.

DHCP Usage Examples

You can configure a *DHCP* server to respond only to specific clients during an installation using `instl_adm` to configure specific *`dhcp_class_id`*s. Your situation might fall into one of these categories:

- The network has a DHCP server and the clients being installed will be requesting IP addresses from it. See the section "Manage Clients That Will Use DHCP During and After Installation" (page 256).
- The network has a DHCP server, but you would like to manage a small group of temporary IP addresses for use in performing installations. When deployed, the clients will be reassigned new addresses. In this case, see "Manage Clients with Temporary IP Addresses During Installation" (page 256)

Ensure DHCP is configured and operating on your Ignite-UX server.

For more information, see *instl_adm*(1M) and *bootpd*(1M).

Manage Clients That Will Use DHCP During and After Installation

This example uses *DHCP* with your Ignite-UX server or boot helper to provide IP addresses during installation and regular use. This section assumes no networking information is being provided interactively or by the configuration used to install the system.

If you have an existing DHCP device group you wish to use for DHCP addresses, make the following changes to the *install file system*, substituting *`YourClassId`* with the class ID you wish to use.

```
dhcp_class_id="YourClassId"
is_net_info_temporary=false
```

The use of `dhcp_class_id` is optional. Use it if you want a lease on a DHCP-allocated IP address associated with a specific DHCP class id. If you do not wish to specify a DHCP class id, no changes are required to the install file system since the `is_net_info_temporary` keyword is set to false by default. If you previously customized the install file system, you should review and change the configuration as appropriate.

You should not use the same DHCP device group as the device group created for installing anonymous Itanium-based clients. Typically, the number of IP addresses available from that device group is small and the IP addresses will be exhausted quickly, leaving you unable to install or recover any systems over the network. If you don't have a suitable DHCP device group available, consider configuring a new DHCP device group in consultation with your network administrator.

After a *cold install*, you will be prompted for networking information. You must indicate the system will use DHCP. See *set_parms*(1M) for more information.

Manage Clients with Temporary IP Addresses During Installation

This example uses *DHCP* with your Ignite-UX server or boot helper to provide temporary IP addresses for installation and recovery.

The following assumes you have created a DHCP device group suitable for booting anonymous Itanium-based clients and that you have made changes to the device group allowing it to recognize the `IgniteDHCPDeviceGroup class id`. See "Isolating Ignite-UX From Noncontrollable DHCP Servers " (page 52) for information on how to change the device group. Make the following changes to the *install file system*.

```
dhcp_class_id="IgniteDHCPDeviceGroup"
is_net_info_temporary=true
```

When choosing to use the same DHCP device group as the one for booting anonymous Itanium-based clients, the system being installed or recovered will continue to use the IP address for a longer period. The number of IP addresses available for network booting in this device group should be sized to allow for the maximum number of systems you expect to be concurrently installing or recovering.

For information regarding the usage of `is_net_info_temporary`, see this important note.

For information regarding creating a DHCP device group for booting anonymous Itanium-based clients, see "Ignite-UX Server and Boot Helper Setup for DHCP" (page 51).

Using bootptab as an Alternative to DHCP

If you want to have more control over the allocation of IP addresses and their mappings to your clients, you can configure entries in `/etc/bootptab` for each client. Because the Bootstrap Protocol (BOOTP) is a subset of the *DHCP* protocol, the client's request for a DHCP server will be satisfied with the BOOTP response.

If you also specify a *boot* file (`bf`) of `/opt/ignite/boot/boot_lif` in the `bootptab` entries, you do not need any additional entries in `/etc/opt/ignite/inst_boottab`. In this case, you would boot the clients using `boot lan` instead of `boot lan install`. Only clients known in `/etc/bootptab` will be able to boot if you do not use `instl_boottab`.

A minimal example `/etc/bootptab` entry follows; you must use your own hostname, IP address, hardware address, and subnet mask. Other networking information may also be specified here or using `instl_adm`. Specify the Ignite-UX server's IP address with the `instl_adm -t` option.

```
sysname:\
  hn:\
  vm=rfc1048:\
  ht=ether:\
  ha=080009352575:\
  ip=10.1.51.82:\
  sm=255.255.255.0:\
  bf=/opt/ignite/boot/boot_lif
```

NOTE: Important information relevant to using `bootptab` to enable network installation of *client*s is found in "Configuring the Ignite-UX Server for Itanium-Based Clients" (page 41).

Background Information on DHCP Design

The *DHCP* protocol is implemented as extensions to the BOOTP protocol, and in fact the HP-UX DHCP server and BOOTP *daemon*s are the same: `bootpd`. This daemon reads two configuration files: `/etc/bootptab` and `/etc/dhcptab`.

The mapping of systems to IP addresses and lease time information is kept in the DHCP database file `/etc/dhcpdb`. Some amount of management of this database is provided by the `dhcptools` command.

On the client side, a command called `/usr/lbin/dhcpclient` is used to contact the server to get an IP address lease. This command has the ability to broadcast out onto the network prior to the network interface being enabled.

The `dhcpclient` also serves as a *daemon* process that sleeps until such time that it needs to renew the IP address lease. At that time, the client will recontact the server from which it got the original lease in order to extend it.

The `dhcpclient` command is not intended to be run by users directly; it is called by other tools during system *boot* and installation.

For more information regarding the networking parameters that DHCP can supply, see *auto_parms*(1M) and *dhcpdb2conf*(1M).

For more general information regarding DHCP, see *bootpd*(1M) and *dhcptools*(1M).

C LIF Volume Contents

The *logical interchange format* (LIF) volume contains portions of content needed for *boot* and installation, and is required on any medium used for Ignite-UX boot, installation and recovery, including network server, root disk (`bootsys`), tape, and CD/DVD.

The `make_medialif` command creates the *LIF volume*. For more information on creating a LIF volume, see *make_medialif*(1M).

A Description of the Files in the LIF Volume.

To list the contents of a LIF volume, use the `lifls -l` *`lif_volume`* command. Example output from `lifls -l` and descriptions of the files that can be in a LIF volume follow:

```
# lifls -l /var/tmp/lif
volume ISL10 data size 1953114 directory size 3 07/03/07 10:12:02
filename   type   start   size     implement  created
================================================================
ISL        -12800 16      242      0          07/03/07 10:12:02
AUTO       -12289 264     1        0          07/03/07 10:12:02
INDEX      BIN    272     1        0          07/03/07 10:12:02
CONFIG     BIN    280     84       0          07/03/07 10:12:02
HPUX       -12928 368     1024     0          07/03/07 10:12:02
FWWKAR6    BIN    1392    1        0          07/03/07 10:12:02
FWWKAR7    BIN    1400    1        0          07/03/07 10:12:02
FWWKAR8    BIN    1408    1        0          07/03/07 10:12:02
WINSTALL   -12290 1416    130092   0          07/03/07 10:12:04
WINSTALLFS -12290 131512  128000   0          07/03/07 10:12:04
IINSTALL   -12290 259512  544598   0          07/03/07 10:12:09
IINSTALLFS -12290 804112  239616   0          07/03/07 10:12:09
RECCMDS    BIN    1043728 1158     0          07/03/07 10:12:09
RECCMDSIA  BIN    1044888 2210     0          07/03/07 10:12:09
INSTCMDS   BIN    1047104 55663    0          07/03/07 10:12:09
SYSCMDS    BIN    1102768 127867   0          07/03/07 10:12:09
INSTCMDSIA BIN    1230640 116929   0          07/03/07 10:12:09
SYSCMDSIA  BIN    1347576 214290   0          07/03/07 10:12:10
SCRIPTS    BIN    1561872 45       0          07/03/07 10:12:10
PAD        BIN    1561920 256      0          07/03/07 10:12:10
#
```

The following list describes the files that can be in a LIF volume and, when applicable, identifies the `make_medialif` options to include them.

- **ISL** — Initial system loader. The ISL implements the operating system independent portion of the bootstrap process on PA-RISC systems. It is loaded and executed after self-test and initialization have completed successfully. If it is run interactively, it issues a prompt and waits for user interaction. Otherwise it looks for the *`AUTO`* file. It is extracted by `make_medialif` from the default *boot* file: `/opt/ignite/boot/boot_lif` See *isl*(1M) for more information.
- **AUTO** — Autoexecute file. A file that defines default boot behavior. For PA-RISC systems the *`AUTO`* file is in the LIF volume. For Itanium-based systems, the `AUTO` file is located in the `/opt/ignite/boot` directory. See *auto_adm*(1M) for more information.
- **INDEX** — Default `INDEX` file. It groups references to *configuration file*s in clauses in order to define installation behavior.
- **CONFIG** — Typically contains all the software configuration information and the default *file system* layout information. It includes default configuration information for the operating system release, user-defined configuration information, and information regarding *archive*s and *depot*s. To create a *`CONFIG`* file, you should begin with the default *configuration file* for the release. For example, for HP–UX 11i v3 use `/opt/ignite/data/Rel_B.11.31/`

`config`. Additional configuration files can be appended using the `-f` option of the `make_medialif` command.

Information in this file allows complete access to all the *archive*s and depots on the media. The *`CONFIG`* file is referenced in the `INDEX` file.

For more information, see "Using Configuration Files" (page 177).

- **HPUX** — HP-UX bootstrap loader. It loads the *kernel* and starts HP-UX. For Ignite-UX, the HP-UX bootstrap loader loads the *install kernel* (*[W|V|I]INSTALL*) and *install file system* (*[W|V|I]INSTALLFS*). The bootstrap loader may also load other LIF content it needs to operate. See *hpux*(1M) for more information.
- **FWWKAR*** — A LIF might include files needed by system firmware and bootloaders such as an `FWWKAR` file.
- **[W|V|I]INSTALL** — The HP-UX install kernel. The kernel is statically linked and includes a wide variety of drivers and modules to support different computer systems.
  - To support PA-RISC 32-bit systems, use the `-o 32 make_medialif` option to include the *`INSTALL`* kernel.
  - To support V-class PA-RISC systems, use the `-o 64v make_medialif` option to include the *`VINSTALL`* kernel.
  - To support wide 64-bit PA-RISC systems, use the `-o 64w make_medialif` option to include the *`WINSTALL`* kernel.
  - To support Itanium-based systems, use the `-o IA make_medialif` option to include the *`IINSTALL`* kernel.
  - Use the `-a make_medialif` option to include all *[W|V|I]INSTALL* kernel and *[W|V|I]INSTALLFS file system* files in the `/opt/ignite/boot/`*`release`* directory.

  NOTE: By default, the *install kernel* and *install file system* selected when a LIF volume is built will match the HP-UX release running on the system where `make_medialif` is run. Optionally, you may select *boot content* for a different HP-UX release via options. See *make_medialif*(1M) for more information.

  For more information, see "Classes of Configuration Files" (page 177).
- **[W|V|I]INSTALLFS** — The Ignite-UX HP-UX *install file system* image. This *file system* is loaded into memory and serves as the initial root file system for the installation or recovery process. Note that this file includes config content. See *instl_adm*(1M) for more information.
  - To support PA-RISC 32-bit systems, use the `-o 32 make_medialif` option to include the *`INSTALLFS`* file system.
  - To support V-class PA-RISC systems, use the `-o 64v make_medialif` option to include the *`VINSTALLFS`* file system.
  - To support wide 64-bit PA-RISC systems, use the `-o 64w make_medialif` option to include the *`WINSTALLFS`* file system.
  - To support Itanium-based systems, use the `-o IA make_medialif` option to include the *`IINSTALLFS`* file system.
  - Use the `-a make_medialif` option to include all *[W|V|I]INSTALL* kernel and *[W|V|I]INSTALLFS* file system files in the `/opt/ignite/boot/`*`release`* directory.

  NOTE: By default, the *install kernel* and *install file system* selected when a LIF volume is built will match the HP-UX release running on the system where `make_medialif` is run. Optionally, you may select *boot content* for a different HP-UX release via options. See *make_medialif*(1M) for more information.

  For more information, see "Classes of Configuration Files" (page 177).
- **INSTCMDS** and **INSTCMDSIA** — A compressed `tar` *archive* of commands required for specifying the system configuration to install on PA-RISC and Itanium-based systems, respectively. It includes commands needed to set up disk volumes and *file system*s.

- **SYSCMDS** and **SYSCMDSIA** — A compressed `tar` *archive* of commands required for installation of PA-RISC and Itanium-based systems, respectively. It includes commands needed to complete installation and recovery.
- **RECCMDS** and **RECCMDSIA** — A compressed `tar` *archive* of commands required for *expert recovery* on PA-RISC and Itanium-based systems, respectively.
- **SCRIPTS** — A *configuration clause* of commands containing scripts that can be selected in the *`itool`* under the **Advanced** tab.
- **PAD** — A file used to reserve space for future content, control the size and position of content, or both.
- **VERSION** — A file containing the Ignite-UX version used to produce the stored content. This is to make sure the Ignite version used to create the content is the same version of Ignite you are currently running. Mismatched versions can result in incorrect operation.

IMPORTANT: The first 8 KB of the *install file system [W|V|I]INSTALLFS* can contain Ignite-UX text. If it does, this information is used first and can override configuration statements in the LIF `INDEX` and *`CONFIG`* files. In particular, the information in the first 8 KB can cause the client to *boot* from media and use an Ignite-UX server to complete installation, and controls if the installation or recovery is interactive or not.

For more information, see "Using Configuration Files" (page 177).

D Using Integrated Lights Out Virtual Media with Ignite-UX

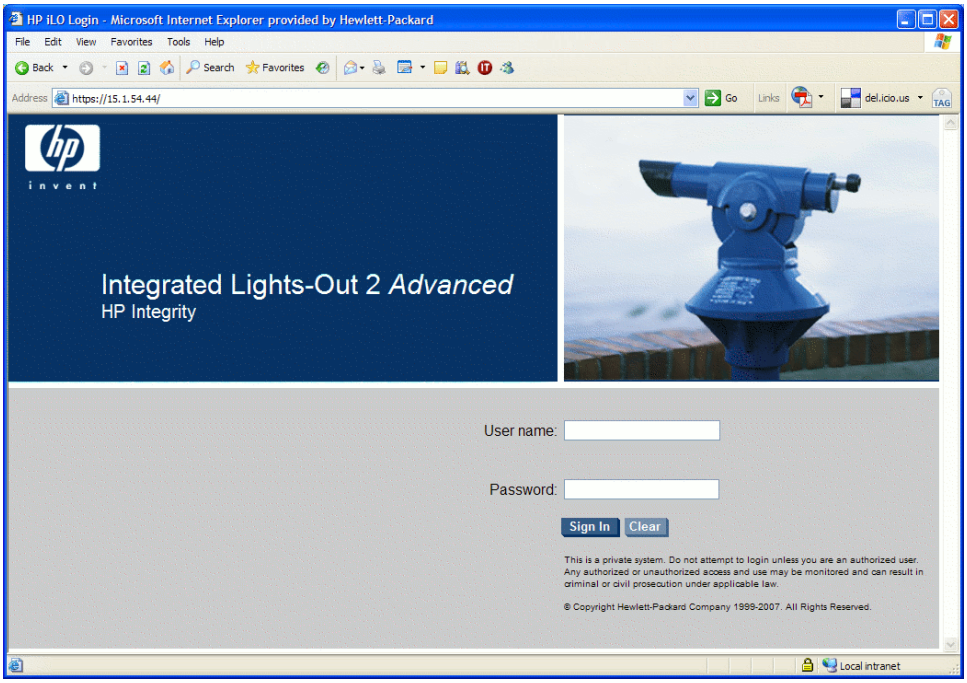
The virtual media (vMedia) functionality is part of the iLO 2 Advanced Pack feature set and is enabled by purchasing the optional iLO 2 Advanced Pack license and granting the vMedia right. You can use iLO vMedia to *boot* and install HP-UX onto an Integrity system from a remote system's DVD device or ISO DVD image. The remote system may be a PC.

This section describes how to use vMedia from a remote PC to *boot* an Integrity system using a DVD, and then install it using an *Ignite-UX server* or *installation media*.

It is assumed the Integrity system is equipped with iLO, and that a valid vMedia license has been obtained and installed. The vMedia feature of iLO requires a Java™ Plug-in. See the *HP Integrity iLO 2 Operations Guide*, available at http://www.hp.com/go/bizsupport for detailed information on iLO vMedia.

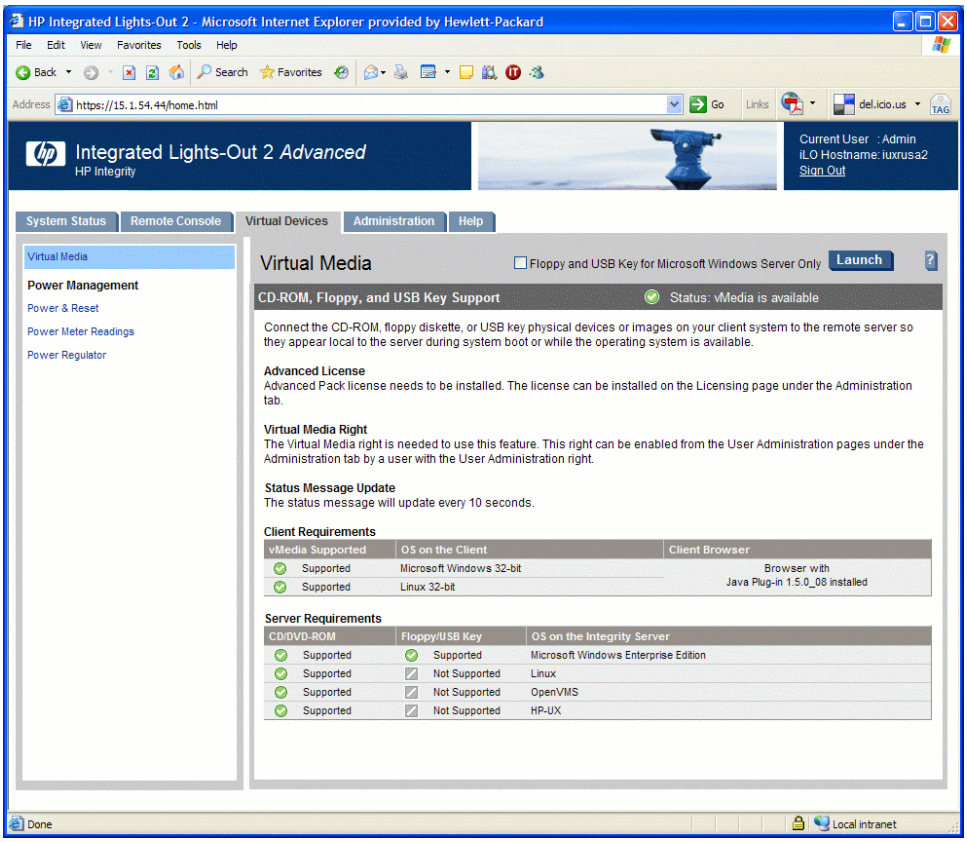
If you use an Ignite-UX server to do the installation, the version of Ignite-UX on the boot media must match the version on the Ignite-UX server.

1. On the remote PC, open a browser window. Enter the IP address or the *DNS* name of the Integrity system's iLO management processor (MP) as the browser Address. Log in to iLO by entering your User Name and Password, and then selecting **Sign In**. (Initially, iLO is configured with the default login user/password pairs of Admin/Admin and Oper/Oper. Password protected user accounts can be configured and assigned levels of access.)



2. Select the **Virtual Devices** tab. Select **Virtual Media** from the left navigation bar. Make sure you have a green Status light reading, “vMedia is available.”

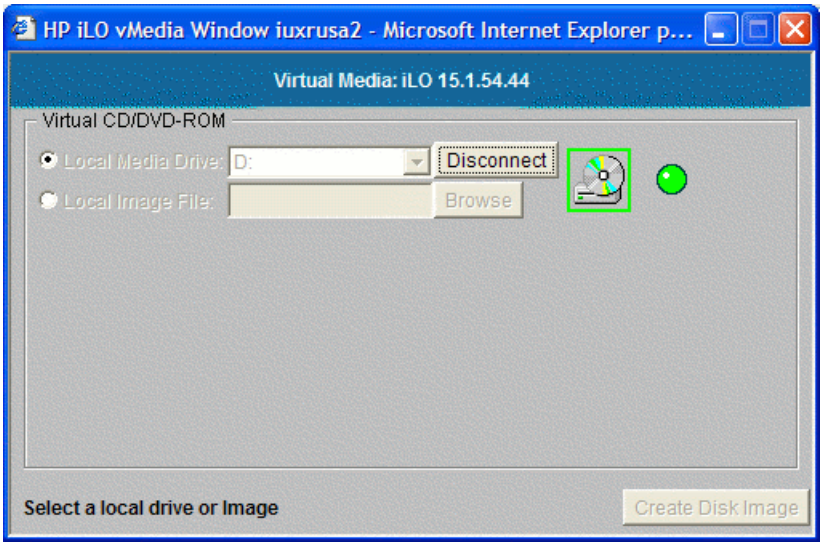
NOTE: Depending on your server, this screen might look slightly different.



3. Select **Launch** to load the vMedia applet. The vMedia feature of iLO requires a Java Plug-in. See the *HP Integrity iLO 2 Operations Guide*, available at http://www.hp.com/bizsupport for detailed information on iLO vMedia.

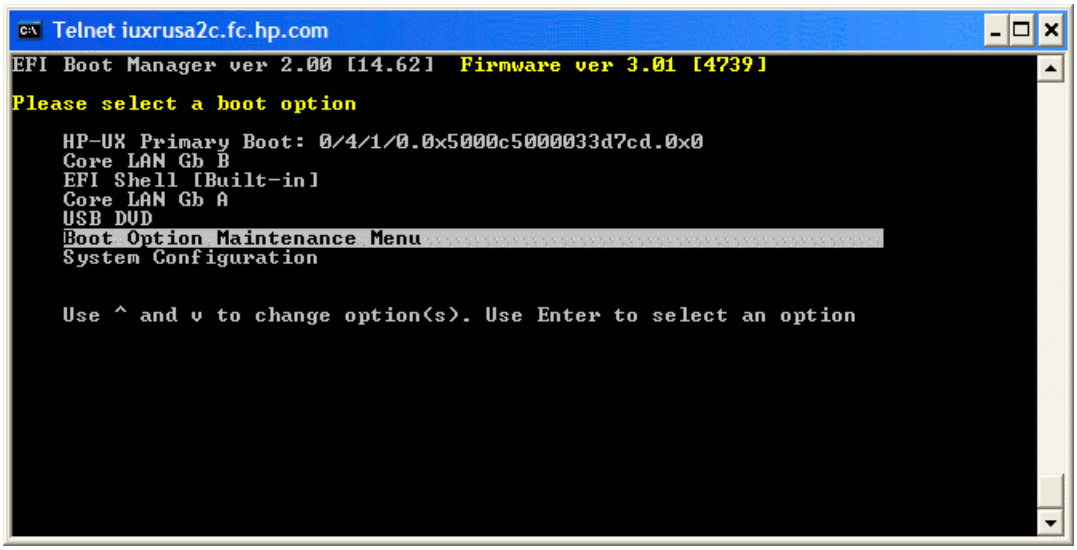
NOTE: Only one user and one device can be connected at a time.

4. Select **Local Media Drive** and then select the correct DVD drive from the pull-down. Click **Connect**. You should get a green light beside the virtual drive, as shown below.

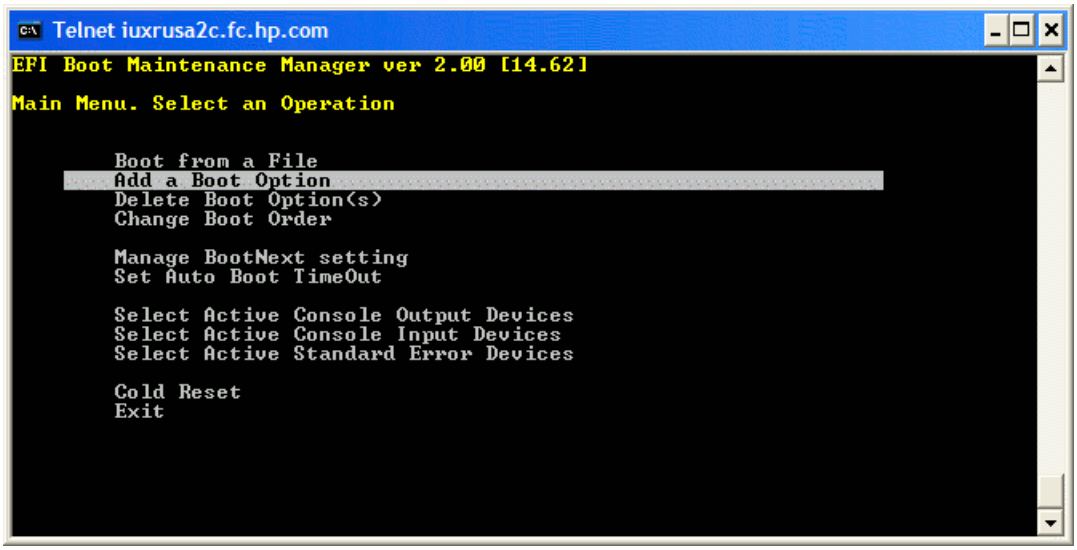


5. Reboot the Integrity system from its console, interrupt the boot process, and then add a boot option for vMedia. The console *EFI* menu sequence for adding a vMedia boot option follows.

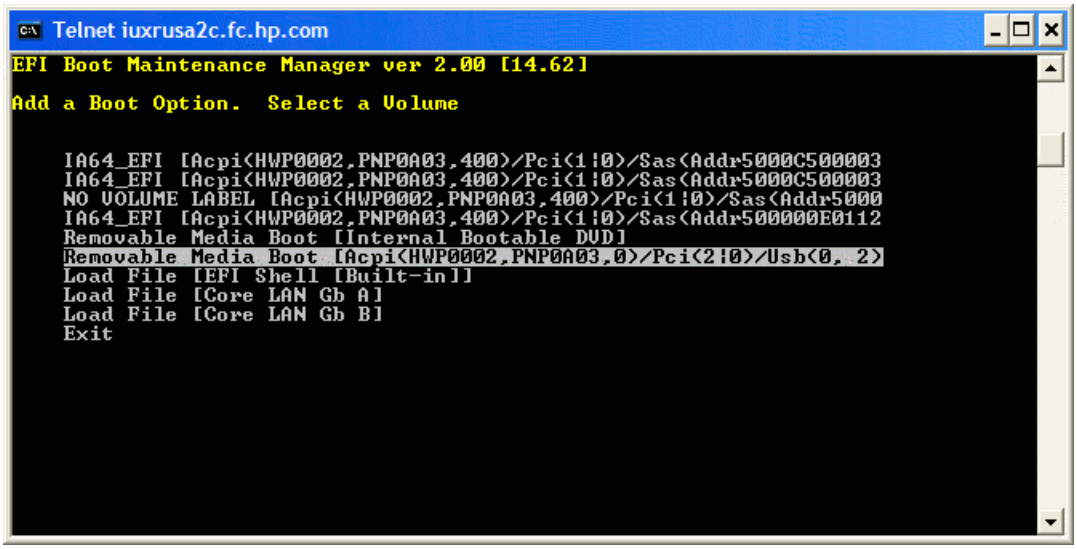
   Select **Boot Option Maintenance Menu** from the EFI Boot Manager.



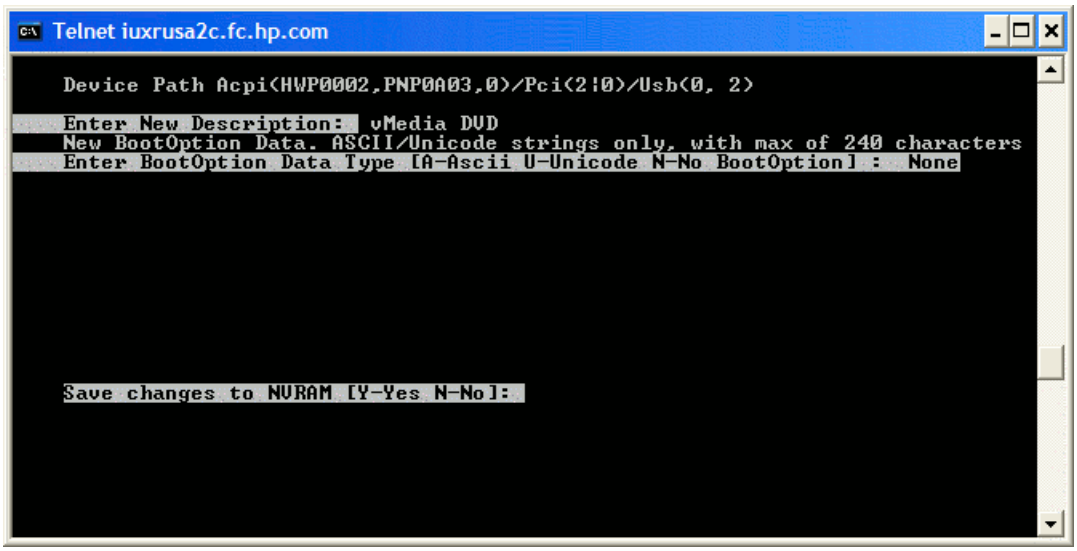
   Select **Add a Boot Option**.



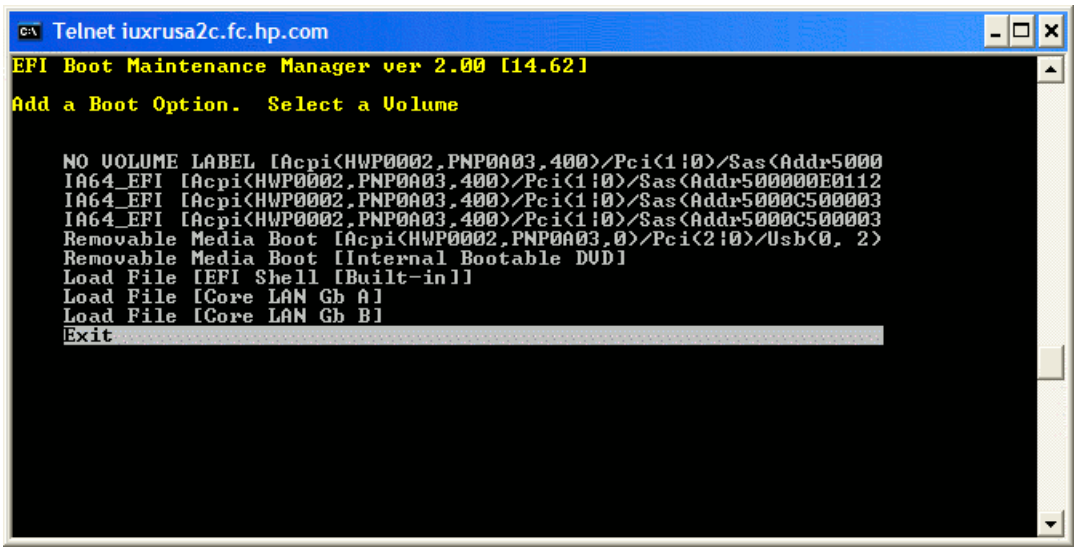
Select the PC DVD vMedia device from the list of available volumes. The vMedia device will appear to be local to the Integrity system.



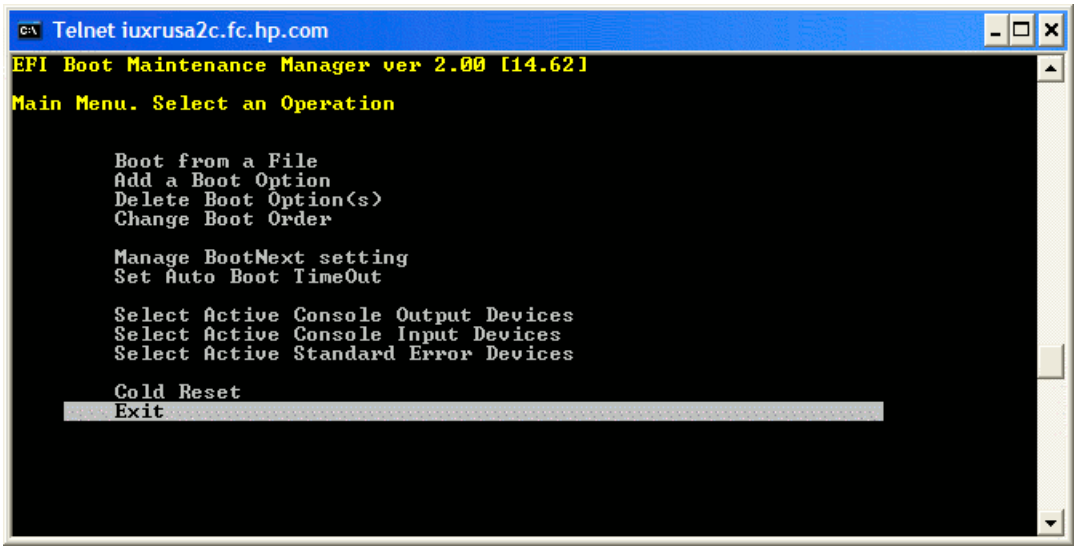
Enter any description of the boot option you prefer. Then, enter **N** (No BootOption) as the BootOption Data Type. Save the new boot option by entering **Y** at the “Save changes to NVRAM” prompt.



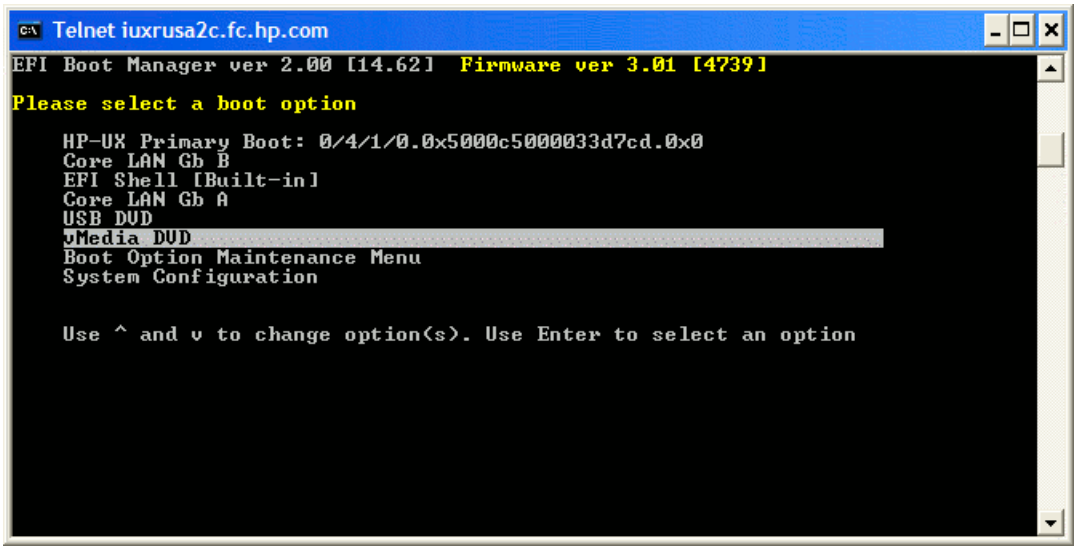
Return to the main menu by selecting **Exit**.



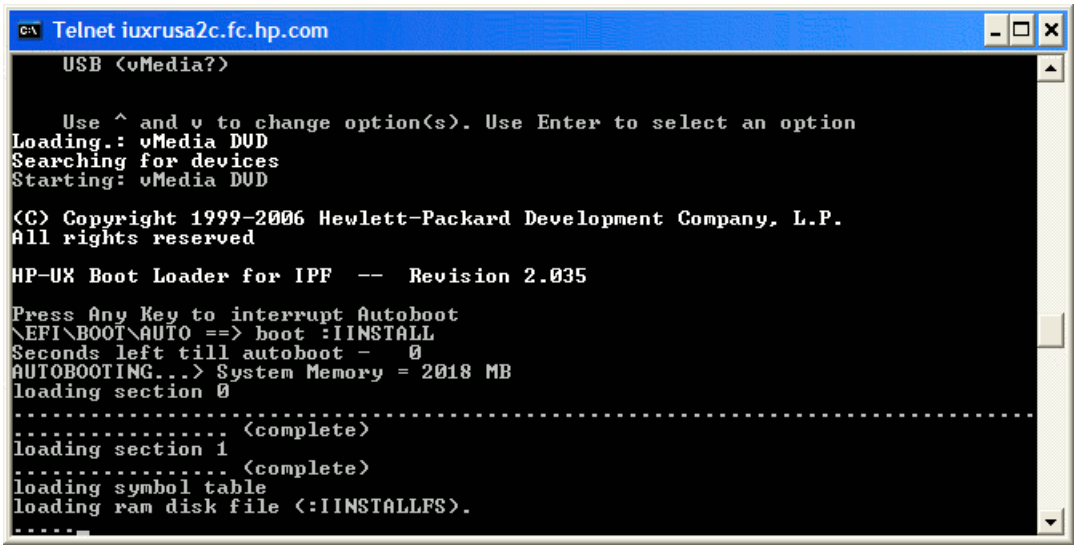
Return to the boot option menu by selecting **Exit** from the main menu.



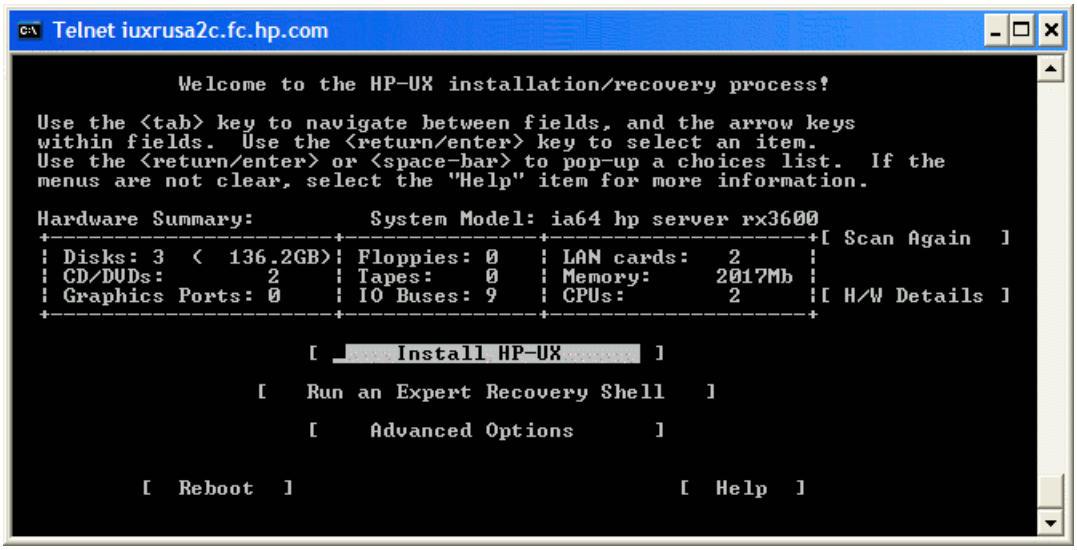
6. Make sure there is bootable media in the vMedia PC DVD drive and then select the boot option you just created. If you are going to use an Ignite-UX server to do the installation, the version of Ignite-UX on the boot media must match the version on the Ignite-UX server.



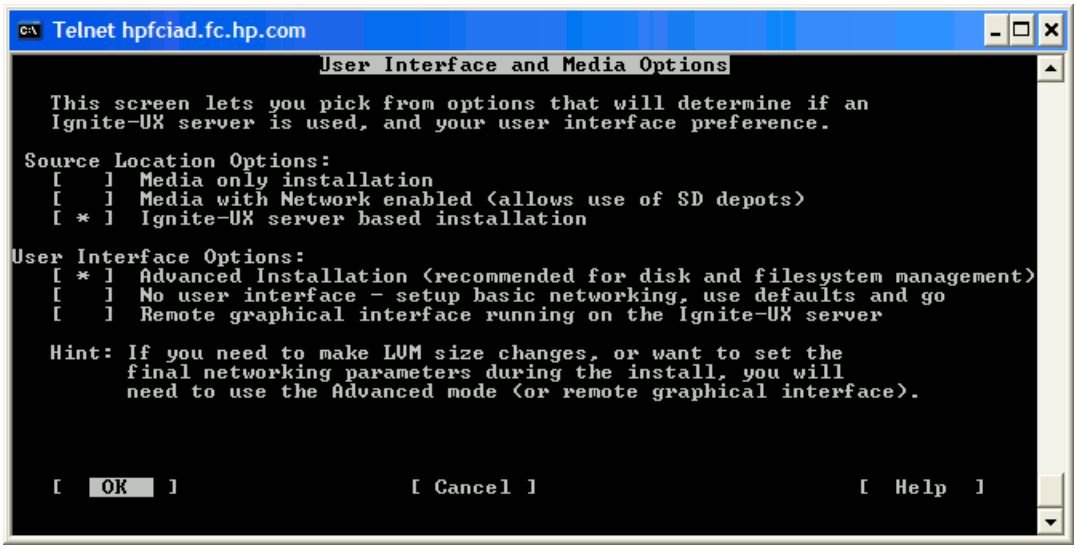
The system then boots from the vMedia DVD as shown below. Depending on network performance, using vMedia can take longer than using physically local DVD media .



7. The client system then boots to Ignite. Select **Install HP-UX**.



8. Now that the Integrity system has been booted with HP-UX, it is time to install the HP-UX system software. When possible, switch to install from an Ignite-UX server since vMedia installations can take significantly longer than direct media or network-based installations. To install from a server, select **Ignite-UX server based installation** and **Advanced Installation** from the User Interface and Media Options screen.



Using vMedia with DVD installation media and ISO images

If you would like to continue using vMedia for the installation, select **Media only installation** from the User Interface and Media Options screen. For installations that require multiple DVDs, simply eject the finished DVD from the remote system and replace it with the next DVD. If you are using an ISO image with vMedia, select a new image file using **Connect**. You should only select **Disconnect** from the vMedia applet when you are completely finished using vMedia since it frees vMedia for use by others.

9. Select **OK** and continue the configuration. When the configuration is complete, select **Go!**.
10. When you are finished using the iLO vMedia device, select **Disconnect** and close the applet. This is important, as it allows others to use vMedia.

E Expert Recovery

With *expert recovery*, you can repair aspects of the operating system on a damaged root disk using an Ignite-UX server or HP-UX OE media to boot the system. There are minor differences in message content between using a server or media, but the process is functionally identical. When using media, enabling and using networking is strictly optional.

Expert recovery can be useful if your system should become compromised or corrupt and will not boot to a login prompt, or if the system boots but critical operating system files are corrupted. At that point, it might be useful to restore system elements made available by booting from an Ignite-UX server or OE media.

While it might be possible to repair LVM and VxVM problems, that would require in-depth knowledge of those subsystems and is not documented here.

Expert recovery is not useful in recovering from hardware failures.

Expert Recovery Preparation

The more you know about the system disk and its layout before you encounter major damage or corruption, the easier it will be for you to recover.

Much of this information, including *file system* types, can be obtained by accessing your online system *manifest*, either using Ignite-UX or by reading the hardcopy that came with your system.

Before you attempt to recover an HP-UX system, you should gather the following information:

- The version of HP-UX on the system you are attempting to recover.

IMPORTANT: The HP-UX version of the system to recover must match the HP-UX version on the boot server or OE media. For example, use HP-UX 11i v1 (B.11.11) OE media to recover an HP-UX 11i v1 (B.11.11) system.

- The hardware path of the `root` file system on the disk (that is, what file system you will be checking/repairing using `fsck`.)
- Whether you have an LVM, VxVM, or whole-disk system.

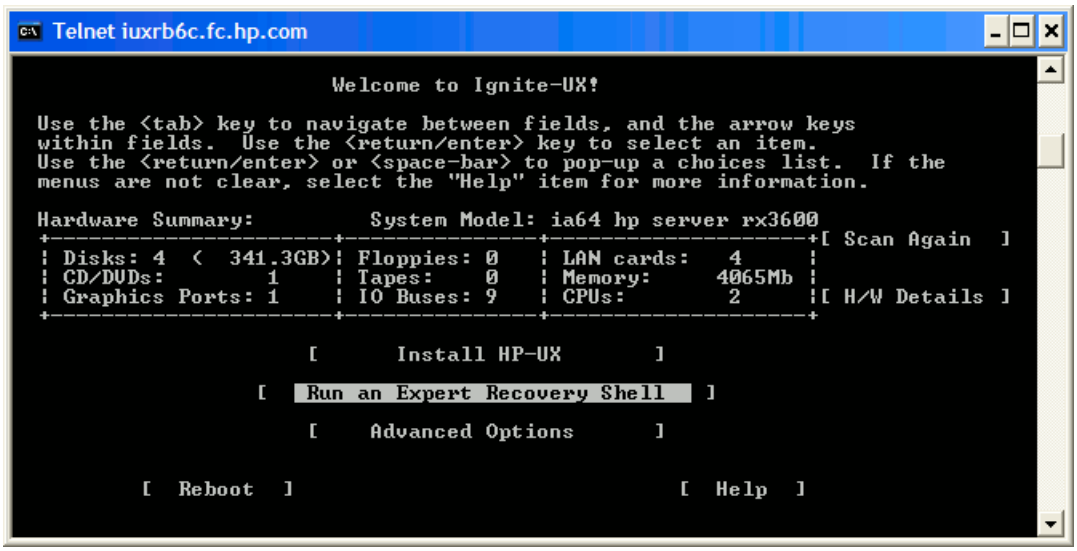
The Expert Recovery Procedure

The following is an example procedure for using expert recovery. Depending on your version of HP-UX, your disk layout, and whether the client is Integrity or PA-RISC, the details of these screens might vary.

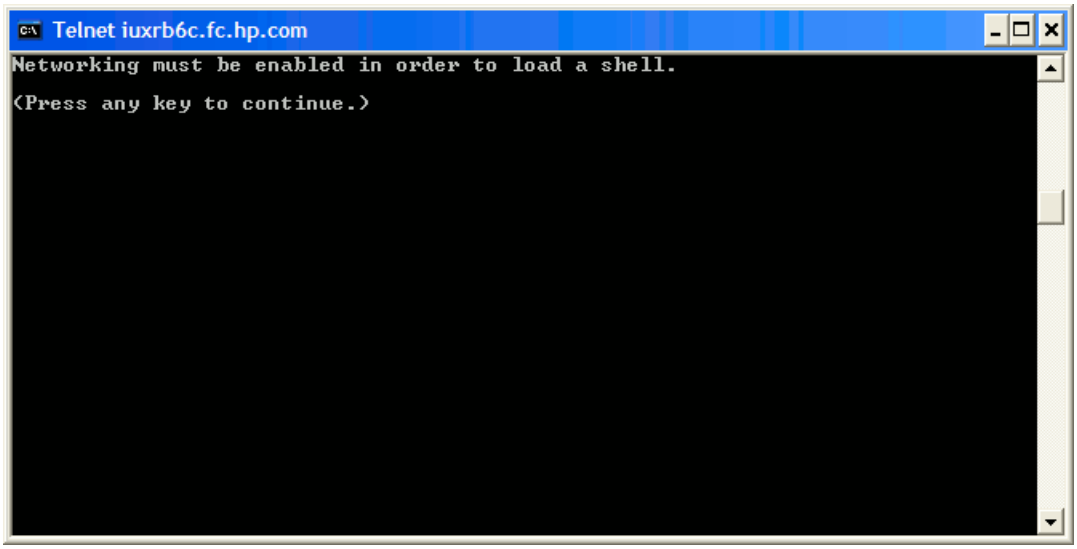
IMPORTANT: This procedure requires the commands `fsck` and `mount` running successfully on the system disk.

1. Boot your system from the Ignite-UX server or the OE media to begin the recovery process. For specific information regarding how to boot your system from an alternate boot source, see the *HP-UX System Administrator's Guide* for HP-UX 11i v3, or the *Managing Systems and Workgroups: A Guide for HP-UX System Administrators* manual.

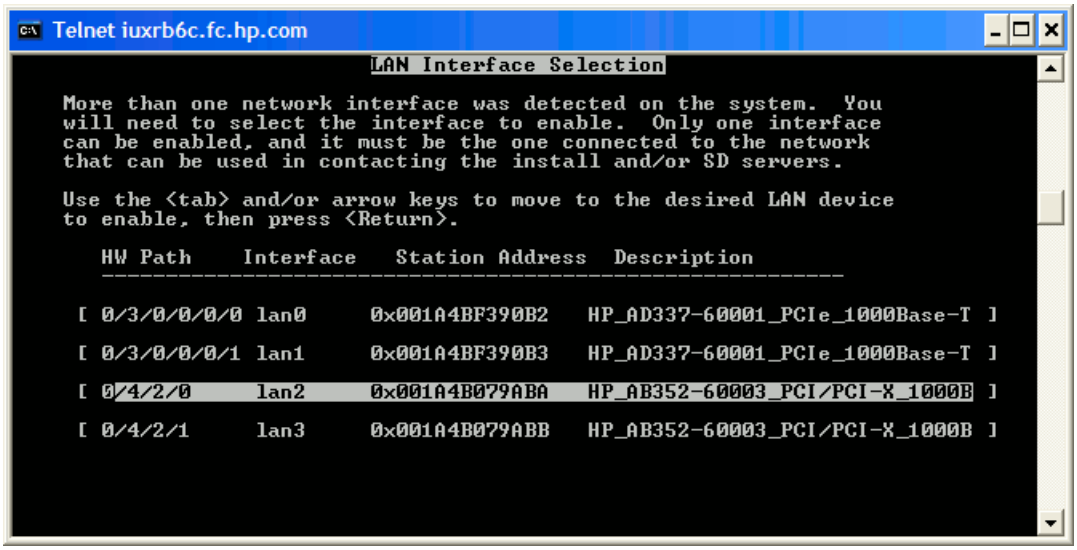
   The following menu is displayed once the booting process is complete. Select **Run an Expert Recovery Shell**.



2. If you booted from a server, you will see the following screen. Press any key to continue.

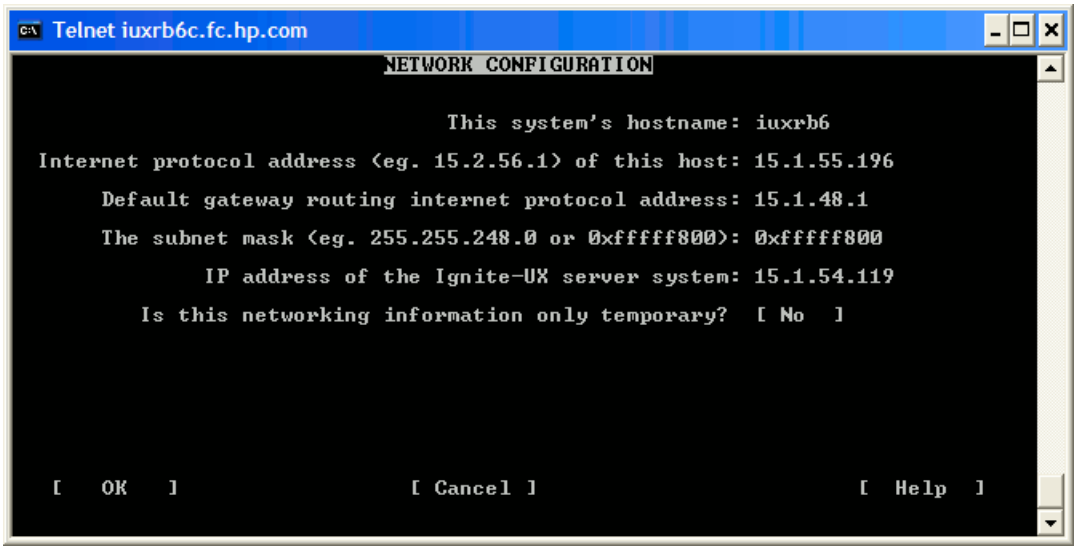


   Expert recovery will detect the interface used to boot the client. That interface will be highlighted in the following screen.

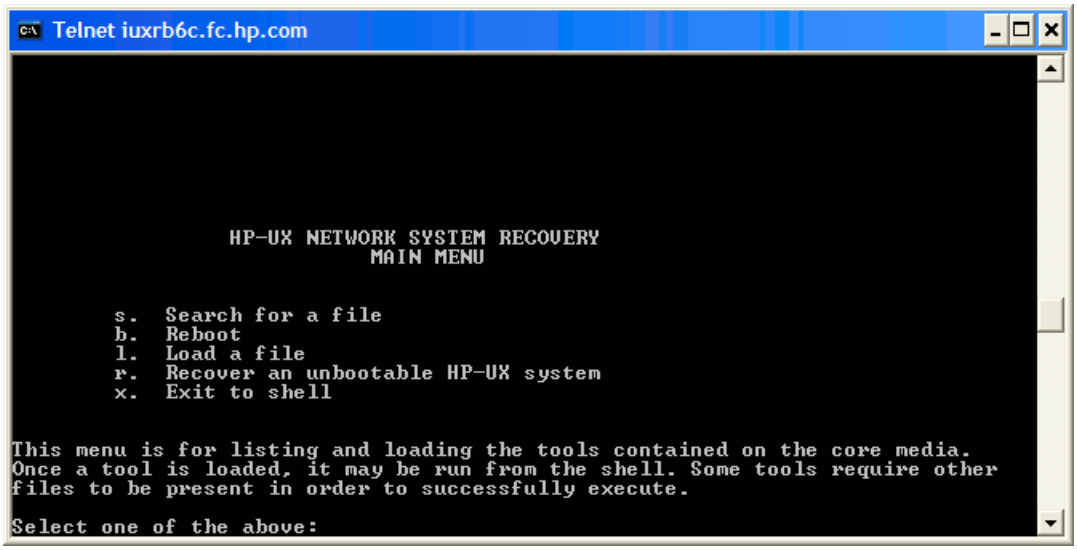


Press **Enter**.

3. As usual, Ignite-UX will attempt to obtain networking information via DHCP. If that is set up correctly, the following screen will be filled-in; otherwise, you must enter the correct networking information. Confirm the networking information by selecting `[ OK ]`.

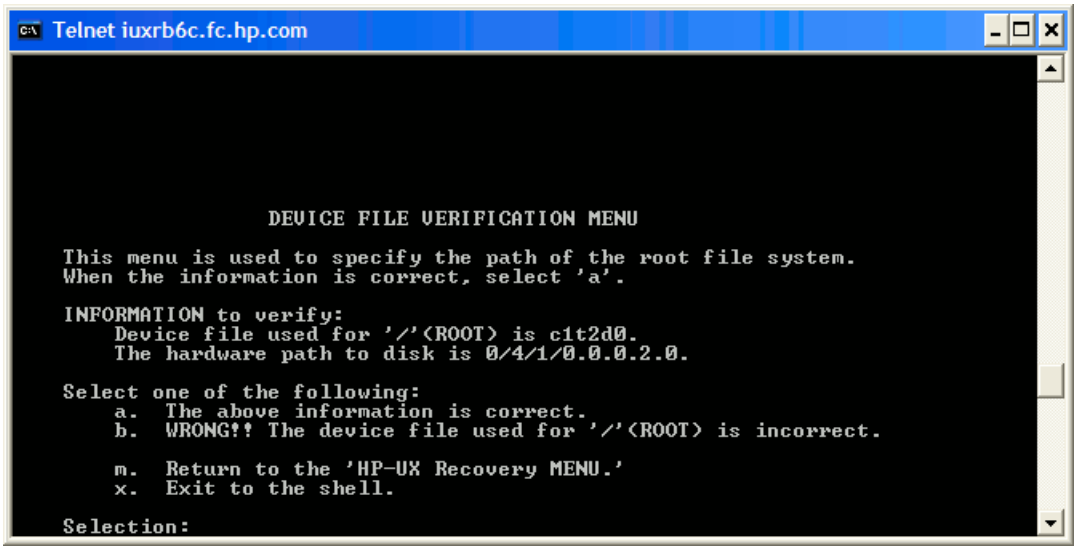


4. Expert recovery will load commands as needed. Upon completion, the HP-UX NETWORK SYSTEM RECOVERY menu is displayed:

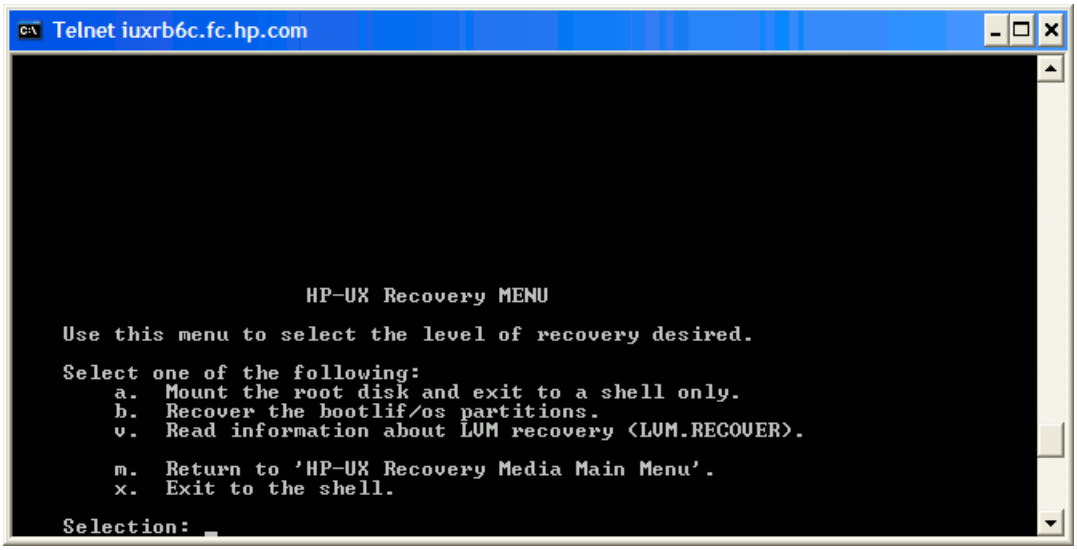


Select **r** and then **Enter** to continue with the recovery process.

5. Verify the root disk hardware path at the DEVICE FILE VERIFICATION MENU. Generally speaking, the information displayed on the menu is correct and **a** followed by **Enter** is the correct selection.

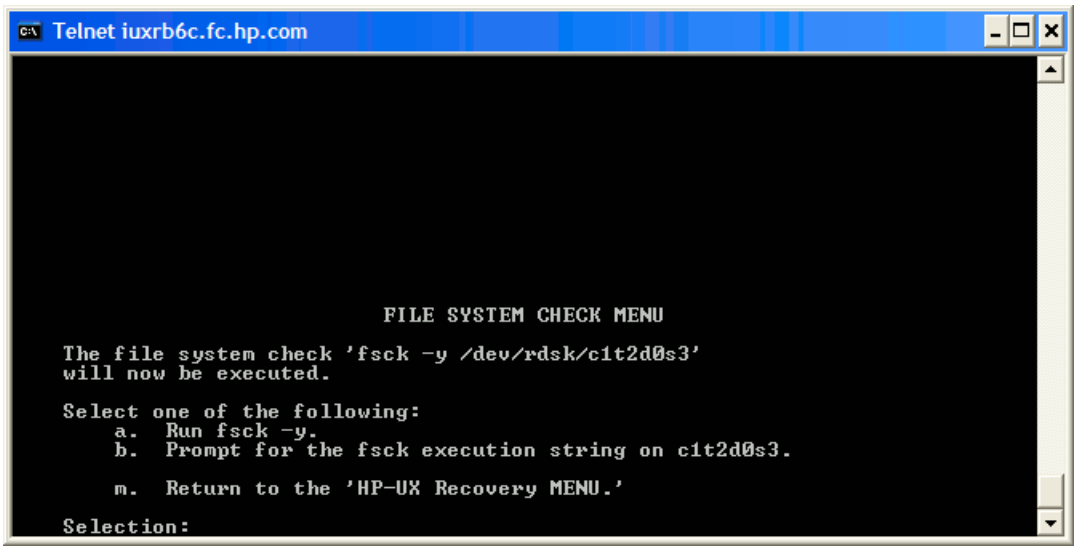


6. Select **a** and then **Enter** from the HP-UX Recovery MENU to mount the root disk and push a shell. From the shell, you'll be able to either restore an accidentally removed `/dev/console` device file or to reset a damaged `/etc/passwd` file back to its *newconfig* state.

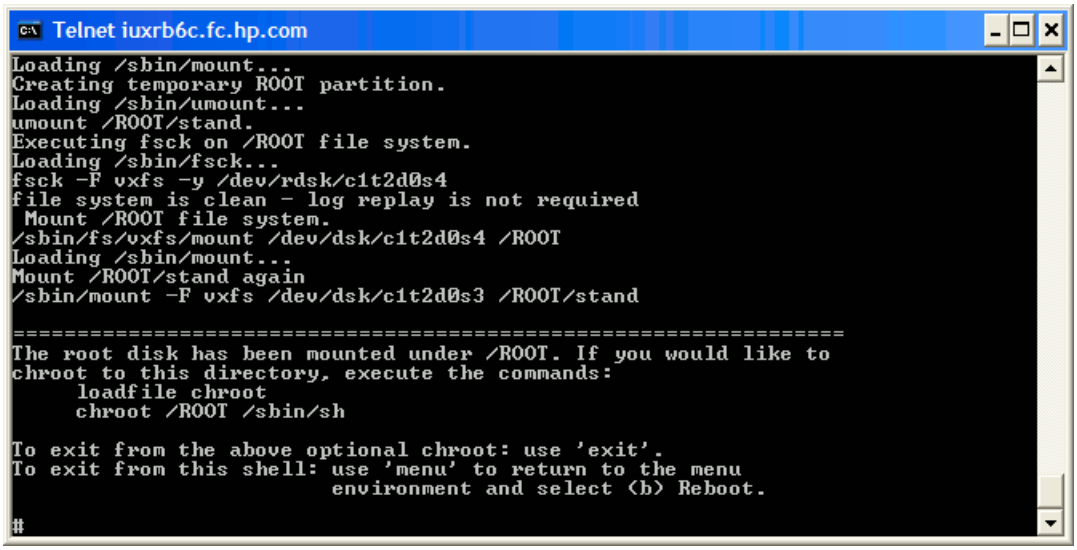


From this point the specifics of your system might cause your screens to appear differently than shown here. Select the first option on all screens to continue through the process.

7. Proceed through the next set of screens to get to the shell prompt. The following screens are for an Integrity LVM disk layout. Your screens might differ depending on your system – especially device files will be different. (The following screen will not appear if you have a VxVM root disk layout.)



Select **a Enter** to run `fsck` and proceed on.



If `fsck` ran successfully, you can now restore an accidentally removed `/dev/console` device file or reset a damaged `/etc/passwd` file back to its *newconfig* state.

8. To restore an accidentally removed `/dev/console` device file, enter the following command:

   `# mknod /ROOT/dev/console c 0 0`

   To reset a damaged `/etc/passwd` file to its newconfig state, enter the following command:

   `# cp /etc/passwd /ROOT/etc/passwd`

9. If you require additional content, it can be loaded using the `loadfile` command. For example, to get the `/etc/magic` file:

   `# loadfile /etc/magic`

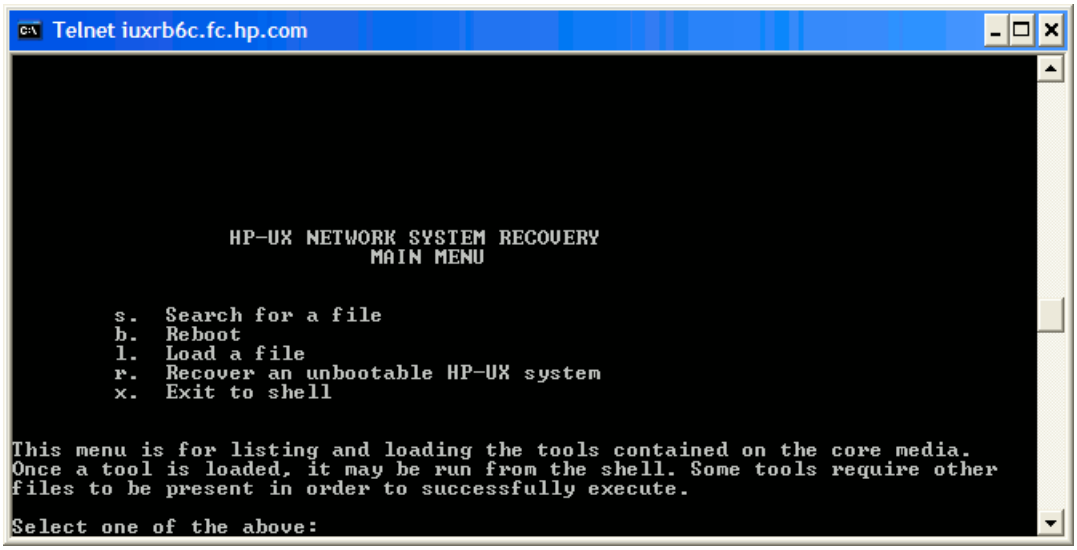
   The `loadfile` command loads content from the `SYSCMDS` archive by default. If the content is only available in the `INSTCMDS` archive, then use the `-l` option. For example:

   `# loadfile -l INSTCMDS /sbin/vx_emerg_start`

10. When the recovery tasks are complete, exit expert recovery by first entering the `menu` command at the shell prompt.

    `# menu`

    You are then returned to the HP-UX NETWORK SYSTEM RECOVERY screen.



Select **b Enter** to reboot the system.

F Terminal Keyboard Shortcuts

The Ignite-UX client-side installation tools support HP terminals, VT100, and Wyse 60 to navigate within the Ignite-UX GUI using a keyboard.

The following sections describe the various keyboard shortcuts that can be used in conjunction with the support terminals.

Basic Keyboard Shortcuts

These basic keyboard shortcuts can be used on both the vt100 and wyse60 terminals.

Table F-1 Basic Keyboard Shortcuts

Action	Keys
Move among window elements.	**Tab**
Move within or scroll a window element.	Arrow keys
Move to the tabs appearing across the top of the window.	**Tab**
Leave the tabs.	**Tab**
Select (highlight) a list item.	Space bar
Deselect a list item.	Space bar
Click a button, radio button or checkbox.	Space bar
Scroll up or down one page.	**Page Up** and **Page Down** keys (See "Advanced Keyboard Navigation" (page 278) if these keys are not available.)
Refresh the window and contents.	**Ctrl**-**L** or **Shift**-**F1**

Advanced Keyboard Navigation

HP Terminals

HP-family terminals display function key labels, also called softkey labels, at the bottom of the window. These labels vary depending on the type of window being displayed, but the functions of the **F1**-**F4** keys are consistent. These four keys function as follows:

Table F-2 HP Terminal Function Keys

Function	Key
Context-sensitive Help	**F1**
ALT (a modifier key)	**F2**
Select/Deselect	**F3**
Menu bar on/off	**F4**

Some of the advanced actions in the following table require a key-combination by pressing the **Alt** key (**F2**), releasing it, and then pressing another key. For example, selecting all items in a list requires first pressing **F2**, releasing it, and then pressing **/**.

Table F-3 Advanced Actions

Action	Key Combination
Scroll a list left one page.	**Prev**
Scroll a list right one page.	**Next**
Select (highlight) all items in a list.	**F2**, then **/** on the main keyboard
Deselect all items in a list.	**F2**, then **** on the main keyboard
Open a menu on the menu bar by using the mnemonic (underlined) letter	**F2**, then the mnemonic (underlined) letter
Press a button if the cursor is not resting on it.	**F2**, then the mnemonic (underlined) letter

TIP: To quickly save changes to a *file system* on the File System tab when using an HP terminal, press **F2** followed by the letter **m**. You save time and effort using this method since you do not have to navigate to the **Modify** button then press it.

vt100 Terminals

Typically the vt100 family keyboards use the same row of keys immediately above the numeric key pad as function keys. These keys may be labeled **[PF1]** -**[PF4]**, but in some cases may be labeled *****, **/**, **+**, and **-**. On some HP workstation (and HP 700 series terminals) keyboards, these keys may be four unlabeled keys above the keypad. If this is the case, **[PF1]** is the left unlabeled most key and **[PF4]** the right most key.

If you are unsure where the **[PF1]** - **[PF4]** keys are located on your keyboard, HP recommends the following:

- Consult the hardware documentation for your terminal to determine the keys that correspond to the **[PF1]** - **[PF4]** keys.
- If you are using X11, you must determine how your X11 key bindings are set. To view your current X11 key bindings, run `xmodmap` with no options.

  To set the keypad keys, *****, **/**, **+**, and **-** to be **[PF1]** - **[PF4]**, you would use the following commands:

```
xmodmap -e "keysym KP_Multiply = KP_F1"
xmodmap -e "keysym KP_Divide = KP_F2"
```

```
xmodmap -e "keysym KP_Add = KP_F3"
xmodmap -e "keysym KP_Subtract = KP_F4"
```

NOTE: Modifying the key bindings in X Window in this manner affects all X clients. The changes are not retained between sessions so you must modify the key bindings each time you login to an X Window session.

- If you are using a PC terminal emulator, consult the emulator application documentation to determine how the **[PF1]** - **[PF4]** keys have been mapped to your PC keyboard; it is possible that no mappings exist. Typically, the **[PF1]** - **[PF4]** keys are mapped to the keys above the key pad on your PC keyboard.

These four keys function as follows:

Table F-4 Function Keys

Function	Key
ALT (a modifier key)	**PF1**
Context-sensitive Help	**PF2**
Menu bar (to/from)	**PF3**
Escape to Shell	**PF4**

Some of the advanced keyboard actions in the following table require a key-combination using the **PF1** key. In other words, you must first press the **PF1** key, release it, and then press some other key. For example, to scroll a list one page requires you must press **PF1**, release it, and then press either the **2**, **4**, **6**, or **8** keys on the numeric keypad.

The advanced keyboard actions for a vt100 are as follows:

Table F-5 Advanced Keyboard Actions

Action	Key Combination
Scroll a list up one page.	**PF1**, then **8** on the numeric keypad
Scroll a list down one page.	**PF1**, then **2** on the numeric keypad
Scroll a list left one page.	**PF1**, then **4** on the numeric keypad
Scroll a list right one page.	**PF1**, then **6** on the numeric keypad
Select all items in a list.	**PF1**, then **/** on the main keyboard
Deselect a list item.	**PF1**, then **** on the main keyboard
Select a range of list items.	**PF1**, then **.** on the first item; move the cursor; **PF1**, then **.** on the last item
Press a button if the cursor is not resting on it.	**PF1**, then the mnemonic (underlined) letter

Glossary

A glossary term appears in **boldface italics** when used for the first time in the text of this manual. *Italicized* terms in the following glossary refer to other terms in the glossary.

0–9

0xLLA See *Link Level Address (LLA)*

A

add_new_client An Ignite-UX command that constructs a *client* directory on an *Ignite-UX server* without requiring the client to be booted from the Ignite-UX server first. See *add_new_client*(1M).

agile addressing The ability to address a *LUN* with the same *DSF* regardless of the physical location of the LUN or the number of paths leading to it. Agile addressing is introduced in HP-UX 11i v3.

agile view The representation of *LUN*s using *lunpath hardware path*s, *LUN hardware path*s, and *persistent DSF*s, introduced in HP-UX 11i v3.

always-installed For HP-UX 11i v1 and 11i v2, the software and driver *bundle*s required for HP-UX. They must be loaded as part of the operating system.

anonymous client A *client* system that requests an IP address for booting when its *MAC address* is not associated with a specific IP address on the *server*. An anonymous client has its IP address allocated from a pool of IP addresses set aside for anonymous clients. Contrast with *registered client*.

ANSI tape label See *standard label tape*.

archive 1. A file made with `make_sys_image` that contains files for *installation* or *recovery* for a system. The file format may be `tar`, `cpio`, or `pax`, and can be compressed. (To use the `pax` format with 11i v2, you must have the PAX-Enh product installed. The `pax` format is not available for 11i v1.) An archive does not include *file system* or *disk layout* information. Archives can be either *recovery archive*s or *golden archive*s. Recovery archives retain host-specific customizations from the system; golden archives have customizable files reset to the *newconfig* state. An archive may include the core HP-UX operating system, and may include application software, patches, and global customizations.

2. A file containing the contents of other files, created and maintained by programs such as `pax`, `tar` and `cpio`.

archive_impact An Ignite-UX command that calculates the disk space required for an *archive* on a per top level directory basis by default. The results are written in Ignite-UX *configuration file* syntax to standard output. See *archive_impact*(1M).

AUTO A file that defines default boot behavior. For PA-RISC systems, the `AUTO` file is in the *LIF volume*.

For Itanium®-based systems, the `AUTO` file is located in the `/opt/ignite/boot` directory. See *hpux*(1M) and *hpux.efi*(1M).

auto_adm An Ignite-UX command that allows you to manipulate *`AUTO`* file contents. See *auto_adm*(1M).

automated installation Any installation done automatically, without manual interaction. This can be done at the command line with a bootsys command, or it can be scheduled to run at a particular time using the cron *daemon*.

B

Bastille A security hardening/lockdown tool that can enhance the security of the HP-UX operating system. It provides customized lockdown on a system-by-system basis.

boot To load the *kernel* and start the operating system that is referenced by the kernel that was loaded. See *boot*(1M).

boot console handler (BCH) The firmware interface on a PA-RISC system. The Itanium-based equivalent is the *extensible firmware interface (EFI)*.

boot content *[W|V|I]INSTALL [W|V|I]INSTALLFS INSTCMDS* or *INSTCMDSIA*, *config file*s, and other files to support *boot* and Ignite-UX functionality needed to switch to another *install* source.

boot helper media *Installation media* with no *archive*. Meant for booting only. *Installation* is completed using an *Ignite-UX server*.

boot helper system A system with minimal Ignite-UX core functionality on a local *subnet* that provides an Ignite *install kernel* to a *client* to assist it with booting from an *Ignite-UX server* on another subnet.

boot image Boot content, including *[W|V|I]INSTALL*, *[W|V|I]INSTALLFS*, *INSTCMDS* or *INSTCMDSIA*, and other content needed to support boot, formatted for a specific media type such as CD, DVD, or tape.

bootsys An Ignite-UX command that allows you to *reboot* and *install client*s that are currently running HP-UX. See *bootsys*(1M).

bundle A package of software available from *Software Distributor*. Bundles are distributed within *depot*s and contain *product*s and *fileset*s. Ignite-UX typically references software at the bundle level, as opposed to product or product.fileset.

C

CD boot image A *boot image* formatted for a CD.

cfg clause See *configuration clause*.

check_net_recovery An Ignite-UX command that compares the files on a currently running system with a network *recovery archive* created by `make_net_recovery`. A report is generated showing those files that have been added, deleted, or changed since the recovery archive was created. See *check_net_recovery*(1M).

check_tape_recovery An Ignite-UX command that compares the files on a currently running system with a tape *recovery archive* created by `make_tape_recovery`. A report is generated showing those files that have been added, deleted, or changed since the recovery archive was created. See *check_tape_recovery*(1M).

CINDEX An *INDEX* file for individual clients, kept in the `/var/opt/ignite/client` directory on the *Ignite-UX server*. Used to *install recovery image*s made with `make_net_recovery`.

client A computer that uses an Ignite-UX *server* for *installation*, *recovery* services, or both.

cloning systems To replicate one computer's software and configuration onto another. This can be accomplished to varying degrees using `make_sys_image` and *make_[tape|net]_recovery*.

cold install Booting, loading HP-UX onto, and then starting a system. This process loads a new copy of HP-UX onto a system.

common configuration System software you want replicated onto other systems, such as: an operating system, patches, and application software.

compressed file A file made smaller with compression software such as `gzip` or `compress` without losing any information. See *gzip*(1) and *compress*(1).

CONFIG A file in the *LIF volume* that typically contains all the software configuration information and the default file system layout information. It includes default configuration information for the operating system release, user-defined configuration information, and information regarding *archive*s and *depot*s.

config file See *configuration file*.

configuration clause Defines a collection of related *configuration file*s used to *install* or *recover* a system. You will find configuration clauses in the `/var/opt/ignite/data/INDEX` file for *installation*s, and in the `/var/opt/ignite/clients/MAC_address/CINDEX` file for recoveries, on an *Ignite-UX server*.

configuration file A file that contains information describing installation behavior, *archive* contents, or the contents of a *depot*. Configuration files are referenced by the *INDEX* and *CINDEX* files. See *instl_adm*(4).

copy_boot_tape An Ignite-UX command that replicates a PA-RISC *recovery* tape made with *make_tape_recovery*. See *copy_boot_tape*(1M).

custom installation An *installation* tailored to your specific requirements, including: *kernel* parameters, the running of user-supplied scripts, host information, and networking information.

custom limited config A *configuration file* that informs Ignite there is no corresponding *archive*. Used with *boot helper media*.

D

daemon A process that runs in the background and is usually immune to termination instructions.

default-installed For HP-UX 11i v1 and 11i v2, software *bundle*s that are installed as default with HP-UX. These bundles can be deselected before *installation*.

depot A repository of software products, organized so *Software Distributor* (SD-UX) commands can use it as a software source.

device identifier A user friendly, readable string, such as "LAB2CAB23LUN15", that is stored on a device. It remains viewable even if the device is moved physically. See *scsimgr*(1M).

DHCP Stands for Dynamic Host Configuration Protocol, and is a way of dynamically allocating IP addresses and other network topology information to *client*s for a specified lease time.

directed boot A *boot* request that is directed to a particular boot *server* or *boot helper system*. *Client* and server networking configuration information is stored on the client system and *DHCP* is not used. A directed boot also allows boot and *installation* from a server on a different *subnet* without requiring a boot helper.

disk group The *VxVM* equivalent of an LVM *volume group*.

disk layout The way hard disks are formatted and information stored on them. There are two general methods of disk layout: physical-storage layout and logical-storage layout. *VxVM* and LVM use logical-storage layout, and use various layout techniques such as mirroring and striping.

disk layout version The version of the *VxFS* private data that is used for its *file system* layout. This term can be abbreviated to DLV*x*, where *x* is the disk layout version number. See *mkfs_vxfs*(1M).

DNS Stands for Domain Name Service, and provides mapping between hostnames and IP addresses.

DSF Stands for Device Special File. A file associated with an I/O device. DSFs are read and written to as ordinary files are, resulting in activation of the associated device.

DUMP A use designation typically for an LVM *logical volume* or a *VxVM* volume for system crash dump storage. See *crashconf*(1M).

DVD boot image A *boot image* formatted for a DVD.

E

EFI See *extensible firmware interface (EFI)*.

El Torito An extension to ISO9660 for creating bootable optical media.

essential The list of files and directories in `/opt/ignite/recovery/mnr_essentials` describing the default minimum contents of a *recovery archive*.

expert recovery An Ignite-UX mode of operation allowing expert users to repair a system with software damage without reinstalling HP-UX.

extensible firmware interface (EFI) The Intel® developed firmware environment on Itanium-based systems that acts as an interface between operating systems and platform firmware. The interface consists of platform related information, as well as *boot* and runtime service calls. The PA-RISC equivalent is the *boot console handler (BCH)*.

F

file system A collection of files and supporting data structures residing directly on a mass storage device or on a virtual or logical disk. There are various file system implementations, such as HFS and VxFS.
See also install file system.

fileset For *SD-UX*, a collection of files within a *product*. The product may be part of a *bundle*. See *sd*(5).

G

gateway The IP address of a system that routes forwarded traffic to a non-local network. A gateway IP address is usually associated with a router.

golden archive An *archive* with files set to the *newconfig* state.

golden image A combination of a *golden archive*, and a *configuration file* describing a system's *disk layout* and *file system*. Use as a *common configuration* to *install* systems.

gzip A command available with HP-UX that compresses and decompresses files in "`.gz`" format. See *gzip*(1).

H

HBA Stands for Host Bus Adaptor. A physical I/O interface that provides I/O processing and connectivity between a server and a storage device.

hierarchical file system (HFS) A particular implementation of a *file system*. See *mkfs_hfs*(1M).

HPUX The HP-UX bootstrap loader. Loads the *kernel* and starts HP-UX. For Ignite-UX, the HP-UX bootstrap loader loads the *install kernel* (for example `IINSTALL`) and *install file system* (for example `IINSTALLFS`). The bootstrap loader may also load other *LIF* content it needs to operate. See *hpux*(1M).

I - J

ignite The `ignite` command name. See *ignite*(5).

Ignite-UX server A *server* from which Ignite-UX is used to *install* HP-UX on *client* systems.

`IINSTALL` The *install kernel* for Itanium-based systems.

`IINSTALLFS` The associated *file system* for the `IINSTALL` *kernel*.

image The current state of your computer, or portion of your computer. Often thought of as a "snapshot" of the state of the machine at any given moment.

INDEX A file on the *Ignite-UX server* and in *LIF volume*s that groups references to *configuration file*s in clauses in order to define *installation* behavior.

initial system loader (ISL) Implements the operating system independent portion of the bootstrap process on PA-RISC systems. It is loaded and executed after self-test and initialization have completed successfully. See *isl*(1M).

INSTALL The *install kernel* for 32-bit enabled PA-RISC systems.

install Perform an *installation*.

install content *Boot content* and other files needed to support Ignite-UX functionality during *install* and *recovery*.

install environment The Ignite *install kernel*, the *install file system*, and *INSTCMDS* or *INSTCMDSIA*, *SYSCMDS* or *SYSCMDSIA*, and *RECCMDS* or *RECCMDSIA* as needed.

install file system See `IINSTALLFS`, *INSTALLFS*, *VINSTALLFS*, and *WINSTALLFS*.

install kernel See `IINSTALL`, *INSTALL*, *VINSTALL*, and *WINSTALL*.

installation Loading the operating system, other software, and configuration information onto a system.

installation media Removable media such as tape, CD, and DVD for *stand alone installation* of a *client* system.

INSTALLFS The associated *file system* for the *INSTALL kernel*.

INSTCMDS A compressed `tar` *archive* of commands in the *LIF volume*, or on an *Ignite-UX server*, required for specifying the system configuration to install on a PA-RISC system.

INSTCMDSIA A compressed `tar` *archive* of commands in the *LIF volume*, or on an *Ignite-UX server*, required for *disk layout* on an Itanium-based system.

instl_adm An Ignite-UX command that checks syntax on Ignite-UX *configuration file*s and manages the configuration file in an *install file system* on an *Ignite-UX server*. See *instl_adm*(1M).

instl_bootd A boot protocol *daemon* for Ignite-UX *client*s that responds to PA-RISC systems requesting *boot* services from the *Ignite-UX server*. See *instl_bootd*(1M).

instl_combine An Ignite-UX command that combines a *CD boot image* or *DVD boot image* with *install content*. See *instl_combine*(1M).

instl_dbg An Ignite-UX command that will parse and debug an Ignite-UX *client*'s *configuration file*s. See *instl_dbg*(1M).

ISL See *initial system loader (ISL)*.

ISO image An ISO9660 formatted file that is to be written to a CD or DVD. Used for a bit-for-bit burn of a CD or DVD.

itool The name of an internal IUX program that presents the Ignite-UX *client installation* configuration user interface. This interface has five tabs: **Basic**, **Software**, **System**, **File System**, and **Advanced**. It is used to customize all or part of the operating system installation on the client before an installation or *recovery*. This command is not invoked from the command line.

K

kernel The HP-UX kernel is the executable code responsible for managing the computer's resources, such as: allocating memory, creating processes, and scheduling programs for execution. The kernel resides in RAM (random access memory) whenever HP-UX is running.
See also install kernel.

L

LANIC See *MAC address*.

largefiles An option available on certain *file system*s that allows file sizes greater than 2 gigabytes.

legacy DSF A *DSF* with the hardware path information such as SCSI bus, target, and *LUN* embedded in the file's minor number and file name, such as `/dev/dsk/c#t#d#`. These are the only DSFs available in releases prior to HP-UX 11i v3.

legacy hardware path The representation of a hardware path as it exists in releases prior to HP-UX 11i v3. It is composed of a series of bus-nexus addresses separated by slashes (/) leading to the *HBA*. After the HBA, additional address elements, such as domain, area, port, target, and *LUN*, are separated by periods (.). The string /0/2/1/0.1.4.0.0.2.7 is an example of a legacy hardware path.

legacy view The representation of *legacy hardware path*s and *legacy DSF*s as in releases prior to HP-UX 11i v3.

LIF See *Logical Interchange Format*.

LIF volume Portions of content needed for *boot* and *install* combined into a LIF file. The LIF file is included in *boot content* and in *install content*.

link level address (LLA) See *MAC address*.

logical interchange format A simple file system implemented on HP computers able to run HP-UX to aid in media transportability. See *lif*(4).
See also LIF volume.

logical volume A virtual subdivision of a *volume group*. See *logical volume manager (LVM)*.

logical volume manager (LVM) A specific *volume manager* type created and managed by the HP LVM product. See *lvm*(7).

LUN An identifier of a SCSI device. This refers to an end storage device such as a disk, tape, floppy, or CD. This is the unit itself and does not represent the path to the unit.

LUN hardware path A hardware path for a SCSI *LUN* that virtualizes all paths to the LUN. The first path element is 64000, followed by a virtual bus instance and a logical unit number. Multipathed LUNs have a single LUN hardware path. The string 64000/0xfa00.0x22 is an example of a LUN hardware path. LUN hardware paths are part of the *agile view* introduced in HP-UX 11i v3.

lunpath hardware path The representation of a hardware path for a mass storage device. It is identical in format to a *legacy hardware path* up to the *HBA*. After the HBA, additional addressing is represented in hexadecimal format. The string 0/2/1/0.0x50001fe1500170ac.0x4017000000000000 is an example of a lunpath hardware path. Lunpath hardware paths are part of the *agile view* introduced in HP-UX 11i v3.

M

MAC address Stands for Media Access Control. A network card's unique hardware number. Used to uniquely identify a network interface connected to a local area network.

make_[tape|net]_recovery
Collectively refers to the `make_tape_recovery` and `make_net_recovery` Ignite-UX commands.

make_boot_tape An Ignite-UX command that creates a bootable tape that contains just enough information for a system to *boot* and connect to an *Ignite-UX server*. See *make_boot_tape*(1M).

make_bundles An Ignite-UX command that creates a *bundle* container for products in a *depot*. See *make_bundles*(1M).

make_config An Ignite-UX command that constructs Ignite-UX *configuration file*s for *Software Distributor depot*s. See *make_config*(1M).

make_depots An Ignite-UX command that builds a *Software Distributor depot* for use by other Ignite-UX tools by copying *bundle*s from a Software Distributor source. See *make_depots*(1M).

make_ipf_tape An Ignite-UX command that creates a bootable tape for an Itanium-based system. The tape will contain *boot* and *installation* components, but not a *recovery archive*. See *make_ipf_tape*(1M).

make_medialif An Ignite-UX command that creates a file containing a *LIF volume* that is used to *boot* PA-RISC systems. This file may then be copied to tape, CD, or DVD to create *installation media*. See *make_medialif*(1M).

make_net_recovery An Ignite-UX command that creates a *recovery archive* and system *config file*, and stores the resultant *recovery image* on a system on the network. See *make_net_recovery*(1M).

make_sys_image An Ignite-UX command that creates an *archive* of a system. See *make_sys_image*(1M).

make_tape_recovery An Ignite-UX command that creates a *recovery archive* and system *config file*, and stores the resultant *recovery image* on a local tape. See *make_tape_recovery*(1M).

manage_index An Ignite-UX command that is used to manipulate an Ignite-UX `INDEX` or `CINDEX` file. See *manage_index*(1M).

manifest Details of a *client*'s *installation*. It is available in a file on the client and the Ignite-UX *server* after successful installation, and it may be displayed or regenerated with the `print_manifest` command. See *print_manifest*(1M).

media image The combined software source, *config file*s, and *boot* information to be written to CD or DVD, used when preparing *installation media*. It is a single large file in the HP-UX operating system that is written to the medium and used for *installation* or *recovery*.

multipathing The detection, correlation, and coordinated usage of multiple hardware paths leading to the same LUN.

N

network boot When a system boots the HP-UX *install kernel* over the network from an Ignite-UX server.

newconfig The state of a file as it is stored in `/usr/newconfig` before it is moved into place and modified to be system specific. Files in a newconfig state contain no information about the personality of the system.

NFS Stands for Network File System. Allows a *client* to perform transparent file access over the network.

NIC address See *MAC address*.

NIS Stands for Network Information Service. It allows access from any system to any system with a single user identification and password.

NIS domain The group of systems sharing NIS configuration information.

O

offline diagnostic environment (ODE) Diagnostic utilities stored in the *boot* area or service partition able to run without the operating system.

optional With 11i v3, software *bundle*s that are not installed with HP-UX by default. You must select these bundles before *installation*. Ignite-UX is an example of an optional software bundle.

P

partition A portion of a disk device that appears to *volume manager*s, *file system*s, and other OS software as a disk device. Partitions allow different portions of a disk to hold content needed for *boot*, OS file systems, diagnostic tools, and other information.

pax An HP-UX file archiving command that extracts, lists, and writes member files to an *archive*. Ignite-UX uses `pax` to produce `tar`, `cpio`, and `pax` formatted archives. See *pax*(1),*tar*(1), and *cpio*(1).

PDC See *processor dependent code*.

persistent DSF A DSF conforming to the naming model introduced in HP-UX 11i v3 to support agile addressing. The device file name contains an instance number, such as `/dev/disk/disk#`, and the minor number has no hardware path information.

physical location Device identifier that uses information such as cabinet, bay, and LUN ID to indicate where the device is actually located. Used to indicate a specific device independent of I/O path changes.

pkg_rec_depot An Ignite-UX command that repackages the Ignite-UX product into a *depot* on an *Ignite-UX server* in order to distribute Ignite-UX software to *client* systems that use the Ignite-UX server for network *recovery*. See *pkg_rec_depot*(1M).

print_manifest An Ignite-UX command that displays a *manifest* of the system it is run on. It can either display the original manifest, or gather information about the current system when it runs. See *print_manifest*(1M).

processor dependent code The firmware that implements all processor-dependent functionality including initialization and self-test of the processor. Upon completion, it loads and transfers control to the *initial system loader (ISL)*. Firmware behavior varies depending on the hardware series. See *pdc*(1M).

product In SD-UX, a collection of *fileset*s, individual filesets, or both.

PXE Stands for preboot execution environment. A protocol built on top of *DHCP* on HP-UX Itanium-based systems used to request *boot* services from a *server* in order to load the operating system.

R

reboot To stop all running code including the *kernel*, restart a computer, and *boot* the system. See *shutdown*(1M) and *reboot*(1M).

RECCMDS A compressed `tar` *archive* of commands required for *expert recovery* on PA-RISC systems.

RECCMDSIA A compressed `tar` *archive* of commands required for *expert recovery* on Itanium-based systems.

recommended With 11i v3, software *bundle*s that are recommended and should be installed with HP-UX because they fulfill required software dependencies, if any exist. You can deselect the bundles before *installation*.

recover Perform a *system recovery*.

recovery See *system recovery*.

recovery archive An *archive* that retains host specific customizations from the system.

recovery image A system-specific snapshot of a system that contains hostname, IP address, networking information, all files and directories essential to bringing up a functional system, as well as optionally including specified data on a disk, *volume group*, file, or directory basis. It is created with *`make_[tape|net]_recovery`*.

recovery shell See *expert recovery*.

registered client A *client* system that has its *MAC address* registered with the *server* in order to always *boot* to the same assigned IP address. For PA-RISC systems, the client's MAC address is assigned to an IP address in the `/etc/opt/ignite/instl_boottab` or `/etc/bootptab` file. For Itanium-based systems, the default is to use the `/etc/bootptab` file. Contrast with *anonymous client*.

required With 11i v3, software and network driver *bundle*s that are required and always installed with HP-UX. Software in this category cannot be deselected.

S

save_config An Ignite-UX command that creates a hardware *configuration file*. It extracts disk and *file system* information along with certain system and networking parameters for the current system, and writes it to a configuration file. See *save_config*(1M).

script hook Predefined mechanisms allowing user-defined scripts to run at specific points during the execution of the *installation*. Within the Ignite-UX installation procedure there are various script hooks: pre-config, pre-load, post-load, post-config, and final. See *instl_adm*(4) under "Command and Script Execution Hooks."

SCRIPTS A configuration clause of commands in the *LIF volume* containing scripts that can be selected in the *itool* under the **Advanced** tab.

SD-UX See *Software Distributor*.

selectable For 11i v1 and 11i v2, software *bundle*s that are not installed by default with HP-UX. You must select these bundles to install them. Ignite-UX is an example of a selectable software bundle.

server A computer that provides software and services to *client*s.

server selection The ability of a *client* system to request *DHCP*/`bootp` *boot* services from a specific boot *server* and ignore all other boot offers. Client networking configuration information is supplied by the boot server.

setup_server An Ignite-UX command that performs administration tasks for an Ignite-UX *server* from a command line interface. The same functionality is available with the Ignite-UX GUI. See *setup_server*(1M).

Software Depot See *depot*.

Software Distributor An HP-UX product that provides a set of tools for centralized HP-UX software management. Software Distributor (SD-UX) commands are included with the *installation* of the HP-UX operating system. SD-UX commands typically use the prefix "sw", such as in `swinstall` and `swverify`.

stand alone In terms of Ignite-UX (as opposed to any other usage) a *client* system that is not network booted, but may still use an Ignite-UX *server* for *installation* or *recovery*.

standard label tape A tape containing ANSI standard labeling. Ignite-UX uses *ansitape*(1) to create a standard label tape.

subnet A separate part of a larger network connected via network gateways.

SWAP A use designation typically for an *logical volume manager (LVM) logical volume* or a *VxVM volume* for paging. See *swapon*(1M).

SWAP+DUMP A use designation typically for an *logical volume manager (LVM) logical volume* or a *VxVM volume* that can be used for *DUMP* or *SWAP*.

SYSCMDS A compressed `tar` *archive* of commands in the *LIF volume* or on an Ignite-UX *server*, required for *installation* of PA-RISC systems.

SYSCMDSIA A compressed `tar` *archive* of commands in the *LIF volume* or on an Ignite-UX *server*, required for *installation* of Itanium-based systems.

system image See *image*.

system recovery The use of a *recovery image* to reestablish a system.

T

two-step media recovery A method of using the Ignite-UX tape *recovery* tool when a system or I/O interface does not support firmware tape boot. The method involves step 1: booting from *installation media* such as DVD or CD, and then step 2: recovering from tape.

U

Unique LUN ID A general term for a *LUN* identifier, which might specifically be a physical disk's WWID, a WWID assigned to a SAN virtual LUN, the uniq_name assigned to a LUN for which a WWID could not be obtained, or some other type of identifier unique to a LUN.

V

Veritas A set of products from Symantec that include *VxVM* and *VxFS*.

VINSTALL The *install kernel* for V-class PA-RISC systems.

VINSTALLFS The associated *file system* for the *VINSTALL kernel*.

vmunix The default *kernel* filename, used during normal system operation. vmunix is built by the `mk_kernel`command, which Ignite calls during system installation. The *AUTO* file typically contains the entry "`boot vmunix`", which references `/stand/vmunix` on the selected boot device.

volume 1. A pool of data storage made up of one or more physical disks. These volumes are created and managed using tools from one of the volume managers, *VxVM* or LVM. See *volume manager*.

2. A tape, especially when ANSI labeled.

volume group An arbitrary grouping of disks for use by LVM. See *volume manager*.

volume manager A subsystem for managing disk space that allows one or more disks to be combined. It can provide increased size, improved reliability via data and path redundancy, greater configuration flexibility, and other features for managing *file system* space.

VxFS A Veritas product from Symantec that is a *file system* implementation.

VxVM A specific *volume manager* type that is a *Veritas* product from Symantec.

W - Z

[W|V|I]INSTALL Refers to the *WINSTALL*, *VINSTALL*, *IINSTALL*, or *INSTALL install kernel* depending on your system.

[W|V|I]INSTALLFS Refers to the *WINSTALLFS*, *VINSTALLFS*, *IINSTALLFS*, or *INSTALLFS file system* depending on your system.

whole disk A volume management selection that treats the entire disk as one volume. This selection does not use a volume manager product. For bootable disks, it is a practical selection only for Itanium-based systems.

WINSTALL The *install kernel* for 64-bit PA-RISC systems.

WINSTALLFS The associated *file system* for the *WINSTALL kernel*.

XNTP Stands in part for network time protocol. The `xntpd` *daemon* maintains system time in agreement with Internet standard time servers. This can be configured from the *itool* **System** tab under Network Services->XNTP.

Index

D

L

M

S

T

U

V

W

X