Ignite-UX Administration Guide

HP Computers
with HP-UX 10.x, 11.0 or 11i

Manufacturing Part Number : B2355-90772

Edition 10, March 2003
Printed in U.S.A.

Legal Notices

The information in this document is subject to change without notice.

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Warranty

A copy of the specific warranty terms applicable to your Hewlett-Packard product and replacement parts can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DOD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

HEWLETT-PACKARD COMPANY
3000 Hanover Street
Palo Alto, California 94304 U.S.A.

Use of this document and any supporting software media supplied for this pack is restricted to this product only. Additional copies of the programs may be made for security and back-up purposes only. Resale of the programs, in their present form or with alterations, is expressly prohibited.

Copyright Notice

Trademark Notices

Itanium® is a registered trademarks of Intel® Corporation.

MS-DOS® and Microsoft® are U.S. registered trademarks of Microsoft Corporation.

Netscape ® is a registered trademark of Netscape Communications Corporation.

OSF/Motif™ is a trademark of the Open Software Foundation, Inc. in the U.S. and other countries.

SunForum® is a registered trademark of Sun Microsystems, Inc. in the United States and other countries.

UNIX® is a registered trademark in the United States and other countries, licensed exclusively through The Open Group.

VERITAS® is a registered trademark of VERITAS Software Corporation.

VERITAS File System™ is a trademark of VERITAS Software Corporation.

X Window System™ is a trademark of the Massachusetts Institute of Technology.

Publication History

The manual's publication date and part number indicate its current edition. The publication date changes when a new edition is released. The manual part number changes when extensive changes are made.

To ensure that you receive the new editions, you should subscribe to the appropriate product support service. See your HP sales representative for details.

- March 1999, Edition 1, HP part number **B2355-90677**.
- December 2000, Edition 2. HP part number **B2355-90704**.
- June 2001, Edition 3. HP part number **B2355-90738**.
- March 2002, Edition 4. HP part number **B2355-90749**.
- June 2002, Edition 5. HP part number **B2355-90750**.
- September 2002, Edition 6. HP part number **B2355-90758**.
- September 2002, Edition 7. HP part number **B2355-90765**.
- October 2002, Edition 8, HP part number **B2355-90770**.
- December 2002, Edition 9, HP part number **B2355-90767.**
- March 2003, Edition 10, HP part number **B2355-90772**.

This guide replaces the *Installing HP-UX 11.0 and Updating from HP-UX 10.x to 11.0* guide, HP part numbers **B23355-90135** and **B2355-90679**.

For the latest version of this and other HP-UX guides, see the HP technical documentation web site at:

`http://docs.hp.com/`

Please direct comments regarding this guide to:

Hewlett-Packard Company
HP-UX Learning Products
3404 East Harmony Road
Fort Collins, Colorado 80528-9599

Or, use this web form to send us feedback:

`http://docs.hp.com/assistance/feedback.html`

Typographic Conventions

We use the following typographical conventions:

audit (5)	An HP-UX manpage. *audit* is the name and *5* is the section in the *HP-UX Reference*. On the web and on the Instant Information CD, it may be a hot link to the manpage itself. From the HP-UX command line, enter "`man audit`" or "`man 5 audit`" to view the manpage. See *man* (1).
Book Title	The title of a book. On the web and on the Instant Information CD, it may be a hot link to the book itself.
Emphasis	Text that is emphasized.
Emphasis	Terms defined for the first time or text that is strongly emphasized.
`ComputerOut`	Text displayed by the computer.
`Command`	A command name or qualified command phrase.
`Computer`	`Computer font` indicates literal items displayed by the computer. For example: `file not found`
`Filename`	Text that shows a filename and/or filepath.
`UserInput`	Commands and other text that you type.
`Variable`	The name of a variable that you may replace in a command or function or information in a display that represents several possible values.
`[ ]`	The contents are optional in formats and command descriptions.
`{ }`	The contents are required in formats and command descriptions. If the contents are a list separated by `|`, you must choose one of the items
. . .	The preceding element may be repeated an arbitrary number of times.
`|`	Separates items in a list of choices.

Contents

Contents

Contents

Contents

1 Introducing Ignite-UX

This chapter introduces you to the features and uses of Ignite-UX:

- About this guide.
- Ignite-UX overview.
- Solving problems with Ignite-UX.

About This Guide

This guide describes installing, configuring and using Ignite-UX to facilitate installing and recovering HP-UX on HP computer systems in your computing environment.

This guide is written for experienced HP-UX systems administrators who are responsible for setting up and maintaining HP-UX workstations and servers. They must be familiar with installing HP computer hardware and software, upgrading software, applying patches and troubleshooting system problems. Ignite-UX will help them install new systems and maintain existing ones with less effort than when using individual install and update tools.

Web papers included here

This guide replaces Ignite-UX information in the *Installing HP-UX 11.0 and Updating from HP-UX 10.x to 11.0* guide previously supplied with HP-UX 11.0 media. This guide also includes information from the following papers available on the Ignite-UX web site:

- *Ignite-UX Startup Guide for System Administrators*.
- *Ignite-UX Network System Recovery*.
- *Customized Install Media*.
- *Ignite-UX FAQ*, as of March 2002 (check the web for newer versions).

Documentation and Training

Check the Ignite-UX web site often for announcements, updates to the *Ignite-UX FAQ*, and to download the latest version of Ignite-UX:

`http://software.hp.com/products/IUX`

Details about recent changes to Ignite-UX are in the *Ignite-UX Release Notes* located on the web site or in the directory */opt/ignite/share/doc/release_note*.

Additional information is available on these web pages:

- Ignite-UX and Mirrored disks

  `http://software.hp.com/products/IUX/docs/diskmirror.pdf`

- Ignite-UX System Recovery

  `http://software.hp.com/products/IUX/docs/recovery_merge.html`

- DHCP FAQ

  `http://www.dhcp.org`

Ignite-UX training

HP offers a 3-day classroom course on Ignite-UX for the experienced HP-UX system administrator. For details, classes scheduled in your area, and class registration, go to:

`www.hp.com/education/courses/h1978s.html`

Ignite-UX Commands and Online Documentation

Ignite-UX provides online information in the `/opt/ignite/share/doc/` directory. Also see these Ignite-UX manpages:

Table 1-1

Ignite-UX Manpage	Description
ignite (5)	Ignite clients remotely from the Ignite-UX screen and provides an overview of all Ignite-UX commands.
instl_adm (1M) *instl_adm* (4)	Manage Ignite-UX config files.
instl_combine (1M), *make_medialif* (1M)	Construct custom, bootable install media.
instl_dbg (1M)	Debug config files.
instl_bootd (1M)	Boot protocol server for Ignite-UX client.
bootsys (1M)	Reboot and install systems using Ignite-UX.
make_bundles (1M)	Package SD bundles into an SD Depot.
make_depots (1M)	Creates SD depots from media.
make_boot_tape (1M)	Create a system boot tape.
make_net_recovery (4)	Create recovery archives on a network system.
make_tape_recovery (1M)	Create recovery tapes. Replaces `make_recovery` available beginning with Ignite-UX A/B 3.2.
check_recovery (1M)	Check recovery tape status since last `make_*_recovery`.
make_sys_image (1M)	Create golden system images.
make_config (1M)	Generate config files for installing software in SD bundles.

Table 1-1 (Continued)

Ignite-UX Manpage	Description
manage_index (1M)	Manage the `INDEX` file.
setup_server (1M)	Perform Ignite-UX client-server administration tasks.
add_new_client (1M)	Construct and populate a client directory without requiring a reboot.

Ignite-UX Overview

Ignite-UX addresses your need to perform system installations and deployment, often on a large scale. With Ignite-UX, you can:

- Create and reuse standard system configurations.
- Archive a standard system configuration and use that archive to replicate systems.
- Create customized processes to allow interactive and unattended installs.
- More-easily recover your operating system (OS) and applications after crashes and hardware failures.

After running an Ignite-UX install session, you have a working HP-UX client system.

Ignite-UX release versions

Ignite-UX is an HP-UX 10.*x*, 11.0, and 11i product, including 11i v1.6 which supports **Itanium Processor Family** (IPF) systems, that facilitates installing and configuring HP-UX systems. Ignite-UX releases are available to install the HP-UX 10.01, 10.10, 10.20, 11.0 and 11i releases on client systems.

Ignite-UX server software is currently available in these versions:

- HP-UX 10.01, 10.10, 10.20 — version A.3.7.95
- HP-UX 11.0 and 11i— version B.4.0 or later

An Ignite-UX B.*x* server runs on HP-UX 11.0 and 11i and can install HP-UX 10.20, 11.0, and 11i OS and applications on target systems. An Ignite-UX A.*x* server runs on HP-UX 10.*x* and can only install HP-UX 10.*x* software on target systems.

Be sure to check the Ignite-UX web site often for announcements and information on new Ignite-UX releases.

Ignite-UX Features

Client/server control

Ignite-UX install sessions for multiple targets can be controlled from a single server in a true client/server model. A graphical user interface (GUI) called the **Ignite-UX screen** helps you manage multiple, simultaneous install sessions. Alternatively, you can run a single install session from the target system if that is more convenient. A single install server can serve multiple HP-UX releases.

Easy-to-use GUI

The Ignite-UX screen uses tabbed dialogs for task navigation. In addition, a wizard mode is available for the novice.

Multi-sourced installations

You could install your base from one SD depot, a set of patches from another depot, and the applications you want from a third depot all in one session.

Multiple archive formats

In addition to supporting SD software sources, Ignite-UX supports tar, cpio, and pax archives. Tools are provided to help you create a **golden system image** if you wish to install from an archive.

Custom installations

It is easy to create a system that is ready to go as soon as the install session completes. Many of the tasks typically done as separate steps after an install are now incorporated into the Ignite-UX installation process. Ignite-UX allows you to specify what kernel parameters you want set and what user-supplied scripts you would like to run as part of the session. Many different script hooks are provided so you can add your own customizations (during and after the installation). Also, the host and networking information which must normally be supplied at first boot can be specified at install-time. You can:

- Create a configuration for your particular needs, save it, and then quickly apply that configuration to multiple install targets.
- Set up a configuration and then install it on a target machine with no further user interaction. This can be done in both the initial installation and the re-installation cases.
- 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 loaded. This report can be customized to meet your needs.
- Construct your own customized bootable install/recovery media using the `make_medialif` command.

System recovery

You have consistent, reliable recovery in the event of a catastrophic failure of the system disk or root volume group using either the `make_tape_recovery` or `make_net_recovery` command.

Support for ServiceControl Manager

Ignite-UX supports installing HP-UX client systems in an HP ServiceControl Manager environment. See the *ServiceControl Manager Installation and Configuration* guide for more details.

Solving Problems with Ignite-UX

Once you have an Ignite-UX server installed and configured for your environment, you'll find that it can help you solve many common installation and recovery problems.

One-step installation

Once you configure a system with a **common configuration** that you want **pushed** to other systems, use Ignite-UX to either manually or automatically Ignite-UX each client system. This common configuration can include any supported HP-UX 10.*x*, 11.0, or 11i OS, plus you can add any required patches and applications. This configuration can be bundled into an OS archive, either on the Ignite-UX server or any system in your environment. You can also **pull** bits from an Ignite-UX server and install the client locally. These processes are explained in Chapters 2 through 5.

Re-install HP-UX or apply patches quickly

You can quickly re-install the OS on an existing system after repair from either an OS archive or SD depots and apply patches. See Chapter 6, Installing Patches with Ignite-UX,and Chapter 7, Using Golden System Images, for details.

Scanning a system

With Ignite-UX, you can easily scan a system, or all systems in your environment, to see what hardware is present, how the disks are used, what kernel modifications have been made and what software has been loaded. See Chapter 5, "Installing HP-UX with Ignite-UX on Clients from a Server," on page 73.

Automate installations

Using Ignite-UX's configuration files allow you to completely automate the OS installation process on any systems in your environment, as explained in Chapter 9, "Automating Installations."

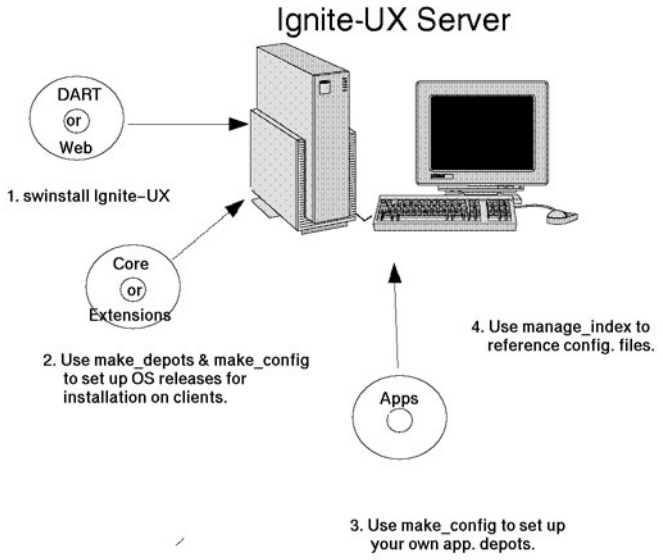
Quick system recovery

In addition to OS archives for initial installations, you can create recovery archives on tape (access from a drive on the client) or on any system in your environment (access via the network). See Chapter 11, System Recovery, for details.

2 Installing and Administering an Ignite-UX Server

This chapter describes installing and configuring an Ignite-UX server.

For on-line information about the Ignite-UX server after it has been installed, see the `/opt/ignite/share/doc/` directory and the *ignite* (5) manpage.



Installation Overview

Ignite-UX functions as a client-server application. Much of the server configuration will be performed for you in the Ignite-UX installation process, but there are also some separate steps you must take after installation. Tools are supplied to help you complete the server configuration.

Installing Ignite-UX will take care of most server configuration tasks. This can also be done outside Ignite-UX by using either the `setup_server` command as a simple interface or by using the Ignite-UX screen, as explained in this chapter.

Installation tasks

The tasks required to set up an Ignite-UX server are:

"A: Obtain Ignite-UX Software" on page 27

"B: Install Ignite-UX Software" on page 28

"C: Update PATH" on page 29

"D: Set Up or Update the Software Source" on page 29

"E: Add Optional Applications" on page 31

"F: Installing Minimal Ignite-UX Filesets" on page 32

"G: Start Ignite-UX for the First Time" on page 33

"H: Set an Initial Ignite-UX Server Configuration" on page 34

"I: Starting Ignite-UX" on page 36

"J: Configuring Server and Session Options" on page 40

Before proceeding to install an Ignite-UX server, review the server's hardware, software and networking requirements explained next.

Ignite-UX Hardware Requirements

IMPORTANT NFS Diskless functionality *is not supported* on HP-UX 11.0 and later versions. Do not update your server to HP-UX 11.0 or later versions if you intend to operate that server as a NFS Diskless server.

Server requirements

You will need the following to install an Ignite-UX server:

- **Computer —** A Series 700/800 system running HP-UX 10.0, 10.10, 10.20, 11.0 or 11i. See Ignite-UX version requirements in"Ignite-UX release versions" on page 18.
- **Memory —** Ignite-UX requires 64MB minimum on each server and client. Your HP support engineer can assist in determining the proper amount of RAM.
- **Source Device —** Make sure that your system has an appropriate source (CD-ROM, DVD, DDS drive or LAN card). Ensure that tape drive heads are clean.
- **Disk Drive —** A server needs at least one hard-disk drive with at least the following capacities (`swinstall` performs an analysis of disk space needed prior to loading the software):
  - — Generally, 2 GB or more for a usable system, 2.2 GB if on HP-UX 11.0 and 4GB if on HP-UX 11i.
  - — Ignite-UX will be loaded under the directory `/opt/ignite`. The data files Ignite-UX creates will be placed under `/var/opt/ignite`. Ignite-UX installation will require approximately 105MB of disk space. You will probably need additional space available under `/var/opt/ignite` for archive and software depot storage.
- **tftp** — Ignite-UX will transfer some of its files using tftp. The minimum directories needed by tftp are set up in the `/etc/inetd.conf` file. Others may need to be added if you place configuration scripts in non-standard locations.
- An additional X11 display server (workstation, X-terminal, PC running an X server, etc). This can be the same system as above.

  - — A separate graphics display may be required if a Series 800 Ignite-UX server is being used. Or:
  - — The display can be redirected to another X-windows system by setting the `DISPLAY` environment variable. For example, in the Korn Shell or Posix Shell, enter:

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

- Product media or link to the web to load Ignite-UX and any software depots you plan to distribute to target systems.
- Client and server must be on the same subnet if you plan to do the initial boot of the client over the network. A **boot-helper** system can be used to get between subnets and the `bootsys` command also works between subnets. See Appendix B, “Using a Boot-Helper System,” on page 261.

NOTE You can boot over the network only from an Ethernet interface.

Installing an Ignite-UX Server

A: Obtain Ignite-UX Software

Via Media and the Web

Ignite-UX is available from these sources in standard Software Distributor (SD) depot format:

- Application CD-ROM or DVD (if specified when ordering) supplied with HP-UX 10.20, 11.0 and 11i OS media.
- An HP-UX 11i CD1 or DVD (if specified when ordering).
- HP's Software Depot:

  `http://software.hp.com/products/IUX`

  Be sure to obtain the correct Ignite-UX version for your system:

  — For HP-UX 11.0 and 11i, download and install Ignite-UX version B.*x*.

  — For HP-UX 10.20, download and install Ignite-UX version A.*x*.

An Ignite-UX version B.*x* server can install HP-UX 10.20 and 11.0/11i OS and applications on target systems. An Ignite-UX version A.*x* server can only install HP-UX 10.*x* software on target systems.

You may load one or more of the individual `Ignite-UX-1`*x*`-`*xx* bundles onto your system to set up a new Ignite-UX server for installing only that HP-UX version on other systems. That is, you can choose to load a release-specific bundle, such as `Ignite-UX-10-20` for HP-UX 10.20, or an entire bundle such as `B5724AA_APZ`.

IMPORTANT

Do not install individual Ignite-UX server bundles to update an existing Ignite-UX Server. Instead, install the complete bundle for your OS, for example, B5724AA_APZ for HP-UX 10.20. To update yours server to HP-UX 11i, also consider using the new `update-ux` command, as explained in the guide supplied with HP-UX 11i OE media.

Via ftp

You can also access HP's SD Depot via ftp, however this access is "blind"; the `ls` command is not available in the `/ftp` directory. Follow these steps:

Step 1. Log on anonymously to HP's Software Depot:

```
ftp www.software.hp.com
```

Step 2. Move to the `swdepot` directory and get the software bundles you need:

```
ftp> cd /dist/swdepot
ftp> get file_name.tar
```

file_name examples for HP-UX 10.20 servers are:

```
ignite_10.01.tar, ignite_10.10.tar, ignite_10.20.tar,
ignite_all.tar
```

file_name examples for HP-UX 11.0/11i servers are:

```
ignite11_10.01.tar, ignite11_10.10.tar, ignite11_10.20.tar,
ignite11_11.00.tar, ignite11_ALL.tar
```

B: Install Ignite-UX Software

Each software bundle contains the Ignite-UX tools plus the data files required for support of the particular HP-UX release indicated by the bundle name. If you do not wish to load the entire Ignite-UX bundle, see "F: Installing Minimal Ignite-UX Filesets" on page 32.

Step 1. If needed, remove NetInstall. Ignite-UX replaces the capability previously supplied by the NetInstall bundle that came with HP-UX releases 10.01, 10.10 and 10.20. (A system cannot be configured as a server for both NetInstall and Ignite-UX.) Loading any of the Ignite-UX software bundles will give an error until you either remove the NetInstall bundle or touch the `/tmp/okay_to_remove_net_install` file.

Step 2. Once the application CD containing Ignite-UX has been mounted or you have downloaded the Ignite-UX bundle from the web, use `swinstall` to load the desired Ignite-UX bundle(s). You can load the entire product, or you can load only a single Ignite-UX depot if you plan on only using Ignite-UX to install a single release, such as HP-UX 10.20, on client systems. For example, if the Applications CD is mounted at `/cdrom` and you want to load the support for installing HP-UX 10.20 clients onto an HP-UX 10.20 server system, enter:

```
swinstall -s /cdrom Ignite-UX-10-20
```

Or, if you want to install the entire Ignite-UX 11.0 product on an HP-UX 11.0 or 11i server from a software depot on your network located at, say, *hpfclc.fc.hp.com:/release/Ignite-UX*, enter:

```
swinstall -s hpfclc.fc.hp.com:/release/Ignite-UX \
B5725AA_APZ
```

Step 3. After loading Ignite-UX bundle(s), unmount and remove the media and mount the media/drive, if needed, to load the Core software.

C: Update PATH

In your login scripts, add `/opt/ignite/bin` to your default search path:

```
export PATH=${PATH}:/opt/ignite/bin
```
for ksh

or

```
set_path = (${path} /opt/ignite/bin
```
for csh

D: Set Up or Update the Software Source

Ignite-UX allows many options for installing software on target systems. The basic option is to install all software from SD depots on the server. This step describes setting up the software to install on the server.

If you plan to use both SD sources and non-SD sources (tar, cpio, or pax), consider each individually:

For SD OS software

Follow these steps to make an SD source available to Ignite-UX:

Step 1. If you do not already have disk depots, create one using the `make_depots` command. For example, to create the necessary disk depots that correspond to the HP-UX 10.20 Core CD-ROM or the HP-UX DVD, enter:

```
make_depots -r B.10.20 -s /dev/dsk/c0t0d0
```

This assumes that the CD-ROM or DVD is connected at: `/dev/dsk/c0t0d0` and creates one or more depots in the directory: `/var/opt/ignite/depots/Rel_B.10.20`

Step 2. If you used `make_depots` as described above to create your depots, use the `make_config` command to create Ignite-UX config files for each of the depots you plan to use:

```
make_config -r B.10.20
```

This command will create config file for all depots found in the `/var/opt/ignite/depots/Rel_B.10.20` directory. It will also add these config files to all `INDEX` entries for the HP-UX 10.20 release. Skip the next step.

Step 3. If you did not use `make_depots` to create your depots, run `make_config` and point it at a specific depot. For example:

```
make_config -s server:/depot_700 \
-c /var/opt/ignite/data/Rel_B.10.20/core_700
```

Now add a reference to the `INDEX` file:

```
manage_index -a -f /var/opt/ignite/data/Rel_B.10.20/core_700
```

See the *ignite* (5) manpage for further examples.

For non-SD OS software

You will need to create a unique config file that represents the non-SD operating system software. Samples of config files that do a core archive can be found in: `/opt/ignite/data/examples/`

After copying this file and making edits to it as instructed in the comments contained in the file, you can use the `manage_index` tool to insert a reference to this configuration in the `/var/opt/ignite/INDEX` file. Use of configuration files is described in Chapter 3, “Using Configuration Files,” on page 49.

E: Add Optional Applications

If you have other SD-packaged software that you would like to have installed on target clients and want to have the software made available for selection in the Ignite-UX interface, run the `make_config` and `manage_index` commands on those depots.

For SD application software

Run the following commands for each depot you plan to load SD software from during the installation. The `make_config` command only handles SD software which is packaged in bundle form. (All HP-supplied software is packaged in this form. See the *make_bundles* (1M)manpage for information on making SD bundles in an SD depot.)

For example, to make compiler depot bundles available, as root enter:

```
/opt/ignite/bin/make_config\ -s hpfcxxx.hp.com:\
/depots/compiler \
-c /var/opt/ignite/data/Rel_B.10.20/compilers_cfg
```

```
/opt/ignite/bin/manage_index \
-a -f /var/opt/ignite/data/Rel_B.10.20/compilers_cfg
```

Replace the depot server name (in this example: `hpfc`*`xxx`*`.hp.com`) with the server you have the SD software on. Note that the depot server can be a different system from the Ignite-UX server.

TIP

Rerun the `make_config` command each time new software is added or modified in the depots.

The `make_config` command constructs Ignite-UX config files which correspond to SD depots. When an SD depot is used as part of the Ignite-UX process, it must have a config file which describes the contents of the depot to Ignite-UX. This command can automatically construct such a config file, when it is given the name of an SD depot to operate on. This command should be run when adding or changing a depot which will be used by Ignite-UX.

The `manage_index` command manipulates the `/var/opt/ignite/INDEX` file. This utility is primarily called by other Ignite-UX tools but can also be called directly.

For non-SD application software

If the source is not an SD depot, the `make_config` command is not applicable. You will need to create a unique config file that references the non-SD software. A sample of a config file that does a non-core archive can be found in: `/opt/ignite/data/examples/noncore.cfg`

NOTE

Do not attempt to use non-core OS archives (such as layered applications) that contain files that get loaded in: `/var/adm/sw/*` Delivering files in this directory in this method may corrupt the SD database.

Step 1. Copy this file to: `/var/opt/ignite/data/Release/configx` Then make the changes to the copy in that directory.

Step 2. After copying and editing this file, use `manage_index` to insert a reference to the copy of the configuration in: `/var/opt/ignite/INDEX`

F: Installing Minimal Ignite-UX Filesets

Depending on what you are using Ignite-UX for, you may be able to reduce the disk space usage by not loading the full product. Below is a list of typical usages and a list of what parts of Ignite-UX you need. If you are not concerned with disk space, just load the bundle(s) for the HP-UX releases you support.

For all cases, the `Ignite-UX.IGNT-ENG-A-MAN` fileset can be omitted or removed if you do not want on-line documentation.

- **Ignite-UX server to install HP-UX on clients**— Load the `Ignite-UX-`*`xx-yy`* bundle(s) for each HP-UX release (*xx-yy*) which you plan to install onto clients. You can omit the `Ignite-UX.OBAM-RUN` fileset if your server is HP-UX 11i or later and you don't plan on using `make_net_recovery` for HP-UX 10.*x* clients.
- **Ignite-UX server to support network recovery for clients**— You will need the full Ignite-UX-*xx-yy* bundle for each version of HP-UX that your clients are running.
- **Using only make_tape_recovery command:** — You only need these filesets:
  - — `Ignite-UX.RECOVERY`
  - — `Ignite-UX.BOOT-KERNEL`

`Ignite-UX.FILE-SRV-`*`release`* where:

`release` is the HP-UX release of the system you are running

— `Ignite-UX.MGMT-TOOLS`

- **Using only make_net_recovery on a client** — The filesets a client needs will normally be pushed out by Ignite-UX to each client from the depot created by the `pkg_rec_depot` command. These are the only filesets required for `make_net_recovery` on the client:

  ```
  Ignite-UX.RECOVERY
  Ignite-UX.MGMT-TOOLS
  ```

- **A network boot-helper system** — To setup a system on a remote subnet that is used just to allow a client to do a network boot and then contact a remote Ignite-UX server, all you need is `Ignite-UX.MinimumRuntime`. See "Setting Up the Boot-Helper" on page 262.

G: Start Ignite-UX for the First Time

To start Ignite-UX, as root enter:

`/opt/ignite/bin/ignite`

You will get a warning screen stating no clients exists as this is the first time that `ignite` has been invoked. This is normal since you do not have any clients waiting.

If you get this error message:

```
ERROR: This machine is not an NFS server (no nfsd running).

The -n option will not be processed.
```

the Ignite-UX server is not currently on the NFS server. The Ignite-UX server must be an NFS server. Exit Ignite-UX and make the Ignite-UX server an NFS server before continuing. You can do this by using SAM, or by editing `/etc/rc.config.d/nfsconf` to set `NFS_SERVER=1` and rebooting. If you do not get the above error, Ignite-UX has modified your `/etc/exports` file to include the `/var/opt/ignite/clients` directory:

```
exportfs -v

/var/opt/ignite/clients -anon=2
```

This directory is exported to allow remote root users to write to the client's directory. This is required for proper Ignite-UX operations. You may need to export additional directories. For example, if you use NFS to transfer your archive, it must be NFS accessible.

A quick tour

After you have Ignite-UX up and running, you will see the Welcome screen and then the Ignite screen. When you have booted each client you will see an icon representing it on the Ignite-UX screen:

- Click once on a client icon to select it for further actions.
- Double-click the client icon to get a Client Status screen.
- Right-click a client icon to get an Actions menu similar to the pull-down **Actions** menu.

To learn more about the server, step through the quick tutorial. To get started, select:

Actions –> **Run Tutorial/Server Setup** and click **Tutorial and Demo**

For more information on these screens, see Chapter 4, "Installing HP-UX with Ignite-UX on Clients Locally," on page 63.

H: Set an Initial Ignite-UX Server Configuration

Follow these steps to complete the initial server configuration:

Step 1. Select: **Options** –> **Server Configuration**

Step 2. Select the **Server Options** tab.

If needed, modify the Server Options to match the following:

- Default Configuration: (your selection)
- Default Printer: (select a default printer to be used by Ignite-UX)
- Client Timeouts: `40` (the number of minutes delay before the Ignite-UX server will inform the administrator of a network problem or client failure)
- Run client installation UI on: `server` (most administration of the install process to be performed only on the Ignite-UX server)

Step 3. Select: **Add Booting IP Address**

During the install process, the clients need to do a network boot from the Ignite-UX server. In order to do this, the clients need to be given a temporary IP address.

Under **Booting Clients**, enter an initial range of available IP addresses. This example allows Ignite-UX to perform 20 simultaneous installations:

```
15.2.73.1       15.2.73.20
```

This IP address is only used when booting over the network during the initial transfer of the kernel to the client. You may only need one or two addresses depending on how many systems do network boots at the same time. For more information see the *instl_bootd* (1M) manpage. If you need to change these addresses later, you will need to edit: `/etc/opt/ignite/instl_boottab`

Permanent IP addresses are distributed via DHCP Services.

Unless you are familiar with DHCP services, for this exercise, do not modify the “DHCP Class ID” or the “DHCP Addresses Temporary” field. The DHCP service is only used for client configurations which do not have predefined system hostnames and IP addresses.

Provide a range of available "permanent" IP addresses. These can only be supplied once here in Ignite-UX. After the initial definition, use SAM’s **Networking and Communications** –> **Bootable Devices** area. For example, we use these IP addresses in our network:

```
15.2.73.21      15.2.73.40
```

Step 4. Select: **Options** –> **Server Configurations** –> **Session Options**

Verify that only these options are set:

```
Confirm new clients
Show the welcome screen for the install server
```

You may wish to de-select **Ask for customer information**, as this installation information is geared to HP and HP distributor-partner manufacturing.

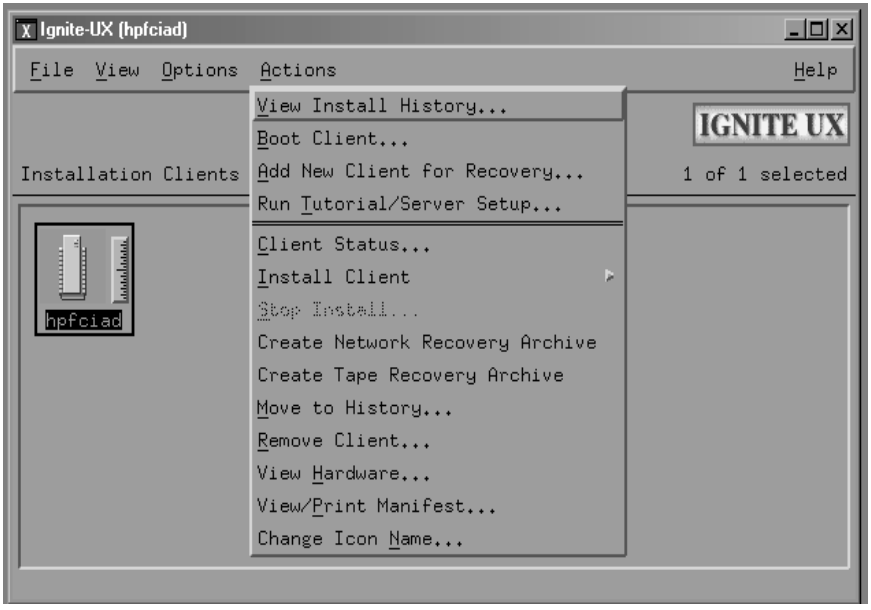
I: Starting Ignite-UX

To start `ignite` on the Ignite-UX server, as root enter:

```
/opt/ignite/bin/ignite
```

Client/server screen

After the Welcome screen is acknowledged by clicking **OK**, Ignite-UX displays its client/server screen as in the following:



Ignite-UX displays each system's installation status via the colored border around each system icon:

- **Green** — OS completely installed, booted and running.
- **Red** — Partially installed or installation stopped. The light blue installation indicator shows the relative progress.
- **No color** — OS not installed.

Client icons represent all booted systems and those systems that can be used for recovery systems. These systems are known to Ignite-UX via `/var/opt/ignite/clients`. If a client is not yet running an OS, see the booting procedure at the end of this chapter. If the client is already

running an OS, this can be accomplished remotely by selecting **Actions -> Boot Client**

Actions menu

Select a client (click its icon) and select the **Actions** menu to review available actions for that client:

- **View Install History** — Lists details of all successfully installed clients.
- **Boot Client** — Allows you to boot the selected client.
- **Add New Client for Recovery** — Allows you to identify a client to be recovered.
- **Run Tutorial/Server Setup** — Displays the Welcome screen and you can choose to run the **Tutorial and Demo** or **Server Setup** options.
- **Client Status** — Allows you to see the status of a given client, see “Review client status” on page 39 for more information.
- **Install Client** — Starts the HP-UX installation process for the selected client. This process is explained in Chapter 5, “Installing HP-UX with Ignite-UX on Clients from a Server.”.
- **Stop Install**— Stops the install process on the selected client. You can now reboot or halt the client.
- **Create Network Recovery Archive** — Initiates creating a network recovery archive using the `make_tape_recovery` command. See Chapter 11, System Recovery, for more details.
- **Create Tape Recovery Archive** — Initiates creating a recovery archive using the `make_tape_recovery` command. See Chapter 11, System Recovery, for more details.
- **Move to History** — Saves critical files for the client, adds them to the history file and removes the client icon. The client must be "complete" (fully installed) for the configuration to be moved to the history file.
- **Remove Client** — Deletes the icon for the selected client configuration. Client data except for the recovery archive is removed.
- **View Hardware** — Lists hardware associated with the selected client.

- **View/Print Manifest** — Allows you to see or print the manifest and/or Software Certificate. The manifest is also available in saved form on the client and server systems after the installation as the manifest files. On the client, the manifest is in:`/var/opt/ignite/local/`

  On the server, it is in: `/var/opt/ignite/clients/0x`*`LLA/`*

  For an example, see "Viewing and Printing a Manifest" on page 104.

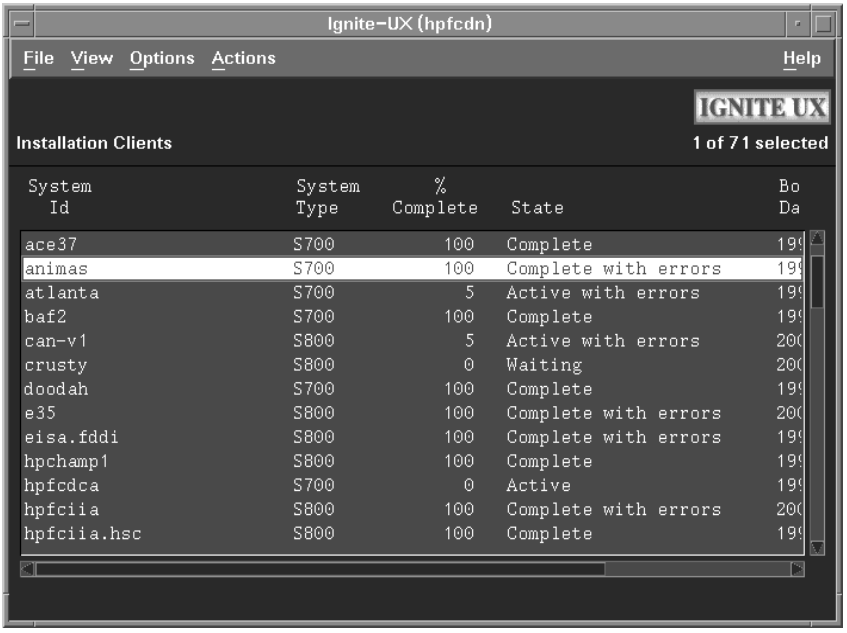
- **Change Icon Name** — Displays a form for renaming the icon for the selected client.

View menu

Use the **View** menu selections to customize the Ignite-UX screen for your needs:

- **Columns**— Re-arrange icons by system attributes.
- **Filter** — View a selected subset of system icons per selected criteria.
- **Sort**— Sort the displayed icons per selected criteria.
- **By Properties** — List clients in a text format rather than with icons. To return to the default icon display, select: **View** –> **By Name and Icon**.

  Using the **By Properties** view along with sorting by **% Complete** can make it easier to quickly scan for clients that have finished installing. Select **Descending Direction** to have all completed systems listed at the top of the display. Here's a portion of a **By Properties** view:



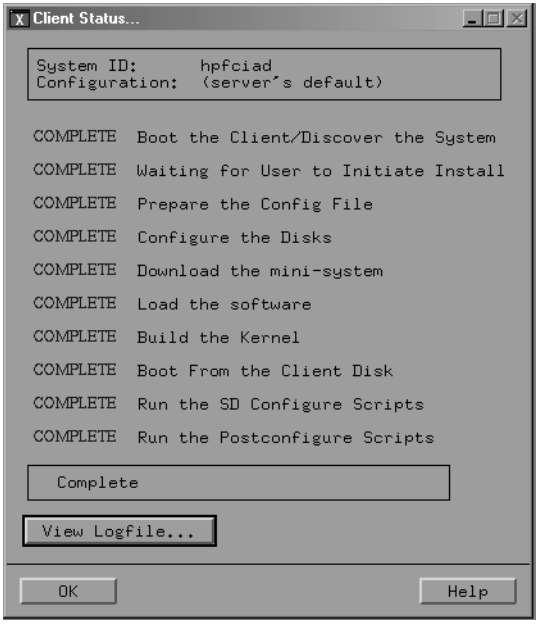
Review client status

After you see client systems displayed on the Ignite-UX screen, you can review the status of any client by:

Step 1. Click once on a client icon to select it for further actions.

Step 2. Double-click the client icon to get the **Client Status** screen, or select Client Status from the Actions menu, or *right-click* a client icon to get a menu similar to the pull-down **Actions** menu.

Any of these actions result in the status of the client being polled and displayed as in the following example:



J: Configuring Server and Session Options

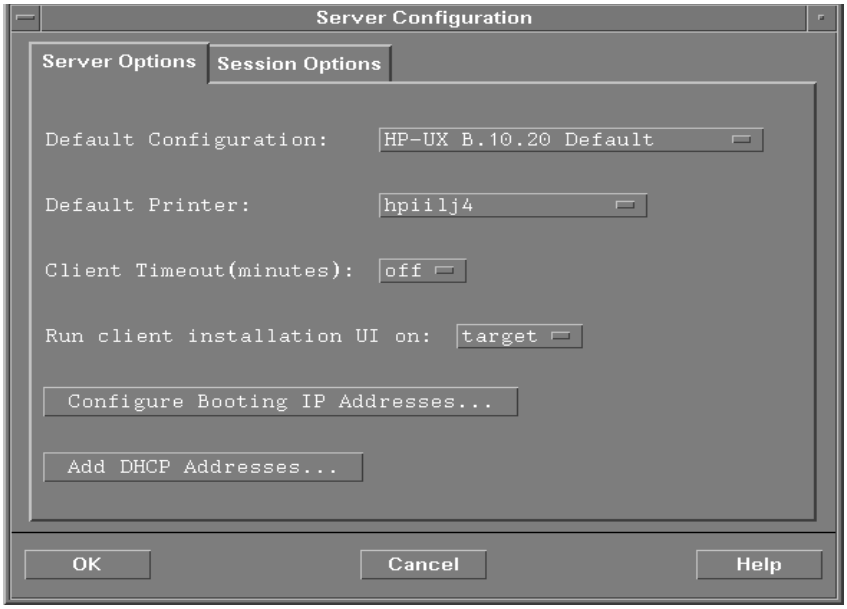
The Ignite-UX server and session options must be configured as described in this section.

Use fields in the **Options** –> **Server Configuration** –> **Server Options** and **Session Options** tabs to:

- Set up your network installation **Precision Architecture Reduced Instruction Set Computing** (PA-RISC or PA)-based or IPF-based server. Network installation details when using Ignite-UX versions B.4.1 and B.4.2 are found in “Release Specific Server Configuration” on page 46.
- Configure the IP addresses to be used for initially booting the install clients (target systems).
- Configure the DHCP address range to be used for directing the client installation process.

Configuring server options

Step 1. Select **Options** –> **Server Configuration**



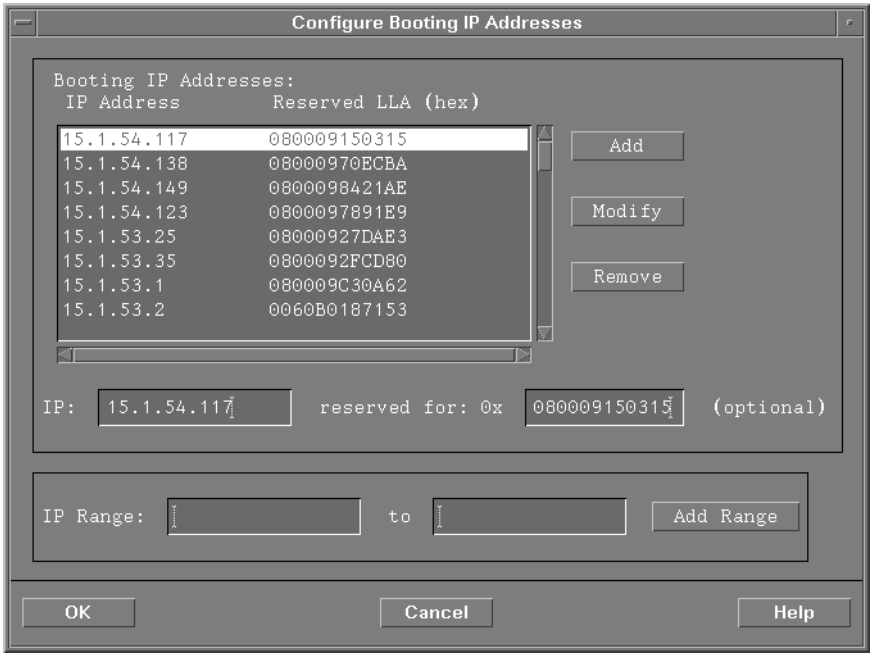
Step 2. Select the **Default Configuration** box and highlight the OS or OE you want. The selected configuration from this list will be installed on to the client's target system. The default setting can be overridden on a per-client basis by Ignite-UX.

Step 3. Click on the **Default Printer** pull-down menu to display the available (configured) printers. Select the one you want to use. If needed, use the **System Administration Manager** (SAM) **Default Printer** area to configure a new printer onto the system. This will be the printer for printing the manifest or installation history. The printer IP address will be checked by Ignite-UX before a job is sent.

Step 4. Select the appropriate **Client Timeout** value, **on** or **off**. This will set a limit on the time since the client install log has been written into. Some points in the installation may require 15 to 30 minutes. A warning note will be displayed if this time is exceeded.

Setting Client Timeout to **off** disables this notification.

Step 5. Use the **Run client installation UI on** menu to designate where you want to view the client UI for this installation. If you have a server configured, you have the choice of running the client installation interface from either the **target** (as a TUI) or **server** (as the Ignite-UX screen). If the client installation is to be non-interactive (no user intervention), select **none**.



The default location for the GUI to be displayed is the Ignite-UX server.

Step 6. If you are using Ignite-UX version B.4.0 or earlier, you can configure which Ignite-UX servers are used to boot client servers in two ways using the GUI: by identifying IP or DHCP addresses. Select one of the following methods:

a. Click **Configure Booting IP Addresses**
Enter the appropriate IP addresses for the initial boot of the target systems. The number of such addresses determines the number of simultaneous boots you can do.

TIP

Be sure that IP addresses entered here are not assigned elsewhere, or you could (re)boot the wrong system.

These IP addresses are used to initially boot target systems. They are used until the system is assigned one of the DHCP boot addresses. One address is required for each simultaneous boot. Typically one to three are needed, depending on your usage.
This data can also be configured by using the `/opt/ignite/lbin/setup_server` command. Or, you can directly edit the `instl_boottab` file; this is necessary for modifying the list order for existing IP addresses.
See the *instl_bootd* (1M) manpage for further details.

Or

b. Click **Add DHCP Addresses** Ensure that the listed IP addresses are not assigned elsewhere. These IP addresses are used during the OS download and application loading. The addresses are in use for most of the Ignite-UX download to a target machine. One address is required for each simultaneous download. You should set more, if the addresses are assigned permanently.

   Click the **Temporary** box if you would like to manage a small group of temporary IP addresses, just for use in doing installations, and then reassign the clients new addresses when they are deployed.

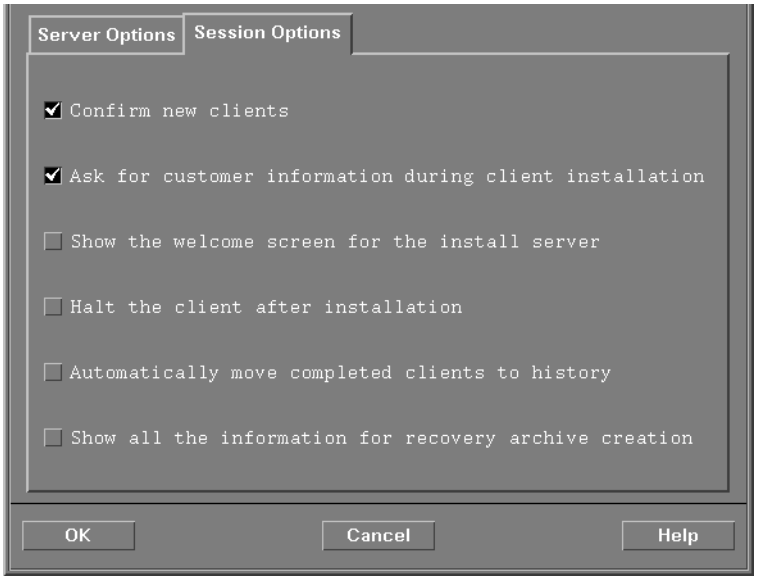
   The provision of DHCP capability is for the purpose of installation only and you may want to limit configurations so that they do not interfere with prior DHCP server functions.

   See Appendix C, Configuring for a DHCP Server, for details on configuring for DHCP. See the *setup_server* (1M) and *instl_adm* (4) manpages for more information on setting up DHCP functions, addresses and class IDs.

Configuring session options

To configure client response behavior, select:

Options –> **Server Configuration** –> **Session Options**



- **Confirm new clients** — Controls the appearance of a dialog window each time a new client is booted from the Ignite-UX server.
- **Ask for customer information during client installation** — Controls the appearance of an input window for Customer Name, System Serial Number, and Order Number. You may want to refrain from using this option as this information is geared to HP and HP distributor-partner manufacturing.
- **Show the welcome screen for the install server** — If selected, Ignite-UX automatically displays the Welcome screen. This is a useful default if many new operators run Ignite-UX.
- **Halt the client after installation** — Controls whether the client system is halted (rather than the default, reboot) after installation.
- **Automatically move completed clients to history** — Select this button to automatically add completed clients to the end of the history log, `/var/opt/ignite/clients/history/history.log`. It will also move their config and manifest files to the history directory on the

Ignite-UX server for future reference. The client icon will be removed from the client/server screen. The client must be `complete` (fully installed) for this to take place.

- **Show all the information for recovery archive creation**.

Your Ignite-UX server is now ready to ignite HP-UX on client systems in your network.

Proceed to Chapter 4, Installing HP-UX with Ignite-UX on Clients Locally, or to Chapter 5, Installing HP-UX with Ignite-UX on Clients from a Server, depending on where you want to execute the Ignite-UX process.

Release Specific Server Configuration

With the release of Ignite-UX B.4.1, unique server configurations have become necessary. The server specific configurations described in this section are considered cumulative with each release unless specifically stated otherwise.

Follow the specific server configuration that is appropriate for the Ignite-UX version you are installing:

Ignite-UX B.4.2 PA/IPF Server Setup

The Ignite-UX B.4.2 and later releases provide enhanced support for server response to anonymous IPF clients with changes to the `instl_bootd` server. This enhancement is available for both PA and IPF server architectures.

The changes to `instl_bootd` requires that the `bootpd` daemon is not running on the given Ignite-UX server, rather the `instl_bootd` daemon is used by Ignite-UX to respond to all boot requests from clients. The `instl_bootd` daemon normally runs on a set of unique network ports, 1067/1068, that are used only for booting IPF clients. However, in this implementation the `instl_bootd` will run on the standard `bootpd` ports, 67/68.

Using `instl_bootpd` to support anonymous IPF clients

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

Step 1. Set up your Ignite-UX server as described in “Installing an Ignite-UX Server” on page 27.

Step 2. Once your server has been setup, ensure that `bootpd` is disabled on ports 67/68 by commenting out the following line in `/ect/inetd.conf` as shown in this example:

```
#boots dgram udp wait root /usr/lbin/bootpd bootpd
```

Step 3. Restart the `inetd` daemon:

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

Step 4. Enable the `instl_bootd` daemon on ports 67/68 by adding the following line to `/ect/inetd.conf`:

```
boots dgram udp wait root /opt/ignite/lbin/instl_bootd
instl_bootd
```

Step 5. You must restart `inetd` again to invoke the change made in the previous step:

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

See the *instl_bootd* (1M) and *inetd* (1M) manpages for more details. Your Ignite-UX server is now configured to respond to anonymous clients.

Ignite-UX B.4.1 IPF Server Setup

Ignite-UX release B.4.1 and later support the installation of IPF systems running HP-UX 11i Version 1.6.

Configuring DHCP support for anonymous IPF clients

Network installation of an IPF system with Ignite-UX B.4.1 requires that you perform the following unique network installation steps.

Step 1. At a minimum, the Ignite-IA-11-22 bundle should be loaded on your system. If it is not, load this bundle with `swinstall`.

Step 2. Add your client's entries to `/etc/bootptab` on the server. The following example is provided in `/etc/bootptab` :

```
ignite-defaults:\
        ht=ethernet:\
        hn:\
        bf=/opt/ignite/boot/nbp.efi:\
        bs=48
System-IPF:\
        tc=ignite-defaults:\
        ha=00d009000000:\
        ip=190.40.101.22:\
        sm=255.255.248.0:\
        gw=190.1.48.1:\
        ds=190.1.48.11
```

All entries in the `ignite-defaults` section can be used without modification.

Duplicate (cut and paste) the `System-IPF` entries and change this section in the following ways:

1. Change `System-IPF` to reflect the client system's hostname.
2. Change the `ha` and `ip` fields for that client.
3. Change the `sm`, `gw`, and `ds` fields to reflect your network.

You can modify the system name, hardware address, IP address and other information for the client.

The following describes fields you may need to change per system and which fields are unique to your network:

- The **`ha`** field requires setting to the hardware address (`Mac` or *`LLA`* address of the client system). This address can be found from the firmware's user interface when adding a boot option. See , "IPF Client Network Booting Option" on page 70 of Chapter 4. If the system is up and running, the `lanscan` command can also be used to find this value.
- The **`ip`** field is the IP address that has been reserved for the client you are about to install and must be an IP address that is valid for your network.
- The **`sm`** field is the network subnet mask and is probably the same for all systems on your network.
- The **`gw`** field is the network gateway. It is optional for booting purposes, but useful to provide the system defaults.
- The **`ds`** field is the domain name server (DNS) and is also optional for booting purposes, but useful to provide as a default.

Step 3. Enable `bootp` services in the `/etc/inetd.conf` file by uncommenting the `bootps` entry.

Step 4. Restart the Internet daemon by entering:

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

See the *bootpd* (1M) and *inetd* (1M) manpages for more details. Your Ignite-UX server is now configured to respond to anonymous clients.

3 Using Configuration Files

This chapter introduces Ignite-UX configuration files and shows example config-file applications. See Chapter 9, “Automating Installations,” on page 135 for more examples of config files.

config Files

Ignite-UX's central data store is called a **config file**. A config file can be thought of as a recipe for how to construct a target system. The config file is expressed in a language designed for this purpose. The language is fully defined in the *instl_adm* (4) manpage. The syntax is human-readable; config files may be created directly by you or via the Ignite-UX screen. The config file language is much like programming languages in that it supports the use of variables and conditional expressions.

Most of the important elements which make up an installed system can be described in the config file:

- Disk and file system layout.
- Software to be installed.
- Target system's identity and network-configuration kernel modifications (additional drivers or tunable parameter settings).
- User-defined scripts which will run at various points in the installation process to further customize the target system.

Types of config Files

For maintenance convenience, the configuration information is split into several types of config files:

- **Default disk and file system layout** — Because the capabilities of each operating system release differ somewhat, HP supplies a different set of defaults for each release. These are located in: `/opt/ignite/data/Rel_`*`release`*`/config`

  where: *`release`* is the result of the **`uname -r`** command. For example, this file contains the default disk layout for HP-UX 10.20: `/opt/ignite/data/Rel_B.10.20/config`

- **Software description of a single SD depot** — Config files which describe software available from SD depots can be automatically generated via Ignite-UX's `make_config` tool. This tool produces one config file per SD depot. Software description config files are located in: `/var/opt/ignite/data/Rel_`*`release`*`/*`

- **Software description of an archive** — Config files can be hand built to allow access to archives (templates are provided with Ignite-UX in `/opt/ignite/data/examples/` to give you a good starting point). Archives may be in either tar or cpio format. Archive software description config files are also located in: `/var/opt/ignite/data/Rel_`*`release`*`/*`
- **Local configuration overrides that apply globally** — It is often convenient to specify defaults which will be applied to every system installed from a particular server. For example, you might want to specify the same NIS domain for all systems. Place overrides in: `/var/opt/ignite/config.local`
- **Boot control parameters and networking information** — It is possible to specify defaults for attributes like the IP address of the Ignite-UX server and whether to run a UI to install a new target. These can be specified in the first 8KB of the install file system, `/opt/ignite/boot/INSTALLFS` . Use the `instl_adm` command to add, change, or delete this information.
- **Client-specific configuration files** — Each client to be installed has a unique configuration file located at: `/var/opt/ignite/clients/0x`*`LLA`*`/config`

  LLA is the link-level address of the client. This file is typically created when using the Ignite-UX user interface to specify the target system configuration.

  This file usually refers to other config files mentioned above. It also contains specific directives to override what may have been defined in the other files. For example, you may wish to customize the disk layout beyond what the defaults allow in: `/opt/ignite/data/Rel_`*`release`*`/config`

  The customizations appear in: `/var/opt/ignite/clients/0x`*`LLA`*`/config`
- **Named configurations created by saving a configuration via Ignite-UX screen**— You can create your own default configurations via the Ignite-UX screen and save them for future use. For example, you might have a large number of users with similar systems who all run CAD tools. You could build a configuration which matches what they need and save it in a configuration called "CAD System". When you need to install a new system of this type, you can select **CAD**

System from the UI and you're done (or you could customize it further using CAD System as the template). Saved configurations are located in: `/var/opt/ignite/saved_cfgs/*`

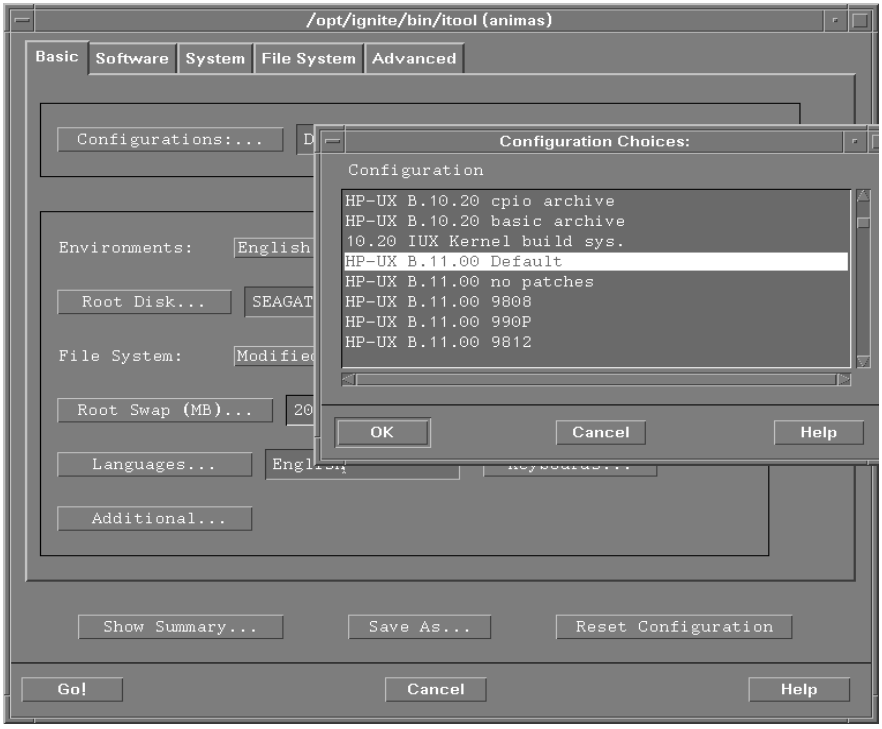
You can build your own config files to specify a particular building block you are interested in, and then combine them in arbitrary ways. Place these building block config files in:
`/var/opt/ignite/data/Rel_`*`release`*`/*`

The next section describes how multiple config files can be combined to define a single configuration.

Combining config Files via INDEX Entries

The grouping of config files into useful configurations is accomplished in `/var/opt/ignite/INDEX` . This file contains a list of valid configurations, each of which is made up of one or more config files. You can view these configurations from the Ignite-UX GUI when installing a new client at the top of the **Basic** tab by selecting:

Actions –> **Install Client -> New Install**



For example, the INDEX file might contain:

```
cfg "HP-UX B.10.20 Default"  {
     description "This selection supplies the default system
       configuration that HP supplies for the B.10.20 release."
     "/opt/ignite/data/Rel_B.10.20/config"
     "/var/opt/ignite/data/Rel_B.10.20/core_700_cfg"
     "/var/opt/ignite/data/Rel_B.10.20/core_800_cfg"
     "/var/opt/ignite/data/Rel_B.10.20/apps_700_cfg"
     "/var/opt/ignite/data/Rel_B.10.20/apps_800_cfg"
     "/var/opt/ignite/data/Rel_B.10.20/patches_700_cfg"
     "/var/opt/ignite/data/Rel_B.10.20/patches_800_cfg"
     "/var/opt/ignite/config.local"
}
cfg "CAD System-10.10" {
     description "This selection is the typical CAD system

     installation for HP-UX B.10.10"
     "/opt/ignite/data/Rel_B.10.10/config"
     "/var/opt/ignite/data/Rel_B.10.10/core_700_archive_cfg"
     "/var/opt/ignite/data/Rel_B.10.10/apps_700_cfg"
     "/var/opt/ignite/data/Rel_B.10.10/patches_700_cfg"
     "/var/opt/ignite/config.local"
} = TRUE
```

With this INDEX file, the Ignite-UX would present two configurations: **HP-UX B.10.20 Default** and **CAD System-10.10**. The **CAD System-10.10** configuration is the default (it is marked TRUE). Once you choose one of these base configurations, you can do further customizations with the UI or accept the config defaults and do the install immediately.

If you selected **CAD System-10.10**, you would get the combination of the five config files listed for that clause. The order of the config files is significant; attributes specified in a later config file can override the same attributes specified in an earlier config file. There are also two config files which are implicitly used every time. Any information stored in the first 8KB of /opt/ignite/boot/INSTALLFS is implicitly appended to each configuration. The client-specific configuration file /var/opt/ignite/clients/0x*LLA*/config is implicitly added as the last config 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 INDEX file directly.
- Use the manage_index file to help automate INDEX file modifications.

Example Config Files

This section shows a few example config files to give you an idea of their look and capabilities. See the *instl_adm* (4) manpage for a complete description of Ignite-UX config files.

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 LVM or VxVM. The disk contains the `/` file system and a swap area. The swap area takes up 64 MB and the file system takes up whatever space is left over:

```
partitioned_disk {
    physical_volume disk[2/0/1.6.0]

    fs_partition {
        mount_point ="/
        usage=HFS
        size=remaining
        file_length=long
    }
    swap_partition {
        usage=SWAP
        size=64Mb
    }
}
```

Combining disks to form a single volume group

In this example, two disks are put together to form a single volume group. Two file systems are defined; both are striped across both disks. The first file system, `/apps1`, is sized by calculating the amount of space required by the software which is to be loaded, and then adding a 30% free-space cushion. The second file system, `/apps2` , gets the remaining space on the disks.

NOTE

The following example shows LVM as the volume manager. However, it is also applicable to VxVM if `usage=LVM` is changed to `usage=VxVM`.

```
volume_group “appsvol” {
    usage=LVM
    physical_volume disk[2/0/1.5.0] {
    }
    physical_volume disk[2/0/1.4.0] {
    }
    logical_volume “apps1”

         size=30% free
         usage=VxFS
         mount_point=/apps1
         minfree=5
         stripes=2
    }
    logical_volume “apps2” {
        mount_point= “/apps2”
        usage=VxFS
        size=remaining
        minfree=5
        stripes=2
    }
}
```

Defining networking

This example defines a few of the network parameters which will be assigned to the system after it has been installed:

```
final system_name = "acorn1"
final ip_addr["lan0"] = "15.99.45.123"
final netmask["lan0"] = "255.255.248.0"
final nis_domain = "nis1"
final route_gateway[0] = "15.99.45.1"
```

Defining an install depot

This 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 impact 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 allows you to group the software so that the user interface can present it in chunks which make sense to you. Since this example references an SD depot, it would have been created by `make_config`:

```
sw_source “ee_apps_depot” {   description = “Electrical Engineering
Application”   source_format = SD
   source_type = “NET”
   sd_server = “15.23.45.6”
   sd_depot_dir = “/var/opt/ignite/depots/Rel_B.10.20/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.10.20_700”
   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.10.20_700”
   impacts = “/usr” 12568Kb
   impacts = “/var” 26788Kb
}
```

Customizations Based on the Target System

The config file syntax provides a large number of system attribute keywords which describe the target system. Some examples are:

- The size of the disk at the specified *hw_path*:

```
disk[hw_path].size
```

- The amount of memory present on the target system:

```
memory
```

- The string returned from **uname -m**:

```
hardware_model
```

- The link-level address of the target system:

```
lla
```

System attribute keywords can be used in expressions in config files so that a particular clause is only included in specific target 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 some kernel tunable parameters if the target system has more than 256 MB of memory:

```
(memory > 256Mb) {
    mod_kernel += "nproc (20+12*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 target system has the appropriate hardware:

```
(graphics[0].planes > 0) {
    post_config_script +=
        "/var/opt/ignite/scripts/multi_plane_graphics"
}
```

You can also specify multiple conditions. This example installs a particular piece of (previously defined) application software if the target system is a workstation (Series 700) having at least two disks. A message lets the user know why it is happening:

```
( (hardware_model ~ "9000/7.*") & (num_disks >= 2) ) {
    note += "Installed apps1 because this is a big series 700."
    init sw_sel "apps1" = TRUE
}
```

It is also possible to add an `else` clause. This example uses a generic variable capability and mathematical expressions to set the primary swap size based on the amount of memory in the target system:

```
(memory > 512Mb) {
    init _hp_pri_swap = 512Mb
}
else {
    init _hp_pri_swap = memory * 2
}
```

Customizations Based on User Selection

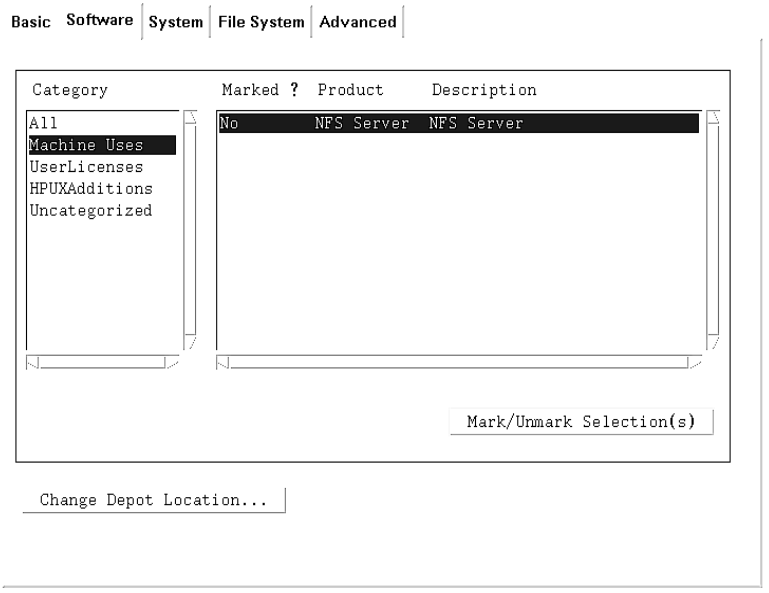
It is sometimes advantageous to be able to select particular customizations independent of the target system's hardware setup. For example, you might have some systems which you intend to use as NFS file servers. You would like the ability to select NFS server capability from the UI when you are configuring the target system.

You have found that NFS file servers are more efficient if some of their kernel parameters are modified. NFS file servers also require some changes to the `/etc/rc.config.d/nfsconf` file via ch_rc.

One solution is to define a custom software selection with a `sw_sel` clause, which Ignite-UX shows on the **Actions** –> **New Install** –> **Software** tab when you are configuring a new installation. For example:

```
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"
}
```

The next figure shows the **Software** tab when the NFS server config file is used. As shown, the selected category is `Machine Uses` as defined in the config file. Choosing a different category would display a different set of software. If you were to select `NFS Server` from this screen, the kernel modifications specified in the config file would be applied during the installation. Likewise, the `ch_rc` commands specified in the config file will be run as part of the installation.



Using install tabs to configure client installations is explained in Chapter 5, “Installing HP-UX with Ignite-UX on Clients from a Server.”.

Debugging config Files

Designing a config file to meet your needs can be a very tedious task. It usually requires a lot of trial and error. Beginning with Ignite-UX version A/B 2.2.4 (May 2000), the `instl_dbg` command is available to help you with config file design. With the `instl_dbg` command you can:

- Parse a client's configuration files for syntax errors.
- Place all relevant configuration information into one file for review.
- Display and set variables, software selections, and use models.
- Detect any other 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 config file, run `instl_dbg` from the Ignite-UX server to parse the specified client's config file as well as any of the server's configuration files referenced by the client's config file. `instl_dbg` first scans for any syntax errors. After syntax is checked, `instl_dbg` substitutes variables, use models, and software selections (`sw_sel`) with real values, and writes a single, unified config file if the `-f` option is specified. You can then compare this file with your original to determine required changes, or use this file as is to install the client. More options are available for more thorough checking or to provide more detail.

Example uses

To debug a client system1 config file and print the debugged config file to stdout and save the debugged config file to `system1_cfg.out`:

```
instl_dbg -D /var/opt/ignite/clients/system1 -d -f
system1_cfg.out
```

Debug the config file for the client named system1, show the effects upon the disk layout when the value of `_hp_disk_layout` and `_hp_pri_swap` are changed, and print the "very, very verbose" (`-vvv`) output to the screen as well as the verbose output to `system1_cfg.out`:

```
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
```

Additional examples can be found in the *instl_dbg* (1M) manpage.

4 Installing HP-UX with Ignite-UX on Clients Locally

You can install the client locally by **pulling** the HP-UX operating system from an Ignite-UX server in **terminal user interface** (TUI) mode, as explained in this chapter.

For multiple target installations, you will generally be executing the installation from an Ignite-UX server. Ignite-UX allows you to install one or more client systems manually from the Ignite-UX screen as explained in Chapter 5, from a remote system by using `bootsys` which is also in Chapter 5, or automatically as explained in Chapter 10. These are called **pushing** installations.

Both installation methods, pushing or pulling, require that a configuration (config) file be created, as explained in Chapter 3. The configuration can include any supported HP-UX 10.*x*, 11.0, or 11i OS, plus any required patches and applications.

This chapter discusses the steps for installing HP-UX on client systems locally. Topics are:

- Preparing Clients for Installation.
- Non-Interactive Installation Using bootsys.
- Booting Client Systems from the Network.

Preparing Clients for Installation

Boot each Series 700 or Series 800 client system that supports network boot either by entering the appropriate command explained in the following pages or by using the Ignite-UX screen. If a client with a known IP address is already running HP-UX, you can use the `bootsys` command (see page 65) from the Ignite-UX server to install a specific configuration without further interaction.

TIP To interrupt the boot process on any HP computer system, press **Esc** on the given system.

The next section provides a brief review of the manual boot process. Boot ROM commands for manual booting are explained in the installation guide supplied with the HP-UX OS/OE media:

- *Installing and Updating HP-UX 10.x, Chapter 3.*
- *HP-UX 11.0 Installation and Update Guide*
- *HP-UX 11i Installation and Update Guide*

If the client cannot find the server to boot from, check these items:

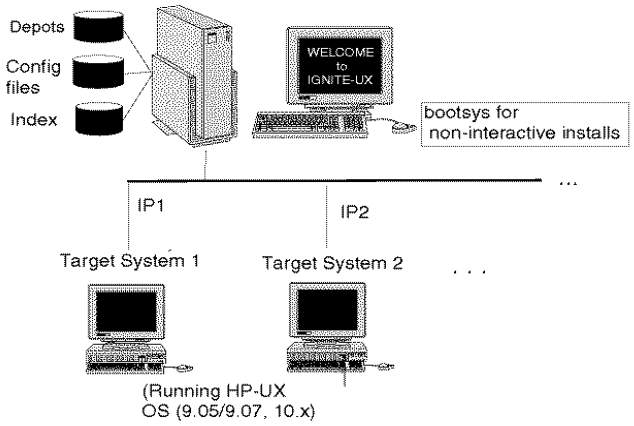
- Client is on the same subnet as the server.
- Any `instl_bootd` errors in: `/var/adm/syslog/syslog.log`
- The `/var/adm/inetd.sec` file to make sure that IP address `0.0.0.0` is not being disallowed.
- If `/etc/services` comes from NIS, make sure that the NIS server has `instl_boot*` entries.

The icons for all clients booted from the Ignite-UX server should now appear on the Ignite-UX client/server screen. If the Ignite-UX server has not been set up completely, or if the client could not obtain enough networking parameters via DHCP, then the client may require interaction on the Ignite-UX client/server screen.

Now that you can view clients on the Ignite-UX client/server screen, you can proceed to Chapter 5.

Non-Interactive Installation Using bootsys

You can use `bootsys` to start an interactive system installation on one or more clients without logging onto the client system, as illustrated in the following diagram.



It can be invoked either from a command shell, or from the Ignite-UX screen by selecting:

Actions –> **Boot Client**

Each client must:

- Be currently booted under HP-UX 10.20 or later.
- Be accessible on the network.
- Have enough disk space in the `/stand` directory to hold these files:

  `/opt/ignite/boot/INSTALL`

  `/opt/ignite/boot/INSTALLFS`

`bootsys` copies the Ignite-UX kernel and RAM file system 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.

Examples

This sample command will boot the client system from the boot1 server and wait for install instructions from the Ignite-UX server:

```
bootsys -w boot1
```

If you have already run an install session from the server, issuing `bootsys` without the `-w` option results in automatic installation without further intervention.

To automatically install `system1` using a different IP address than what is currently assigned and without waiting for server interaction, use this command:

```
bootsys -a system1:1.2.3.45
```

More information...

See the *bootsys* (1M) manpage for more information and examples. Common problems using `bootsys` with Ignite-UX are covered in Appendix A, “Troubleshooting,” on page 233.

TIP

To prevent a critical client from being inadvertently booted via bootsys, create the file, `/.bootsys_block`. For example, you can create the file with:

```
touch /.bootsys_block
```

Booting Client Systems from the Network

NOTE Network boot applies to HP Workstations and HP servers except D, K, and R-class servers that do not support the remote network booting feature. For more details on supported systems, see "Ignite-UX Hardware Requirements" on page 25. See Appendix A "Booting older workstations" on page 239 for more information.

This section provides an overview of booting HP computer systems if you have HP computer systems that may not be running HP-UX.

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

```
BOOT ADMIN> help boot
```

If the client system is already running an OS, use this procedure or the `bootsys` command as described in the previous section.

Step 1. Determine your network server address for the install. If necessary, see your system administrator for this information.

Step 2. Turn the power ON for the target system.

Step 3. When you see a message about stopping the boot search, quickly press and hold **Esc** down to stop the boot-selection process.

Booting Current Workstations and Servers

After the power is turned on, you will see a GUI screen (workstations) that displays instructions to press **Esc** to stop the boot process. (On servers, the TUI is used.)

Step 1. Press **Esc** to view the BOOT ADMIN menu:

```
Command                                 Description
-------                                 -----------
Auto [boot|search] [on|off]             Display or set auto flag
Boot [pri|alt|scsi.addr] [isl]          Boot from primary,alternate or
SCSI
Boot lan[.lan_addr] [install] [isl] Boot from LAN
Chassis [on|off]                        Enable chassis codes
Diagnostic [on|off]                     Enable/disable diagnostic boot
```

```
mode
Fastboot [on|off]                    Display or set fast boot flag
Help                                 Display the command menu
Information                          Display system information
LanAddress                           Display LAN station addresses
Monitor [type]                       Select monitor type
Path [pri|alt] [lan.id|SCSI.addr]    Change boot path
Pim [hpmc|toc|lpmc]                  Display PIM info
Search [ipl] [scsi|lan [install]]    Display potential boot device
Secure [on|off]                      Display or set security mode
------------------------------------------------------------
BOOT_ADMIN>
```

Step 2. If your network only has one Ignite-UX server available, enter:

```
BOOT ADMIN> boot lan install
```

Step 3. Otherwise, to make sure you boot from the correct server, either make the system search for servers and pick one or explicitly tell the system where to boot, as follows:

1. To search for servers type the following (workstations only):

   ```
   BOOT ADMIN> search lan install
   ```

2. The list of servers will be displayed with IP addresses. You may need to run the `nslookup` command on another running system to determine which address corresponds to your Ignite-UX server, if this information isn't already available.

3. Once you know the IP address of your server (as provided by the search or the `nslookup` command), boot the system by entering:

   ```
   BOOT ADMIN> boot lan.nn.n.nn.n install
   ```

   where: *nn.n.nn.n* is your server's IP address.

   The system then begins to load the install kernel from the network server. This should take 3 to 5 minutes.

Booting Older Series 700 Workstations

On older Series 700 systems, you will eventually see this menu:

```
b)    Boot from specified device
s)    Search for bootable devices
a)    Enter Boot Administration mode
x)    Exit and continue boot sequence
?)    HelpSelect from menu:
```

Do one of the following:

- If your network has only one install server and your system is not configured as a diskless client, then type:

  `boot lan`

  The boot may fail the first time because of an intentional delayed response by the install server. If it fails, try it again. If it fails more than three times, check for problems on the install server (see Appendix A, Troubleshooting,) *OR...*

- If your network has multiple install servers, make sure you boot from the network server address specified by your system administrator.

Search for servers

Step 1. Enter: `BOOT ADMIN>` **`search lan`**

Step 2. If your Ignite-UX server does not appear during the search, exit by entering: **`x`**

1. If necessary, enter the search command again:

   `BOOT ADMIN>` **`search lan`**

TIP It typically takes two or three searches before the Ignite-UX server will be found, due to a built-in delayed response from the server system.

2. Identify your LAN server from the listing.
3. If three attempts result in no response from the desired server, see Appendix A, “Troubleshooting,” on page 233.

Step 3. If you know the Ethernet™ address of your server and can specify where to boot without going through the search process, enter:

`BOOT ADMIN>` **`boot lan.080009-`*`nnnnnn`***

where: `080009-`*`nnnnnn`* is the Ethernet address of the install server. (Some newer systems may not use the `080009` prefix.) This number can be found by running the `lanscan` command on the server.

Step 4. If your server is listed during the search, you can boot the system by entering p and the index number of the server. For example:

```
BOOT ADMIN> p1
```

This will cause the boot to begin. Or, exit this screen by entering:

```
BOOT ADMIN> x
```

```
BOOT ADMIN> boot p1
```

IPF Client Network Booting Option

Step 1. From the EFI Boot Manager menu, you will see a prompt to select a boot option. Select **Boot option maintenance menu**.

```
EFI Boot Manager ver 1.10 [14.54]  Firmware ver 0.0 [4209]

Please select a boot option

    EFI Shell [Built-in]
    Boot option maintenance menu
    Security/Password Menu (*** Prototype ***)

Use up and down-arrows to change option(s).
Use Enter to select an option
```

Step 2. The Main Menu appears and prompts you to choose an operation. Select **Add a Boot Option**.

```
EFI Boot Maintenance Manager ver 1.10 [14.54]

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
```

Step 3. The following menu displays. Select an appropriate network card for network boot. For example, look for entries with a MAC followed by the Mac/*LLA* address of the LAN card.

```
EFI Boot Maintenance Manager ver 1.10 [14.54]

Add a Boot Option.  Select a Volume

    Removable Media Boot
[Acpi(HWP0002,0)/Pci(2|0)/Ata(Primary,Maste
    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
```

Step 4. Enter an appropriate boot option name at the message prompt. For this example, new boot options are named LAN1 and LAN2.

Step 5. Exit to the main menu. The new boot option will now appear in the EFI Boot Manager main menu.

```
EFI Boot Manager ver 1.10 [14.54]  Firmware ver 0.0 [4209]

Please select a boot option

    SCSI2-HPUX
    EFI Shell [Built-in]
    LAN2
    LAN1
    Boot option maintenance menu
    Security/Password Menu (*** Prototype ***)

Use up and down-arrows to change option(s).
Use Enter to select an option
```

Step 6. Select the new `boot option you created`. The following is an example of a successful boot using the new boot option.

```
Loading.: LAN1
Running LoadFile()

CLIENT IP: 15.1.52.128  MASK: 255.255.248.  DHCP IP: 15.1.53.37
GATEWAY IP: 15.1.48.1
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...
```

5 Installing HP-UX with Ignite-UX on Clients from a Server

This chapter discusses the steps for installing HP-UX on client systems from an Ignite-UX server. Topics are:

- Methods of Installing Client Systems.
- Installing from the Ignite-UX Server.
- Configuring the Installation.
- Advanced Tab.
- Executing the Installation: Go!.
- Viewing and Printing a Manifest.

Methods of Installing Client Systems

Ignite-UX allows you to install client systems manually from the Ignite-UX screen as explained here, or automatically as explained in Chapter 10. These are called pushing installations. You can also install clients from a remote system by using `bootsys`, as explained at the end of this chapter, or install the client locally by pulling an OS from an Ignite-UX server as explained in Chapter 4.

Each installation method requires that a configuration (config) file be created, as explained in Chapter 3. The configuration can include any supported HP-UX 10.*x*, 11.0, or 11i OS, 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.

Supported Peripherals

All disk drives supported on HP computer systems are supported for installation. Fibre channel, tape devices, and LAN cards are also supported.

Disk arrays can be installed with HP-UX, but the installation tasks do not support configuring an array. See your array documentation for configuration information.

The HP-UX client-side installation tools support VT100 and Wyse 60 terminals, compatible emulations, and all HP terminals.

Network Requirements

If you are loading your server depots or client software from a remote system, your target system will also need a network card. If the target system has multiple LAN cards, select the card that is configured onto the correct network by navigating to the **System** –> **Additional Interfaces** menu. Only one LAN card can be used during the installation, configured on the Ignite-UX screen or handled automatically by `bootsys`.

Your server system will need to be configured. In addition you will need:

- If you plan to perform a network boot for a target client then the server must be on the same subnet as the target system that will be installed. Other options include using a boot-helper system on each subnet from which to boot clients. See Appendix B, “Using a Boot-Helper System,” on page 261 to set up a boot-helper system or using the `bootsys` or `make_tape_recovery` commands.
- If you have more than one LAN connection, you must select the one to be used for the install process.

TIP You can only boot over the network from the system’s built-in Ethernet interface. FDDI is also supported, but for non-booting only.

Installing from the Ignite-UX Server

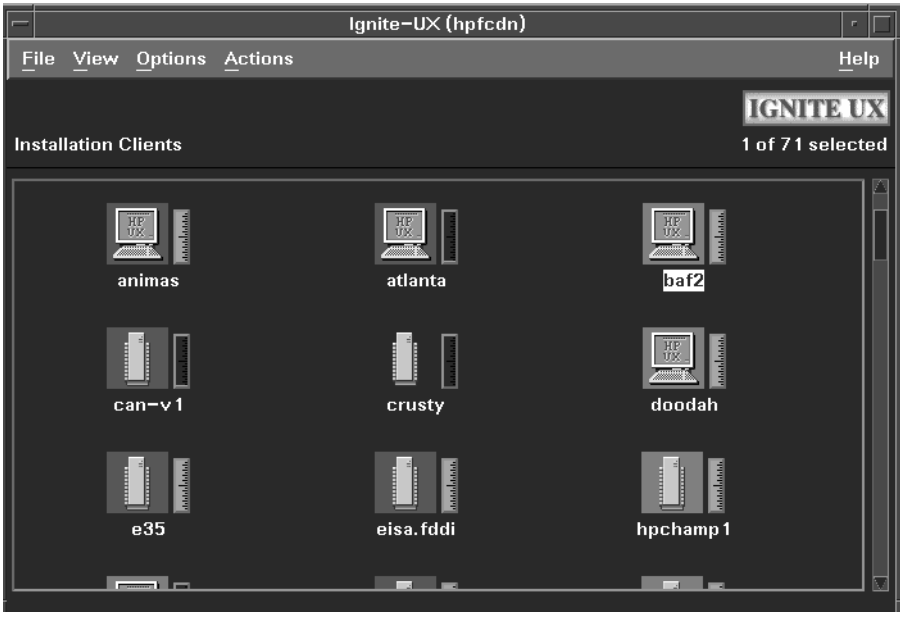
Starting Ignite-UX

If you have not already done so, run Ignite-UX on the server as root:

```
/opt/ignite/bin/ignite
```

Running Ignite-UX on the server is explained in the following procedures. Running Ignite-UX remotely from a client or other system provides a TUI with equivalent keyboard navigation.

Ignite-UX screen

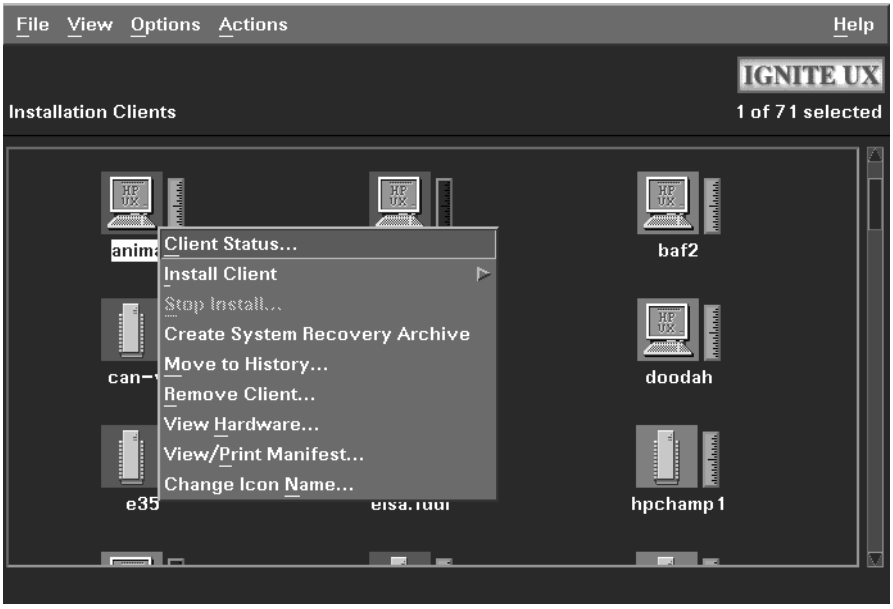


Before any new clients are represented as icons on the Ignite-UX screen, they must first be booted from the Ignite-UX server. If the client is already running an OS, this can be accomplished remotely via the server using: **Actions** –> **Boot Client**

If the client is not running, see “Booting Client Systems from the Network” on page 67.

After the client icons display, you may:

- Click once on the client icon to select it for further actions.
- Double-click the client icon to get a Client Status screen.
- Right-click the selected client icon to get an Actions menu, which is very similar to the pull-down **Actions** menu:



For more about the available Ignite-UX selections, see Chapter 4, “Installing HP-UX with Ignite-UX on Clients Locally,” on page 63 or click **Help**.

Configuring the Installation

To begin the installation, first select a client icon. Then, from the **Actions** menu, select either:

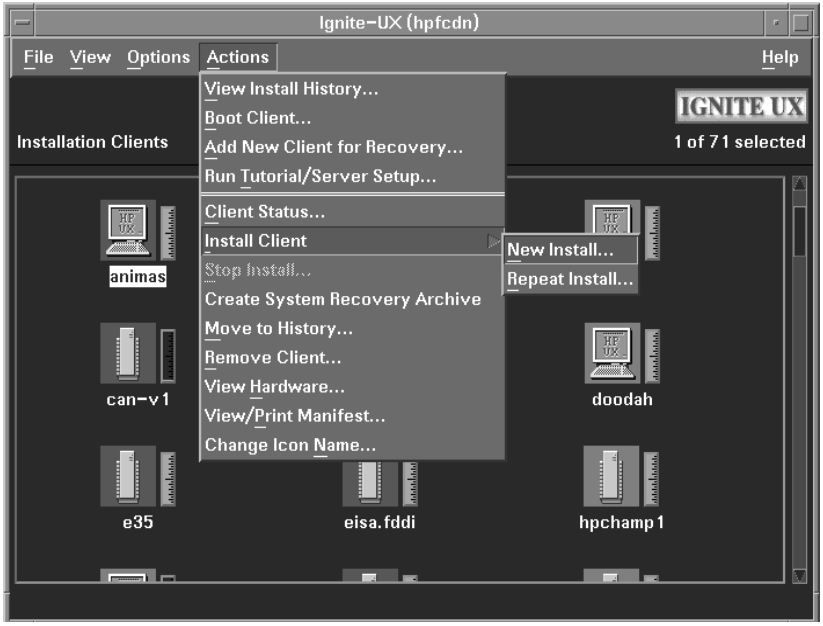
- **Install Client** –> **New install** to install a new client, OR
- **Install Client** –> **Repeat install** to use another clients configuration.

If you have previously installed a client, you will be asked if you want to use the same configuration data again.

If the following message displays:

```
Settings from a previous installation session were found at
startup.  Do you wish to retain these settings for the
current session?
```

Respond **Yes** to re-use some or all of the configuration used in the previous session. Respond **No** to use an entirely new configuration.



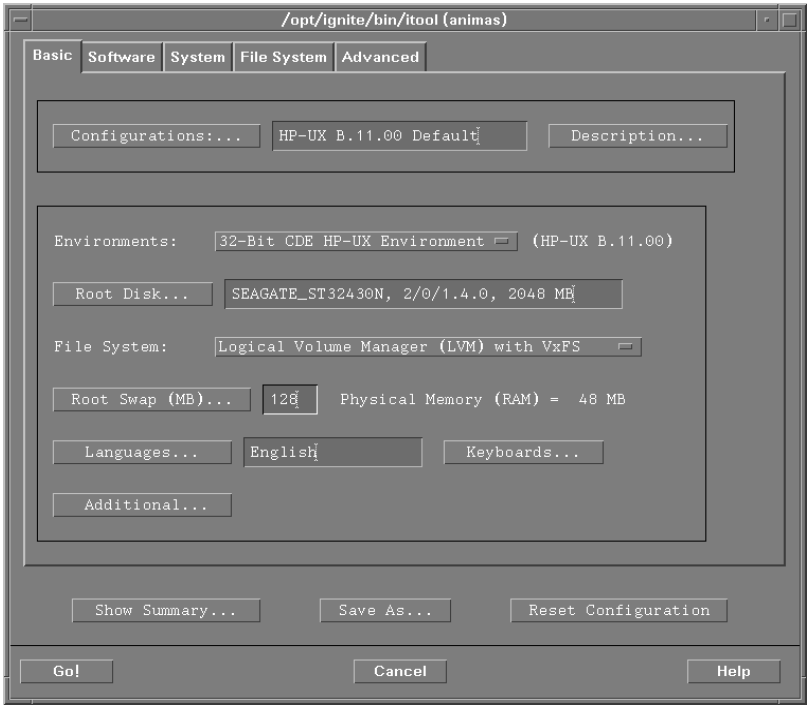
config file parameters

All configuration parameters from an installation are identified and saved as a config file in: `/var/opt/ignite/clients/0x`*`LLA`*`/`

You can use config files in a non-interactive installation using the `bootsys` command. You can choose a preset configuration in the **Repeat Install** selection list to repeat a previously installed configuration and execute it within Ignite-UX, without further intervention.

Basic Tab

After you choose to install a system, you see the **Basic** screen:



This screen shows all the basic information for setting up the file system and for loading the OS environment and selecting an HP-UX 11i OE. It also allows you to configure languages, locale, and keyboard requirements. A **Save As...** button also allows saving configurations for later use.

Configurations

Click this selector to display a list of available OS configurations. Then select the one you want to use for this installation. The **Description** button shows more information about each configuration.

Your configuration files are stored in a server location referenced by the `/var/opt/ignite/INDEX` file. The `INDEX` file defines the list of configurations.

OS and HP-UX 11i OE Environments

Select the OS or HP-UX 11i OE environment from the choices available in the list. For HP-UX 11.0/11i, this may include 64-bit or 32-bit OS version. The choices and defaults depend on the releases available on the server, and may include, for example, Common Desktop Environment (CDE) as the default.

File System

Select one of the following:

- **Whole Disk (not LVM)** — This may be the appropriate choice for single-disk systems with disks less than 2GB.
- **Logical Volume Manager (LVM) with HFS** (High-Performance File System) — This selection will format single or multi-disk systems to combine the disk space into a single, large disk pool, and then allocate volumes as needed. The root volume in this case and the swap must be on the same physical volume, and will be configured in this manner by Ignite-UX. The File System tab will provides additional opportunities to configure the LVM volumes. In the File System tab, you can edit the sizes of LVM partitions, or use the values that Ignite-UX computes for you.
- **Logical Volume Manager (LVM) with VxFS** (Veritas File System) — This will format single or multi-disk systems to combine the disk space into a single, large disk pool, and then allocate volumes as needed. VxFS is the same as the Journaled File System (JFS) and allows file system size to be changed after installation. With the optional HP OnlineJFS product you can resize, defragment, or make a "snapshot" of a mounted file system.

- **VERITAS Volume Manager (VxVM) with VxFS** — This will format single or multi-disk systems to combine the disk space into a single, large disk pool, under VxVM, and then allocate volumes as needed. The root, boot, and primary swap volumes must be on the same physical disk and will be configured in this manner by Ignite-UX. The File System tab provides you additional opportunities to configure the VxVM volumes. VxFS is the same as the Journaled File System (JFS) and allows file system size to be changed after installation. With the optional HP OnlineJFS product you can resize, defragment, or make a "snapshot" of a mounted file system.

NOTE VxVM 3.5 is currently only available for HP-UX 11i Version 1.0. VxVM 3.1 is available for HP-UX 11i Versions 1.5 and 1.6.

See “File System Tab” on page 92 for detailed information on File System configuration.

Root disk To change root disks, select this button, select another disk from the list of available disks, and click **OK** in that screen.

For example, a root disk is usually located at SCSI bus location 6.

Root swap The amount of root swap space depends on the applications being loaded. You can choose to use the default which Ignite-UX computes, based on available memory on the target system. Or you can select **Root Swap** and select from the choices that appear in the list. You can also edit the field directly and type in the amount of swap space you wish. The swap will be rounded to a multiple of 4MB or the LVM extent size.

Computing swap space is explained in these HP-UX guides:

- HP-UX 10.*x* — *System Administrator Tasks*.
- HP-UX 11.0 and 11i — *Managing Systems and Workgroups*.

Languages

The languages available in your HP-UX system will be shown when you select this field. Select the item(s) you want, if it is other than the default. The dialogue screen allows you to select more than one

language. Highlight the additional items by double-clicking on each. You can also drag the pointer down the screen to highlight a range of items; then press the mark/unmark button.

You can make any of the selections the system default language. This will become the system default language after it is installed.

Locale

Each language will have a corresponding locale (language variant). A locale describes the system management of a language for doing the following:

- Messaging
- Representing numbers
- Displaying monetary values
- Telling time
- Generating characters
- Sorting text

HP-UX can have more than one installed language. The "default language" is the language environment represented on the target system at boot, unless you select another installed language using the HP-VUE or CDE login screen, reset the `LANG` environment variable, or use geocustoms (HP-UX 10.30 and after) to change it.

Default language choices

Click **Default Language...** to see the **Default Language Choices**. 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.

Languages may be activated is several ways:

- **ASK_AT_FIRST_BOOT** allows you to leave the language setting open (unset) until the client system is first booted. At that time, the user will be prompted. The language setting will be performed as part of the initial system configuration. (This applies only to HP-UX 10.30 and later).
- **SET_NULL_LOCALE** creates a `NULL` language environment, with the locale variables set to `NULL` by default. A null locale allows programs to execute without using localized message catalogs. This can increase system performance. All HP-UX messages appear in English if the locale is set to **NULL**.

Keyboards… Select the type of keyboard to be used, from the adjacent field. Otherwise, you can use the default selection.

Additional… Click **Additional...** to select among certain pre-configured use-models and variables from your current configuration files. The buttons available are determined from the variables in your config file. When using LVM, you will see selections for easily setting up multiple disks, striping, and file system creation. For details, see the *instl_adm* (4) manpage.

Functions Available on All Tabs

Save as… In server mode, when you have finished your configuration for all tabs, you can save the configuration as a specific file. The saved configurations will then appear under the **Configurations** menu for use in future installations. This function is not available when running the Ignite-UX interface on the install client.

Show summary… Click **Show Summary...** to display the current HP-UX, the basic disk layout, hardware inventory, and other software that will be installed.

Reset configuration Click **Reset Configuration** to change the configuration settings for the currently-selected configuration back to the default settings. You can do this from any tab.

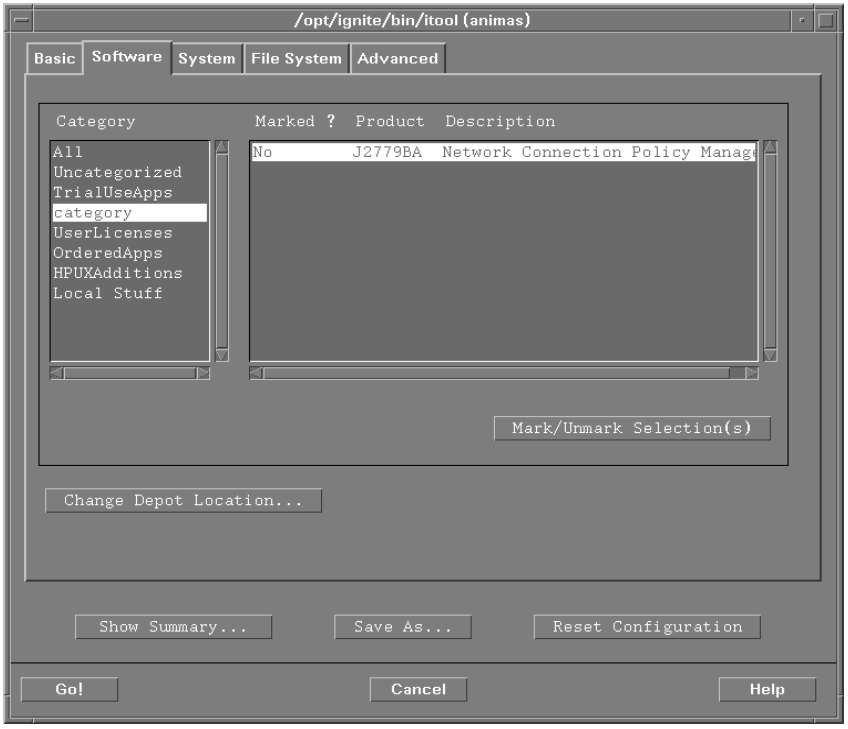
Go! Click **Go!** to initiate an installation. Since **Go!** is always available, click it from any of the tabs. If you don't need to do any customization, click **Go!** now to begin the installation. Then see “Executing the Installation: Go!”, later in this chapter.

After clicking **Go!**, you will still have the opportunity to cancel out of the install sequence.

Cancel Click **Cancel** to exit installing this system.

Help Help information is available on all screens, and you can get context-sensitive help for specific areas by pressing the **F1** function key.

Software Tab



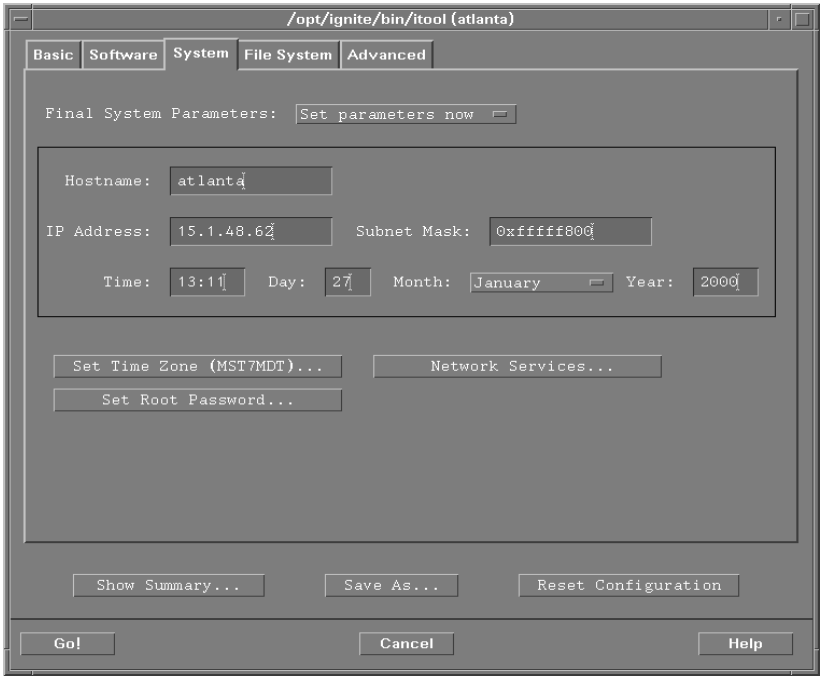
This tab allows you to choose licensing level and additional applications that you configured when you set up your server. To access a specific depot, you can also change depot locations. This display does not dynamically update from a newly-selected depot. When choosing a new depot, it must be identical in content to the current one. If not, use `make_config` on the server to configure the new depot.

- **Category** — Select a topical category to display products available.
- **Product List** — Double-click on a product in the list to select (highlight) it and to toggle its "marked" status (**Yes** or **No**). Alternately, use **Mark/Unmark Selection(s)** to toggle the "marked" status for a selected item.

If patches are kept in a separate depot, by default they are loaded after Core software. If there is more than one non-Core software to be loaded, you may need to specifying the load order for the patch(es) in a config file.

System Tab

You will see a choice selection allowing you to set parameters now, or at first boot of the target system. If you choose to set these parameters now, you see this screen:



Hostname

Your system must have a unique system name (a "hostname"), which can be a simple name.

A system name must fulfill the following conditions:

- It must contain no more than eight characters
- It must contain only letters, numbers, underscore (_), or hyphen (-).
- It must start with a letter. Upper case letters are not recommended.

- The first component of a host name should contain no more than eight characters, for compatibility with the uname command.

IP address

Use this field to enter the IP address of the form:

n.n.n.n

where: each *n* can be `0` through `255`.

For example:

`15.1.48.140`

To determine an existing IP address, use:

`nslookup` ***`hostname`***

Subnet mask

This field sets the subnet mask. The subnet mask will typically be provided by your network administrator, and is of the IP address form or a corresponding hex number. For example:

`255.255.248.0`

Time and date information

If necessary, enter information for the **`Time`**, **`Day`**, **`Month`**, and **`Year`** fields: For time, use the 24-hour format: *hh*:*mm*

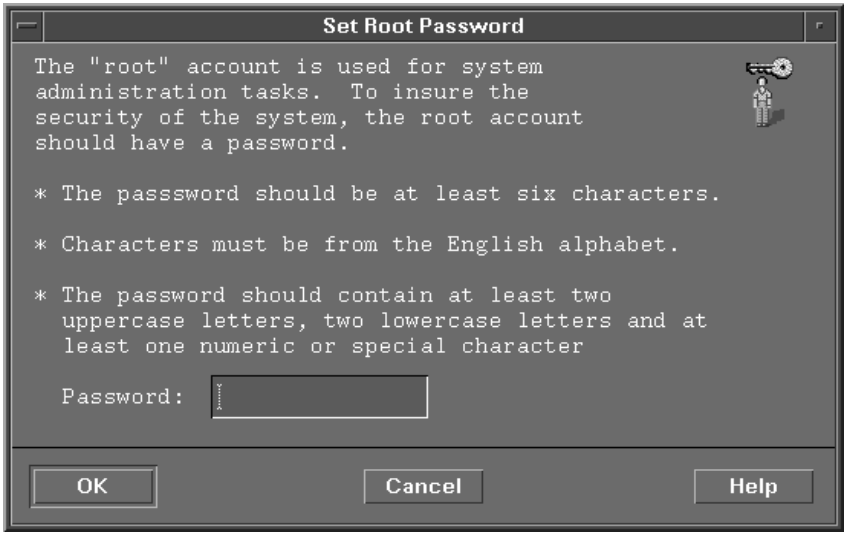
Select the correct month by clicking on the button and selecting from the list. Edit other fields by using the **Backspace** and **Delete** keys.

Set Time Zone screen

Select this button to bring up a display of time zone selections. You will see two selector lists: the first consists of general locations, and the second has corresponding time zones.

Select an item and click **OK** to make a choice.

Set Root Password screen



The root account is used for system administration tasks. To insure the security of the system, the root account should have a password.

You should observe the following requirements when setting a password:

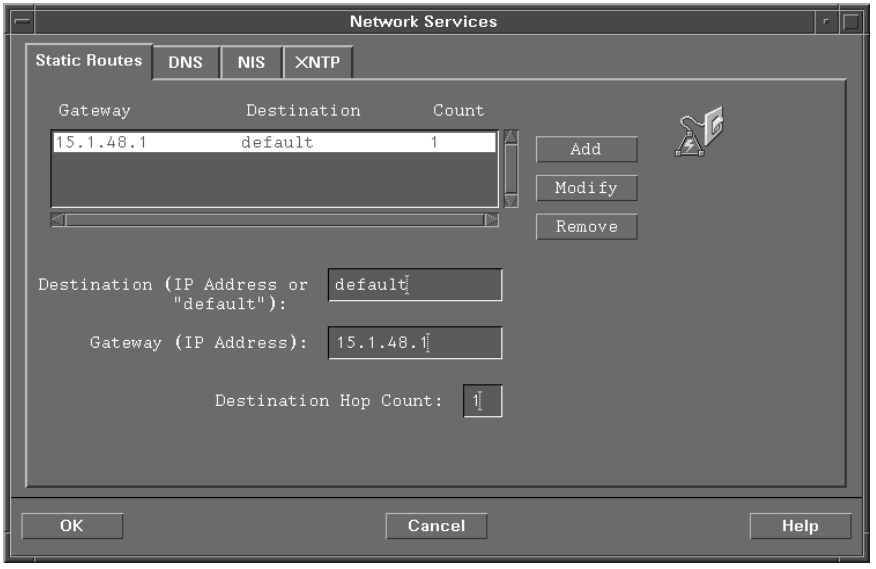
- The password must be at least six characters long.
- Characters must be from the English alphabet.
- The password should contain at least two uppercase letters, two lowercase letters and at least one numeric or special character.

Network services

Click **Network Services** to access tabs used to enter information on:

- **Static Routes**
- **DNS**
- **NIS**
- **XNTP**

Static routes screen



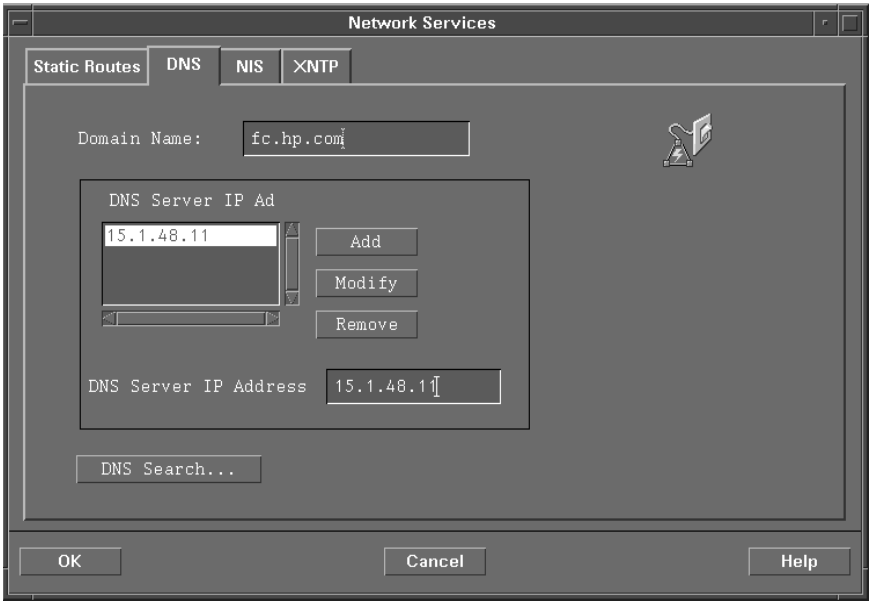
If your network is divided into subnets, you will probably need to specify a gateway system to reach other subnets:

- **Destination** — The field has the word **default** or the IP address of the destination network.
- **Gateway** — 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 is not your system's own IP, this is usually set to `1`. If your gateway IP is the same as your system's, then the Hop Count is `0`.

Once the appropriate fields have been completed on this screen, click the **Add** button.

For more information, see the *routing* (7) manpage.

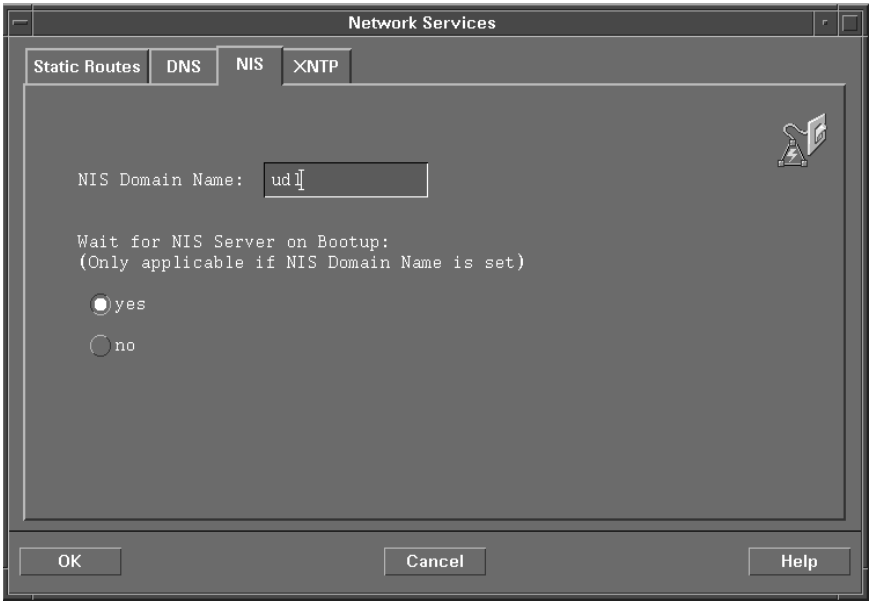
DNS screen



On this screen, you can configure the **Domain Name** (an extension to the host name, such as `fc.hp.com`) and the IP address of the Domain Name Server. The listing of current servers is displayed, if they are predefined in the Ignite-UX server. Use the `nslookup` command on a running system to find this information.

After entering a DNS server, click **Add**. Use **Modify** if you are changing an existing entry.

NIS screen



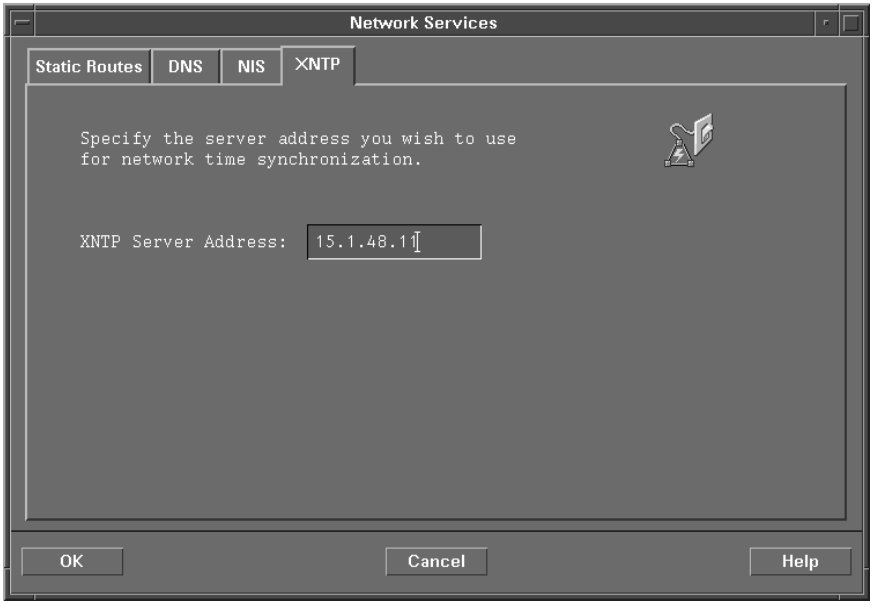
Typically, the (non-server) hosts in a network are NIS clients. Whenever a process on an NIS client requests configuration information, it calls NIS instead of looking in its local configuration files. The set of maps shared by the servers and clients is called the **NIS domain**.

For more information on NIS, see the *domainname* (1M) manpage or the guide *Installing and Administering NIS Services*.

Wait for NIS server on bootup

Select **yes** or **no**, depending on your configuration for NIS.

XNTP screen



The xntpd daemon maintains system time, in agreement with Internet standard time servers. It does all computations in fixed point arithmetic and clock adjustment code is carried out with high precision.

For more information, see the *xntpd* (1M) manpage.

Additional (network) interfaces

Use this button to bring up a screen for entering information identifying additional LAN interface cards in the target system.

This screen enables you to enter or change IP and Subnet information, as needed, and designate the Primary Interface.

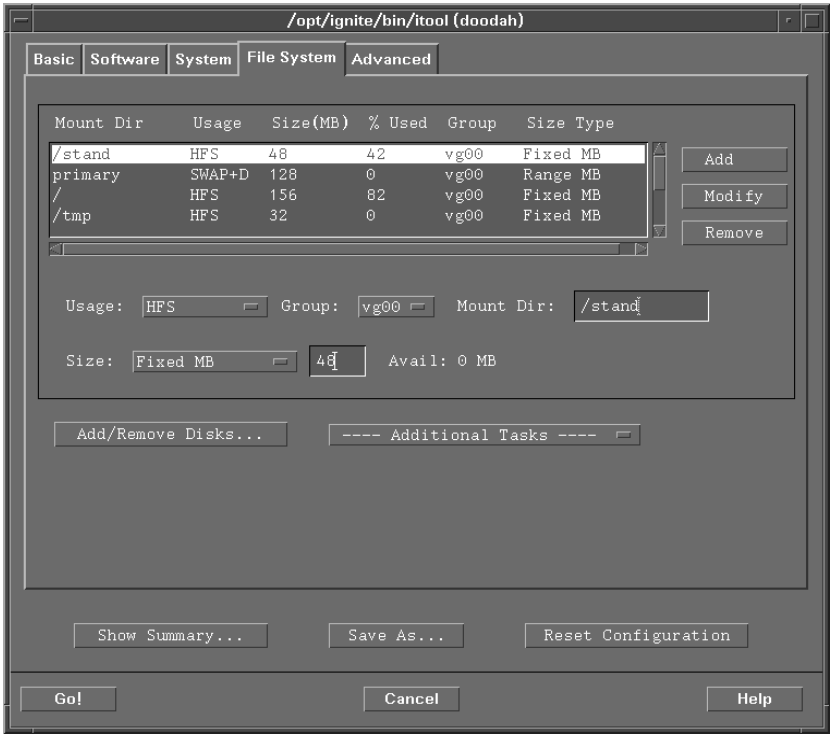
NOTE

If the target system has more than one interface, the Primary LAN card will be associated with the host name of the system in `/etc/hosts`.

1. Select an **Interface** card from the selection list.
2. Enter or modify the **IP Address**, as needed.
3. Enter or modify the **Subnet Mask**, as needed.

4. Activate **Primary Interface**, depending on the status you want for this interface.
5. Click **Modify** when you have finished with changes for each interface.

File System Tab



This tab enables you to do a variety of file-system and disk-configuration tasks and will differ in appearance, depending on whether you previously selected LVM, VxVM or whole disk, on the **Basic** tab. This illustration is what you would see if you had picked LVM on the **Basic** tab.

Adding and changing file system configuration

To add or change any configurations on the display of file systems:

1. Enter the information in an appropriate field below the display.
2. Select one of the buttons to the right: **Add**, **Modify** or **Remove**.

3. To see more information on the file system display, use the horizontal scroll bar or resize the screen.
4. The **Avail:** indication shows how much space is unallocated in the volume group of the highlighted volume.

For LVM:

- One of the volumes must be root (`/`).
- A swap volume (primary) is required.
- Directory names must have valid HP-UX names (`/usr`, `/database`, etc.).

For VxVM:

- One of the volumes must be root (`/`).
- 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 (`/usr`, `/database`, etc.).

The buttons which activate changes are **Add**, **Modify** and **Remove**.

Generally, changes are not put into effect until you select one of these. If you make a change and then leave the tab without using one of these buttons, your changes may not be applied.

Usage

Select the **Usage** field to see list of file system usage types. If you want to change file system type or usage for the selected item, select an item in this list. The usages are as follows:

- **HFS** — Select this item to create a High-Performance File System.
- **SWAP** — Select this item to create swap.
- **SWAP-Dump** — Select this item to create an area for both swap and system dump.
- **VxFS** — Select this item to create a JFS. This is an extent-based, journaled file system featuring high-reliability, fast recovery time and on-line administration.
- **Unused** — This means the logical volume will be created, but not used.

Group

Click on the **Group** field to open a selection list. You can choose a volume group name from the list.

IMPORTANT

Renaming or changing the disk file-system structure causes the old file system on that disk to be lost (a warning message will remind you of this).

- If you want to add a new/unused disk and give it a different volume group name or create a new volume group, select the **Add/Remove Disks...** field and follow the procedure.
- If you want to reconfigure the volume group in general, including renaming it, click **Additional Tasks** –> **Group Parameters**, where you can fill in a custom group name, and change other disk parameters.
- Click **OK** when you are finished with the sub-screens for any of these tasks. You will be returned to the **File System** tab.

Mount dir

For the root disk, use 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`".

Size

For setting up each selected file system (as shown in the Mount Dir display), the following choices are available:

1. First select an item in the directory display for the file system you want to change. The current selection will show in the **Mount Dir** field.
2. The sizing method (such as "Fixed Size") currently used for that particular file system will appear in the **Size** field. To change the Sizing Method:
   a. Make sure the file system you want to change is selected in the directory display list.
   b. Select the sizing method field to open the list of sizing methods.
   c. Select one of the items (such as **Size Fixed MB**). It will then remain displayed in that field.
   d. Click **Modify** to execute the change on the selected file system.

The types of sizing are:

Fixed The selected (highlighted) file system is set to this size.

All Remaining The selected file system automatically takes over all remaining file system space on the disk or volume group.

Free Size Use this selection when you know how much free space you wish the volume to have after the system is installed. The size of the volume will be the specified amount plus the amount the selected software requires.

Free Percent This category is similar to free size, but expressed in percent. It is used if you know how full you wish the volume to be, in percentage of the volume size. If you indicate "20%", then the volume would be 80% full after the installation of the selected software.

Size Range Select this category in the list to set a maximum size for the file system (the minimum is determined by the software impact on the volume).

NOTE `/usr` must have sufficient space to accommodate an OS update. The absolute minimum is 324 MB for a 64-bit system. See the installation guide supplied with your HP-UX media.

Add/remove disks This opens a display which allows you to do the following:

- Add a new disk and configure its file system type and volume group designation, if any.
- Remove a disk from current usage on the target system by designating it as **Unused**.
- Determine your current disk usage.

To change a disk usage status:

- Select a disk in the displayed list.
- Click **Usage** to set a new usage. If you select LVM or VxVM the **Disk Group**: button appears.
- Click **Disk Group:** to see the volume group choices or type in a new volume group name in the entry field.
- Click **Modify** to execute any changes.

Additional Tasks

This button enables you to configure advanced information in the following categories, as needed. Click on the field to see the following menu items:

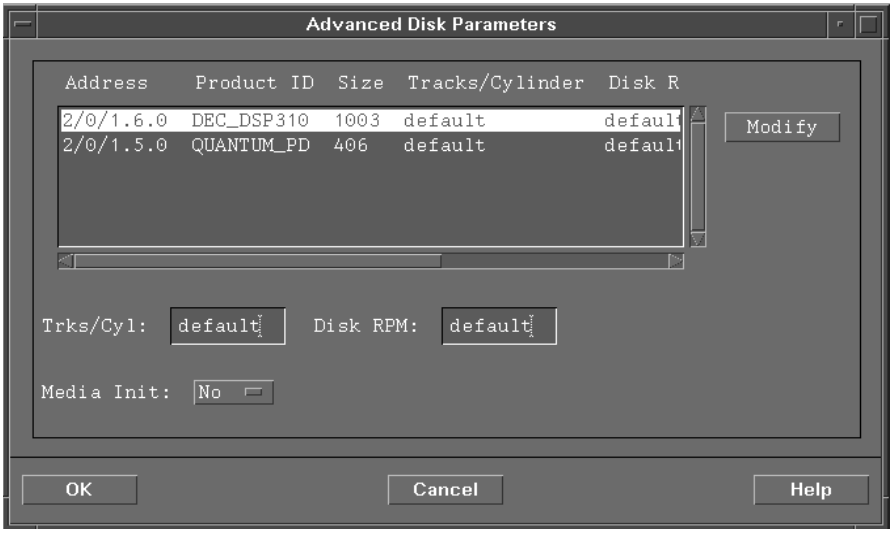
- **Disk Parameters**
- **File System Parameters**
- **Volume Parameters**
- **Group Parameters**

Clicking on one of these will open a screen which will enable you to change advanced parameters. The button will retain the label of the area you are currently working in.

NOTE

Screen choices differ depending on the file system choices you made on the **Basic** tab.

Advanced Disk Parameters screen



Step 1. Highlight a disk in the selection list to select it.

Step 2. Configure the **Trks/Cyl** and **Disk RPM** by direct editing, as needed.

Step 3. Indicate whether Media Init is required by clicking on the selection box and selecting a choice.

Step 4. Click **Modify** to configure changes.

Step 5. Click **OK** to leave **Advanced Disk Parameters** and return to the **File System** tab.

Tracks per cylinder

Step 1. Select a disk by clicking on its entry.

Step 2. Edit the **Trks/Cyl** field as needed and click **Modify** to execute any changes.

Step 3. Click **OK** to leave this screen and return to the **File System** tab.

Disk RPM

Step 1. Select a disk by clicking on its entry.

Step 2. Edit the **Disk RPM** field as needed and click **Modify** to execute any changes.

Step 3. Click **OK** to leave this screen and return to the **File System** tab.

NOTE Running `MediInit` is not recommended unless hardware damage is suspected.

Media init

Step 1. Select a disk by clicking on its entry in the list displayed.

Step 2. Click **MediaInit** to open the selection list.

Step 3. Click **Yes** or **No**. If this is set to **Yes**, you will also see the **Interleave** field.

Step 4. Click **Modify** to execute any changes.

Step 5. Click **OK** to leave this screen and return to the **File System** tab.

More information...

- *mkfs_vxfs* (1M)
- *mkfs_hfs* (1M)
- *mediainit* (1)

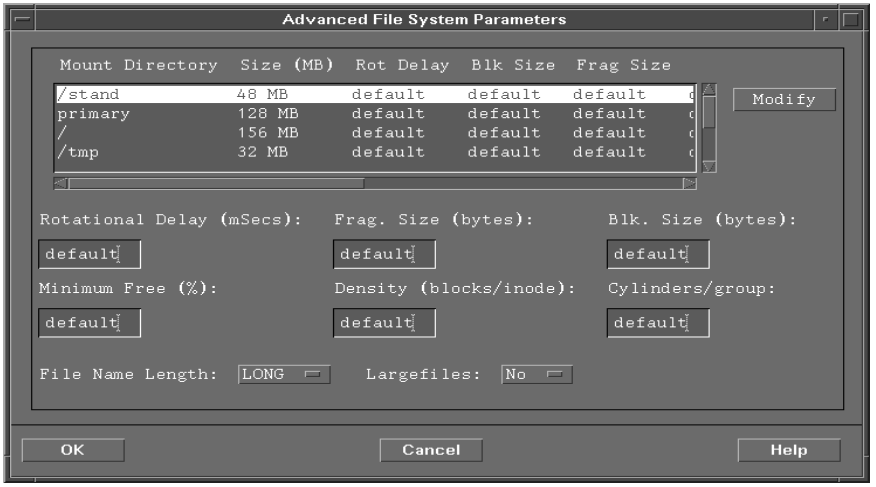
Intrlv This field is available if **Media Init** is set to **Yes**.

The interleave factor, “interleave”, refers to the relationship between sequential logical records and sequential physical records on the disk. It defines the number of physical records that lie between the beginning points of two consecutively numbered logical records. The choice of interleave factor can have a substantial impact on disk performance.

For more information, consult the guide for your disk hardware.

Also see the *mediainit* (1) manpage.

Advanced File System Parameters screen



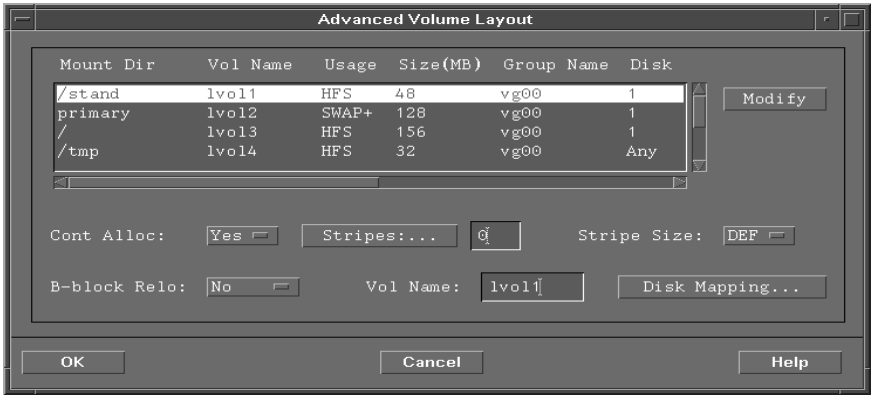
These parameters apply only to HFS file systems:

- **Rotational Delay**
- **Fragment Size**
- **Block Size**
- **Minimum Free**
- **Disk Density**
- **Cylinders/Group**

You can use the default values computed by Ignite-UX, or change them, as needed. Selecting **default** means it will use the default defined by the `mkfs` command. When you have finished with this area, click **OK** to return to the **File System** tab.

To get more details, see the *mkfs_hfs* (1M) and *mkfs* (1M) manpages.

Advanced Volume Layout screen (Volume Parameters)

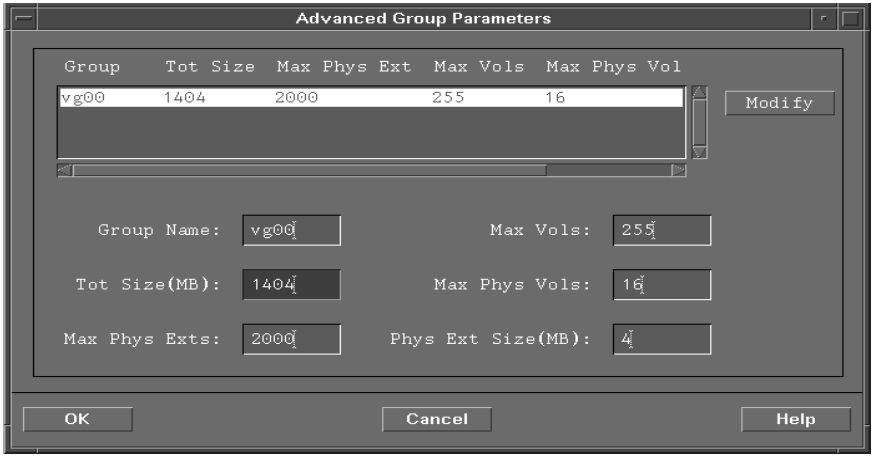


Example is shown using LVM.

Use this screen to perform detailed configuration of volumes, as needed, in the following areas. For more detailed information, see the *lvcreate* (1M) manpage for LVM or the *vxassist* (1M) manpage for VxVM.

- **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 allowed 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.**
- **B-block Relo** (Bad-Block Relocation)
- **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 default stripe size is 64Kb.
- **Vol Name** — Enter the name you want for the selected volume.
- **Disk Mapping** — Displays a screen which allows you to restrict the disk drives on which the volume data will reside. Normally, the data will be allocated over these disks sequentially.

Advanced Group Parameters screen

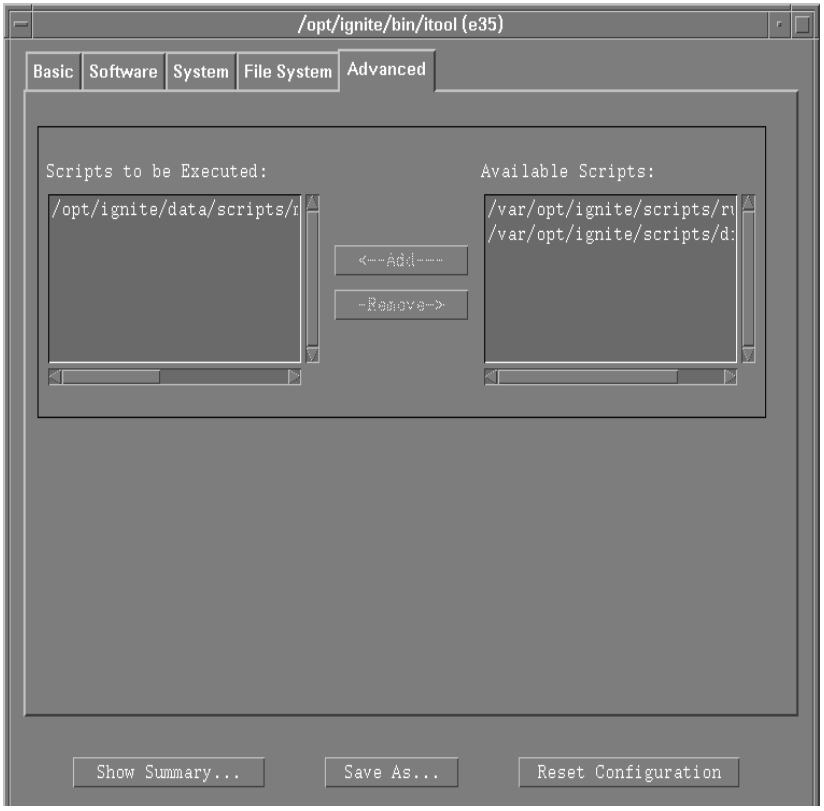


Example is shown using LVM.

Use this screen perform detailed configuration of volumes, as needed, in the following areas. For more detailed information, see the *vgcreate* (1M) manpage for LVM or the *vgdg* (1M) manpage for VxVM.

- **Group Name** — Use to rename existing volume groups.
- **Max Vols** — Maximum number of logical volumes.
- **Total Size** — Total size of all volumes.
- **Max Phys Vols** — Maximum number of volumes.
- **Max Phys Exts** — Maximum physical extents.
- **Physical Ext Size** — Physical extent size in MBs.

Advanced Tab



Use this screen to activate any HP or custom scripts which you might want to run as part of your installation. Note that the scripts listed are those with a "scripts" keyword in the `/var/opt/ignite/INDEX` file.

Adding a script To add an item, select the item from **Available Scripts** and click: **Add**

Removing a script To remove an item, select the item in **Scripts to be Executed** and click: **Remove**. The item is deactivated, but remains in the **Available Scripts** list.

Executing the Installation: Go!

Select **Go!** in any Ignite-UX tab to initiate the installation. You do not need to examine all tabs, if you simply want to do a generic installation.

A confirmation screen lists disks that will be written on during the installation process and a log of any warnings or errors.

- If you do not wish to proceed with the installation at this time, click: **Cancel**
- The pre-install analysis display screen is scrollable. Be sure to inspect this information and check to see that the disk(s) described in the display list is the one you intend to install on.
- Any errors which are listed must be corrected before you proceed.

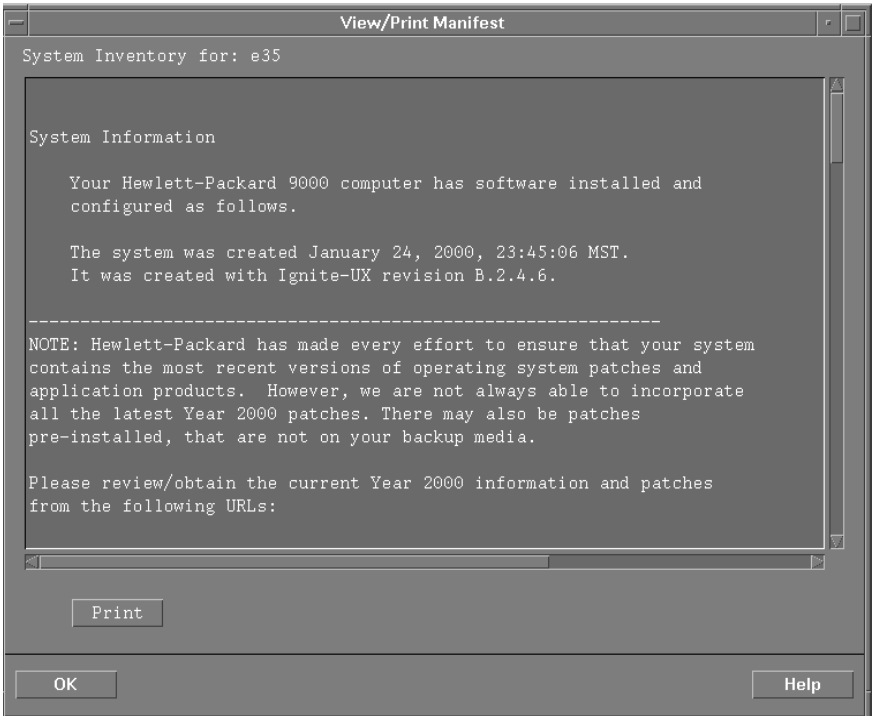
As the installation proceeds, you will see a log including the warnings and errors which may need to be addressed before proceeding.

When the installation is complete, you can print a manifest. Then either save the client data in a history directory, remove the client and its data from the server, or just leave the data on the server.

Viewing and Printing a Manifest

From the Ignite-UX screen

To view a system manifest from Ignite-UX, select a client icon and click: **Actions** –> **View/Print Manifest**. This is also available from the Client Actions menu (right-click on client icon). The system search may take a moment.



The manifest provides customer order information for the selected target system.

You can view or print the manifest when a target client is "Complete", as indicated by the Client Status screen. The online information is scrollable.

The manifest contains the following information:

- Customer information, if this has been entered on the individual client configuration screen.
- Hardware connected to the system.
- Storage Devices.
- Installed Software.
- Disk layout.
- File System layout.
- Swap Configuration.
- Kernel Configuration.
- System Information.

Using the print_manifest command

To print the system manifest from the server command line, enter:

```
/opt/ignite/bin/print_manifest
```

The ASCII file is printed to stdout using format instructions from the manifest template file (explained below).

Location of manifest files

Manifest files are saved on the server in:

```
/var/opt/ignite/clients/LLA/manifest/manifest.info
```

and on the target client system in:

```
/var/opt/ignite/local/manifest/manifest.info
```

If the client data is moved to history, that data includes both the client's manifest and config file. Both these files can be recalled at a later time.

Customizing manifest output

Add printer formatting — Include the `-e` option to add PCL control codes to the output, adding bold headings, etc. to the output.

Print an existing manifest file — Include the `-s` option to use previously stored data, rather than starting a new system search.

Specify a template file —Use the `-t` option to specify a template file to customize the manifest output to your needs. A sample template file is: `/var/opt/ignite/local/manifest/template.def`. The template uses pcl3 formatting commands (similar to `printf`), allowing you to structure the output as desired. To create your template, be sure to edit a copy of this file, not the original.

For example, if you want a condensed, machine-readable output, you can remove all blank lines and headings from your template. This will also speed-up the manifest generation. This command prints a condensed manifest using the existing manifest file and referencing a template you created named `condensed.def`:

```
print_manifest -s  -t
/var/opt/ignite/local/manifest/condensed.def
```

You can also access the raw manifest data via a script or program. This file is updated on the Ignite-UX server each time print_manifest is run without the `-s` option:
`/var/opt/ignite/clients/LLA/manifest/manifest.info`

6 Installing Patches with Ignite-UX

Ignite-UX uses existing SD depots to distribute software. To distribute patches with Ignite-UX, you need to first bundle them into SD depot format. Patches can be installed along with the Core software being patched.

This chapter shows how to create a patch depot containing HP-UX 11.0 patches, create a single patch bundle of the contents of the depot, and add this bundle to an existing Ignite-UX configuration.

For more details, see the “Managing Patches” chapter in the *Software Distributor Administration Guid*e available on the HP-UX Instant Information CD and on the web:

`http://docs.hp.com`

Creating a Patch Depot

Follow these steps to create a patch depot on an HP-UX system.

Step 1. Obtain the set of patches you want to place and manage in an SD depot. For example:

```
PHCO_7891 PHCO_9348 PHKL_9361 PHSS_7726 PHSS_8966 PHSS_9400
PHCO_8353 PHKL_8376 PHKL_9569 PHSS_8667 PHSS_9201
```

HP patches delivered by the Response Center or the web are `shar` files consisting of a serial depot and a `ReadMe` file.

Step 2. Unshar the patches using:

```
for i in PH*
do
sh $i
done
```

Step 3. Combine the separate depots into one depot:

1. Create the directory to store the patches:

   ```
   mkdir /var/opt/ignite/Patches
   ```

2. Copy the individual patch depots into the target depot:

   ```
   for i in PH*.depot
   do
   swcopy -s ${PWD}/$i \* @ /var/opt/ignite/Patches
   done
   ```

Step 4. Verify the contents of the depot:

```
swlist -d @ /var/opt/ignite/Patches
```

Here's the output for the example list of patches above:

```
Initializing...
Contacting target "interop1"...
Target: interop1:/var/opt/ignite/Patches
No Bundle(s) on interop1:/var/opt/ignite/Patches
Product(s):
PHCO_7891 B.10.00.00.AA allows mount to turnon hfs-specific opts
PHCO_8353 B.10.00.00.AA cumulative awk(1) patch
PHCO_9348 B.10.00.00.AA cron(1M) and at(1) patch
PHKL_8376 B.10.00.00.AA Fix vmtrace bug.
PHKL_9361 B.10.00.00.AA Fix panic caused by MP race
PHKL_9569 B.10.00.00.AA NFS and VxFS (JFS) cumulative patch
PHSS_7726 B.10.00.00.AA CDE Dtterm August 96 patch
PHSS_8667 B.10.00.00.AA CDE Runtime Nov96 Patch
PHSS_8966 B.10.00.00.AA LIBCL cumulative patch
PHSS_9201 B.10.00.00.AA fix for aC++ dld.sl
PHSS_9400 B.10.00.00.AA ld(1) cumulative patch
```

The output shows that the depot has "No Bundles." HP-UX Patches are SD "products", but Ignite-UX can only manage SD "Bundles."

Step 5. Convert the individual patches into a single bundle and put the bundle in the Patches depot:

```
make_bundles -B -n Misc_Patches \
-t "HP-UX 11.00 Patches" /var/opt/ignite/Patches
```

Step 6. Rerun `swlist` on this depot to verify that the bundle has been created:

```
swlist -d @ /var/opt/ignite/Patches
```

Here's the output assuming the example patches:

```
Initializing...
Contacting target "interop1"...
Target: interop1:/var/opt/ignite/Patches
Bundle(s):
Misc_Patches HP-UX 11.00 Patches
```

By default, `swlist` shows only the higher level software bundles. This command shows the patch "products" contained in the bundle:

```
swlist -l product -d @ /var/opt/ignite/Patches
```

NOTE If you need to add additional patches to the depot in the future, simply unshar the patches as described above, `swcopy` them into the Patches depot, and rerun `make_bundles`. This will repackage the depot.

If you would like to remove a patch from the depot, simply use the `swremove` command. You can either run `swremove` and use its friendlier user-interface, or run `swremove` in command-line mode. This example removes the PHKL_8376 patch from the depot:

```
swremove -d Misc_Patches.PHKL_8376 @ \
/var/opt/ignite/Patches
```

Step 7. If you inadvertently create a bundle of a bundle (for example, if you add an HP product you want distributed with the patch depot), use `swremove` interactively to examine and delete the extra bundle.

Step 8. Create a config file for the newly-created `Misc_Patches` bundle. Follow the steps outlined in "Adding a SD Bundle to the Archive Environment" in Chapter 8. Use `/var/opt/ignite/Patches` for your source depot and specify a new configuration file:

```
make_config -s /var/opt/ignite/Patches -a 700 \
-c /var/opt/ignite/data/Rel_B.11.00/misc_patch_bundle_cfg
```

Step 9. Modify the `/var/opt/ignite/INDEX` file to include the new bundle in our "HP-UX B.11.0 archive" configuration:

```
cfg "HP-UX B.11.00 archive" {
description "The ARCHIVE B.11.00 release with patches."
"/opt/ignite/data/Rel_B.11.00/config"
"/var/opt/ignite/data/Rel_B.11.00/core_700_archive_cfg"
"/var/opt/ignite/data/Rel_B.11.00/patch_bundle_cfg"
"/var/opt/ignite/data/Rel_B.11.00/misc_patch_bundle_cfg"
"/var/opt/ignite/config.local" }
```

Step 10. To force the installation of the new `Misc_Patches` bundle with the golden-image archive, add this line to the `sw_sel` clause for the patch bundle in `/var/opt/ignite/data/Rel_B.11.00/misc_patch_bundle_cfg`:

```
load_with_any = "golden image"
```

(This file was created with `make_config` in Step 8.)

IMPORTANT Most software distributed by HP, such as applications on DART CDs, are already bundles and will not need (and should not be) bundled again!

Avoiding Backup Patch Files

When loading HP-UX patches from SD depots, the files that are patched are normally saved, just 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 may want to turn this feature off.

The way you control this feature depends on whether you are loading HP-UX 10.*x* or 11.0/11i. It also differs if the patches are coming from the Core depot and being controlled by the `hw_patches_cfg` config file. See `/opt/ignite/share/doc/ace_hwe_setup` for more info on `hw_patches_cfg`.

For HP-UX 10.*x* releases

Control this feature by the existence of the file `/var/adm/sw/patch/PATCH_NOSAVE`. If you don't want to save the patched files, then you need to have a `pre_load_cmd` that touches this file. `pre_load_cmd` can be at the global level or in the `sw_source` for the patch depot. You can remove this file in a `post_load_cmd` if you want this feature re-enabled after the load is done. For example:

```
pre_load_cmd += "
   mkdir -p /var/adm/sw/patch
   touch /var/adm/sw/patch/PATCH_NOSAVE"
# Put PATCH_NOSAVE back to the way it was.
post_load_cmd += "
   rm -f /var/adm/sw/patch/PATCH_NOSAVE
"
```

For patches in the core depot that are loaded via the `hw_patches_cfg` config file, `PATCH_NOSAVE` is always created and put back the way it was after the core load is complete. See this file for details: `/opt/ignite/data/Rel_B.10.20/hw_patches_cfg`

For HP-UX 11.0/11i releases

Control this feature by this option in the `swinstall` command: `-xpatch_save_files=false|true`

You can use the `sd_command_line` keyword, either at the global level or within individual `sw_source` clauses depending on if you want it specified for all loads or just certain ones.

For patches in the Core depot, this option is specified by the `/opt/ignite/date/Rel.B.11.*/hw_patches_cfg` file. It is controlled by the config file variable: `_hp_patch_save_files` and is listed on the Additional Configuration Controls screen.

To specify this option at the global level (for example in the `/var/opt/ignite/config.local file`), you can add the line:

```
sd_command_line += " -xpatch_save_files=false "
```

To default the variable controlling the Core patches to "NO", add the following to `config.local` (which must be listed after `hw_patches_cfg` in the INDEX file):

```
init _hp_patch_save_files = "NO"
```

Avoiding Problems with Superseded Patches

When you are loading HP-UX 10.20 systems from multiple depots that contain patches, it's easy to run into the situation where patches in one depot supersede patches that have already been loaded from another. The superseded patches will prevent themselves from loading by giving an error. Ignite-UX indicates this failure by changing the client icon in the client-server screen to red.

What to do

To work around this problem, use the Ignite-UX `/opt/ignite/bin/fix_patches` script on each depot that contains patches. This script modifies the patch's `checkinstall` script so that it will "EXCLUDE" itself from loading without giving an ERROR.

See `/opt/ignite/share/doc/ace_hwe_setup` for more details. Although there is no manpage for `fix_patches`, enter the following to see command-line syntax:

```
/opt/ignite/bin/fix_patches -?
```

7 Using Golden System Images

This chapter describes how to build and install your own Ignite-UX installation media. Topics include:

- Installing from System Images.
- Creating an OS Archive.
- Configuring Ignite-UX Server to Recognize the OS Archive.
- Enable the Target System.
- Install the OS Archive on the Target.
- Restoring OnlineDiag LIF Volumes.

Examples in this chapter create a **golden system image** or **OS archive**, which is a snapshot of a known, good installation for use to copy to other systems. The copied (source) system is called the golden system image. The OS archive is a compressed tar or cpio archive that will be installed on other client machines.

Ignite-UX does not require creating a golden image to install a new client OS. Installing a golden image, however, is much faster (typically under 30 minutes) than installing the OS via `swinstall`.

Installing from System Images

In addition to supporting the standard OS installations from an SD depot, Ignite-UX supports installing from system images. This method recognizes that many, if not all, target nodes in a network may be identical (or almost identical) to each other. It is possible to take advantage of this fact by building an archive which contains all of the files you want installed on each of the targets and then using Ignite-UX to install them.

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 will depend on the size of the installation being done and the capacity of the network, but a typical system image can be extracted in about 20 minutes compared to about an hour for an SD install.
- Instead of troubleshooting a target, it is often more cost-effective to completely re-install with a known, good system image.
- When coupled with dataless nodes (all volatile data is on a separate file server), system replacement time or move time is drastically reduced.
- Once a system image has been created, it is simple to apply it to multiple target systems. Very little or no user interaction is required during these subsequent installs, reducing the chance of error.

Building this golden system image is done by setting up a single system the way that you want all of your systems to look, and then creating an archive of that system. Follow instructions in this chapter to set up the first system.

Creating an OS Archive

In general, the golden image is simply a system configured with all the software and customizations needed to distribute to a group of target systems. The golden image can be saved on tape or CD from the golden system and installed on individual systems. Or, the golden image can be stored on another system and installed remotely over the network.

Most large HP-UX sites already have the equivalent of a golden system, that is maintained by the IS certification or QA department. This system is configured with customer modifications on top of a base HP-UX system. Critical patches which all users need are installed onto the OS. Local, common software that all users use are also layered on the OS. The resulting system is tested to ensure proper operation in the customer's environment.

These systems represent a prototype or starting point for all users. The steps needed for install customizations are normally captured and are well known. They make good candidates for a golden image archive as explained here. If a golden system already exists, skip to "Configuring Ignite-UX Server to Recognize the OS Archive" on page 121.

Creating a golden system from scratch involves the following steps described in this section:

A. Install the HP-UX OS from media.

B. Load critical patches onto the OS.

C. Load optional HP and third-party software.

D. Customize the system.

Once you have a golden system with the base OS, use Ignite-UX to create an OS archive. It's up to the administrator, to define exactly what constitutes a golden system. You may choose to place patches, applications, kernel configurations, etc. on the golden system, or just include the Core OS. In our example, we only include the Core OS. For speed, you may want to place all of your common applications, patches and tools onto the golden system.

Ignite-UX is capable of installing systems from SD depots and/or archives. You may want to use this capability when setting up your golden system, since you will need to have a system installed before you can get an image.

A: Install HP-UX OS

Although this can be performed without an Ignite-UX server by using `swinstall` from CD or tape, this example uses Ignite-UX and a network depot as the source of our software.

Step 1. On the Ignite-UX server, set up the Core software to be distributed:

```
make_depots -r B.11.00 -a 700 -s
hpfclc:/release/S700_11.00/B3782EA
/opt/ignite/bin/make_config -r B.11.00
```

`make_depots` copies HP-UX B.11.00 software at the SD depot pointed to by the `-s` option (this pathname depends on the setup of the SD depot you are accessing) onto the local Ignite-UX server. (You can also run `make_config` and point it to the remote depot directly.)

`make_config` then adds this software as a configuration available for Ignite-UX installations.

Step 2. Begin installing HP-UX onto the target golden system by booting the target from the Ignite-UX server:

- If the target is currently running HP-UX, enter:

```
bootsys -v -w -f -i "HP-UX B.11.00 Default" target_hostname
```

- If the target is not currently running HP-UX, enter this on the target console:

```
boot lan install
```

Step 3. Select the configuration you've just set up, "HP-UX B.11.00 Default", and continue with the next section.

B: Load Critical Patches onto the OS

At this point you should have a target system with the basic HP-UX 11.0 release. If you have patches which you wish to distribute to all users, install them now. This is normally done using the standard SD tools.

For example, to install patch PHSS_8375:

Step 1. Download and unshar PHSS_8375 to obtain two files: `PHSS_8375.depot` `PHSS_8375.text`

Step 2. Install the patch non-interactively:

```
swinstall -x autoreboot=true -x match_target=true \
-s /PHSS_8375.depot
```

These instructions can also be found in the `PHSS_8375.text` file.

C: Load Optional Software

Load any optional HP and third-party software you want to make available to *all* users. Keep in mind that we are creating a golden system, and anything put on this will be distributed to all systems installed using the golden image. You'll need to keep in mind licensing restrictions, as well.

HP software (such as compilers) are normally loaded using SD from media or a network SD depot. Third-party software installation varies depending on the vendor.

D: Customize the System

Perform any customizations that you want to distribute to all users. These might include customized CDE login screens, base `/etc/passwd` files, additional phone tools and manpages, or corporate-wide default DNS and NIS setup. It would *not* include system, work-group or site-specific changes such as gateways, user accounts, or machine-specific networking. These will be taken care of by Ignite-UX later.

Use the next steps to create the golden image from the golden system, and configure Ignite-UX to use it. The `make_sys_image` command is provided to assist in creating the OS archive. See the *make_sys_image* (1M) manpage for details.

Step 1. Copy `/opt/ignite/data/scripts/make_sys_image` to `/tmp` on the golden system. Make sure it is an executable file. `/var/tmp` is the default location where `make_sys_image` stores the archive image. You can also have it save the image to a remote server that allows remote access from this client. Whichever method you choose, you will need to have sufficient disk space to hold the image. The amount of disk space will be approximately 1/2 the amount of data contained on your golden system (assuming about 50% compression ratio provided by 1).

IMPORTANT Do not use the system while `make_sys_image` is running in the next step. Device files are removed, and the host and/or networking information on the system are reset. After the command is complete these files are put back.

Step 2. On the golden system, run:

`/tmp/make_sys_image [`*`options`*`]`

By default, this will create a gzip-compressed archive in `/var/tmp` with the name *`hostname`*`.gz` , and all specific host information, device files, log files, and network information will be removed. Optionally, if you do not have enough disk space, or you would like for the archive to be 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`*

For example:

`/tmp/make_sys_image -d \`
`/var/opt/ignite/archives/Rel_B.11.00 -s 15.2.72.150`

The `make_sys_image` command can also build an archive containing any combination of tar, cpio, gzip and compress formats. We recommend tar and gzip formats.

Step 3. On the Ignite-UX server, create an archives directory to store the golden image:

`mkdir -p /var/opt/ignite/archives/Rel_B.11.00`

The `-p` option creates intermediate directories. It's best to keep the naming conventions `Rel_B.11.00` (or the release you're using.) This directory will need to be NFS exported if you'll be using NFS to transfer the archive to the target.

Step 4. Move the OS archive. For example, if hpfcnjm2 is the hostname: */var/opt/ignite/archives/Rel_B.11.00/hpfcnjm2.gz*

Configuring Ignite-UX Server to Recognize the OS Archive

To create an Ignite-UX configuration file for the OS archive, use the example file: `/opt/ignite/data/examples/core11.cfg`

Step 1. Create a copy of the example config file:

```
cp /opt/ignite/data/examples/core11.cfg \
/var/opt/ignite/data/Rel_B.11.00/core_700_archive_cfg
```

The destination file name is arbitrary. You can store configuration files anywhere on the system you chose. Ignite-UX manages the names and locations via the `INDEX` file (see Step 3 below). This file must be accessible via tftp.

Step 2. Modify the `core_700_archive_cfg` section to set up the OS archive for NFS transfer. Key changes are:

a. In the `sw_source` clause, change the following:

```
nfs_source =
"15.2.72.150:/var/opt/ignite/archives/Rel_B.11.00"
```

(This points to directory where the archive lives and must be NFS exported.)

b. In the `init sw_sel` clause, change the following:

```
description = "Archive HP-UX 11.00 CDE"
```

(This will now appear in the Environments section of the Ignite-UX user-interface as a menu choice).

```
archive_path = "hpfcnjm2.gz"
```

(This points to the actual file in combination with the `nfs_source` line).

c. Add `impacts` lines in the `init sw_sel` clause by executing:

```
/opt/ignite/lbin/archive_impact -t -g archive_file
```

and including the results in the file, replacing the example `impacts` lines. By default, this assumes that we created a tar archive that was gzipd.

Here is the complete `sw_sel` clause (some extra clauses in the example have been deleted for simplicity):

```
init sw_sel “golden image” {
description = “Archive HP-UX 11.00 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 = “hpfcnjm2.gz”
# ftp/remsh sources can use a full path:
# archive_path = “/pub/IUXarchives/B.11.00_700_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
```

Step 3. Add the new configuration file to Ignite-UX:

Edit the `/var/opt/ignite/INDEX` file to install a new “configuration” to Ignite-UX. For this example, add a new “cfg” clause as follows:

```
cfg “HP-UX B.11.00 archive” {
description “some description of this archive...”
“/opt/ignite/data/Rel_B.11.00/config”
“/var/opt/ignite/data/Rel_B.11.00/core_700_archive_cfg”
“/var/opt/ignite/config.local” }
```

The line of most interest is the one containing the `core_700_archive_cfg`, which is the config file we added in Step 2.2. The “config” and “config.local” are standard configurations.

`/var/opt/ignite/config.local` should be last. The last config file has the highest priority and can override values in prior config files.

The file `/opt/ignite/data/Rel_B.11.00/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.11.00 archive` will now appear as a valid configuration.

Step 4. Ensure NFS file system is exported correctly.

In the above `sw_source` clause, we specified the location of the OS archive to be a file on an NFS server. You need to ensure target systems have access to this directory.

Make sure the NFS configuration is correct. To view the current status and ensure the directory containing the archive is correctly exported, enter:

```
exportfs -v
```

Ignite-UX will automatically try to export `/var/opt/ignite/clients` for its use. In our example, `/var/opt/ignite/archives/Rel_B.11.00` must also be exported because that is where we placed the OS archive.

Here's our `/etc/exports` file:

```
/var/opt/ignite/clients -anon=2
/var/opt/ignite/archives/Rel_B.11.00 -ro,anon=2
```

If these are not correct, use SAM to set them up correctly.

Enable the Target System

Since the Ignite-UX server now knows about your new OS archive, you can use Ignite-UX to load the OS archive onto a target system. To do this, you need to get the target system to inform Ignite-UX that it is ready to install a new OS. There are two methods for doing this.

Method 1:

If the system is currently running HP-UX 10.x or higher — From the Ignite-UX server, use `bootsys` to reboot the target for which you wish to install the new OS. The target system can be booted in a mode in which it can be controlled by the Ignite-UX user interface.

```
/opt/ignite/bin/bootsys -w -v system_name
```

This will cause the target system to boot a copy of the Ignite-UX kernel and file system that `bootsys` copies to the target. An icon representing the system will appear in the Ignite-UX user interface on the server when the system has completed boot. (This may take several minutes.) An icon appears on the Ignite-UX UI when each client is booted.

If the server cannot resolve the system name, specify to `bootsys` the *system_name* and *IP_address*:

```
/opt/ignite/bin/bootsys -w -v system_name:IP_address
```

Method 2:

If the system does not have an OS — Manually reboot the system. Interrupt the boot process and select the Ignite-UX server as the lan boot source. This command will be slightly different depending on your target system. As an example, to install to a Model 712 workstation, enter the following from the boot admin mode:

```
boot lan.15.2.72.150 install
```

Older Series 700 workstations that use the RMP (rbootd) protocol instead of BOOTP require that you use the hardware LAN address of the server, and omit the `install` keyword:

```
boot lan.080009-123456
```

Replace the above IP/Ethernet addresses with the correct value for your Ignite-UX server. When prompted with a message about interacting with the IPL, respond **NO**

Install the OS Archive on the Target

In this section, we will use Ignite-UX to customize an OS install. Chapter 9 explains how this can be done with no user/administrator interaction.

Step 1. Run Ignite-UX by executing this as root:

```
/opt/ignite/bin/ignite
```

When the target has rebooted (using either `bootsys` or manual network boot), and is ready for installation, it will appear as an icon, labeled either as its original hostname (if rebooted using `bootsys`), or by the hostname supplied by DHCP, or at Ignite-UX screen.

Step 2. Select or click the icon of the system you wish to install.

Step 3. Select **Actions** –> **Install Client** –> **New Install**

You should now see the Ignite-UX screen with five tabs across the top. (If you see the System Hardware Inventory screen, simply select **OK** to bypass it).

Step 4. In the **Basic** tab (the top tab at startup), select:
Configurations: –> **HP-UX B.11.00 archive**

Ensure that the Root Disk, Root Swap and other fields are correct for your installation. *Any disks you select here will be over-written!* If you have a disk with existing user information you don't want to modify, add it manually after Ignite-UX has installed the OS.

Step 5. In the **Software** tab: Because there is only an archive at this point, the screen is blank. We'll add a patch and application depot later.

Step 6. In the **System** tab, select:
Final System Parameters –> **Set parameters now**
Fill in the blanks with the correct data.

Also fill in the appropriate data under **Set Time Zone**, **Network Services...** and, optionally, **Set Root Password**

Step 7. In the **File System** tab: Verify the correct disk usage. You can also add disks at this point or modify the disk and file system parameters. *A newfs will be performed on all selected disks!*

Step 8. In the **Advanced** tab: Nothing to specify here at this time. Later, we'll add post-process scripts to execute.

Step 9. When finished entering data, select **Go!** Review the data in the configuration dialog box and select **Go!** again.

Step 10. To display target installation status, double-click the target system icon on the Ignite-UX screen during execution.

Done!

In less than 30 minutes, the target system should have the new OS installed, a new kernel built, and the system rebooted and ready for use. Status of the target system will be shown on its icon, and in the status screen.

When using the Ignite screen

Ignite-UX determines the state of a target by reading the files in the `/var/opt/ignite/clients/`*`LLA`* directory. Seeing an icon on the Ignite-UX screen does not mean that the target actually exists, only that its config and control files exist in the Ignite-UX directories. We can use this behavior to our advantage to reinstall systems. This means that if you are reinstalling a system that Ignite-UX has already installed, you may need to either re-execute `bootsys` or boot the client from the Ignite-UX server.

Restoring OnlineDiag LIF Volumes

To restore OnlineDiag LIF volumes after installing from a golden image, set up a script in the `INDEX` file to be run as a post-configure script. For example, add this stanza to the `/var/opt/ignite/INDEX` file:

```
scripts {
   "/var/opt/ignite/scripts/diag.sh"
}
```

The `diag.sh` script is included as an example script in the `/opt/ignite/data/examples` directory. It runs this command which copies the OnlineDiag LIF volumes onto the root disk:

```
/usr/sbin/diag/lif/lifload -f /usr/sbin/diag/lif/updatediaglif
```

8 Customizing Your Installation

This chapter shows how to do local customizations using scripts:

- Using Post-installation Scripts.
- Installing Netscape® as a Post-config Step.

Other example uses of scripts to customize installations are in the *make_medialif* (1M) manpage.

Using Post-installation Scripts

Any number of tasks may be performed on the target system after the OS is installed by providing a script to be run on the target system. This section touches on some common examples, but scripts can easily be written to mount additional disk drives, add additional software, modify configurations based on system use, etc.

There are a number of points in the install process in which you can force scripts or commands to be run. Check the "Command and Script Execution Hooks" section on the *instl_adm* (4) manpage for specifics. One point to note is that `post_config_script` will run after all software has been loaded and the system has been booted with its final kernel, but *before* any of the normal `/etc/rc startup` scripts have been run.

Adding a Post-install Script

Example task

1. Create a script to perform the desired task. When Ignite-UX runs this script as a post-configuration, it will be run on the target system.
2. Add the script to your configuration file. Ignite-UX post-configuration scripts are defined using the `post_config_script` variable. For example, you can place this line into your `core_700_archive_cfg config` file:

   ```
   post_config_script += \
   "/var/opt/ignite/scripts/install_default_printer"
   ```

   The line above will define the `install_default_printer` script to be run as a post-installation process on the target system. The line should stand alone, placed *outside* of any clause (such as a `sw_sel` clause). By default, the script will always be run on the targets. You can change the behavior by navigating to **Install Client** –> **New install** –> **Advanced** tab.
3. If you want to make a script available under all configurations, add it to the `/var/opt/ignite/INDEX` file. Add the following to the end of this file:

   ```
   scripts {"/var/opt/ignite/scripts/install_default_printer"}
   ```

   It will then show up in the **Advanced** tab for all configurations.

NOTE Ignite-UX accesses scripts via tftp. Make sure the directory the script resides in is available to tftp by examining and/or changing the `/etc/inetd.conf` file.

Managing Network Printers

Example task One task an administrator generally needs to perform after a new OS installation is setting up printers. To automate this process, write a script which performs the HP-UX commands for adding a printer. Here is a script for adding a remote printer named “printbob”, and turning on the lp scheduler. The script turns SAM logging on for “commands-only”, performs the tasks desired, and extracts those commands from the SAM log file.

```
#!/sbin/sh
# Post process IUX script to add a local default printer
# Performing task “Add Remote Printer”: Adding “printbob”
#
/usr/sbin/lpadmin -pprintbob -ormhpfcmgw.fc.hp.com -orptsslj \
-mrmodel -v/dev/null -orc -ocmrcmodel -osmrsmodel
/usr/sbin/lpadmin -dprintbob
/usr/sbin/accept printbob
/usr/bin/enable printbob
# Turn on lp scheduler
#
lpsched -v
```

Installing Netscape® as a Post-config Step

Here is an example of using Ignite-UX post-installation scripts to load software on new installs. Netscape is one of those tools which seems to have a new version every six months. Due to the frequency of the changes, this tool may not make sense to include on the "golden system".

This example shows one way of accomplishing the task using a `post_config_script`. Another way would be to create a software selection (`sw_sel`) that would reference the tar archive, and then a `post_config_script` (or `post_config_cmd`) associated with the `sw_sel` that would be run only if the selection was picked for loading. Using a `sw_sel` would have the advantage of making it appear in the UI as just another software selection, and would have the `sw_impact` statements to ensure sufficient file system space. For more examples, see the files in `/opt/ignite/data/examples`.

NOTE Be sure the Netscape Navigator product is appropriately licensed prior to installation.

Step 1. Get Netscape Navigator — Netscape Navigator is typically pulled from one of the Netscape ftp server sites. The pulled files are gzip compressed tar images with an encoded name similar to:

```
netscape-v30-export.hppa1.1-hp-hpux.tar.gz
```

Step 2. Special Considerations for Netscape — In order to run Navigator, each user needs the correct network preferences. Unfortunately, these preferences cannot be defaulted, and must exist in every users `$HOME/.netscape` directory. To get around this limitation, we have supplied a "run-netscape" script. Instead of running "netscape", the user can run a link to "run-netscape" which will install the default preferences at first invocation.

A sample "run-netscape" script is shown below. You will also need to create a default configuration file. Merely take an existing one and remove all user and host specific information.

Step 3. Write an install and customization script — Attached below is a script we used for installing Netscape in our environment. The script does the following:

1. Remote copies from a server to the local target netscape, a default-preferences file, and the special run-netscape script.
2. Unpacks Netscape.
3. Makes `/usr/local/bin/netscape` a link to "run-netscape" to ensure user defaults will be installed.
4. Performs the special netscape customization.
5. Cleans up.

We named and placed our script under:

`/var/opt/ignite/scripts/install_netscape`

Step 4. Add the install script to Ignite-UX customization — Add a line like this to one of your config files (not in a clause):

```
post_config_script="/var/opt/ignite/scripts/install_netscape"
```

For details of adding a post configuration script, see Chapter 9. This script will need to be accessible using tftp.

Example script

Here's an example post-install script for Netscape:

```
# !/usr/bin/ksh
#
# Post Ignite-UX installation script used to install Netscape
version 3.0.
# This installation assumes HP-UX 11.00 because it
depends on gzip
# already loaded on the system.
#
PATH=${PATH}:/usr/sbin:/sbin:/usr/contrib/bin
IUX_SERVER=interop1.fc.hp.com
IUX_ARCHIVE_DIR=/var/opt/ignite/archives/Netscape
NETSCAPE_GZIP=netscape-v30-export.hppa1.1-hp-hpux.tar.gz
NETSCAPE_INSTALL_DIR=/opt/Netscape
NETSCAPE_RUN_DIR=/usr/local
echo "* Loading Netscape"
mkdir ${NETSCAPE_INSTALL_DIR} cd ${NETSCAPE_INSTALL_DIR}
rcp ${IUX_SERVER}:${IUX_ARCHIVE_DIR}/${NETSCAPE_GZIP}
${NETSCAPE_GZIP} rcp ${IUX_SERVER}:${IUX_ARCHIVE_DIR}/run-netscape
. rcp
```

```
${IUX_SERVER}:${IUX_ARCHIVE_DIR}/default-preferences .
gzip -dc ${NETSCAPE_GZIP} | tar -xvf -
echo "* Finished loading Netscape"# # Configure netscape
runtime # echo "* Configuring Netscape"
chmod 755 ${NETSCAPE_INSTALL_DIR}/run-netscape ln -s
${NETSCAPE_INSTALL_DIR}/run-netscape
${NETSCAPE_RUN_DIR}/bin/netscape
# # Install java_30 # mkdir ${NETSCAPE_RUN_DIR}/lib/netscape
ln -s ${NETSCAPE_INSTALL_DIR}/java_30 \
${NETSCAPE_RUN_DIR}/lib/netscape/java_30
# # Install plugins library # mkdir
${NETSCAPE_RUN_DIR}/lib/netscape/plugins ln -s
${NETSCAPE_INSTALL_DIR}/libnullplugin.so
${NETSCAPE_RUN_DIR}/lib/netscape/plugins/libnullplugin.so
mkdir ${NETSCAPE_RUN_DIR}/lib/netscape/mime.types mkdir
${NETSCAPE_RUN_DIR}/lib/netscape/mailcap
rm -f ${NETSCAPE_GZIP}
echo "* Finished configuring Netscape"
Example run time script for Netscape
#!/bin/sh
# # Put this script in /usr/local/bin/netscape
set -e
# Set this to the location of the real Netscape executable #
REAL_NETSCAPE=/opt/Netscape/netscape
# Set this to the location of the default preferences file. #
DEF_PREFS=/opt/Netscape/default-preferences
if [ ! -e $HOME/.netscape/preferences ]; then echo '(installing
default Netscape preferences...)' mkdir $HOME/.netscape
cp -p $DEF_PREFS $HOME/.netscape/preferences echo '(done)' fi
# The "-name" option is to avoid confusing the users' X resources.
# exec $REAL_NETSCAPE -name netscape $*
```

9 Automating Installations

This chapter shows how to use `bootsys` and configuration files to automate the Ignite-UX install process.

Setting up your Ignite-UX server so that the default configuration is correct for any given system will save you time and allow you to easily automate installations. This chapter discusses setting up the defaults the way you like them, as well as setting up a configuration for a specific target system.

Ignite-UX can install HP-UX on a target system with no additional configuration information (the default configuration as specified in the `/var/opt/ignite/INDEX` file will be used). You can, however, select from other configurations listed in the `INDEX` file on the `bootsys` command line.

Starting an Automatic Installation with bootsys

To start an automatic installation, enter:

`bootsys -a -v [-i` *`configuration`*`] [-f]` *`target_hostname`*

`-a` specifies an automatic install.

`-v` specifies verbose mode.

`-f` forces Ignite-UX to disregard prior configuration info for that target.

`-i` selects an alternate configuration. If not set, the default is used.

See the *bootsys* (1M) manpage for details on how to select a configuration and to force its use. The default is set in the Ignite-UX server options menu, or can be set manually with the `=TRUE` statement after a `cfg` clause in the `/var/opt/ignite/INDEX` file.

Ignite-UX will contact the target system and extract its hostname, IP address and default gateway. The default configuration is installed. Post install, Ignite-UX will reset the hostname, IP address and gateway to their original values. (remsh access to the target is required. If not available, `bootsys` will prompt the user for the root password on the target.)

This is the quickest way to install a system. The drawback is that you will receive the default config, which may have incomplete networking information unless you are using a previously “saved” configuration, or you specify the defaults in the `/var/opt/ignite/config.local` file as shown later.

Using a “Saved” Configuration

When using Ignite-UX during an install session, you may choose to save the result as a named configuration when finished specifying the configuration for particular target. This will save any changes that you made during the session for use in subsequent sessions. Then either specify the configuration as the default, and/or just use the name you give it to `bootsys` using the `-i` option.

Specifying Defaults in the config.local File

The /var/opt/ignite/config.local file is normally included in every cfg clause in the INDEX file. This provides a convenient location to store default parameters that are the same for all configurations. Typically this will be networking, default software selections, kernel modifications.

It may be easiest to cut and paste information written to the files /var/opt/ignite/clients/*/config by the user interface. However you can do more here than with the Ignite-UX screen. See the *instl_adm* (4) manpage for more details. Below is an example of what a config.local file could look like. The sw_sel's will depend on what you have defined in config files on the server.

```
dns_domain="fc.hp.com"
dns_nameserver[0] = "15.2.72.2"
nis_domain="udl"
wait_for_nis_server=FALSE
root_password="rlW2xSrugUvi2"
timezone="MST7MDT"
ntpdate_server="15.1.48.11"
init sw_sel "Misc_Patches"=TRUE
init sw_sel "B3919DA_AGP"=TRUE
mod_kernel += "maxuprc 100"
mod_kernel += "dbc_max_pct 80"
```

Always run this after making manual edits to verify that the syntax is correct:

```
instl_adm -T
```

See "Setting Install Parameters Dynamically" on page 143 to see how default information may be specified dynamically depending on the target system's configuration.

Setting defaults with instl_adm

Some network parameters need to be known by the target clients when they first boot. bootsys or DHCP/BOOTP can supply the hostname and IP address; however, the netmask and gateway need to be supplied in the RAM filesystem (INSTALLFS). This can done by using the instl_adm command, which has options to set netmask, gateway, Ignite-UX/tftp server, etc. Or you can dump the current settings to a file and edit it, then load the settings back. Just loading Ignite-UX sets some of the parameters.

For example, you may want to set the keyboard language so that it never prompts you for it when booting from Ignite-UX. The file you store using **instl_adm -f** may look like this:

```
# instl_adm defaults:
server="15.2.72.150"
route_gateway[0]="15.2.70.1"
route_destination[0]="default"
netmask[]="255.255.248.0"
# end instl_adm defaults.
kbdlang="PS2_DIN_US_English"
```

Using the per-target client config file

Until now, we have discussed specifying default parameters that all target systems may use. If you would like to specify a specific configuration for an individual target system, you may use the following procedure.

When Ignite-UX begins an install session, it scans the directory `/var/opt/ignite/clients` for a directory matching the *LLA* of the target system. As an example, if the *LLA* of the target is `0x08000992E346`, Ignite-UX looks for a file named `config` in: `/var/opt/ignite/clients/0x08000992E346/config`

Ignite-UX keeps the last configuration installed to the respective system in this file so it can perform a repeat install.

If found, the configuration data in this file is used to overwrite the default values. This file has the highest precedence over all other config files listed in the `INDEX` file.

CAUTION Ignite-UX will write over this file at the end of the install, so you may want to keep an original copy elsewhere.

The easiest way to create the config file is to use one already built by Ignite-UX. If you've previously installed a system (it's best to use one from a similar system to your target,) you can find a config file in the `/var/opt/ignite/clients/`*LLA* directories. Use this as the basis for your new file. Copy it to: `/var/opt/ignite/clients/`*LLA*`/config`

Edit its contents to correspond to your new system.

Example config Ffle

Here is an example config file:

```
cfg “HP-UX B.11.00 archive”=TRUE
_hp_cfg_detail_level=”ipvs”
#
# Variable assignments
# init _hp_disk_layout=”Whole disk (not LVM) with HFS”
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 “golden image”=TRUE
init sw_sel “English”=TRUE
#
# System/Networking Parameters
# hp_custom_sys+={“Current System Parameters”, “Original Defaults”}
init _hp_custom_sys=”Current System Parameters”
_hp_custom_sys help_text “Final System/Networking Parameters”
(_hp_custom_sys==”Current System Parameters”)
{
final system_name=”hpfcnjm2”
final ip_addr[“lan0”]=”15.2.75.14”
final netmask[“lan0”]=”255.255.248.0”
final dns_domain=”fc.hp.com”
final dns_nameserver[0]=”15.2.72.254” TIMEZONE=”MST7MDT”
is_net_info_temporary=TRUE
}
# end “Current System Parameters”
```

Typically, you would want to change the networking parameters to the correct values. For example:

```
final system_name=”system11”
final ip_addr[“lan0”]=”15.2.75.193”
```

The values specified should be self explanatory, and should be edited to the desired new values. It is also possible to add kernel parameters to this file. See “Setting Install Parameters Dynamically” on page 143 later in this chapter.

You should also update the variable, `_hp_cfg_detail_level` when adding new types of parameters or they will get lost by the UI or by the file rewrite.

To perform an automatic install with a config file:

Step 1. Determine the *LLA* of the target system, either through the `boot_admin` commands (at bootup) or with the `lanscan` command.

Step 2. Create the following directory (assuming the *LLA* is `0x08000992E346`) and copy in your config file:

```
mkdir /var/opt/ignite/clients/0x08000992E346
```

```
cp config /var/opt/ignite/clients/0x08000992E346/config
```

Step 3. Since these files will be accessed using NFS, make sure they have the correct permissions:

```
chown bin:bin /var/opt/ignite/clients/0x08000992E346
```

```
chown bin:bin \
/var/opt/ignite/clients/0x08000992E346/config
```

Step 4. Run `bootsys`:

```
bootsys -a -v target_hostname
```

Ignite does not need to be running. Ignite-UX will install the default configuration (or the configuration specified with the `-i` option) and will include the specific changes provided in the config file.

The target system should boot into the Ignite-UX install process and complete the install automatically. Errors will be reported on the client screen and in the `install.log` file.

Monitor the install process via this file:

```
/var/opt/ignite/clients/0x08000992E346/install.log
```

Setting the Local Time Zone

The TZ environment variable governs what time zone the message in the `install.log` contain. The "time zone" config file keyword does not have any effect on the messages that occur during the install, but does determine the time zone setting on the target system (the two can be independently set).

To set the TZ environment variable, it is best to do so in the INSTALLFS file so that it is set as early in the process as possible. However, the first message will still be in EST since it is produced before the config file contents of INSTALLFS are read. The procedure for setting this to MST7DT is:

```
/opt/ignite/bin/instl_adm -d > /tmp/cur_cfg
echo 'env_vars += "TZ=MST7MDT"' >> /tmp/cur_cfg
/opt/ignite/bin/instl_adm -f /tmp/cur_cfg
```

Scheduling Installations

Client installations are easily automated via the `cron` daemon. For repeated installations, add `crontab` entries (see *cron* (1M) and *crontab* (1) manpages). For single installations, use the `at` command. For example, to perform an installation on a target system at 8:00 PM using the `at` command, as root enter:

```
at 8:00pm
```

```
bootsys -a -v target_system
```
(press **Ctrl-D**)

Setting Install Parameters Dynamically

Ignite-UX can make intelligent decisions about install parameters when it runs, based on information it reads from the target system. Instead of forcing static values for example, swap size or kernel parameters, the best values for these can be determined based on the characteristics of the target system.

This can make configurations set up by the system administrator 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. The variables and syntax are documented in the *instl_adm* (1M) manpage.

Example

This example sets the primary swap size of the target system root disk dynamically at install time based on the size of the disk, and on the size of the target system RAM. The algorithm will set swap to 125MB if the disk is large (> 500MB) and if the amount of system RAM is greater than 64MB. If the disk is small, make the swap very small to maximize the amount of space available for HP-UX.

Step 1. Add these lines to the end of the file `/var/opt/ignite/config.local` to be the default for all configurations:

```
# default to very minimal swap of 25MB
# unless the disk is larger than 500 MB
# and we have more than 64MB ram
(disk[_hp_root_disk].size > 500MB & memory > 64MB)
{
   init _hp_pri_swap=125MB
}
else
{
   init _hp_pri_swap=125MB
}
```

You could also put this in a separate file, say, `/var/opt/ignite/data/Rel_B.11.00/custom_cfg`, and add the file name to the `INDEX`.

This could also be added to the `config` file created for automatic installs. Note that if the `_hp_pri_swap` parameter is set later in the order of files searched in the "cfg" definition, this setting will be overwritten. The order the files are evaluated is documented in the *instl_adm* (1M) manpage and in Chapter 3. Also be aware that the `config` file used for automatic installs is overwritten as part of the install process.

Step 2. To force the load of a patch bundle if the target system matches the regular expression `71*`, such as a `710` or `712`, add the following lines to the end of the file:

```
/var/opt/ignite/data/Rel_B.11.00/custom_cfg
# check for H/W model 71x
# and add the Misc_Patches bundle if true
(hardware_model ~ "9000/71*") {init sw_sel "Misc_Patches" = true}
```

Step 3. Run a previously created post-install script and increase a tunable kernel parameter if we determine our target system is a Model 755. If not, it sets a default value for the kernel parameter:

```
post_config_script += "/var/opt/ignite/scripts/755special"
(HARDWARE_MODEL == "9000/755") {
   mod_kernel += "maxuprc 300"
} else {mod_kernel += "maxuprc 100"}
```

Step 4. Select an entirely different default configuration based on the size of the system RAM and disk. For this to have effect, it must go into the `INSTALLFS` file by using `instl_adm` as described earlier:

```
# For a system with only one disk and small memory, select
# the "small system configuration"
(num_disks == 1 & memory < 64MB )
{cfg "small system configuration" = true}
```

Step 5. To check the syntax of all configuration files that are listed in the `/var/opt/ignite/INDEX` file, enter:

```
instl_adm -T
```

Step 6. To check the syntax of a file that is not yet in the `INDEX` file, enter:

```
instl_adm -T -f file
```

10 Creating Your Own Install Media

This chapter explains how to create custom installation media to use with Ignite-UX. It's assumed here that you have a basic knowledge of Ignite-UX operations, as explained in the previous chapters.

Why Use Custom Install Media?

You may want to build customized install media if:

- You have a large number of systems that are basically identical, and:
  - — The systems do not have network boot capability, or...
  - — The networking will not allow easy or fast access to an Ignite-UX server, or...
  - — The systems are geographically widespread.
- You have HP servers that lack network boot, and so a boot medium is required to contact the Ignite-UX server.
- You want to deliver media to a technician or operator and have the entire install process automated without human intervention or interaction.
- You want a single media that contains all the desired parts of the operating system (HP-UX, applications, patches, diagnostics and local customizations).

Using customized install media also provides both system standardization and customization simultaneously. The standardization comes from using golden system images which contain a base operating system, applications, patches, third-party software and local customizations, already packaged into an archive. The entire system has been tested, verified and tuned before creating the image. This image can be the starting point for all installs to ensure standardization.

The customization comes from using config files to load additional software, change kernel parameters, and run scripts. Software bundles can be:

- Interactively chosen.
- Pre-selected, unconditionally or conditionally.
- Invisible.

There are also parameters that control the environment in which Ignite-UX operates. The most important parameters are `run_ui` and `control_from_server`. When `run_ui` is false, no interaction will occur and the load will proceed according to all the configuration information

provided to Ignite-UX. When `control_from_server` is true, an attempt will be made to contact the Ignite-UX server as defined in the configuration information. These modifications are explained in the procedures in this chapter.

Using a custom install media allows you to chose how things work, what you will leave up to the end user, what will happen automatically, and so forth.

Building Example Install Media

The remainder of this chapter describes building custom install media that meets these requirements:

- It will be shipped worldwide so that systems can be installed with a golden image.
- Both a Series 700 workstation and a Series 800 server golden images will be built with the `make_sys_image` command.
- There is a set of applications that are to be chosen interactively by the end user.
- All software will come from the media, and there will be no contact with nor need for an Ignite-UX server during installation. However, you do need an Ignite-UX server to create the media and execute Ignite-UX commands.

Building an Install Tape

This section describes the golden image layout and building an example install tape. We recommend using only 90 metre DDS-1 tapes to make install tapes with Ignite-UX to ensure that the tape will work with any DDS drive. For details on other supported tape formats, see the release notes available from the Ignite-UX Web site.

The Golden Image

Golden system images in these examples were created on Series 700 and Series 800 systems running HP-UX 10.20 using:

`/opt/ignite/data/scripts/make_sys_image`

The archives are in tar format and are gzipped. The files are:

```
/var/tmp/archive_700.gz
/var/tmp/archive_800.gz
```

This command has been run on these archives to get disk-space usage information (impacts) so that configuration information can be supplied:

`/opt/ignite/lbin/archive_impact`

DDS tape layout

A DDS install tape is constructed logically like this:

Table 10-1 DDS Install Tape Construction

LIF	A1/E	D/A2	A3	A4	...

LIF — A bootable tape starts with a Logical Interchange Format (LIF) volume containing all the components required to boot off the tape. It also includes the Ignite-UX toolset and configuration information that controls how Ignite-UX will operate. It includes config file information about the SD depot on the tape (should there be one) and all archives on the media.

A1/E — The next portion is either the first OS Archive (A1) or is Empty (E) if the installation is solely from the software depot.

D/A2 — The next portion is either a serial depot (D) or another OS archive (A2). There can only be one depot on a tape, and it must be the third file on the tape due to a SD restriction.

A3, A4,... — Beyond this, there may be other archives, limited only by the capacity of the tape. If more archives are needed, they can be put on a second medium.

LIF volume content

The `make_medialif` command is used to create the LIF volume. A typical LIF volume looks like this:

```
volume ISL10 data size 175771 directory size
filename   type   start   size     implement  created
ISL        -12800 16      240      0          98/02/10 14:06:38
AUTO       -12289 256     1        0          98/02/10 14:06:38
INDEX      BIN    264     1        0          98/02/10 14:06:38
CONFIG     BIN    272     58       0          98/02/10 14:06:38
HPUX       -12928 336     800      0          98/02/10 14:06:38
INSTALL    -12290 1136    57503    0          98/02/10 14:06:45
INSTALLFS  -12290 58640   31774    0          98/02/10 14:06:48
INSTCMDS   BIN    90384   9873     0          98/02/10 14:06:51
SYSCMDS    BIN    100264  45901    0          98/02/10 14:07:02
SCRIPTS     BIN    146168   30       0           98/02/10 14:07:02
```

ISL — Initial System Loader. 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`

AUTO — Autoexecute file defines the default (possibly automatic) boot behavior.

INDEX — Default `INDEX` file (it has the same function as `/var/opt/ignite/INDEX` does on an Ignite-UX server). The file `CONFIG` is referenced in this file.

CONFIG — Contains all software configuration information. You should begin with the default config file for that release (for example, for the Ignite-UX B.10.20, look in: `/opt/ignite/data/Rel_B.10.20/config`) Additional config files can be added via the `-f` option of the `make_medialif` command. Information in this file will allow complete access to all the archives and depots on the media.

HPUX — HP-UX bootstrap utility. It is also extracted from the default boot file.

INSTALL — 32-bit kernel booted by 32-bit install clients. With a 64-bit kernel, use the `make_medialif -o 64` option to create a LIF volume called VINSTALL for V-class systems. For other 64-bit kernels, use the `-o 64w` option to create WINSTALL files. Use the `make_medialif -a` option to create all `NSTALL` and `INSTALLFS` files.

INSTALLFS — The RAM file system used by install clients. Configuration information stored in the first 8KB of this file is accessible using `instl_adm`. If this is a 64-bit LIF volume, the file is called `VINSTALLFS`. For 64-bit systems, the file is `WINSTALL` .

INSTCMDS — gzip archive of the commands needed for disk layout. These commands run on the install kernel and inside the `INSTALLFS`.

SYSCMDS — gzip archive of commands used to load the software onto the system. There are different archives for each release.

SCRIPTS — gzip archive of all `post_load` and `post_config` scripts that are required. By default when loading a core archive (`load_order` is zero, which means it gets loaded first), the two scripts in `/opt/ignite/data/scripts` called `os_arch_post_l` and `os_arch_post_c` are executed.

Scripts like these are discussed in depth in the *instl_adm* (1M) manpage.

For more information on what happens during system boot and what files do what, see “System Bootup Sequence” on page 162.

Important Config Files

You need to consider two important config file concepts when building archives and bundles to be loaded onto a target system:

- `sw_source` — specifies the access method to either an archive or a depot.
- `sw_sel` — specifies the path of an archive or a bundle in a depot.

For details on these objects, see the *instl_adm* (1M) manpage.

Be sure to pass all user-generated config files through this command to check for syntax errors:

`instl_adm -T -f cfg_file`

Accessing a DDS Tape Archive

The best place to start when creating a config file for an archive is to use the template file supplied by Ignite-UX in `/opt/ignite/data/examples/`. Other files are available for HP-UX 11.0/11i 32- and 64-bit systems. This file can be copied elsewhere, say, to `/var/tmp/archive.cfg`, and then edited to suit your situation.

Assume that our tape will be used more for installing Series 700 systems than Series 800 systems. Hence, we will put the Series 700 archive on the tape first. In the diagram above, it will be `A1`. Since there will be a serial depot on the tape, the Series 800 archive will be located at `A3`.

To modify the config file to access the Series 700 archive, the following attributes need to be changed in `/var/tmp/archive.cfg` in the `sw_source` core archive stanza:

Table 10-2

Attribute	Old Value	New Value
`source_type`	NET	MT
`change_media`	#change_media=FALSE	change_media=FALSE
`nfs_source`	nfs_source=*IP*:*depot*	#nfs_source=*IP*:*depot*

These changes will modify the source type from network (NET) access (which is either NFS, ftp or remsh) to magnetic tape (MT). Since the archive is going to reside on the same media, `change_media` is set to false by un-commenting that attribute. To avoid trying to NFS mount that directory, the `nfs_source` is commented out.

Since the template file already has conditional logic that provides for different Series 700 and Series 800 archives, we will use that to our advantage.

Inside the stanza enabled by `HARDWARE_MODEL ~ 9000/7.*` the following fields must be changed:

Table 10-3

Attribute	Old Value	New Value
`archive_path`	B.10.20_700_CDE.gz	1
`impacts`	/ 27KB	(as reported by `archive_impact`)

The change in `archive_path` indicates that there is one EOF mark to skip on the tape and the archive will begin right after that mark. The archive will be the second file on the tape after the LIF volume. The impacts lines must be replaced with whatever was reported by `archive_impact` for the Series 700 archive.

Inside the same stanza the `sw_sel` and `description` strings can be changed to something more descriptive and applicable to your situation. The text inside the double quotes can be changed to whatever you like. They will be visible on the Ignite-UX UI. `archive_type` must match what was done by `make_sys_image`. See *instl_adm* (4) for more about `archive_type`.

Since we have only one Series 700 archive the entire stanza called `golden image2` can be deleted. It was included in case you had two different types of archives, for example one for VUE and one for CDE. If more than one archive per architecture is on the media, it is advisable to use an exrequisite attribute between them so only one archive can be selected at one time.

In the stanza for the Series 800 (it is an else clause later in the file) the same sort of changes must be made. However remember that the Series 800 archive is in a different location on the tape. So these attributes need to change:

Table 10-4

Attribute	Old Value	New Value
`archive_path`	B.10.20_700_CDE.gz	3
`impacts`	/ 27KB	(as reported by `archive_impact`)

The change in `archive_path` indicates that there are three EOF marks to skip on the tape (LIF volume, Series 700 archive, and the serial depot). The Series 800 archive is the fourth file on the tape. The impacts lines must be replaced with whatever was reported by `archive_impact` for the Series 800 archive.

It is important not to 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.

Accessing the Serial Depot on a DDS Tape

Assume there is a depot (`/var/tmp/depot`) that contains all the applications you wish to install on top of the archive. It can be a mixture of Series 700-only applications, Series 800-only applications, and applications that can be loaded on both architectures. Use the `make_config` command to create config file information for this depot, and the config file is modified to reflect the ultimate destination of the depot.

Step 1. Create config files by entering these commands:

```
make_config -s /var/tmp/depot -a 700 -c
/var/tmp/depot_700_cfg

make_config -s /var/tmp/depot -a 800 -c
/var/tmp/depot_800_cfg
```

On a tape, the depot must be the third file so there is no need to specify a path to the depot.

Step 2. Change both config files by removing these attribute lines:

```
sd_server = IP_address
sd_depot_dir = /var/tmp/depot
```

Step 3. Change this attribute in both files:

Table 10-5

Attribute	Old Value	New Value
source_type	NET	MT

The deleted information is not needed when accessing a serial depot on a tape. The change to source_type indicates that the depot is located on a tape instead of over the network.

Step 4. Create the Serial Depot. The depot put on a DDS tape is known as a serial depot. It can exist as a regular file, but it cannot be accessed remotely.

To create a serial depot from /var/tmp/depot and store it in /var/tmp/serialdepot, enter:

```
swpackage -s /var/tmp/depot -x target_type=tape \
@ /var/tmp/serialdepot
```

Step 5. Assembling the DDS Tape. Now that all the components of the tape are done, create the LIF volume in /var/tmp/lifvol using the make_medialif command:

```
make_medialif -f /opt/ignite/data/Rel_B.10.20/config \
-f /var/tmp/archive.cfg -f /var/tmp/depot_700_cfg \
-f /var/tmp/depot_800_cfg -l /var/tmp/lifvol
```

This creates the LIF volume that includes all the configuration information, including defaults Ignite-UX provides and information on the archives and depot. For HP-UX 11.0 systems, also use the -a option to include all INSTALL and INSTSALLFS files.

Step 6. Modify INSTALLFS Config. Change configuration information in INSTALLFS. To set run_ui and control_from_server variables using instl_adm to TRUE and FALSE, respectively, based on our scenario:

```
instl_adm -d -F /var/tmp/lifvol > /var/tmp/cfg

vi /var/tmp/cfg      #Add/change the two variables

instl_adm -T -F /var/tmp/lifvol  #Check syntax

instl_adm -d -F /var/tmp/lifvol #Verify changes
```

Step 7. Make a New Device File. DDS-1 density is used so that the tape is more readily readable by all DDS tape devices, which are notorious for being finicky at times. To create a device with these characteristics, enter:

```
ioscan -fC tape  #get the hardware path
mksf -v -H hardware_path -b DDS1 -n -a
```

You can also create this file using SAM:

1. Click: **Peripheral Devices** –> **Tape Drives**
2. Select (highlight) the tape drive you want to use:
   Actions –> **Create Device Files** –> **Create Custom Device File**
3. Change **DENSITY** to **DDS1**; turn off **Compressed Mode** and **Rewind on Close**
4. Click: **OK**

Step 8. Create the Install Tape. Create the tape using a DDS-1 density, no compression, no rewind device file (for example `/dev/rmt/c0t3d0DDS1n`):

```
mt -t /dev/rmt/c0t3d0DDS1n rew

dd if=/var/tmp/lifvol of=/dev/rmt/c0t3d0DDS1n obs=2k

dd if=/var/tmp/archive_700.gz \
of=/dev/rmt/c0t3d0DDS1n obs=10k

dd if=/var/tmp/serialdepot \
of=/dev/rmt/c0t3d0DDS1n obs=10k

dd if=/var/tmp/archive_800.gz \
of=/dev/rmt/c0t3d0DDS1n obs=10k

mt -t /dev/rmt/c0t3d0DDS1n rew
```

The tape is now ready for installations.

Building an Install CD-ROM

CD-ROM layout

There are similarities between putting a CD-ROM together and putting a tape together, as explained in the previous pages. One major difference, however, is in disk space usage. You have to create a logical volume (or provide a whole disk) large enough to hold the archives and the depots. This paper will assume that a logical volume is used. You need that much space again to copy the raw logical volume to a regular file. So you probably end up using around three times the disk space consumed by your archives and depots.

A bootable CD-ROM is not a serial device like a tape. It has a file system on it, and it also has a LIF volume that contains the same information as above except for the config files which describe the archives and depots. Access to these objects is somewhat different.

The file system on the CD-ROM can be either HFS or CDFS. You can create an HFS file system using standard HP-UX commands. Various third-party applications are available for a CDFS file system. Note that there is less capacity on a CD-ROM (650 MB) than on a 90 meter DDS-1 tape (2GB).

Step 1. Create the logical volume. Assume that the logical volume that will be used is `/dev/vg00/image`, and it is mounted at `/var/tmp/image`. Also assume that an HFS file system will be used. Using HFS and standard HP-UX commands, create the logical volume (assume everything fits in 500 MB):

```
lvcreate -L 500 -n image vg00

newfs -F hfs -f 2048 /dev/vg00/rimage

mkdir -p /var/tmp/image

mount /dev/vg00/image /var/tmp/image
```

For CDFS file systems there are similar commands. Check the software supplier documentation.

Step 2. Access a CD-ROM Archive. The file system in the logical volume will contain both the archives and the depots. Place the archives in the CD-ROM image by copying them into the file system just created:

```
cp /var/tmp/archive_700.gz /var/tmp/image

cp /var/tmp/archive_800.gz /var/tmp/image
```

Step 3. Creating config file information for the archives is similar to what was done for the DDS tape. Start with a new copy of `/opt/ignite/data/examples/core.cfg` in `/var/tmp/archive.cfg`

To modify the config file to access the Series 700 archive, the following attributes need to be changed in `/var/tmp/archive.cfg` in the `sw_source` core archive stanza:

Table 10-6

Attribute	Old Value	New Value
source_type	NET	DSK
change_media	#change_media=FALSE	change_media=FALSE
nfs_source	nfs_source=*IP*:*depot*	#nfs_source:*IP:depot*

These changes will modify the source type from a network (`NET`) access to CD-ROM (`DSK`). The other changes are as described before.

Step 4. Inside the stanza enabled by `HARDWARE_MODEL ~ 9000/7.*` the following fields must be changed:

Table 10-7

Attribute	Old Value	New Value
archive_path	B.10.20_700_CDE.gz	archive_700.gz
impacts	/ 27KB	(as reported by archive_impact)

The change in archive_path indicates that the archive will be found in the pseudo-root of the CD-ROM in a file called archive_700.gz . Ignite-UX will prepend the mount point it uses to access the archive. As before, the correct set of impacts lines need to be included.

Step 5. Again you can change the sw_sel and description strings to something more descriptive and applicable to your situation. The text inside the double quotes can be changed to whatever you like. Note that archive_type must match what was done by make_sys_image.

Step 6. Delete the entire stanza called golden image2 again.

Step 7. In the stanza for the Series 800 (it is an else clause later in the file) the same sort of changes must be made. These attributes need to change:

Table 10-8

Attribute	Old Value	New Value
archive_path	B.10.20_700_CDE.gz	archive_800.gz
impacts	/ 27KB	(as reported by archive_impact)

The change in archive_path indicates that the archive will be found in the pseudo-root of the CD-ROM in a file called archive_800.gz. Ignite-UX prepends the mount point it uses to access the archive. As before, the correct set of impacts lines need to be included. It cannot be emphasized enough not to change anything else in the file.

Step 8. **Create and Access the CD-ROM Depot.** Tape is restricted to a single depot. That restriction does not apply to a CD-ROM. However, we will use a single depot for simplicity sake. Create the depot using swcopy to a target in the logical volume:

```
swcopy -s /var/tmp/depot \* @ /var/tmp/image/depot
```

Step 9. Once again, use make_config to create the start of the config files for the depot:

```
make_config -s /var/tmp/depot -a 700 -c
/var/tmp/depot_700_cfg

make_config -s /var/tmp/depot -a 800 -c
/var/tmp/depot_800_cfg
```

Step 10. Edit config Files. Since the SD server is the system that is being installed, remove this attribute. Change both config files by removing this attribute line:

```
sd_server=IP_address
```

Step 11. Change these attributes in both files:

Table 10-9

Attribute	Old Value	New Value
source_type	NET	DSK
sd_depot_dir	var/tmp/depot	depot

The change to source_type indicates that the depot is located on a CD-ROM instead of over the network. The change in sd_depot_dir indicates that the depot will be found in the pseudo-root of the CD-ROM in a depot called depot. Ignite-UX prepends the mount point it uses to access the depot.

Step 12. Assemble the CD-ROM. The raw file system just created must be copied into a regular file so it can be written to the CD:

```
umount /var/tmp/image

dd if=/dev/vg00/rimage of=/var/tmp/fs_image bs=1024k
```

Step 13. Create the LIF Volume. Now that most of the components of the CD-ROM are complete, use make_medialif to create the LIF volume:

```
make_medialif -f /opt/ignite/data/Rel_B.10.20/config \
-f /var/tmp/archive.cfg -f /var/tmp/depot_700_cfg \
-f /var/tmp/depot_800_cfg -l /var/tmp/lifvol
```

This creates the LIF volume that includes all the configuration information. It includes the defaults Ignite-UX provides and provides the access to the archives and the depot.

Step 14. Modify INSTALLFS Config. Change configuration information in INSTALLFS. To set run_ui and control_from_server variables using instl_adm to TRUE and FALSE, respectively, based on our scenario:

```
instl_adm -d -F /var/tmp/lifvol > /var/tmp/cfg

vi /var/tmp/cfg   #Add/change the two variables

instl_adm -T -F /var/tmp/lifvol   #Check syntax

instl_adm -d -F /var/tmp/lifvol   #Verify changes
```

Step 15. These two objects (the raw file system and the LIF volume) must be combined using instl_combine. The result is a single file with the LIF volume wrapped around the file system, which can then be written to the CD:

```
/opt/ignite/lbin/instl_combine -F /var/tmp/lifvol \
-C /var/tmp/fs_image
```

Step 16. Complete the CD config. Using your CD-ROM writer software, copy /var/tmp/fs_image to the CD.

The CD-ROM is now ready for installations. You can test the image out before burning a CD by copying it to an unused raw disk and rebooting the system off that disk.

Media from `make_tape_recovery`

There are many similarities between a tape produced by `make_tape_recovery` and one constructed by the method described here. In fact, `make_tape_recovery` performs many of the same steps. But there are some important differences as well. The primary purpose of a recovery tape is to restore enough of a system to get it going following some catastrophe, so the rest of the system can be recovered from backups.

Additional Archives and Depots Media

If there is insufficient space on either a tape or a CD to hold all the archives and depots, it is possible to put them onto separate media. In this case, the config file describing these archives or depots would have `change_media` set to true. Ignite-UX would prompt the user for the new medium. If this is a CD, the `instl_combine` step is needed only for the first CD.

System Bootup Sequence

This sequence of events occurs when an HP computer system boots up:

Step 1. The firmware determines from which device to boot via either user input or primary path.

Step 2. The firmware looks for a LIF header on that device, and if it finds it, it looks in the LIF header for where the ISL starts.

Step 3. The firmware loads the ISL into memory from the boot device and executes it. It passes a flag to it that indicates whether to run interactively or to autoboot.

Step 4. If the ISL is interactive then it gives the `ISL>` prompt and waits for user input before proceeding.

Step 5. If the ISL is not interactive, then it looks for the `AUTO` file on the boot device to determine what to run next.

Step 6. The `AUTO` file or user input usually supplies the `hpux` *`args`* command. This tells ISL to load the program `HPUX` from the LIF header on the boot device and to run it with the given arguments.

Step 7. The hpux program (also known as the secondary loader) figures out what HP-UX kernel to load, and what arguments to pass to it (like init state).

Step 8. `hpux` loads the kernel and starts running it.

Step 9. For the INSTALL kernel, the kernel looks at its name and realizes that it fits the pattern `*INSTALL` and then loads the matching `*INSTALLFS` file from the boot device.

In all these cases the firmware (Processor Dependent Code, or PDC) API services are used when accessing the boot device.

11 System Recovery

This chapter describes important system recovery tools available with Ignite-UX:

- Creating a Bootable Recovery Tape.
- Duplicating make_tape_recovery Tapes.
- Creating a Recovery Archive via the Network and Tape.
- Archive Creation Steps.
- Verifying Archive Results.
- Retaining “Known-good” Archives.
- Making config File Additions.
- Selecting File Systems During Recovery.
- Tape Recovery with make_net_recovery.
- Tape Recovery with No Tape Boot Support
- Notes on Cloning Systems.
- Expert Recovery Using the Core Media.
- System Recovery Questions and Answers.

NOTE The `make_tape_recovery` tool replaces `make_recovery` for creating recovery tapes, beginning with Ignite-UX A/B 3.2, March 2001. With the `make_tape_recovery` command, you can make recovery archives on local and remote systems, as explained in this chapter. For more details, see the *make_tape_recovery* (1M) manpage.

Overview

HP-UX provides two recovery methods as part of the standard product: system recovery and expert recovery. The method you use depends on the situation.

System Recovery via Network and Tape

The system recovery tools available with Ignite-UX allow you to quickly recover from a failed disk (root disk or disk in the root volume group). The failure can be either a hardware failure or a catastrophic software failure.

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: use the `make_net_recovery` command to create an archive on another system or the `make_tape_recovery` to create an archive on tape.

The `make_tape_recovery` and `make_net_recovery` commands both create a bootable recovery (install) archive which is customized for your machine. The archive contains your system's configuration information (disk layout, etc.) and files on your root disk or root volume group. You can exert some control over which files are saved as part of the archive.

Once you have a recovery archive on tape or another system, 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, recover the latest copies of files from the last system backup.

Expert Recovery

Expert recovery, formerly called Support Media Recovery, allows you to recover a slightly damaged root disk or root volume group. With this method, you repair the boot/root disk and root volume group from the network or HP-UX Core media. Once the recovery system has been booted, you can:

- Put a known, good kernel in place.
- Fix the LIF volume on the disk.
- Copy essential files and commands into place.

Expert recovery does not require that you do any preparation before you use it. The media used is supplied by HP; it is not customized to your site. In addition to the media, you can also boot from your Ignite-UX server. However, this also means that any customization you have are not reflected in the files you recover via expert recovery. Depending on the failure cause, expert recovery gives you enough capabilities to get your system back up again. At that point, you need to use your normal restore tool to recover your system to the state it was in before the problem occurred. Expert recovery is not useful to recover from hardware failures.

System Recovery Tools

Comparing Features

The `make_net_recovery` and `make_tape_recovery` tools share many features in common with few differences that exist, mainly due to the different media that are used and ways of handling them. Both system recovery tools share the same basic archive creation options, data structures, archive file content, and installation dialogues.

To determine which system recovery tool is best suited for your needs, consider the following:

Use `make_tape_recovery` if:

- Managing a single or limited number of systems locally
- Cloning a “like system”
- Systems are not networked
- Suitable tape drive exists.

- Tape media is needed for an off-site recovery system

Use `make_net_recovery` if :

- Managing central, networked systems
- Cloning a “like system”
- Avoid tape issues (media cost and handling, multi-tape archives, etc.)
- Suitable disk space for archive storage
- Performing unattended backups without tape handling

The following table summarizes and compares some of the features of the `make_tape_recovery` and `make_net_recovery` tools:

Table 11-1 Comparing System Recovery Tool Features

Feature	`make_tape_recovery`	`make_net_recovery`
Minimum hardware configuration	Stand-alone system; local tape drive	Two networked systems; sufficient disk space to hold archive
Archive Creation Interface	Client command line; Server GUI; Client TUI	Server GUI; Client command line; Client TUI
Archive	Self contained image; written to the client’s tape drive	Requires an Ignite-UX server to install; written to NFS mounted file system

Archive Contents

The `make_net_recovery` and `make_tape_recovery` commands allow you to view and control archive contents:

- The list of essential files to be included in the archive is available as a simple text file: `/opt/ignite/recovery/mnr_essentials`. This file allows you to see what files and directories are included by default in the archive.
- 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/LLA/recovery/archive_content` or

using command line options. You are not restricted to one or two volume groups. You can create a complete multi-volume group file system archive if you want.

- You can use the user interface 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 archive 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 "Archive Configuration Policies" on page 175.

`make_tape_recovery` creates a bootable tape that can be used to restore a system via the system's tape drive. `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 the previous good archives before creating new ones, you need to remove the old tapes and insert different tapes on each system.
- If an archive 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 need to check the log files on every system.
- Tape drives are more error-prone than a local network.

Dependency on Ignite-UX Server for Recovery

make_tape_recovery

The tape created by `make_tape_recovery` is completely self-contained and does not require an Ignite-UX server. The `make_tape_recovery` archive contains a specially prepared LIF volume. The config file in the LIF volume is the configuration file for the archive. The `INDEX` file in the LIF volume specifies the recovery configuration as the default for the system. The `INSTALLFS` in the LIF volume contains additional configuration information so no user interaction will take place. Additional files needed for booting and installing are copied from

/opt/ignite/boot and /opt/ignite/data to the LIF volume, so that everything the system needs to recover is there. You could use your make_tape_recovery tape even if you removed your Ignite-UX server.

make_net_recovery The archives created by make_net_recovery are designed to work with an Ignite-UX server; you could not remove your server and still use your recovery archive.

Creating a Bootable Recovery Tape

IMPORTANT `copyutil` is a diagnostic tool for HP-UX 10.*x* or later, and should not be used for system recovery. Instead, use one of the tools described in this chapter.

Ignite-UX's `make_tape_recovery` command can create a system recovery tape. This tape can be used to boot and recover a system which has become unbootable due to corruption of the root disk or volume group. A system can be booted and installed from the tape without user intervention for configuration, customization, software selection, hostname, or IP address.

NOTE A bootable recovery tape can also be created from the Ignite-UX server. However, the client must have a local tape drive.

The `make_tape_recovery` tool creates a system recovery archive and stores the archive on a local tape. `make_tape_recovery` is capable of creating system recovery tapes for tape devices, with the ability to span multiple tapes. The archive created by `make_tape_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 archive can be installed via tape to restore the system.

The contents of the system recovery archive will always include all files and directories which are considered essential to bringing up a functional system. This **essential** list is pre-defined by `make_tape_recovery` and is located in the following file: `/opt/ignite/recovery/mnr_essentials`. By running `make_tape_recovery` in interactive mode, the directories and files which make up the essential list can be displayed. In addition to the essential list, data can be included in the archive on a disk/volume group, file, or directory basis. Non- essential files and directories can also be excluded.

NOTE It is preferable to use the Ignite-UX GUI menu command on the Ignite-UX server when running an interactive `make_tape_recovery` session. Running it from Ignite-UX causes any additional server configuration of NFS mounts to be performed. It also provides a better progress report and an easier to use interface.

Logging

On a server, progress and errors are logged to:

`/var/opt/ignite/clients/<LLA>/recovery/<datetime>/recovery.log`

On a local system, progress and errors are logged to:

`/var/opt/ignite/recovery/<datetime>/recovery.log`

Task: Recover a minimal OS

Preform these operations as root.

To create a minimal operating system recovery tape at `/dev/rmt/0mn`, containing only the OS elements required to boot the system, perform the following steps:

Step 1. Load a writable tape in the default tape drive for your system.

Step 2. Enter: **`make_tape_recovery`**

A tape will be created without further interaction.

System recovery from this tape would involve booting from the tape to recover the minimum Core OS. Then you would follow up with data recovery of all user files newer than those restored from the recovery tape.

Task: Create a system recovery archive of entire root disk volume

To create a system recovery tape at the default device `/dev/rmt/0m`, and that includes the entire root disk in the archive, perform the following steps:

Step 1. Load a writable tape in the default tape device for your system.

Step 2. Enter the appropriate command:

`make_tape_recovery -x inc_entire = vg00`

A tape will be created without further interaction.

Task: Create a system recovery archive tape with the -A option

To create a system, you can create a system recovery tape as follows:

Step 1. Load a writable tape in the default tape device for your system.

Step 2. Enter: `make_tape_recovery -A`

A tape will be created without further interaction. You can boot this tape on your new system.

Task: Install a system recovery from an archive tape

To install a system recovery from an archive tape:

Step 1. Mount the system recovery tape on the tape drive.

Step 2. Boot the system.

Step 3. Interrupt the boot sequence to redirect it to the tape drive by pressing **Esc**.

Step 4. Cancel the non-interactive installation by pressing any key when given the opportunity.

Step 5. Allow the install process to complete.

For more information on creating recovery tapes, see the *make_tape_recovery* (1M) manpage.

Duplicating make_tape_recovery Tapes

A tape created with `make_tape_recovery` contains two tape "files": a 2KB LIF file and a 10 KB tar archive. If you have two tape drives on a system, you can easily duplicate the tapes using two `dd` commands with a no-rewind-on-close device file for the first command. For example:

```
dd if=/dev/rmt/0mn of=/dev/rmt/1mn bs=2k
dd if=/dev/rmt/0m of=/dev/rmt/1m bs=10k
```

If you only have one tape drive, and have enough disk space to hold the contents of both tape files, use something like this:

```
dd if=/dev/rmt/0mn of=/var/tmp/f1 bs=2k
dd if=/dev/rmt/0m of=/var/tmp/f2 bs=10k
```

(Insert blank tape now)

```
dd if=/var/tmp/f1 of=/dev/rmt/0mn bs=2k
dd if=/var/tmp/f2 of=/dev/rmt/0m bs=10k
```

Also see the *copy_boot_tape* (1M) manpage.

Creating a Recovery Archive via the Network and Tape

Ignite-UX A.2.0, B.2.0 and later versions allow you to create recovery archives via the network onto the Ignite-UX server system, or any other specified system. You can either use the Ignite-UX `/opt/ignite/bin/ignite` or run `/opt/ignite/bin/make_net_recovery` on a client system. Use Ignite-UX to recover specified systems on the net. Systems can be recovered across subnets from a boot tape using `make_boot_tape`, local boot server or the `bootsys` tool from an Ignite-UX server.

The `make_net_recovery` tool creates a system recovery archive and stores the archive on the network. The archive 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 archive can be installed via Ignite-UX to restore the system.

The contents of the system recovery archive will always include all files and directories which are considered essential to bringing up a functional system. This essential list is pre-defined by `make_net_recovery`. By running `make_net_recovery` in interactive mode, the directories and files which make up the essential list can be displayed. In addition to the essential list, data can be included in the archive on a disk/volume group, file, or directory basis. Non- essential files and directories can also be excluded.

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 files which describe the client configuration and location of the recovery archive. The second mount point is made to the *`archive_server:archive_dir`* (see the `-a` option) and is used to store the recovery archive of the client system. After successful or unsuccessful completion of the system recovery archive, the NFS mount points are un-mounted.

The NFS mount for the archive directory may be exported on a per-client bases. A separate archive directory is used for each client. This allows you to NFS export each directory only to the individual client owning the archive, which provides security.

If the client system does not have the most recent versions of Ignite-UX tools, the Ignite-UX GUI uses `swinstall` to install the "recovery package" which includes all necessary files to perform the recovery.

Create a Recovery Archive from the Ignite-UX Server

To create a system recovery archive from an Ignite-UX server:

Step 1. On your host system, allow the Ignite-UX server access to your to display by adding the Ignite-UX server hostname to your xhost list:

```
xhost + Ignite-UX_server_hostname
```

Step 2. Set the DISPLAY variable to your local host system, if necessary. For example:

```
export DISPLAY=Elvis:0
```

Step 3. On the Ignite-UX server, as root run:

```
/opt/ignite/bin/ignite
```

Step 4. Select: **Actions** –> **Add New Client for Recovery** You will need to respond to a dialogue to identify the system.

Step 5. Click the client icon when its appears on the Ignite-UX screen.

Step 6. Select: **Actions** –> **Create Network Recovery Archive**. You may be prompted for the root password for the targetsystem.

The network recovery tools needed on the client will automatically be installed.

After some information screens, you will see an Include/Exclude selection screen. To view the essential files, click the **Show** button. Essential files cannot be excluded, but you can customize the archive by specifying additional volumes, directories, or files. In case an item is duplicated as both Include and Exclude, the Exclude category dominates.

Archive Configuration Policies

When specifying archive content for both `make_net_recovery` and `make_tape_recovery`, either via the Ignite-UX GUI or the command line, the following rules apply:

- No essential file or directory can be excluded.
- Files and directories inside an included directory will be included recursively.
- If a symbolic link to a file or directory is included, only the link will be included in the archive, not the actual file or directory, unless it, too, is included. A warning will be given when the item itself is a symbolic link.
- If a directory is included which 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 will not be included, by default. This policy can be waived by specifying the option `inc_cross` (include directory and cross-mount points), in the selection interface or command line.
- In case of conflicting entries in the selections, Exclusions take precedence over Inclusions.

IMPORTANT If there are mount points below `/etc`, `make_net|tape_recovery` will not restore these files until the recovery generates errors. The recovery does not fail, but the mount points under `/etc` are missing.

More Examples

The follow examples apply for system recovery using either `make_net_recovery` or `make_tape_recovery`:

Create recovery from the client

This command creates a system recovery archive 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/0xLLA/recovery/*:

```
make_net_recovery -s myserver
```

Create recovery from the client that includes volume group files

To create a system recovery archive from the client that includes files from all file systems in the `vg00` volume group, enter:

```
make_net_recovery -s myserver -x inc_entire=vg00
```

Create recovery archive file to replace mnr_essentials

To create a system recovery archive with all the files/directories on the disk(s)/volume group(s) containing the files specified by the default essentials file list `/opt/ignite/recovery/mnr_essentials` or the user defined version of this file, that replaces this file, `/var/opt/ignite/recovery/mnr_essentials`, enter:

```
make_net_recovery -s myserver -A
```

Preview system recovery

To preview the creation of the system recovery archive enter:

```
make_net_recovery -p
```

Large File Support

Specific support for large files is needed if archives greater than 2GB are to be created. This requires ensuring that both the file system and the NFS mount on the archive server will support large files.

The `fsadm` command can be used to determine whether large files are currently supported on a specific file system. The *fsadm* (1M) manpage has an example of how to change the file system to support large files. If you use `fsadm` to convert a file system, re-run **`exportfs -a`**, if it is already exported, in order for clients to be affected by the change.

To support NFS mount and network data transfer of large files, you will need to have NFS PV3 installed on both the client and server. If the client or server is running HP-UX 10.20, the Networking ACE patch

(containing the NFS PV3 software) should be installed and updated with a patch cited in the Ignite-UX Release Notes. HP-UX 11.0 and later versions come with PV3 by default.

The Ignite-UX Release Notes (`/opt/ignite/share/doc/release_note`) identifies which patches are required for NFS support of archives greater than 2GB for HP-UX 10.20, 11.0 and later.

Recovering via the Network for PA Clients

To recover a failed disk or volume group using the system recovery archive:

Step 1. Boot the failed system using one of these ways (see "Booting Client Systems from the Network" on page 67):

- Using Ignite-UX after booting with: **`boot lan install`**
- Booting from an Ignite-UX server, using `bootsys` if the client OS is running.
- Booting the failed client locally by using a boot tape previously created with `make_boot_tape`.

Step 2. Do not interact with ISL.

Step 3. From the main menu, select **Install HP-UX**
At the client:

1. Respond to Network configuration dialogue screen.
2. Respond to the UI display options (run at Server or at Client).
3. If working from the Ignite-UX server, select the client icon for the system to be recovered.

Step 4. Select **Install/New Install**

Step 5. Select the recovery configuration to use.

Recovering via the Network for IA Clients

To recover a failed disk or volume group using the system recovery archive:

Step 1. From the EFI Boot Manager menu, you will see a prompt to select a boot option. Select **`Boot option maintenance menu`**.

```
EFI Boot Manager ver 1.10 [14.54]  Firmware ver 0.0 [4209]

Please select a boot option

    EFI Shell [Built-in]
    Boot option maintenance menu
    Security/Password Menu (*** Prototype ***)

Use up and down-arrows to change option(s).
Use Enter to select an option
```

Step 2. The Main Menu appears and prompts you to choose an operation. Select **`Add a Boot Option`**.

```
EFI Boot Maintenance Manager ver 1.10 [14.54]

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
```

Step 3. The following menu displays. Select an appropriate network card for network boot. For example, look for entries with a MAC followed by the Mac/*LLA* address of the LAN card.

```
EFI Boot Maintenance Manager ver 1.10 [14.54]

Add a Boot Option.  Select a Volume

    Removable Media Boot
[Acpi(HWP0002,0)/Pci(2|0)/Ata(Primary,Maste
    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
```

Step 4. Enter an appropriate boot option name at the message prompt. For this example, new boot options are named LAN1 and LAN2.

Step 5. Exit to the main menu. The new boot option will now appear in the EFI Boot Manager main menu.

```
EFI Boot Manager ver 1.10 [14.54]  Firmware ver 0.0 [4209]

Please select a boot option

    SCSI2-HPUX
    EFI Shell [Built-in]
    LAN2
    LAN1
    Boot option maintenance menu
    Security/Password Menu (*** Prototype ***)

Use up and down-arrows to change option(s).
Use Enter to select an option
```

Step 6. Select the new boot option you created. The following is an example of a successful boot using the new boot option.

```
Loading.: LAN1
Running LoadFile()

CLIENT IP: 15.1.52.128  MASK: 255.255.248.  DHCP IP: 15.1.53.37
GATEWAY IP: 15.1.48.1
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...
```

Step 7. From the main menu, select **Install HP-UX**
At the client:

1. Respond to Network configuration dialogue screen.
2. Respond to the UI display options (run at Server or at Client).
3. If working from the Ignite-UX server, select the client icon for the system to be recovered.

Step 8. Select **Install/New Install**

Step 9. Select the recovery configuration to use.

Create a Bootable Archive Tape via the Network

This section explains how to create a self-contained recovery tape for a recovery configuration already stored on an Ignite-UX server via network system recovery. See the *make_net_recovery* (4) manpage for more details. It is important that the archive fit onto a single tape.

These instructions assume that:

- The hostname of the machine the archive was created for is `sys1`.
- The archive was created at “2001-03-12,09:00”.
- The archive will fit onto a single tape.

Step 1. Build the LIF file. The LIF file will contain the Ignite-UX tools and environment, the config files produced for the recovery archive, and the scripts used during recovery.

1. Use `make_medialif` to build the LIF file:

   ```
   cd
   /var/opt/ignite/clients/sys1/recovery/2001-03-12,09:00

   make_medialif -f system_cfg -fcontrol_cfg -f archive_cfg\
   -C “2001-03-12,09:00 sys1 recovery image” \
   -l /var/tmp/my_lif -a -r os_rev
   ```

2. Modify the LIF file for use on the tape:

   ```
   instl_adm -d -F /var/tmp/my_lif > /var/tmp/cfg
   ```

3. Add these lines to the end of the `/var/tmp/cfg` file:

   ```
   control_from_server=FALSE

   run_ui=TRUE
   ```

   OR, if you just want the recovery to proceed without any interaction, make:

   ```
   run_ui = FALSE
   ```

   and specify to allow warnings as shown here:

   ```
   env_vars += “INST_ALLOW_WARNINGS=10”
   ```

4. Enter:

   ```
   instl_adm -F /var/tmp/my_lif -f /var/tmp/cfg
   ```

Step 2. Create the Tape. To write the LIF and archive to a tape:

1. Determine which tape device file you can use to write the tape. The device file must match the tape drive type you will use when actually recovering the system. Also, the tape device file must be the no-rewind type. For the rest of this example, assume that `/dev/rmt/c1t0d0DDS1n` is a no-rewind DDS-1/no compression device file.

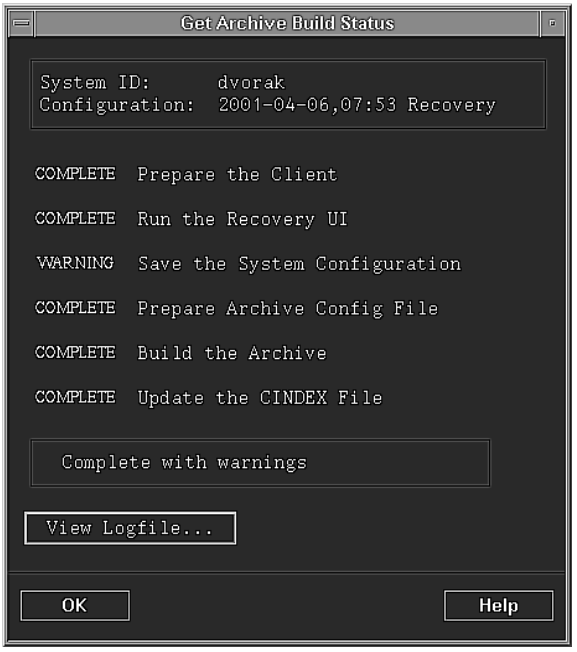
2. Rewind the tape, write out the LIF and archive, and rewind again:

```
mt -t /dev/rmt/c1t0d0DDS1n rew

dd if=/var/tmp/my_lif of=/dev/rmt/c1t0d0DDS1n obs=2k

dd if=/var/opt/ignite/recovery/archives/sys1/2001-03-12,
09:00 \ of=/dev/rmt/c1t0d0DDS1n obs=10k

mt -t /dev/rmt/c1t0d0DDS1n rew
```

In this example, the archive is retrieved from the standard location on the Ignite-UX server for this host. If you have chosen to put the archive elsewhere, refer to that location instead.

Archive Creation Steps

Ignite shows the steps as an archive is being created with `make_net_recovery`:



Step 1. **Prepare the client.** `make_net_recovery's` primary tasks are to check that the recovery tools installed on the client are compatible with the versions on the Ignite-UX server, and to create the necessary directories and files. If the client was not previously installed using the Ignite-UX server, `make_net_recovery` creates a new directory for the client in `/var/opt/ignite/clients` on the server.

`make_net_recovery` also generates a timestamp for naming the archive, the configuration, and the configuration directory. The directory containing the configuration files for the archive will be something like:

`/var/opt/ignite/clients/`*`LLA`*`/recovery/2001-03-09,00:27`

The corresponding archive will be:

```
/var/opt/ignite/recovery/archives/client/2001-03-09,00:27
```

The timestamp is important for coordinating configuration files and archives, and for ongoing archive management.

Here's an overview of files located in the client's directory for network recovery (for a local tape the path would be `/var/opt/ignite/recovery`):

```
/var/opt/ignite/clients/<LLA>
        install.log
        CINDEX
        Client_status
        recovery/
                archive_content
                defaults
                latest
                2001-02-09,00:27/
                        archive_content
                        system_cfg
                        archive_cfg
                        config_cfg
                        recovery.log
                        flist
                        manifest
                2001-03-09,00:27/
                        archive_content
                        system_cfg
                        archive_cfg
                        config_cfg
                        recovery.log
                        flist
                        manifest
```

Step 2. Run the recovery Interface. If `-i` is specified on the command line, the Recovery user interface is run next. The interface enables users to set or change the following default values for the archive:

- Long description of the archive. This description may be used to add identifying information that can help to distinguish archives when the timestamp is not sufficient. This information is shown by clicking **Description** on the **Basic** tab during installation configuration.

- Maximum number of archives to keep. When the number of archives in the destination directory reaches this maximum, `make_net_recovery` removes the oldest archive. It uses the timestamp in the name of the archive to determine which one to remove.
- Destination host for the archive.
- Destination directory for the archive.

The user interface also gives users the opportunity to review and edit the `archive_content` file as mentioned above. When the user exits the recovery user interface, the default values that the user entered are written to:

`/var/opt/ignite/clients/`*`LLA`*`/recovery/defaults`

The archive content is written to:

`/var/opt/ignite/clients/`*`LLA`*`/recovery/archive_content`

Step 3. Save the system configuration. For all volume groups, even ones that are not included in the archive, `make_net_recovery` now backs up volume group configuration information and stores in the `system_cfg` file. It also obtains map files for volume groups that are not part of the archive using `vgexport`. The volume group configuration files and the map files generated at this stage are stored in `/etc/lvmconf`. This directory is included by the list of essential files, so the lvm files are included in the archive.

After the volume group information is saved, `make_net_recovery` creates the `control_cfg` file. This file includes the `post_config_cmds` to import all volume groups that were not included in the archive 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.

Step 4. Prepare the config file. Once the archive is created, `make_net_recovery` calls `make_arch_config` to create the `archive_cfg` file to reference it. `make_arch_config` uses `archive_impact` to calculate the file system impacts for the archive, and includes these in the `sw_sel` stanza it writes.

Step 5. Build the archive. make_net_recovery calls make_sys_image to create the archive. make_sys_image passes a pre-built 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 pax.

Step 6. Update the CINDEX file. make_net_recovery uses manage_index to update the /var/opt/ignite/clients/*LLA*/CINDEX file for the client. This file contains a list of all the recovery cfgs available for the client. An entry for the most recently created archive looks something like this:

```
cfg "1999-03-10,00:27 Recovery" {
   description "This cfg is a pure mnr_essentials recovery
   archive."
   "recovery/1999-03-10,00:27/system_cfg"
   "recovery/1999-03-10,00:27/control_cfg"
   "recovery/1999-03-10,00:27/archive_cfg"
}=TRUE
```

Verifying Archive Contents

What files will be archived?

To list the files and directories that will be included in a make_net_recovery archive, enter:

/opt/ignite/lbin/list_expander -l -f *input_file*

You can examine the list of files that will be re-created during an installation from a make_net_recovery configuration, by viewing the /var/opt/ignite/clients/cca/recovery/fhist file. For example:

pg /var/opt/ignite/clients/cca/recovery/fhist

If make_tape_recovery was used, the list of recovery files resides in /var/opt/ignite/recovery/latest/fhist and can be viewed in the same manner as the above example.

What disks or volume groups will be recovered?

To list disks or volume groups that will be re-created during an installation from a make_net_recovery configuration, enter this from the client:

/opt/ignite/lbin/list_expander -d -f *input_file*

where: *input_file* is a file specifying what is to be archived. See the *make_net_recovery* (4) manpage for details on the format of the *input_file*. make_net_recovery can take input from an input file, no input, or input from the command line with the -x option. list_expander can take input from an input file, or no input, but does not have an x option like make_net_recovery does, so to see the result of using x options, put them in a file and pass list_expander the file name.

If you used the Ignite-UX to specify what is to be included in the archive, then the input file can be found on the server in:

/var/opt/ignite/clients/*client*/recovery/archive_content

You can copy this file from the server to the client, then run list_expander against that file itself.

Omitting -f *input_file* causes list_expander to use only the essential files as input. This will show what disks or volume groups will get re-created for the minimal archive. Here's an example 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 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. The In? column shows, for each disk or volume group, if it will be:

2 = included in full (INC_ENTIRE dsk/vg specified),

1 = included in part (some files included, some not), or

0 = not included at all (no files from this dsk/vg are included).

0 means the disk or volume group will *not* be touched. 1 or 2 means that the disk or volume group *will* be re-created, and files from the archive will be restored during a recovery operation.

Verifying Archive Results

During a system recovery, Ignite-UX strives to restore the system back to the way it was. However, Ignite-UX is a general-purpose installation tool, and can modify many system configuration files.

When you run `make_net_recovery`, a lot of system configuration information is gathered and saved in config files that are used later when the system is recovered. During the system recovery the user is allowed to make changes to this information, in which case Ignite-UX will make the appropriate changes to the system configuration. If a user does not make any changes, then it simply re-applies the same information and you should see no change to the system in the end.

Most of the system configuration files that Ignite-UX will modify are listed in the script: `/opt/ignite/data/scripts/os_arch_post_l`. The `os_arch_post_l` script checks for the system recovery case by checking the `$RECOVERY_MODE` variable. When this variable is TRUE, the os_arch_post_l script causes some configuration files to be protected from modification by using the "save_file" function. os_arch_post_l uses the "merge_file" function on files that Ignite-UX knows how to intelligently merge information into.

The files operated on by "merge_file", as well as those that have a commented out "save_file" line are those that 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_l file on your system to see which files are saved as-is and which are merged with information from the Ignite-UX config files.

Retaining “Known-good” Archives

You may want to prevent known-good archives from being deleted from your system. `make_net_recovery` provides the `-n` option to specify the number of archives to save. To preserve disk space, the oldest archive(s) are removed as new archives are created. The number of archives that get removed is based on the number of archives specified to be saved.

One way to ensure that known-good archives are saved would be to specify the number of archives to save to be greater then the maximum number of archives you plan to store on the system at any given time. This would cost disk space.

An alternative and better approach to saving known-good archives is to rename the archive and edit the configuration file to include the new archive name. Follow these steps:

Step 1. Login to the system where the archive is to be stored (this could be different than the Ignite-UX server).

Step 2. Rename the archive. (The path to the archive may be different than the example below). The name of the archive to save can be anything unique, but it should be outside the naming convention: *`yyyy-mm-dd,hr:min`*

```
cd /var/opt/ignite/recovery/archives/system_name
mv old_archive_name saved_archive_name
```

For example:

```
mv 2001-05-11,15:14 Recovery_Archive.0511.save
```

Step 3. If the archive server is different from the Ignite-UX server, login to the Ignite-UX server system.

Step 4. Edit this file to reference new archive name:

```
/var/opt/ignite/clients/client/recovery/  \
old_archive_name/archive_cfg
```

Change the `archive_path` variable inside the `(source_type == "NET")` conditional to the name of the saved archive. For example:

```
(source_type == "NET") {
  archive_path = "Recovery_Archive.0511.save"
}else {
  archive_path = "1"
}
```

Step 5. Optionally, edit the `cfg` tag entry in the file:

/var/opt/ignite/clients/`client`*/CINDEX*

so that configuration will be unique and descriptive when it is viewed via the Ignite-UX screen. For example:

Change from:

```
cfg "2001-05-13,06:51 Recovery Archive" {
description "Weekly System Recovery Archive"
...

}
```

To:

```
cfg "Saved Recovery Archive" {
description "Weekly System Recovery Archive"
...

}
```

Making config File Additions

To make configuration file additions to all recovery configurations for a given client, create a new Ignite-UX configuration file called:

`/var/opt/ignite/clients/0x{LLA}/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 into your recovery configuration for this client each time you run `make_net_recovery`. (`make_net_recovery` is run for you when you use Ignite-UX for network recovery).

If you already have recovery configurations for this client and would like them to include the `config.local` file, edit the `/var/opt/ignite/clients/0x`*`LLA`*`/CINDEX` file to include a reference to "recovery/config.local" in all of the configuration clauses.

Selecting File Systems During Recovery

It is possible to change the way your disks are configured when you recover from an image saved by `make_net_recovery`. If you want to use a standard HP filesystem layout, you can specify the disk configuration using Ignite-UX:

Install Client –> **New Install** –> **File System**

If you do not want to use a standard HP filesystem layout, you can modify the `/var/opt/ignite/clients/0x`*`LLA`*`/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 that you intend to recover from. If you want to add HP's standard 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.10.20/config`

would be added for a client with the HP-UX 10.20 operating system. This new configuration file entry should be the first entry in the clause you are modifying.

When you bring up the user interface during recovery, select the File System type you wish to use on the **Basic** tab.

Tape Recovery with make_net_recovery

There are two ways you can recover from a tape with `make_net_recovery`. The method you choose depends on your needs.

Use make_medialif This method is useful when you want to create a totally self-contained recovery tape. The tape will be bootable and will contain everything needed to recover your system, including the archive of your system. During recovery, no access to an Ignite-UX server is needed. Using `make_medialif` is described beginning on page 180 and also on the Ignite-UX server in the file: `/opt/ignite/share/doc/makenetrec.txt`

Use make_boot_tape This method is useful when you do not have the ability to boot the target machine via the network, but are still able to access the Ignite-UX server via the network for your archive and configuration data. This could happen if your machine does not support network boot or if the target machine is not on the same subnet as the Ignite-UX server. In these cases, use `make_boot_tape` to create a bootable tape with just enough information to boot and connect with the Ignite-UX server. The configuration files and archive are then retrieved from the Ignite-UX server. See the *make_boot_tape* (1M) manpage for details.

Tape Recovery with No Tape Boot Support

You can use the Ignite-UX tape recovery tool to archive your system even if there is no tape boot support on the system. This support is provided in the Ignite-UX B.4.1 release and later only.

IMPORTANT Be sure to locate the December 2002 version or later of Ignite-UX (B.4.1 or later) media (CD or DVD) that can be used to boot your system to the interface screens to guide you through tape recovery using a tape drive.

Step 1. Insert the December 2002 version or later of Ignite-UX (B.4.1 or later) media into the appropriate drive, then boot from it.

The following interface screen appears:

```
                    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:
   [ * ]  Guided Installation   (recommended for basic installs)
   [   ]  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  ]
```

Step 2. Click the **OK** button to continue and you are advanced to the next screen:

```
                        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 ]
```

Step 3. Select the **Boot from CD/DVD, Recover from Tape** and click **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
    to enable, then press <Return/Enter>.

       HW Path              Device File
Description

------------------------------------------------------------

     [ 0/18/1/0/0.0.3.0    /dev/rmt/c0t0d0        HP C5683A ]
     [ 0/18/1/0/0.1.0.0    /dev/rmt/c1t1d1        HP A5580A ]
```

Step 4. Select the tape drive that contains the archive tape then press **Enter** to start the installation of the recovery tape archive from the chosen tape drive.

Notes on Cloning Systems

Ignite-UX offers two main options for replicating (cloning) systems. The traditional Ignite-UX method makes use of `make_sys_image` to create an archive of the source system, followed by manually modifying config files to meet your needs. A much simpler (but less flexible approach) uses `make_net_recovery` or `make_tape_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 target systems. This means that the version of HP-UX, patches, drivers, etc., must be sufficient for all systems involved. This often requires loading a superset of software and drivers onto the source system that will be used on all potential targets.

Using make_sys_image

Using the traditional 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:

- Ability to install systems from network, tape, or CD-ROM from either an Ignite-UX server, or local clients.
- 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 target system's hardware.
- Ability to load additional software or patches on top of the system archive from an SD depot. This reduces the need to recreate the archive, and allows you to add support for new hardware that requires new patches, or drivers without making a new archive.

Using make_net_recovery and make_tape_recovery

The `make_tape_recovery` and `make_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 for cloning in various ways:

- You can change hostname/networking information.
- You can make changes to disks and file systems during the recovery.

- 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.

The next section shows how to clone a system using `make_net_recovery`. System cloning using `make_tape_recovery` begins on page 169.

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. Follow these steps:

Step 1. Use `make_net_recovery` or Ignite-UX to create a system recovery archive of the source system.

Step 2. Login to the Ignite-UX server.

Step 3. If the target system to be installed does not currently have a directory in `/var/opt/ignite/clients` but is up and running, then use Ignite-UX to create that directory using **Actions** –> **Add New Client for Recovery**. If the system is not running, you will either need to boot the client from the Ignite-UX server (or from a tape made with `make_boot_tape` in order for this directory to be created.

Step 4. Copy the `CINDEX` and recovery directory from the source client to the target client directory. If the target client has previously used `make_net_recovery` then it will already have a `CINDEX` file. If the `CINDEX` file for the target system exists already, you may want to save a copy, and/or hand edit the file to add the desired entries from the source client. The commands below copy the required files. You may specify `src_client` and `target_client` using either the LAN addresses (such as 0x0060B04AAB30), or by using the client's hostname (which is a symlink to the LAN address):

```
cd /var/opt/ignite/clients/src_client
find CINDEX recovery | cpio -pdvma ../target_client
```

Step 5. Give the *target_client* NFS access to the archive of the source system. To do this, login to the server that holds the archive (normally the Ignite-UX server).

Typically each client has its own directory for storing the archives, and the directory is exported only to the individual client. In this case, you will need to edit the `/etc/exports` file to allow access to both the source and target clients:

1. Enter: **`vi /etc/exports`**
2. Append **`:target-client`** to the end of the source-client's line.
3. Enter: **`exportfs -av`**

Step 6. Boot the target-client from the Ignite-UX server (using any method you wish). Then when you install the system, you can select from the recovery configurations of the source system.

Step 7. Change the system networking parameters for the target system during the installation.

Expert Recovery Using the Core Media

If your system should become so compromised or corrupt that it will not boot at the login prompt, or the system boots, but critical files are corrupted, adversely affecting overall system performance, it may be useful to restore system elements with core recovery media.

Before you attempt to recover an HP-UX system, you should have the following information about your system disk available.

Much of this information, including file system types, can be obtained by accessing your on-line system manifest, either via Ignite-UX, or by reading the hardcopy that came with your system

- Revision of the HP-UX system which you are attempting to recover.

CAUTION

Only attempt to recover HP-UX systems that match the version number of the recovery tools you are using, currently HP-UX 11.0. For example, you can use HP-UX 10.20 Core media to attempt to recover an HP-UX 10.20 file system.

- The hardware path of the root filesystem on the disk (that is, what file system you will be checking/repairing using `fsck`).
- The address of the bootlif path of that disk.
- What the autofile in the bootlif should contain.
- Whether you have an LVM, VXVM, or whole-disk system.

The more you know about the system disk and its partitioning scheme, *before you encounter major damage or corruption*, the easier it will be for you to recover.

The procedures which follow assume that both `fsck` and `mount` can be run successfully on the system disk; otherwise, the following procedures are not applicable.

Automated Recovery Procedures

There are four possible expert recovery situations, each of which has its associated recovery procedure:

- If, after a system problem, you can't get the system to the `ISL>` prompt from the system disk, you will want to rebuild the bootlif on the system disk, and install all critical files required to boot on the root filesystem.
- If you can get the system to the `ISL>` prompt, but cannot boot `vmunix`, the system disk is corrupted; you will want to install *only* the critical files required to boot on the root filesystem.
- If you can't get to the `ISL>` prompt, but you know that the root file system is good, you will want to rebuild the bootlif on the system disk.
- If you believe your kernel is corrupted, you will want to replace only the kernel on the root filesystem.

The following subsections describe these procedures in detail.

Rebuilding the bootlif and Installing Critical Files

Following is an example of the detailed procedure for rebuilding the bootlif of the system disk, and for installing all the critical files necessary to boot from the root filesystem:

Step 1. Have the Core OS CD for the appropriate HP-UX ready.

Step 2. Reset the System Processor Unit (SPU) using the reset button, or keyswitch, as appropriate.

The console displays boot path information. If Autoboot is enabled, the system console eventually displays messages similar to this:

```
Autoboot from primary path enabled
To override, press any key within 10 seconds.
```

Step 3. With older systems, press any key within 10 seconds. The system console displays:

```
Boot from primary boot path (Y or N)?>
```

Enter **n** at the prompt. The next prompt is:

```
Boot from alternate boot path (Y or N)?>
```

Step 4. If the alternate boot path specifies the address of the CD device where the Core CD is mounted, enter: **y**

If the alternate boot path does not specify the address of the CD device where the HP-UX Core media is mounted, enter **n** at the prompt. The next prompt is:

```
Enter boot Path or ?>
```

Step 5. Enter the address of the CD device where the HP-UX Core media is mounted. The next prompt is:

```
Interact with IPL (Y or N)>
```

Step 6. Enter **n** at the prompt.

After several minutes and several screens of status information, the this is displayed:

```
Welcome to the HP-UX installation/recovery process!
Use the <tab> and/or arrow keys to navigate through the
following menus,and use the <return> key to select an item. If
the menu items are not clear, select the "Help" item for more
information.

                [    Install HP-UX       ]
                [  Run a Recovery Shell  ]
                [   Advanced Options     ]
                           [ Help ]
```

Step 7. Select: **Run a Recovery Shell**. The next prompt is:

```
Would you like to start up networking at this time? [n]
```

Step 8. Unless you need networking to ftp to other systems, enter: **n**

```
* Loading in a shell...
* Loading in the recovery system commands...
             HP-UX SYSTEM RECOVERY CORE MEDIA
             WARNING:  YOU ARE SUPERUSER !!
NOTE: Commands residing in the RAM-based file system are unsupport
ed 'mini'commands. These commands are only intended for recovery p
urposes.

Loading commands needed for recovery!

Press <return> to continue.
```

Step 9. Press **Return** or **Enter**. The next prompt is:

```
Loading commands needed for recovery!
```

This menu is displayed:

```
HP-UX CORE MEDIA RECOVERY
   MAIN MENU
 s.  Search for a file
 b.  Reboot
 l.  Load a file
 r.  Recover an unbootable HP-UX system
 x.  Exit to shell

This menu is for listing and loading the tools contained
on the core media.  Once a tool is loaded, it may be run
from the shell. Some tools require other files
to be present in order to successfully execute.
Select one of the above:
```

Step 10. To load a file or files, enter **l** at the prompt.

```
Filesystem kbytes used avail %cap iused ifree iused Mounted on
/                 2011    1459 552  73%   137    343 29%   ?
/duped_root       2011    1418 593  71%    49    431 10%   ?
Enter the filename(s) to load:
```

Step 11. Enter the name(s) of the damaged/corrupted file(s) you wish to load. For example:

```
sh vi date grep
```

The following example lists two files (`ex` and `egrep`) which must be loaded before the files `vi` and `grep` can be loaded. It also lists a file (`date`) which is not in the load list.

```
NOTE :
  Since ./usr/bin/vi is linked to ./usr/bin/ex
  './usr/bin/ex' must precede './usr/bin/vi' in the load list.
The file 'date' is NOT in the LOADCMS archive.
<Press return to continue>
NOTE :
Since ./usr/bin/grep is linked to ./usr/bin/egrep
'./usr/bin/egrep' must precede './usr/bin/grep' in the loadlist.
********  THE REQUESTED FILE(S): ***********
./sbin/sh ./usr/bin/vi ./usr/bin/grep
Is the above load list correct? [n]
```

Step 12. This load list is incorrect, because `./usr/bin/ex` does not precede `./usr/bin/vi` in the list of requested files. So you would enter: **n**

```
Nothing will be loaded!
<Press return to return to Main Menu>
```

Step 13. Press **Enter** and the Main Menu appears. To search for a file you wish to load, select: **s**

```
Either enter the filename(s) to be searched for,
or 'all' for a total listing.
```

Step 14. Enter:

```
vi awk /sbin/sh date
```

You will receive this response:

```
./usr/bin/vi linked to ./usr/bin/ex
./sbin/awk
./usr/bin/awk
./sbin/sh
**** The file  'date' was not found in the LOADCMDS archive. ****
<Press return to continue>
```

Step 15. Press **Enter** to return to the Main Menu.

Step 16. To begin the actual system recovery and invoke the Recovery Menu, select: **r**

```
 HP-UX Recovery MENU
 Select one of the following:
a. Mount the root disk and exit to shell only.
b. Recover the bootlif/os partitions.
c. Replace the kernel on the root file system.
d. Both options b and c

v. Read information about VxVM/LVM recovery

m. Return to ‘HP-UX Recovery Media Main Menu’.
x. Exit to the shell.
```

Step 17. To install both the bootlif and critical files, select: **a**

```
DEVICE FILE VERIFICATION MENU
This menu is used to specify the path of the root file sysem.
When the information is correct, select 'a'.

INFORMATION to verify:
       Device file used for '/'(ROOT) is c1t6d0
       The path to disk is 56/52.6.0
 Select one of the following:
  a.  The above information is correct.
  b.  WRONG!! The device file used for '/'(ROOT) is incorrect.
  m.  Return to the 'HP-UX Recovery MENU.'
  x.  Exit to the shell.
```

Step 18. Assuming the root device file is incorrect, select: **b**

```
Enter the device file associated with the '/'(ROOT)

file system. (example: c1t6d0):
```

On a system with hard-sectored disks, the prompt and response might look like this:

```
Enter the device file associated with the '/'(ROOT) file system
(example: c0t1d0s1lvm ) :  c0t0d0s13
/dev/rdsk/c0t0d0s13 not a special file
<Press return to continue>
Enter the address associated with the '/'(ROOT) file system
(example: 4.0.1) :  4.0.0
NOTE: if your '/'(ROOT) is not part of a sectioned disk layout
    enter a 'W' for whole disk layout
       or
    enter a 'l' for an LVM disk layout
  instead of a section number.
Enter the section associated with the '/'(ROOT) file system
(example: 13 ):  13
 making rdsk/c0t0d0s13 c 214 0x00000d
 making dsk/c0t0d0s13 b 26 0x00000d
```

Step 19. If you entered **c1t1d0** as the root device filename, you would see:

```
DEVICE FILE VERIFICATION MENU
This menu is used to specify the path of the root file
system When the information is correct, select 'a'.
INFORMATION to verify:
       Device file used for '/'(ROOT) is c1t1d0
       The path to disk is 56/52.1.0
 Select one of the following:
    a.  The above information is correct.
    b.  WRONG!! The device file used for '/'(ROOT) is incorrect.
    m.  Return to the 'HP-UX Recovery MENU.'
    x.  Exit to the shell.
```

Step 20. Since c1t1d0 is the correct root device filename, select: **a**

```
BOOTLIF PATH VERIFICATION MENU
This menu must be used to determine the path to the bootlif(ISL, H
PUX and the AUTO file).
When the information is correct, select 'a'.
INFORMATION to verify:
      Path to the bootlif is 56/52.1.0
Select one of the following:
      a.  The above information is correct.
      b.  WRONG!! The path to bootlif is incorrect.
      m.  Return to the 'HP-UX Recovery MENU.'
      x.  Exit to the shell.
     Selection:
```

Step 21. Assuming that the bootlif path is correct, enter: **a**

```
FILE SYSTEM CHECK MENU

The file system check'/sbin/fs/hfs/fsck -y /dev/rdsk/c1t10'
will now be run.
Select one of the following:
      a.  Run fsck -y .
      b.  Prompt for the fsck run string on c1t1d0.
      m.  Return to the 'HP-UX Recovery MENU.'
     Selection:
```

Step 22. Select **a** to run **`fsck -y`** to check your file system for corruption.

```
** /dev/rdsk/c1t1d0
** Last Mounted on /ROOT
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
6256 files, 0 icont, 149423 used,1563824 free(928 frags,195362 blo
cks)
Mounting c1t1d0 to the HP-UX Recovery Media /ROOT directory...
<Press return to continue>
```

Step 23. Assuming your file system is not corrupted, and you wish to continue with the system recovery, press **Return** to mount your root file system under the / directory. You'll see messages like this:

```
   ***** Downloading files to the target disk *****
x ./sbin/lvchange, 528384 bytes, 1032 tape blocks
./sbin/lvcreate linked to ./sbin/lvchange
./sbin/lvdisplay linked to ./sbin/lvchange


Filesystem kbytes used avail %cap iused ifree iused Mounted on
/ROOT    1713247  149426 1392496  10%  6261 275339  2%   ?

Should the existing kernel be
'left', 'overwritten', or 'moved'?[overwritten]
```

Step 24. To overwrite the existing kernel with your new file system, enter **`overwritten`** or **`over`** at the prompt.

```
 downloading INSTALL to /stand/vmunix
    **** Creating device files on the target disk ****
    ******* Renaming the following files: *******
     '/.profile' has been renamed '/.profileBK'
      *********** Installing bootlif  ***********
mkboot -b /dev/rmt/1m  -i ISL -i HPUX /dev/rdsk/c1t1d0
mkboot -a  hpux (56/52.1.0;0)/stand/vmunix /dev/rdsk/c1t1d0
```

Step 25. Complete the recovery process by selecting: **a**

```
NOTE:    System rebooting ...
-
PDC - Processor Dependent Code - Version   1.3
(c) Copyright 1990-1993, Hewlett-Packard Company, All rights
reserved.
   -
   16 MB of memory configured and tested.
   Primary boot path:    56/52.5   (dec)
   Alternate boot path:  56/52.3   (dec)
Manufacturing permissions ON
-    Main Menu
Command                       Description
        -                       -
BOot [PRI|ALT| &<path>]      Boot from specified path
PAth [PRI|ALT|][ &<path>]    Display or modify a path
SEArch [DIsplay|IPL][&<path>] Search for boot devices
COnfiguration menu     Displays or sets boot values
INformation menu       Displays hardware information
SErvice menu          Displays service commands
MFG menu              Displays manufacturing commands

DIsplay                Redisplay the current menu
HElp [&<menu>|&<command>] Display help for menu or command
RESET                  Restart the system
-
Main Menu: Enter command or menu item.
```

Step 26. Enter **bo pri** at the prompt to boot from the primary boot path. The next prompt is:

```
Interact with IPL (Y or N)?>
```

Step 27. Enter **n** for unattended boot. Several screens of status information are displayed, followed by this warning:

```
THIS SYSTEM HAS BEEN BOOTED USING A TEMPORARY KERNEL!
DO NOT ATTEMPT TO INVOKE MULTI-USER RUN-LEVEL USING THIS KERNEL!
Type the following command from the shell prompt for more
information about completing the recovery process:

cat /RECOVERY.DOC
```

Step 28. To obtain more information on the recovery process, enter:

```
cat /RECOVERY.DOC
```

```
1) Restore valid copies of the following files (either from backup
 or
   from the filename.BK files created during the recovery process)
.
   /etc/fstab,       /etc/inittab,  /stand/ioconfig,
   /etc/ioconfig,     /etc/passwd,  /sbin/pre_init_rc,
   /.profile,   and /etc/profile
  NOTE:  The backup archive may be extracted using '/sbin/frecover
' or'/sbin/pax' (for backups made with 'tar' or 'cpio').
If using '/sbin/pax', linking it to 'tar' or 'cpio' will
force'pax' to emulate the respective command line interface.
2) Replace /stand/vmunix from backup, since the present kernelis p
robably missing desired drivers.
3) If you have an lvm root, refer to the /LVM.RECOVER text file.
```

Step 29. If you have an LVM system, and want more information on recovery procedures, enter:

```
cat /LVM.RECOVER
```

Follow the displayed instructions shown below.

NOTE If a card has been added to, or removed from, your system since the original installation was completed, there is a chance that the device file for the root disk has changed. Consequently, before you run the LVM script *./lvmrec.scrpt* (Step 2 in the displayed instructions below), you should first recover */stand/ioconfig* from backup, and reboot.

```
INSTRUCTIONS to complete your LVM recovery:
The system must now be up now in "maintenance mode".
NOTE: In order for the following steps to lead to a successful
      lvm recovery the LVM label information must be valid.
      If the bootlif was updated from the RAM-based recovery syste
m,
      then "mkboot -l" has already been run to repair this label.
step 1. If the autofile was altered to force the system to boot in
 maintenance mode, use "mkboot -a" to remove the "-lm" option.
 Example:
    to change  "hpux -lm (52.6.0;0)/stand/vmunix"
    to   "hpux   (52.6.0;0)/stand/vmunix"
```

```
    use
   mkboot -a  "hpux (52.6.0;0)/stand/vmunix"  /dev/rdsk/<device fi
le>
```

(Use `lssf /dev/rdsk/*` to match the device file with the boot address.)

```
step 2. Run '/lvmrec.scrpt' to repair the following LVM
 configuration information:
        a. LVM records (lvmrec)
        b. BDRA (Boot Data Reserve Area)
        c. LABEL information
 Requirement:  The following files must reside on disk before
        the script can complete:
        a. /etc/lvmtab
        b. /etc/fstab
        c. /etc/lvmconf/<rootvg>.conf
        d. all device files specified in /etc/fstab
 To run '/lvmrec.scrpt' provide the device filename used to
 access the bootlif as an argument to the script.
 Example:
  /lvmrec.scrpt c0t6d0
  In this example 'c0t6d0' is the device file used to
  access the bootlif.
step 3. Once '/lvmrec.scrpt' completes, issue the command "reboot"
 and bringthe system fully up.
 The recovery of the root LVM is complete.  If the'/lvmrec.scrpt'
issued the following warning:
  "************ I M P O R T A N T *****************         "Root
 logical volume has been repaired, but......."
  "you need to reboot the system and repair the Swap"
  "logical volume using the following LVM command:  "
  "   lvlnboot -A n -s /dev/<root lv>/<swap lvol>   "
  "because Recovery has no way to find out what is  "
  "the Swap logical volume information at this point"
  "************************************************"

The Swap and Dump logical volumes will need to be re-configured
The BDRA contains the "root", "swap" and "dump" logical volumeinfo
rmation. '/lvmrec.scrpt' only fixes the root logical volume
information in the BDRA.The "swap" and "dump" areas can be updated
via the "lvlnboot" command.

 Example:
    lvlnboot -s /dev/<vg00>/lvol2
    lvlnboot -d /dev/<vg00>/lvol3
 In this example 'lvol2' and 'lvol3' are the "swap" and "dump"
  logical volumes respectively.

step 4. Perform any further data recovery deemed necessary.
```

```
*** NOTE ***
If the same volume group contains more than one corrupted bootdisk
, Repeat the above steps for each disk that needs to be repaired.
```

This completes the process for rebuilding the bootlif and installing critical files.

Installing Critical Root Files Only

Following is an example of the detailed procedure for installing all the critical files necessary to boot on the target root file system:

Boot the Core media, following the steps in Chapter 2. You will see some status messages, and then a menu:

```
Welcome to the HP-UX installation process!
Use the <tab> and/or arrow keys to navigate through the following
menus,and use the <return> key to select an item.  If the menu ite
ms are not clear, select the "Help" item for more information.

                    [    Install HP-UX       ]
                    [  Run a Recovery Shell  ]
                    [   Advanced Options     ]
                                [ Help ]
```

Step 1. Select **Run a Recovery Shell**, the screen clears, and the following question appears:

```
Would you like to start up networking at this time? [n]
```

Step 2. If you have no need to access the net, enter: **n**

```
* Loading in a shell...
* Loading in the recovery system commands...
(c) Copyright 1983, 1984, 1985, 1986 Hewlett-Packard Co.
(c) Copyright 1979 The Regents of the University of Colorado,
a body corporate
(c) Copyright 1979, 1980, 1983 The Regents of the
      University of California
(c) Copyright 1980, 1984 AT&T Technologies.  All Rights Reserved.
                  HP-UX SYSTEM RECOVERY CORE MEDIA
                  WARNING:  YOU ARE SUPERUSER !!
NOTE: Commands residing in the RAM-based file system are unsupport
ed 'mini' commands. These commands are only intended for
recovery purposes.
Loading commands needed for recovery!
WARNING: If ANYTHING is changed on a root (/) that is mirrored,
‘maintenance mode’ (HPUX -1m) boot MUST be done in order to force
the mirrored disk to be updated!

Press <return> to continue.
```

Step 3. Press: **Return**

```
Loading commands needed for recovery!
```

Then the following menu will be displayed:

```
HP-UX CORE MEDIA RECOVERY
   MAIN MENU
 s.  Search for a file
 b.  Reboot
 l.  Load a file
 r.  Recover an unbootable HP-UX system
 x.  Exit to shell
 c.  Instructions on chrooting to a lvm /(root).
This menu is for listing and loading the tools contained
on the core media.  Once a tool is loaded, it may be run
from the shell. Some tools require other files to be preset
in order to successfully execute.
Select one of the above:
```

Step 4. To begin the actual system recovery, select **r** to see the HP-UX Recovery Menu:

```
 HP-UX Recovery MENU
 Select one of the following:
a. Mount the root disk and exit to shell only.
b. Recover the bootlif/os partitions.
c. Replace the kernel on the root file system.
d. Both options b and c

v. Read information about VxVM/LVM recovery

m. Return to ‘HP-UX Recovery Media Main Menu’.
x. Exit to the shell.
```

Step 5. To install critical files only, select: **b**

```
DEVICE FILE VERIFICATION MENU
   This menu is used to specify the path of the root file system.
   When the information is correct, select 'a'.
 INFORMATION to verify:
       Device file used for '/'(ROOT) is c1t6d0
       The path to disk is 56/52.6.0
 Select one of the following:
       a.  The above information is correct.
       b.  WRONG!! The device file used for '/'(ROOT) is incorrect
       m.  Return to the 'HP-UX Recovery MENU.'
       x.  Exit to the shell.
```

Step 6. Assuming the root device file is incorrect, select **b**; you will be prompted to enter the correct device filename:

```
 Enter the device file associated with the '/'(ROOT) file system
 (example: c1t6d0):
```

On a system with hard-sectored disks, the prompt and response might look like this:

```
Enter the device file associated with the '/'(ROOT) file system
(example: c0t1d0s1lvm ) :  c0t0d0s13
/dev/rdsk/c0t0d0s13 not a special file
<Press return to continue>
Enter the address associated with the '/'(ROOT) file system
(example: 4.0.1) :  4.0.0
   NOTE: if your '/'(ROOT) is not part of a sectioned disk layout
    enter a 'W' for whole disk layout
       or
    enter a 'l' for an LVM disk layout
  instead of a section number.
 Enter the section associated with the '/'(ROOT) file system
 (example: 13 ):  13
 making rdsk/c0t0d0s13 c 214 0x00000d
 making dsk/c0t0d0s13 b 26 0x00000d
```

Step 7. If you were to enter, for example, **`c1t1d0`** as the root device filename, you would see this:

```
DEVICE FILE VERIFICATION MENU
   This menu is used to specify the path of the root file system.
   When the information is correct, select 'a'.
 INFORMATION to verify:
       Device file used for '/'(ROOT) is c1t1d0
       The path to disk is 56/52.1.0
 Select one of the following:
       a.  The above information is correct.
       b.  WRONG!! The device file used for '/'(ROOT) is incorrect
       m.  Return to the 'HP-UX Recovery MENU.'
       x.  Exit to the shell.
```

Step 8. Since `c1t1d0` is the correct root device filename, select: **`a`**

```
FILE SYSTEM CHECK MENU
   The file system check '/sbin/fs/hfs/fsck -y /dev/rdsk/c1t1d0'
   will now be run.
 Select one of the following:
      a.  Run fsck -y .
      b.  Prompt for the fsck run string on c1t1d0.
       m.  Return to the 'HP-UX Recovery MENU.'
```

Step 9. Select **a** to run **`fsck -y`** to check your file system for corruption.

```
** /dev/rdsk/c1t1d0
** Last Mounted on /ROOT
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
6256 files, 0 icont, 149423 used, 1563824 free (928 frags, 195362
blocks)
Mounting c1t1d0 to the CORE media /ROOT directory...
<Press return to continue>
```

Step 10. Assuming your file system is not corrupted, and you wish to continue with the system recovery, press **Return** to mount your root file system under the Core media / directory.

```
   ***** Downloading files to the target disk *****
x ./sbin/lvchange, 528384 bytes, 1032 tape blocks
./sbin/lvcreate linked to ./sbin/lvchange
./sbin/lvdisplay linked to ./sbin/lvchange
./sbin/lvextend linked to ./sbin/lvchange
                ...
Filesystem      kbytes    used   avail %cap iused  ifree iused Mou
nted on
/ROOT      1713247  149426 1392496  10%  6261 275339  2%   ?
Should the existing kernel be
'left', 'overwritten', or 'moved'?[overwritten]
```

Step 11. To overwrite the existing kernel with your new file system, enter **`overwritten`** or **`over`** at the prompt.

```
 downloading INSTALL to /stand/vmunix
    **** Creating device files on the target disk ****
    ******* Renaming the following files: *******
     '/.profile' has been renamed '/.profileBK'
    RECOVERY COMPLETION
    MENU
     Use this menu after the recovery process has installed all re
quested
     files on your system.
 Select one of the following:
      a.  REBOOT the target system and continue with recovery.
       b.  Return to the CORE Media Main Menu.
```

Step 12. Complete the recovery process by selecting: **a**

```
NOTE:    System rebooting...
   PDC - Processor Dependent Code - Version   1.3
   (c) Copyright 1990-1993, Hewlett-Packard Company,
```

```
       All rights reserved
   16 MB of memory configured and tested.
   Primary boot path:    56/52.5   (dec)
   Alternate boot path:  56/52.3   (dec)
Manufacturing permissions ON
- Main Menu -
        Command                       Description
        -                        -
        BOot [PRI|ALT|<path>]        Boot from specified path
        PAth [PRI|ALT] [<path>]       Display or modify a path
        SEArch [DIsplay|IPL] [<path>] Search for boot devices

        COnfiguration menu        Displays or sets boot values
        INformation menu          Displays hardware information
        SERvice menu              Displays service commands
        MFG menu                  Displays manufacturing commands
        DIsplay                   Redisplay the current menu
        HElp [<menu>|<command>]   Display help for menu or command
        RESET                         Restart the system
- Main Menu: Enter command or menu >
```

Step 13. Enter **bo pri** at the prompt to boot from the primary boot path.

```
Interact with IPL (Y or N)?>
```

Step 14. Enter **n** for unattended boot; several screens of status information will be displayed, followed by this warning:

```
THIS SYSTEM HAS BEEN BOOTED USING A TEMPORARY KERNEL!
DO NOT ATTEMPT TO INVOKE MULTI-USER RUN-LEVEL USING THIS KERNEL!
Type the following command from the shell prompt for more informat
ion
about completing the recovery process:
cat /RECOVERY.DOC
```

Step 15. To obtain more information on the recovery process, enter:

```
cat /RECOVERY.DOC
```

```
1) Restore valid copies of the following files (either from backup
 or
   from the <filename>BK files created during the recovery process
).
   /etc/fstab,       /etc/inittab,  /stand/ioconfig,
   /etc/ioconfig,     /etc/passwd,  /sbin/pre_init_rc,
   /.profile,   and /etc/profile
  NOTE:  The backup archive may be extracted using '/sbin/frecover
' or
   '/sbin/pax' (for backups made with 'tar' or 'cpio').
 If using '/sbin/pax', linking it to 'tar' or 'cpio' will force 'p
ax'
```

```
    to emulate the respective command line interface.
2) Replace /stand/vmunix from backup, since the present kernel is
probably missing desired drivers.
3) If you have an lvm root, refer to /LVM.RECOVER .
```

Step 16. If you have an LVM system, and want more information on recovery procedures, enter:

`cat /LVM.RECOVER`

The file contains the following information:

```
If a card has been added to, or removed from, your system
since the original installation was completed,
there is a chance that the device file for the root disk has chang
ed.
Consequently, before you run the LVM script ./lvmrec.scrpt
(Step 2, below), you should first recover /stand/ioconfig
from backup and reboot.
INSTRUCTIONS to complete your LVM recovery:
The system must now be up now in "maintenance mode".
NOTE: In order for the following steps to lead to a
successful lvm recovery the LVM label information must be valid.
If the bootlif was updated from the RAM-based recovery system,
then "mkboot -l" has already been run to repair this label.

step 1. If the autofile was altered to force the system to boot in
 maintenance mode, use "mkboot -a" to remove the "-lm" option.
 Example:
    to change  "hpux -lm (52.6.0;0)/stand/vmunix"
    to   "hpux   (52.6.0;0)/stand/vmunix"
    use
   mkboot -a "hpux (52.6.0;0)/stand/vmunix"/dev/rdsk/<device file>
```

NOTE To match device file with boot address, use: **`lssf /dev/rdsk/*`**

```
step 2. Run '/lvmrec.scrpt' to repair the following LVM
 configuration information:
        a. LVM records (lvmrec)
        b. BDRA (Boot Data Reserve Area)
        c. LABEL information

 Requirement:  The following files must reside on disk before
        the script can complete:
        a. /etc/lvmtab
        b. /etc/fstab
        c. /etc/lvmconf/<rootvg>.conf
```

```
       d. all device files specified in /etc/fstab
 To run '/lvmrec.scrpt' provide the device filename used to
 access the bootlif as an argument to the script.
 Example:
  /lvmrec.scrpt c0t6d0
  In this example 'c0t6d0' is the device file used to
  access the bootlif.
step 3. Once '/lvmrec.scrpt' completes, issue the command "reboot"
 and bring the system fully up.
 The recovery of the root LVM is complete.  If the '/lvmrec.scrpt'
 issued the following warning:
             "************ I M P O R T A N T ******************"
             "                                                 "
             "Root logical volume has been repaired, but......."
             "you need to reboot the system and repair the Swap"
             "logical volume using the following LVM command:  "
             "   lvlnboot -A n -s /dev/<root lv>/<swap lvol>   "
             "because Recovery has no way to find out what is  "
             "the Swap logical volume information at this point"
             "                                                 "
"*************************************************"

 The Swap and Dump logical volumes will need to be re-configured.

 The BDRA contains the "root", "swap" and "dump" logical volume
 information.  '/lvmrec.scrpt' only fixes the root logical volume
 information in the BDRA.  The "swap" and "dump" areas can be upda
ted
 via the "lvlnboot" command.

 Example:
    lvlnboot -s /dev/<vg00>/lvol2
    lvlnboot -d /dev/<vg00>/lvol3

  In this example 'lvol2' and 'lvol3' are the "swap" and "dump"
  logical volumes respectively.

step 4. Perform any further data recovery deemed necessary.

*** NOTE ***
  If the same volume group contains more than one corrupted boot d
isk,
  repeat the above steps for each disk that needs to be repaired.
```

This completes the process for installing critical files only.

Rebuilding the "bootlif" Only

Boot the Core media, following the steps in Chapter 2. You will see some status messages, and then a menu:

```
Welcome to the HP-UX installation process!

    Use the <tab> and/or arrow keys to navigate through the follow
ing menus,
    and use the <return> key to select an item.  If the menu items
 are not
    clear, select the "Help" item for more information.

                          [    Install HP-UX       ]
                          [  Run a Recovery Shell  ]
                          [   Advanced Options     ]
                               [ Help ]
```

Step 1. Select: **Run a Recovery Shell** . The screen clears, and this is displayed:

```
Would you like to start up networking at this time? [n]
```

Step 2. Enter: **n**

```
* Loading in a shell...
* Loading in the recovery system commands...

(c) Copyright 1983, 1984, 1985, 1986 Hewlett-Packard Co.
(c) Copyright 1979 The Regents of the University of Colorado,
    a body corporate
(c) Copyright 1979, 1980, 1983 The Regents of the
    University of California
(c) Copyright 1980, 1984 AT&T Technologies.  All Rights Reserved.

                  HP-UX SYSTEM RECOVERY CORE MEDIA
                    WARNING:  YOU ARE SUPERUSER !!

NOTE: Commands residing in the RAM-based file system are unsupport
ed 'mini'commands.

These commands are only intended for recovery purposes.

Loading commands needed for recovery!

      WARNING: f ANYTHING is changed on a root(/) that is mirrored
               a 'maintenance mode'(HPUX -lm) boot MUST be done in
                order to force the mirrored disk to be updated!!

     Press <return> to continue.
```

Step 3. Press: **Return**

```
Loading commands needed for recovery!
```

After boot steps, this message appears:

```
 HP-UX CORE MEDIA RECOVERY
   MAIN MENU
 s.  Search for a file
 b.  Reboot
 l.  Load a file
 r.  Recover an unbootable HP-UX system
 x.  Exit to shell
 c.  Instructions on chrooting to a lvm /(root).
Select one of the above:
```

Step 4. To begin the actual system recovery, select: **r**

```
       HP-UX Recovery MENU
 Select one of the following:
a. Mount the root disk and exit to shell only.
b. Recover the bootlif/os partitions.
c. Both options b and c

v. Read information about VxVM/LVM recovery

m. Return to 'HP-UX Recovery Media Main Menu'.
x. Exit to the shell.
```

Step 5. Select **c** to rebuild the bootlif.

```
    BOOTLIF PATH VERIFICATION
       MENU

     This menu must be used to determine the path to the bootlif

(ISL, HPUX and the AUTO file).
    When the information is correct, select 'a'.

INFORMATION to verify:
     Path to the bootlif is 56/52.1.0

Select one of the following:
     a.  The above information is correct.
     b.  WRONG!! The path to bootlif is incorrect.

     m.  Return to the 'HP-UX Recovery MENU.'
     x.  Exit to the shell.

    Selection:
```

Step 6. BOOT STRING VERIFICATION MENU

```
    This menu must be used to verify the system's boot string.
    When the information is correct, select 'a'.

 INFORMATION to verify:
      The system's boot string should be:
       'hpux -lm (56/52.5.0)/stand/vmunix'

 Select one of the following:

     a.  The above information is correct.
      b.  WRONG!! Prompt the user for the system's boot string.

      m.  Return to the 'HP-UX Recovery MENU.'
      x.  Exit to the shell.

 NOTE: For an LVM '/'(ROOT) the '-lm' option MUST be specified
(example: 'hpux -lm (2.3.4)/stand/vmunix' )

 Selection:
```

Assuming that the bootlif path is correct, enter: **a**

Step 7. Assuming the boot string is incorrect, enter **b** at the prompt. You will see a message similar to the following:

```
AUTO FILE should be (replacing 'hpux (56/52.5.0)/stand/vmunix'):
```

Step 8. Enter the correct information (for example, **hpux**); you will then see the BOOT STRING VERIFICATION MENU displayed again:

```
BOOT STRING VERIFICATION MENU

    This menu must be used to verify the system's boot string.
    When the information is correct, select 'a'.

 INFORMATION to verify:
      The system's boot string should be:
       'hpux'

 Select one of the following:
      a.  The above information is correct.
      b.  WRONG!! Prompt the user for the system's boot string.

      m.  Return to the 'HP-UX Recovery MENU.'
      x.  Exit to the shell.

 NOTE: For an LVM '/'(ROOT) the '-lm' option MUST be specified
```

```
        (example: 'hpux -lm (2.3.4)/stand/vmunix' )

 Selection:
```

NOTE Use the **-lm** option to enter LVM administration mode only when recovering an LVM system.

Use the **-V** option to enter VxVM administration mode only when recovering a VxVM system.

Step 9. Assuming the information is now correct, enter: **a**

- For an LVM system, you will see something like the following:

```
      *********** Installing bootlif  ***********
mkboot -b /dev/rmt/1m  -i ISL -i HPUX /dev/rdsk/c1t1d0
mkboot -a  hpux (56/52.5.0;0)/stand/vmunix /dev/rdsk/c1t1d0

 RECOVERY COMPLETION MENU
  Use this menu after the recovery process has installed all
requested files on your system.
 Select one of the following:
   a.  REBOOT the target system and continue with recovery.
   b.  Return to the CORWE Media Main Menu.
  Selection:
```

Step 10. Complete the recovery process by selecting **a**, rebooting your system.

This completes the process for rebuilding the bootlif only.

Replacing the Kernel Only

Boot the Core media, following the steps in Chapter 2. This menu appears after some status messages:

```
                Welcome to the HP-UX installation process!

    Use the <tab> and/or arrow keys to navigate through the follow
ing menus,and use the <return> key to select an item.  If the menu
 items are not clear, select the "Help" item for more information.

                          [    Install HP-UX       ]
                          [  Run a Recovery Shell  ]
                          [   Advanced Options     ]
                              [ Help ]
```

Step 1. Click: **Run a Recovery Shell**

```
Would you like to start up networking at this time? [n]
```

Step 2. Enter: **n**

```
* Loading in a shell...
* Loading in the recovery system commands...

HP-UX SYSTEM RECOVERY CORE MEDIA
WWARNING:  YOU ARE SUPERUSER !!

NOTE: Commands residing in the RAM-based file system are
unsupported 'mini' commands. These commands are only intended for
recovery purposes.

Loading commands needed for recovery!

      WARNING:If ANYTHING is changed on a root(/) that is mirrored
              a 'maintenance mode'(HPUX -lm) boot MUST be done in
               order to force the mirrored disk to be updated!!

     Press <return> to continue.
```

Step 3. Press **Return** and the following status message is displayed:

```
Loading commands needed for recovery!
```

Step 4. You will see the following menu:

```
HP-UX CORE MEDIA RECOVERY
   MAIN MENU
 s.  Search for a file
 b.  Reboot
 l.  Load a file
 r.  Recover an unbootable HP-UX system
 x.  Exit to shell
 c.  Instructions on chrooting to a lvm /(root).
This menu is for listing and loading the tools contained
on the CORE media.  Once a tool is loaded, it may be run
from the shell. Some tools require other files to be present
in order to successfully execute.
Select one of the above:
```

Step 5. To begin the actual system recovery, select: **r**

```
 HP-UX Recovery MENU
 Select one of the following:
a. Mount the root disk and exit to shell only.
b. Recover the bootlif/os partitions.
c. Replace the kernel on the root file system.
d. Both options b and c

v. Read information about VxVM/LVM recovery

m. Return to ‘HP-UX Recovery Media Main Menu’.
x. Exit to the shell.
```

Step 6. Select **d** to replace only the kernel on the root file system.

```
DEVICE FILE VERIFICATION MENU
   This menu is used to specify the path of the root file system.
   When the information is correct, select 'a'.
 INFORMATION to verify:
       Device file used for '/'(ROOT) is c1t6d0
       The path to disk is 56/52.6.0
 Select one of the following:
      a.  The above information is correct.
      b.  WRONG!! The device file used for '/'(ROOT) is incorrect.
      m.  Return to the 'HP-UX Recovery MENU.'
      x.  Exit to the shell.
```

Step 7. Assuming the root device file is incorrect, select: **b**

```
 Enter the device file associated with the '/'(ROOT) file system
 (example: c1t6d0):
```

On a system with hard-sectored disks, the prompt and response might look like this:

```
 Enter the device file associated with the '/'(ROOT) file system
 (For example: c0t1d0s1lvm ) :  device_file
/dev/rdsk/device_file not a special file
<Press return to continue>
 Enter the address associated with the '/'(ROOT) file system
 (For example: 4.0.1) :  address
   NOTE: if your '/'(ROOT) is not part of a sectioned disk layout
    enter a 'W' for whole disk layout
       or
    enter a 'l' for an LVM disk layout
  instead of a section number.
 Enter the section associated with the '/'(ROOT) file system
 (For example: 13 ):  13
 making rdsk/c0t0d0s13 c 214 0x00000d
 making dsk/c0t0d0s13 b 26 0x00000d
```

Step 8. If you were to enter **c1t1d0** as the root device filename.:

```
DEVICE FILE VERIFICATION MENU
  This menu is used to specify the path of the root file system.
  When the information is correct, select 'a'.
INFORMATION to verify:
      Device file used for '/'(ROOT) is c1t1d0
      The path to disk is 56/52.1.0
Select one of the following:
    a.  The above information is correct.
     b.  WRONG!! The device file used for '/'(ROOT) is incorrect.
     m.  Return to the 'HP-UX Recovery MENU.'
     x.  Exit to the shell.
```

Step 9. Since c1t1d0 is the correct root device filename, select: **a**

```
    FILE SYSTEM CHECK MENU
  The file system check '/sbin/fs/hfs/fsck -y /dev/rdsk/c1t1d0'
  will now be run.
Select one of the following:
     a.  Run fsck -y .
     b.  Prompt for the fsck run string on c1t1d0.
     m.  Return to the 'HP-UX Recovery MENU.'
    Selection:
```

Step 10. Select **a** to run **`fsck -y`** to check your file system for corruption; you will see a display similar to the following:

```
** /dev/rdsk/c1t1d0
** Last Mounted on /ROOT
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
6256 files, 0 icont, 149423 used, 1563824 free (928 frags, 195362
blocks)
Mounting c1t1d0 to the CORE media /ROOT directory...
Filesystem      kbytes    used   avail %cap iused  ifree iused Mou
nted on
/ROOT               434773  352461   38834  90% 15241  54647 22%
  ?
Should the existing kernel be
'left', 'overwritten', or 'moved'?[moved]over
```

Step 11. To move the existing kernel with your new file system, enter **`move`** at the prompt.

Step 12. Complete the recovery process by selecting **a**, REBOOT the target system.

This completes the process for replacing the kernel only.

System Recovery Questions and Answers

Question: **Can I use a network recovery archive if my system is not network bootable or is not on the same subnet as the Ignite-UX server?**

Yes! One of the new tools developed with the network recovery toolset is `make_boot_tape`, a command that creates a minimal tape that can be used by any client. The tape contains just enough information to boot a client and then connect to the Ignite-UX server where the tape was created. If that is the server where the client's recovery configuration files are stored, the client can then be recovered.

If you initiate recovery archive creation from the Ignite-UX server, the server will warn you if a boot tape is needed for a client. If you ignore this warning, misplace your boot tape, or find that your tape is for the wrong Ignite-UX server, you can always create a new boot tape on the server that you want to use. There is no client-specific information on the tape.

Notice that a tape created by `make_boot_tape` is useful not only for recovery situations, but also for ordinary installations. If you do not want to set up a boot helper for systems not on the same subnet as the Ignite-UX server, you can now simply use `make_boot_tape`.

Question: **What happens if I make changes to a recovery configuration before clicking Go! ?**

In the past, Ignite-UX could not be used to override values in recovery configuration files. Any change made to a recovery configuration from one of the interface screens would result in recovery_mode being turned off, that is, set to false.

This was because Ignite carefully set aside the archive versions of files like */etc/fstab,* */etc/mnttab*, and */etc/hosts* when it was running in recovery mode, and restored them after system configuration was complete. If Ignite-UX allowed the user to make changes to file systems, networking, etc. and continue in recovery mode, the files restored from the archive would be out of sync with the reconfigured system.

Beginning with the Ignite-UX A/B 2.0 releases, you can edit values in any of the interface screens without turning off recovery mode. This is possible because critical system files are no longer automatically overlaid by ones from the archive.

Recall that the */etc/fstab* file that Ignite-UX created while configuring the disks was saved to a backup copy prior to extracting the archive. After Ignite-UX boots from the client disk, this fstab is merged with the fstab file extracted from the recovery archive. This merging allows Ignite-UX to preserve information from the old fstab, like entries for file systems not included in the archive, NFS mounts, etc., while accurately representing the configuration changes requested.

Special care needs to be taken when making changes affecting disks not included in the archive. Imagine that you include `vg00` and `vg01` in your recovery archive but not `vg02`. You intend to leave vg02 untouched and simply let Ignite-UX import it for you.

You have also been planning on adding a new disk to the root volume group, and you decide that this is as good a time as any to do it. You choose your recovery configuration from the list and then go to the **Add/Remove Disks** dialogue, where you choose a disk to add. By mistake you choose a disk already in `vg02` instead of your new disk. When your system comes back up, `vg00` and `vg01` are fine, but `vg02` is corrupted.

To avoid scenarios like this, Ignite-UX now "hides" disks intended to be imported again. If you want to see these disks, click the **Additional...** button on the **Basic** tab and change the setting there to show the disks.

Question: **What happens if the pre-install checks fail after I make changes to a recovery configuration?**

In the past if the results of the pre-install checks included one or more ERRORS, Ignite-UX would have to turn off `recovery_mode` to allow users to interact with the user interface and resolve the errors. Unfortunately after the errors were resolved, it was not possible to turn `recovery_mode` back on for the reasons described above.

This created some frustrating situations for users. Imagine that the root disk on a system before it crashes is a 1GB disk with a whole disk layout (not LVM). The only replacement disk available is a 4GB disk. The system administrator therefore uses the 4GB disk, selects the recovery configuration or the system, and gets the following error message from the pre-install checks:

```
ERROR: The selected root disk is too large to use whole-disk
(non-LVM) partitioning. You must either switch to a smaller
root disk, or switch to using LVM to partition the root disk.
```

An experienced Ignite-UX user could do a little handcrafting to configuration files to work around the problem, but a novice user would be stuck.

With the new changes to Ignite-UX, it is no longer necessary to turn `recovery_mode` off when the pre-install checks fail. Now you can simply resolve the problem using the interface and continue running in recovery mode.

Question: **How can I change my setup so that a network recovery archive 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 now be changed using the interface and will not be overwritten by files extracted from the archive, it is natural to think about sharing recovery archives for systems with identical or nearly identical hardware. But unlike "global" configurations that appear in the configuration list of all clients, network recovery configurations only appear in the configuration list of the client for which they were created.

The source for shared cfgs is the */var/opt/ignite/INDEX* file that is shipped with Ignite-UX, and the source for client-specific cfgs is the CINDEX file that is created by `make_net_recovery` in the */var/opt/ignite/clients/LLA* 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 archive, to the directory of the client without the archive. The fact that the entries in `CINDEX` use relative paths means that you do not have to change the `CINDEX` file when you copy it. (You may need to export the directory containing the archive to the sharing client.)

Question: **I do not want to interact with the user interface after I reboot the client. How can I have my latest network recovery archive chosen automatically?**

As long as the client is currently booted, use **`bootsys -a`** to start the install 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/LLA/config* exists, use the cfg specified there.
- If */var/opt/ignite/clients/LLA/config* does not exist, use the default cfg for the client.

The default cfg for the client is the last `cfg` entry set to true in the CINDEX file if it exists. Otherwise the default `cfg` is the last `cfg` set to true in the INDEX file. Because `make_net_recovery` sets the most recently created recovery `cfg` to true in CINDEX whenever it creates a new archive, it will be the default unless it is manually changed.

Start Task

To have Ignite-UX choose the latest network recovery archive automatically:

1. Rename or remove the config file currently in the client's directory.
2. Run this from the Ignite-UX server:

   `bootsys -a client`

Question: **Why does `ioscan` core dump when running `make_tape_recovery` or `make_net_recovery`?**

It is possible for `ioscan` to core dump due to the value set for the `COLUMNS` environment variable in versions prior to HP-UX 11i. When the `-n` option is used `ioscan` uses the `COLUMNS` variable to determine how wide the screen is when formatting the names of device files. If a core dump occurs, you will see messages similar to these:

```
sh: 13061 Floating exception(coredump)
WARNING: pclose of ioscan returned: 34816
```

Should this happen, unset COLUMNS then rerun make_tape_recovery or make_net_recovery. If you are using an xterm, hpterm, or dtterm to run make_tape_recovery or make_net_recovery, do not resize the terminal window to a smaller size instead minimize it if is in the way.

Question: **Can I run make_net_recovery from a PA server to create an archive for an IPF client or vice versa?**

Yes, this is possible though additional steps must be taken. Although the B.4.0 or later versions of Ignite-UX can be installed on both IPF and PA systems, make_net_recovery command cannot automatically swpackage the network recovery tools into /var/opt/ignite/depots/recovery_cmds depot for a client with mismatched hardware architecture.

For example, a server is an IPF system with the B.4.0 version of Ignite-UX installed and is being used for PA clients. When make_net_recovery is run from the GUI to create a recovery archive of a particular PA client, an error similar to this appears:

```
Error Message: Swinstall(1M) failed to install recovery tools from
: "funhouse: /var/opt/ignite/depots/recovery_cmds"
```

To work around this issue, you have to execute the following steps manually:

1. Run /opt/ignite/lbin/pkg_rec_depot command to generate the IUX-Recovery bundle in /var/opt/ignite/depots/recovery_cmds on the IPF server:

   /opt/ignite/lbin/pkg_rec_depot

2. Run swcopy the entire Ignite-UX product with the right architecture for the client.

   swcopy -x layout_version=0.8 -s /[*depot path*] B5725AA @ \ /var/opt/ignite/depots/recovery_cmds

3. Ensure that the correct bundles are now in

   /var/opt/ignite/depots/recovery_cmds, by entering:

   swlist -l bundle -s /var/opt/ignite/depots/recovery_cmds

   Messages similar to the following should appear:

   ```
   # Initializing...
   ```

```
# Contacting target "funhouse"...
#
# Target:  funhouse:/var/opt/ignite/depots/recovery_cmds
#


B5725AA              B.4.0.203      HP-UX Installation
Utilities    (Ignite-UX)


IUX-Recovery         B.4.0.203      Ignite-UX network
recovery tool subset
Ignite-IA-11-22      B.4.0.203      HP-UX Installation
Utilities for
Installing 11.22 IA Systems
Ignite-UX-10-20      B.4.0.203      HP-UX Installation
Utilities for
Installing 10.20  Systems
Ignite-UX-11-00      B.4.0.203      HP-UX Installation
Utilities for
Installing 11.00  Systems
Ignite-UX-11-11      B.4.0.203      HP-UX Installation
Utilities for
Installing 11.11  Systems
```

4. Re-launch Ignite-UX, enter:

   ignite

Now, you can create archives for different hardware architectures from this IPF server. The same steps can be used on PA servers to accomplish the same thing.

A Troubleshooting

Likely problem areas described in this appendix are:

- Errors and Warnings.
- Ignite-UX Server Problems.
- Installing Systems with Ignite-UX.
- Installing from Media.
- Installing from Archives (golden images).
- Running swinstall.
- Adjusting File System Size.
- Troubleshooting Large Systems.
- Media Recovery.
- Network Recovery.

This appendix includes troubleshooting hints from the Ignite-UX FAQ. For the latest on Ignite-UX problems and workarounds, see the Ignite-UX FAQ:

`http://www.software.hp.com/products/IUX/faq.html`

To receive the most-recent FAQ via email, send an empty message (no subject or content required) to:

`iux_faq@igniteux.fc.hp.com`

Errors and Warnings

As an HP-UX install progresses, you will see messages relating to the progress being entered into the log file. Usually these messages are related to normal behavior. `ERROR` and `WARNING` messages have the following significance:

`ERROR`	This indicates a serious problem, usually requiring action from the user 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 see a message, or experience unusual behavior, you can use the following sections as prioritized lists of likely problems and their solutions. They are grouped by the following topics, with the problems you are most likely to encounter near the beginning of each section.

Ignite-UX Server Problems

GUI core dumps

Ignite-UX GUI core dumps.

If your system has patch PHSS_12824 (Motif), remove it or install PHSS_13743. Patch PHSS_12824 was found to be bad.

Mixed versions of Ignite-UX

Can't find /d_cfg_mnt_sb61/monitor_bpr after updating the server.

This is caused by having a mix of Ignite-UX fileset revisions on your server. In most cases it happens when you update only one release bundle (like Ignite-UX 10.20) 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 revision, if not then you need to install all consistent versions. If you have **boot-helper** systems (see Appendix B), they also need to have the Ignite-UX product updated to match the same revision as the server that they reference.

Cannot determine dump size limit

Ignite-UX servers "Cannot determine dump size limit."

If you have a saved client config created using a version of Ignite-UX prior to 1.51, and then update Ignite-UX to 1.51 or later, and if you initiate the install from the Ignite-UX server GUI, it will give an ERROR looking something like this:

```
ERROR: Unable to determine dump size limit for disk (8/4.8.0),
release (B.10.20). Internal Ignite-UX error.
```

The problem is that an old version of the clients' `hw.info` file is being examined by the new Ignite-UX. To fix things, merely boot up the client system using:

```
boot lan.IP install
```

or whatever syntax your system supports from the boot prompt. *IP* is the IP address of your Ignite-UX server. The client `hw.info` file will be updated, and everything should proceed normally.

Installing Systems with Ignite-UX

Installs stop after 30 INDEXes

Installs are broken after 30 INDEX entries.

A modification was made to the **Basic** tab in Ignite-UX 1.42 so that if the `INDEX` file has more than 30 configurations (18 in the terminal user interface), then a list with scrolling is used. For example:

```
[Configurations:...] HP-UX B.11.00 Default [Description...]
```

One of the side effects of this is that sometimes a chosen configuration from a large list doesn't get parsed correctly.

Either update to the current Ignite-UX version or make sure that the text name for each configuration (when there are more than 30) is 25 characters or less. Then the chosen configuration will be correctly parsed. For example, this config name:

```
cfg "HP-UX B.10.10 long configuration name" {
description "long configuration name."
"/opt/ignite/data/Rel_B.10.10/config"
"/var/opt/ignite/config.local"

}
```

has a configuration of 37 characters, `"HP-UX B.10.10 long configuration name"`

and will not be correctly parsed. Shortening this will solve the problem:

```
cfg "HP-UX B.10.10 l.c.n." {
```

Another work-around is to use 30 configurations at most.

samreg errors

Igniting from an archive returns numerous "samreg" errors.

The problem is that the SAM filesets haven't been configured when certain products are trying to register themselves with sam. The workaround is to place the following config stanza in `/var/opt/ignite/config.local` or directly in the config file with the "core" `sw_source`:

```
sw_source "core"
{
post_load_cmd += "

swconfig -xautoselect_dependencies=false
-xenforce_dependencies=false SystemAdmin.SAM "
}
```

Problem installing clients on multiple subnets

Problems installing clients on multiple subnets.

There are a couple of problems with having an Ignite-UX server that is multi-homed (connected to multiple subnets):

- The `instl_bootd` daemon allocates IP addresses from the `instl_boottab` file without knowing which IP addresses are valid for the subnet that the client is requesting to boot from. 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 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 LLA address. This will ensure that `instl_bootd` will allocate 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` .

  — Configure a boot-helper system on each subnet that the client can boot from before contacting your central Ignite-UX server. See Appendix C.

- 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 install 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", then 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 screen 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 needs more file system space than expected.

Ignite-UX adds in minfree (normally 10%) to the amount required by the software impact. You may have software bundles that have overlapping contents (filesets and/or files). `make_config` 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-10-*xx* bundles all overlap quite a bit, so that when you load all of them via Ignite-UX, it estimates too much space. To find the space needed, add the `sw_impact` of all the `sw_sel`s that you are loading.

Debugging SD during cold install

How do I monitor SD operation during cold installs from the Ignite-UX server?

The first level debug of SD produces copious output. Because the logs are captured on the servers as well as the client this would run Ignite servers out of disk space rapidly if every install had this turned on.

It is fairly straight forward to do this on a per-session basis without modifying the config files. From the initial Ignite-UX menu, select: **Advanced Options -> Edit config file** This will run vi and you could add, for example:

```
env_vars += "SDU_DEBUG_RPC=1"
sd_command_line += " -x logdetail=true -x loglevel=2
```

Booting older workstations

An HP 9000 715/50 workstation won't network boot off of the Ignite-UX server.

Older Series 700 workstations require `rbootd` running on the server. If the server is FDDI, `rbootd` won't run. In that case boot from media then switch the source to the Ignite-UX server. Older ones use `RMP` not `BOOTP` and require rbootd to translate and hand off to `bootpd`.

`RMP` clients are the older Series 700 workstations: 705, 710, 715, 720, 725, 730, 735, 750, 755.

`BOOTP` clients are the Model 712 and future workstations, as well as K-Class, D-Class and newer Series 800 servers.

Booting errors

Error: `IPL error: bad LIF magic`

Possible problems are:

- Not having tftp access to `/opt/ignite` and `/var/opt/ignite`, the `/etc/inetd.conf` file on the server should have an entry such as:

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

  If not, fix `inetd.conf` and run **`inetd -c`**. Kill any `tftpd` processes that may be running. Loading Ignite-UX should set up inetd.conf. If not, check SAM.

- Using a bootptab entry for the client that is referencing a non-existent boot file (bf).
- A corrupted `/opt/ignite/boot/boot_lif` file.
- Perhaps some remnants of the old cold-install product (NetInstall) conflicting with Ignite-UX (old `instl_bootd` running)
- A defect in the `rbootd` daemon delivered in patch PHNE_10139. If you have this patch loaded and do not need it for DTC devices, try removing it or updating to patch PHNE_11017 (10.20) or PHNE_11016 (10.10).

Booting on IPF with /etc/bootptab error

Error: PXE-E16: Valid PXE offer not received.

Exit status code: Invalid Parameter

When using `/etc/bootptab` to define Ignite-UX boot servers, 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 on the `tftp` line.
   - Check to see if `inetd` was restarted or given an option to re-read the configuration files (`inetd -c`), after the files were edited? Is the `inetd` process running?
   - Check for entries in `/var/adm/inetd.sec` that may cause `inetd` to deny service to certain clients.
   - Check `/var/adm/syslog/syslog.log` to make certain `inetd` was restarted, and that no bad messages are found.
   - Check for messages from `bootpd` and `tftpd`.
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 insure 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.
   - 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 ensure that any router between the two allows the forwarding of `bootp` requests (Note that the configuration is router specific.).

3. Check `tftpd`

— Check the `tftp` line in `/etc/inetd.conf` to make certain `/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

I put `control_from_server=true` and `run_ui=false` in the `INSTALLFS`, but I still get prompted for information on the client.

Possible problems are:

- If the screen is showing the client name in an editable field and a cancel button at the bottom of the screen, then all is well and there should be an icon waiting for you on the Ignite-UX screen. The text screen allows you to change the icon name, or switch to a client side install.
- If the screen is showing two or more lan interfaces to select from, then there wasn't enough information in the config files to tell it which LAN to use. Once you select a LAN and select **Install HP-UX**, you should be set.
- If the screen is prompting you for networking information, then either DHCP didn't respond or there isn't an entry in `/etc/bootptab` for the client. Enter the network information, select **Install HP-UX** and continue the install.

Mount errors when igniting 10.20 systems

Errors regarding mounting a file system occur when igniting 10.20 systems.

Patch PHCO_18317 supplies a new version of `/sbin/mount` but is not compatible with Ignite-UX. If this patch is loaded via either an archive or SD, then the next `swinstall` session will have fatal errors that appear like this

```
ERROR:   "c02380:/":  One or more filesystems that appear in the
filesystem table are not mounted and cannot be mounted.
ERROR:   Entry for filesystem "/dev/vg00/lvol1" in "/etc/fstab"
could
not be mounted.  If you do not want this file system mounted,
comment it out of the "/etc/fstab" file, or set the
"mount_all_filesystems" option to "false".
ERROR:   Cannot continue the Analysis Phase until the previous
errors
are corrected.
```

One workaround is to add the following to your configuration:

```
sd_command_line += " -xmount_all_filesystems=false "
```

However this has the unpleasant side effect that each swinstall session produces a warning message stating that file systems will not be mounted.

Patch PHCO_19694 replaces PHCO_18317. Note that recovery methods are unaffected because they are solely OS archives, and no SD activity takes place.

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 daemons `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 is 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 the user 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 each system in `/var/statmon/sm` and inform them to flush their cache regarding this client.

When you re-install a system, the `/var/statmon/sm` directory is wiped out. In this case, if the reinstalled system tries to re-contact a server that has cached information, the server will try to communicate over a 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 thus 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, and thus rebooting or restarting the daemons will tell the server to flush it's 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` which contains the server's name. This will cause the first boot to generate a crash recovery notification message to each server, causing them to purge the stale port information. Below 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
"
```

bootsys seems to work in reverse

With `bootsys -w client`, the client doesn't wait for the server. With `bootsys client`, the client waits for the server.

This was probably due to your running through the UI 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 boots it picks that up and goes. Ignite-UX tells you that the install will happen the next time that **`bootsys -w`** is used, but does not really say it will happen automatically. And, the next time you did a bootsys, you had not used the UI without the client being booted from the server.

Booting diskless clients

bootsys does not work on diskless clients.

bootsys does not support rebooting HP-UX 9.x and 10.x diskless clients from the Ignite-UX server.

If you need to remotely reboot diskless clients from the Ignite-UX server, follow these steps:

1. If you need to duplicate the behavior of the `-w` or `-a` options to bootsys, you will need to modify the INSTALLFS file using instl_adm to set the keywords `run_ui` and/or `control_from_sever` appropriately. Or you can do this using Ignite-UX under the **Options -> Server Configuration** menu (Run client installation UI option).
2. Copy the `/opt/ignite/boot` directory and contents to the diskless server as `/opt/ignite/boot`:

   `rcp -r /opt/ignite/boot  diskless-server:/opt/ignite/boot`
3. Edit the client's entry in /etc/bootptab on the diskless server to set the `bf` (boot file) flag to be `/opt/ignite/boot/boot_lif`:

   ```
   bf=/opt/ignite/boot/boot_lif
   ```

   You may also need to set the bootptab entries for the gateway (`gw`), and subnet-mask (`sm`). The networking information in the bootptab file will satisfy the client's DHCP request for networking information when it boots. So it will need everything required to contact the Ignite-UX server.
4. Run `setup_tftp` on the diskless server to allow tftp access to `/opt/ignite`:

   `/usr/sam/bin/setup_tftp /opt/ignite # on 9.X systems`

   `/usr/sbin/setup_tftp /opt/ignite # on 10.X systems`

5. With this setup, the next time you reboot the client from the diskless server it will load the INSTALL kernel and INSTALLFS file system from the diskless server. The client will then contact the Ignite-UX server and the installation can proceed as usual.

Server not listed

search lan install doesn't list the server.

Check these items on the Ignite-UX server from which you are trying to boot:

- Messages from `instl_bootd` in `/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 that the rbootd daemon is running. rbootd always runs, where as `instl_bootd` is started via `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 install 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.

bootsys fails with insufficient space

bootsys fails due to insufficient space in the /stand volume

bootsys needs to copy the two files: `/opt/ignite/boot/INSTALL` and `/opt/ignite/boot/INSTALLFS` from the server into the client's `/stand` directory. This error indicates that there is not enough space in /stand. To fix this, you may need to remove any backup kernels. Also check for kernels in the `/stand/build/` directory (like `vmunix_test`).

bootsys failure ***bootsys -i configuration [sys1...] fails.***

A defect in Ignite-UX A/B.2.2 releases prevents bootsys from successfully pushing a specific configuration out to a client. To fix this:

1. Enter: **`vi /opt/ignite/bin/bootsys`**
2. Move to line 848 in the file:

   ```
   if [[ "c_opt" != "$PUSH_MODE" ]]; then
   ```

3. Change `c_opt` to `$c_opt` and save the change. bootsys will now work correctly.

This defect will be fixed in the Ignite-UX A/B 2.3 release.

Installing from Media

DCE/RPC errors

DCE / RPC errors (RPC exceptions) occur during the configuration stage, plus 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 workaround 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.

Patch installation hangs

swinstall hangs during patch software analysis.

If you have created a CD-ROM that uses depots containing patches and the swinstall command that is loading the patches hangs, then you may be running into a defect in the `df` command.

To be sure, type **`^C^C^C`** until you get a prompt asking if you want to stop the install. Answer **`yes`**, then answer **`yes`** to push a debug shell. From the shell, run **`ps -ef`** and look for a hung `df` command.

The problem is caused by a defect in the df command. The defect is that it hangs whenever it sees a mount entry with a one-character device string (in this case the mount device is "`/`").

The workarounds for this problem are:

- If the core OS is loaded from an archive, make sure that the latest `df` command patch is part of that archive (PHCO_15344 or its successor)

- If the core OS and patches are both in the same depot, and you are using the `hw_patches_cfg` config file to cause loading of the patches, then add the following to your config file:

```
sw_source "core patches" {
   pre_load_cmd += "
      sed '/^. /d' /etc/mnttab > /tmp/mnt.fix &&
      cp /tmp/mnt.fix /etc/mnttab
      rm -f /tmp/mnt.fix
   "
}
```

- Install the latest version of Ignite-UX.

Installing from Archives (golden images)

Can't 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 target machine. Check these possibilities:

- File has a different name (check your config files).
- File has the wrong permissions such that it is not readable (check `/etc/exports`).

Missing `.conf` files

`/etc/nsswitch.conf` and `/etc/resolv.conf` files from the archive don't end up on the install target.

Ignite-UX changes some files during the configuration process, including `resolv.conf` and `nsswitch.conf`. Ignite-UX's `os_arch_post_l` and `os_arch_post_c` scripts place these files on the target system 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 config file which describes the archive to point to the new script.

pax_iux errors

```
pax_iux: X: Cross-device link
pax_iux: X: File exists
```

Both of these errors may occur when loading 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, in which case the root file system is all one file system. And, say you have two files: `/usr/local/bin/f1` and `/opt/myprod/bin/f2` are hard links. This error will occur 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 loaded. This may happen for example if the original system that the archive was made from has a symlink like */opt/myprod -> /extra/space*. And then when you are loading a system from the archive you decide to create a mounted file system as `/opt/myprod`. The pax command will fail to create the symlink because a directory exists in it's place.

When the error happens you will be asked if you want to push a shell (on the target's console). Answer **yes**, and from the shell, enter **`exit 2`** to ignore the error and it will continue on. 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 systems.

Running `swinstall`

Tape not readable

`swinstall` cannot read the tape. For example you may see:

```
Source connection failed for "ignitesvr:/dev/rmt/0m".
```

Possible causes and fixes are:

- Wrong device file — Use the **Actions** menu in SAM's **Peripheral Devices/Tape Devices** area to show the device files for the tape drive.
- Failure reading the contents of the tape:
  - — No device file present for the tape — Use the **Actions** menu in SAM's **Peripheral Devices/Tape Devices** area to create the device files for the tape drive.
  - — Bad/wrong tape — Verify label on tape. Check the contents. SD tapes are in tar format:

    `tar tvf` *`device_file`* `| more`

    For example, if the tape device is `/dev/rmt/0m`, enter:

    `tar tvf /dev/rmt/0m | more`

    You should see a tar format table of contents. If you do not see this, the tape is corrupt.
  - — Dirty head in DDS tape drive — Use a DDS tape cleaning cartridge to clean the tape head.

Adjusting File System Size

The absolute minimum /usr file-system sizes needed to update to HP-UX 11.0 are:

- For 32-bit: 300 MB.
- For 64-bit: 324 MB.

If the required file-system size for the bundle you copy to a depot exceeds that file system limit set by your disk installation, you will get an error condition during the copy process. Use lvextend and extendfs in this situation to create a larger file system. You might have a problem updating your system(s) if the /usr or /var volume is too small.

If you try an update, swcopy determines how much disk space is required. If there isn't sufficient space, swcopy reports an error:

```
ERROR: The used disk space on filesystem "/var" is estimated
to increase by 57977 Kb.
This operation will exceed the minimum free space for this
volume.  You should free up at least 10854 Kb to avoid
installing beyond this threshold of available user disk space.
```

In this example, you would need to increase the file system size of /var by 10 MB, which actually needs to be rounded up to 12 MB.

Follow these steps to increase the size limit of /var:

Step 1. Determine if any space is available by entering:

/sbin/vgdisplay

You should see a display like this:

```
         - Volume groups -
VG Name                     /dev/vg00
VG Write Access             read/write
VG Status                   available
Max LV                      255
Cur LV                      8
Open LV                     8
Max PV                      16
Cur PV                      1
Act PV                      1
Max PE per PV               2000
VGDA                        2
PE Size (Mbytes)            4
```

```
Total PE                  249
Alloc PE                  170
Free PE                   79
Total PVG                      0
```

Free PE indicates the number of 4MB extents available, in this case this is 79 (equivalent to 316 MB).

Step 2. Shutdown the system: **`/sbin/shutdown`**

Change to single user state. This will allow `/var` to be unmounted.

Step 3. Enter: **`/sbin/mount`**

You will see a display similar to this:

```
/ on /dev/vg00/lvol1 defaults on Sat Mar 8 23:19:19 1997
/var on /dev/vg00/lvol7 defaults on Sat Mar 8 23:19:28 1997
```

Step 4. Determine which logical volume maps to */var*. In this example, it is `/dev/vg00/lvol7`.

Step 5. Enter: **`/sbin/umount /var`**

This is required for the next step, since extendfs can only work on unmounted volumes.

Step 6. Extend the size of the logical volume:

`/sbin/lvextend -L` ***`new_size_in_MB`*** **`/dev/vg00/lvol7`**

For example, this makes the volume 332 MB:

`/sbin/lvextend -L 332 /dev/vg00/lvol7`

Step 7. Extend the file system size to the logical volume size:

`/sbin/extendfs /dev/vg00/rlvol7`

Step 8. Enter: **`/sbin/mount /var`**

Step 9. Either go back to the regular init state,`init 3` *or*`init 4`, or reboot.

Troubleshooting Large Systems

During system analysis

On a large system such as a T500 with a very large number of disk drives (such as 50 or more), you may see messages such as these during the system analysis phase of cold install:

```
Out of inode- can't link or find disk
or
Write failed, file system is full.
or
File system full.
```

To reduce the likelihood of this problem occurring, before you do the installation you should *turn off any disks you don't plan to use for the installation process and start over.*

After the system is cold-installed, you may wish to add back all the file systems that existed under the previous installation, either manually or using SAM. However, for a large number of file systems (for example, over a hundred), some tables in the kernel may be too small to allow correct booting. This is because the newly-installed kernel contains default values for kernel table sizes, and does not allow for special configurations made to the kernel installed previously.

For example, the first boot after adding the file systems may result in error messages displayed to the console, such as the following:

```
inode: table is full
proc: table is full
file: table is full
```

Boot failures

The boot may fail in various ways. You may be have to do file system repair manually. If this is not possible, the kernel may need to be re-configured before booting. The following settings should allow the kernel to be booted, but may not be optimal for the system:

```
ninode = 2048 (default is 476)
nproc = 1024 (default is 276)
nfile = 2048 (default is 790)
```

Alternatively, you can re-configure the kernel by either raising `maxusers` to a large value, such as `200`, or selecting an appropriate bundle of SAM-tuned parameters from the SAM **Kernel Configuration Actions** menu. Be sure to determine the correct configuration for your system.

Media Recovery

tftp access

When running the recovery system option from a client booted in Ignite-UX, errors seem to point to files not being accessible via tftp.

Only `/opt/ignite` and `/var/opt/ignite` should be needed for tftp access.

`make_sys_image` broken

Level 2 make_sys_image is broken at version 1.51.

Yes, but there is a quick fix:

Change line 1303 of `/opt/ignite/data/scripts edit make_sys_image` from:

```
if [[ ${recovery_mode} != "TRUE" ]]; then
```

to:

```
if [[ ${recovery_mode} = "TRUE" ]]; then
```

Hot-swapping disks

Problems hot swapping disks during recovery.

Ignite-UX supports only hot swappable disks that are completely installed, 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 in the middle. The LVM command `lvlnboot` used by `save_config` does not work when a disk is removed and the system is in this odd state. If this command is not working, then `make_recovery` has no chance of succeeding.

Volume group error

Error: `The minor number of the volume group exceeds the value IUX can support`.

The `make_net_recovery` command check to ensure that it does not back up a system that Ignite-UX will be unable to recreate.

Ignite-UX version A.*x*.*x* can only create volume groups with group numbers in the range 0 to 10. This is due to the *`maxvgs`* kernel tunable being set to 10 in the INSTALL kernel. In order to continue to have Ignite-UX work on systems with 32 MB of memory, the kernel cannot have this parameter increased.

Ignite-UX B.*x*.*x* does not have this restriction due to reductions in the amount of memory LVM consumes.

`make_net_recovery` can operate on non-root volume groups so it is not uncommon to see the error with this tool.

Possible workarounds:

- If you got into this situation by manually recreating the LVM device files, then consider renumbering them to something less than 10.
- If using `make_net_recovery`, exclude that volume group from the archive.
- Use Ignite-UX B.*x*.*x* on HP-UX 11.0/11i systems.

`vgcreate` error

vgcreate error during recovery of a 9GB disk.

If you used Ignite-UX 1.48 or 1.49 to create a recovery tape of a system with a 9GB disk, you may experience a failure when you try to use this tape. The failure would result in the message similar to the following:

```
* Creating volume group "vg00".
vgcreate: Not enough physical extents per physical volume.
Need: 2170, Have: 2169.
ERROR:   Command "/sbin/vgcreate -A n -e 2169 -l 255 -p 16 -s 4
/dev/vg00 /dev/dsk/c1t15d0" failed.

The configuration process has incurred an error, would you like to
push a shell for debugging purposes? [y/n]
```

Either update to the latest Ignite-UX release and recreate the tape, or work around the problem when it happens. To work around the problem, answer **`yes`** to push a shell. From this shell, enter the command as shown in the error you get, but add 1 to the value shown for the `-e` option. When you successfully run `vgcreate`, enter **`exit 2`** from the shell to continue the install.

Online diagnostics not restored

Online Diagnostics LIF files are not restored during a recovery.

Ignite-UX destroys the old LIF area on the boot disk and lays down new LIF volumes every time the system is installed. At no point during the installation are the old LIF volumes copied and restored to the disk.

To restore the LIF volumes to the disk, reinstall the application, or look at the SD configure scripts for the application and rerun the commands which put the LIF volumes in place on the disk. For example, for the OnlineDiag bundle, the `/var/adm/sw/products/LIF-LOAD/LIF-LOAD-MIN/postinstall` script puts the OnlineDiag LIF volumes onto the root disk. It uses this command:

```
/usr/sbin/diag/lif/lifload -f /usr/sbin/diag/lif/updatediaglif
```

Bad IPL checksum error

"bad IPL checksum" error when booting B1000, C3000, and J5000.

The 1.8 revision of firmware on the B1000, C3000, and J5000 workstations has a defect that causes them to give a "bad IPL checksum" error when booting from a make_recovery tape (and possibly other bootable tapes as well).

If you have one of these systems, your options are to update the system firmware once a new version with a fix is made available, or to use Ignite-UX version A/B.2.0 or later which works around the problem.

Network Recovery

Failure when archiving large volumes

make_net_recovery fails when the archive is 2GB or more.

`make_net_recovery` uses NFS to write/read the system archive from the client to/from the server. To manage archives greater than 2GB requires that both the client and server use NFS protocol version 3 (PV3). NFS PV3 is available for HP-UX 10.20 when the Networking ACE set of patches are loaded, and is standard on HP-UX 11.0.

If you know you have NFS PV3 and are having problems, check:

- You must be running Ignite-UX version A.2.1(on HP-UX 10.*x*) or later, or B.2.0 (on HP-UX 11.*x*) or later.
- Some NFS patches in the past have caused problems with >2GB files. These problems have been fixed in patches:

  ```
  10.20: PHNE_17619 (S700), PHNE_17620 (S800)
  11.00: PHNE_17247
  ```

- If your NFS server is running HP-UX 10.20 with the newer NFS patches, then the `/etc/rc.config.d/nfsconf` file has a configurable parameter (`MOUNTD_VER`) which determines if the default mount should be PV2 or PV3. This must be set to 3.

  If your clients are running HP-UX 10.20 with the newer NFS patches, the `/etc/rc.config.d/nfsconf` file must have the parameter `MOUNT_VER` set to 3.

Volume groups erased

make_net_recovery version 2.0 erases volume groups that contain only unmounted and raw logical volumes.

`make_net_recovery` version 2.0 has a bug which causes volume groups that contain only unmounted and raw logical volumes to be re-created but not restored, causing loss of data. When recovering a system, the user can specify to not recreate these volume groups, so that data is not lost. However, the user will need to manually import these volume groups after recovery. This problem has been fixed with version 2.1.

Core dumps

make_net_recovery version 2.0 and 2.1 core dumps when archiving more than 8 volume groups.

This problem is fixed with Ignite-UX 2.2.

Problem with NFS mounts

`make_net_recovery` version 2.0 and 2.1 crosses and archives NFS mounts if an essential directory has a symbolic link to something that is NFS mounted, and the path to the NFS mounted directory contains a directory which is a symlink.

This problem is fixed with version 2.2.

LAN address changes

After replacing the client system, the LAN address is now different.

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 (LANIC, LLA, MAC address, etc). If you replace the client hardware, or even the LAN card 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, which you can do from the Boot-ROM console command LanAddress (actual command may vary from system to system). Once you have the new address, then 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 upper-case, and begin with `0x`.

If you already booted the client from the server which caused it to create a new directory, you can just remove that directory before renaming the old directory. Be careful not to remove the original directory or else you will loose the recovery information. For example:

```
cd /var/opt/ignite/clients
mv 0x00108300041F 0x00108300042A
rm old_hostname
```

Volume group error

`Error: The minor number of the volume group exceeds the value IUX can support`.

See "Volume group error" on page 255.

Hot-swapping disks

Problem with hot swappable disks during recovery.

See "Hot-swapping disks" on page 255.

B Using a Boot-Helper System

A system running HP-UX 10.*x*, 11.0, or 11i can use the Ignite-UX server across a gateway if the target system is booted via the `bootsys` command. If the system is booted manually, it will need a helper system to help it boot across a gateway, and enabling the target system to perform this to the local boot-helper system:

`boot lan.`*`IP_address`* `install`

This chapter describes how to configure the boot-helper system.

To boot HP-UX across a gateway, you need a system on the local subnet to provide the target with a minimum core kernel. The helper system can run either HP-UX 10.*x*, 11.0 or 11i. The setup is much simpler if the helper system is running HP-UX 10.*x*, 11.0 or 11i.

Setting Up the Boot-Helper

Follow these steps to setup and use a system on a remote subnet to allow a client to do a network boot and then contact a remote Ignite-UX server:

Setting up an HP-UX 10.*x*, 11.0 or 11i helper

Step 1. Install the Ignite-UX minimum core functionality onto the helper system:

```
swinstall -s /dev/rmt/0m Ignite-UX.MinimumRuntime
```

Step 2. On the helper, run the following command to point the INSTALLFS at the correct Ignite-UX server:

```
/opt/ignite/bin/instl_adm -t Ignite-UX_server_IP
```

Verify that INSTALLFS is referencing the correct Ignite-UX server, and gateway for your subnet:

```
/opt/ignite/bin/instl_adm -d
```

Step 3. Specify a temporary IP address for the helper. On the helper, in the /etc/opt/ignite/instl_boottab file, add the IP addresses that clients can use to boot. The remote subnet needs to have temporary IP addresses to use during an initial bootup. These are located in the /etc/opt/ignite/instl_boottab file, and were provided during the initial Ignite-UX server installation. But, the remote gateway systems cannot use these, so the boot-helper system must supply its own. Therefore create an /etc/opt/ignite/instl_boottab file on the boot-helper system containing lines of the following format. (See /etc/opt/ignite/instl_boottab on the Ignite-UX server for more details). For example:

```
15.1.53.180
15.1.53.181
15.1.53.182
```

Install Using a Boot-helper

Boot up the target machine to the `boot admin` menu, and boot from the helper system. For example:

`boot lan.`*`helper_IP_address`* `install`

If there's only one install server available on the subnet, then just enter:

`boot lan install`

At that point, the install should proceed, controlled from the server by default.

C Configuring for a DHCP Server

HP-UX 10.20 and Ignite-UX supports retrieving network information via the Dynamic Host Configuration Protocol (DHCP). This appendix describes setting up DHCP:

- Overview of DHCP Services.
- Setting Up a DHCP Server.
  - — Details of the DHCP Services.
  - — Enabling DHCP on a System Not Initially Configured with DHCP.
  - — DHCP Usage Examples.
  - — Using bootptab as an Alternative to DHCP.

Ignite allows for setting up DHCP for use during system installation. This appendix is for the user who wishes to use DHCP for ongoing IP address management, as well as for system installation.

Overview of DHCP Services

DHCP provides these features:

- Allows for dynamically allocating IP addresses and hostnames.
- Automatically supplies most of the networking defaults that are requested during a system installation or first time boot.
- Provides for on-going IP address maintenance via 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 request a new IP address lease.
- Assists in re-establishing 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 bring-up.
- 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 the person installing the systems does not desire to choose a hostname for the system, but rather accepts the one already registered for the IP address supplied by DHCP. This will ensure that the system will be recognized immediately by its hostname.

- When existing systems that did not use DHCP before will continue not to use it. Or, if they did, they would be willing to accept an arbitrary hostname and IP address. This is the same as with a new system. There currently is no tool available for pre-loading the DHCP database with existing IP addresses and identifying the systems they belong to. A tool to do this may be available in a future release.

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, please see “Using bootptab as an Alternative to DHCP” on page 275.

Setting Up a DHCP Server

Once you have decided that using DHCP will provide a benefit, you will need to follow the steps below to set up a DHCP server. *Note that only one DHCP server per network subnet is required*. On the server system:

- Allocate a set of currently unused IP addresses (preferably a contiguous block of addresses). For example:

  ```
  15.1.48.50 -> 15.1.48.80
  ```

- Pre-assign and register hostnames to the IP address allocated above. Using the `-h` option to the dhcptools command may be useful. For example, this command creates a `/tmp/dhcphosts` file that can be incorporated into your `/etc/hosts` or DNS/NIS database:

  ```
  dhcptools -h fip=15.1.48.50 no=30 sm=255.255.255.0 hn=devlab##
  ```

- Designate a system to act as the DHCP server for your network. This should be a system that is "always" available to it's clients.

Use SAM to configure the DHCP services on this server:

Step 1. As root, enter: **`sam`**

(you may need to set your `DISPLAY` variable to use the graphical version)

Step 2. Double-click **Networking and Communications**

Step 3. Double-click **Bootable Devices**

Step 4. Double-click **DHCP Device Groups Booting From this Server**

You should now see a screen that lists any DHCP groups already defined (there may not be any if DHCP is not already configured).

Step 5. To add the new group of IP addresses which you allocated in Step 1, click **Action** –> **Add DHCP Group**

Step 6. Now fill in the information on this screen. Some information may require additional research if you are not familiar with the terms or with your network.

- **Group Name —** This can be any name that isn't already defined as a DHCP group. For example: `group1`

- **Subnet Address —** This is the portion of an IP address that is not masked off by the subnet mask (see below). If you don't want to figure this out, then just enter one of the IP addresses in the range you picked along with the correct subnet mask and SAM will take care of the calculation. For example: `15.1.48.50`
- **Subnet Mask —** This depends on the "class" of your network, and basically determines how an IP address is separated into a network number and a host specific number. Press **`F1`** in this field for more information. For example: `255.255.255.0`
- **Subnet Address Pool —** Press this button to select the range of IP addresses that you allocated in Step 1. A new screen will display where you can enter the Start and End address. If there are addresses within the range that you picked that you do not want allocated via DHCP, you can use the Reserved Addresses button to specify those (or ranges of them).
- **Allow Any Device Class —** The SAM default allows any type of DHCP device to use the group of IP address you are configuring. This may be undesirable if you use a different method (or a different DHCP server or group) for managing systems such as PCs running Windows98™ or NT™.

  If you want this range of addresses to be used only by HP-UX systems, then unselect this button, and in the text field provided enter: **`HewlettPackard.HP-UX`**

  When using Ignite-UX to set up DHCP, it will set a class specific to the server, and will set the `dhcp_class_id` string to match. For more detail, see the *instl_adm* manpage.
- **Automatic Allocation to Bootp Clients —** Leave this option disabled. Enabling it will cause problems for bootp devices such as printers and terminals which rely only on their preconfigured server to respond to their boot request.
- **Accept New Clients —** Leave this option enabled.
- **Address Lease Time —** The lease time should be set sufficiently long so that if a client system is temporarily out of service (`off`) for a time, its lease will not expire too soon.

  Infinite leases will never expire and disable the IP-address reclamation features of DHCP. For example: 2 weeks.

- **Boot file name —** You can leave this field blank.
- **Additional Params —** There are many parameters that can be specified in this screen for such things as the default routers, time server, DNS server, and NIS domain. You can specify as much or as little as you like in this area.
- **Callback Routines —** None are necessary.

Step 7. Once the parameters are all filled in, then press **OK** on the Add DHCP Group screen. SAM will then make the modifications to the `/etc/dhcptab` file.

Step 8. Click **Action** –> **Enable Boot Server** (if it is not already enabled).

New systems that are installed with HP-UX 10.20 or newer version or booted with a pre-installed HP-UX 10.20 or newer version should now contact this server to get an IP address lease and other network information provided by the server.

Details of the DHCP Services

- When cold-installing HP-UX 10.20 or newer:

  The installation tools will broadcast out on the network for any available DHCP servers. The first server to respond will be chosen to provide the default network information that the user is presented with.

  In the network parameters screen during a cold install, you see the question: `"Is this networking information only temporary?"` Responding `yes` or `no` answer to this question implies the following:

  — `no` (the default) means that if the IP address and hostname were leased from an DHCP server, then that lease will be retained after the install is done, so that the first boot of the system will attempt to renew the same lease.

  — `yes` implies that the IP address and hostname lease should be returned to the server after the installation is complete. In this case, the first system boot will try to get a new lease. This is most useful when the system is being installed on a network that is different from its final destination.

This answer to the question can also be set in the configuration file with the `instl_adm` command using the keyword: `is_net_info_temporary`

When automating system installations, the DHCP services allows systems to get networking information without mapping the Ignite-UX configuration files. For more information, see *instl_adm* (1M) and *instl_adm* (4).

- When a system boots for the first time (either after a cold install or the first boot of a pre-loaded (Instant Ignition) system):

  With HP-UX 10.20, the `auto_parms` and `set_parms` tools (they let you configure the system identity and basic configuration parameters) will invoke the `dhcpclient`, which will broadcast out to find a DHCP server. The server, in turn, provides a default set of networking parameters.

  With HP-UX 11.0, you are asked if DHCP should be enabled and used (by set_parms). The default is to *not* use DHCP.

  In both cold install and a first boot of a pre-loaded system, if the user chooses not to use the IP address given by the DHCP server, the tool will inform the DHCP server that it can release the lease on it and give it to someone else.

- At each system boot:

  If a client system was initially set up using an IP address that was leased by a DHCP server, that client will check to ensure that the lease is still valid at each boot. In addition, the system will start a daemon process (`dhcpclient -m`) that will maintain and renew that lease while the system is running.

  If a system cannot contact the DHCP server from which it originally got the IP address lease, it will try to contact other DHCP servers in order to determine if it has been moved to a different network. If this is the case, the system will write a message to the `auto_parms` log file (*`/etc/auto_parms.log`*) indicating that it has detected a move to a new subnet and that it is attempting to request a new lease. If the new lease request is successful, new networking configuration values supplied by the DHCP server will automatically be applied.

Enabling DHCP on a System Not Initially Configured with DHCP

If a system has been set up without using DHCP, but you would like to start using it, the following steps may be taken.

NOTE The system's hostname and IP address may change based on what the DHCP server assigns to it the first time it boots.

There are two methods for enabling DHCP on a system that is not currently using it:

Enable DHCP using SAM

Step 1. As root, run: `sam`

Step 2. Double-click **Networking and Communications**

Step 3. Double-click **Network Interface Cards**

Step 4. Highlight the card on which you wish to enable DHCP.

Step 5. Click **Actions** –> **Configure**

Step 6. Click **Enable DHCP**

TIP If **Enable DHCP** appears grayed-out, use the alternate method below.

Step 7. Click **OK** and exit SAM.

Your system will now begin using DHCP after the next reboot. Please note that all of the current networking parameters will be overridden with new values supplied by the DHCP server. *If for some reason the system cannot contact a DHCP server during the next reboot, it will continue to use its current networking parameters*.

If you suspect that your system had problem contacting the DHCP server, examine `/etc/auto_parms.log` to determine if the lease request was successful.

Alternate method

You can also enable DHCP over a particular network interface using a text editor such as `vi` or `emacs` to edit the `/etc/rc.config.d/netconf` file. In the header of this file, you will find some brief instructions regarding a variable named `DHCP_ENABLE`. This variable is tied by an index number to an individual network interface. For example:

```
INTERFACE_NAME[0]=lan0
IP_ADDRESS[0]=15.1.50.76
SUBNET_MASK[0]=255.255.248.0
BROADCAST_ADDRESS[0]=""
DHCP_ENABLE[0]=1
```

Here, the variables are instructing the system to use the `lan0` interface when attempting to contact a DHCP server. Similarly, if the lease request is successful, the above `IP_ADDRESS` variable would be updated to reflect the new value supplied by the DHCP server.

If the `DHCP_ENABLE` variable was set to `0` or if the variable did not exist, no DHCP operations would be attempted over the corresponding network interface.

As noted in the first method of enabling DHCP, if the variable `DHCP_ENABLE` does not exist for a particular interface, SAM will display a grayed out DHCP enable button.

In this case, you will need to add the variable definition to a specific interface variable block. As an example, you would need to add `DHCP_ENABLE[2]=1` to the following interface variable block in order to enable DHCP on the `lan1` interface:

```
INTERFACE_NAME[2]=lan1
IP_ADDRESS[2]=15.1.50.89
SUBNET_MASK[2]=255.255.248.0
BROADCAST_ADDRESS[2]=""
```

The contents of `/etc/rc.config.d/netconf` for this definition block should now look like the following:

```
INTERFACE_NAME[2]=lan1
IP_ADDRESS[2]=15.1.50.89
SUBNET_MASK[2]=255.255.248.0
BROADCAST_ADDRESS[2]=""
DHCP_ENABLE[2]=1
```

Correspondingly, you could disable DHCP over a particular interface by setting the variable to `0`.

Again, as in the first method, the system will only begin using DHCP after the next reboot.

DHCP Usage Examples

To enable a DHCP server to respond only to specific clients during an installation, use `instl_adm` to configure specific *`dhcp_class_id`*s.

Your situation might fall into one of these categories:

- The network has a DHCP server that manages the whole network, and the clients doing installations will be using the addresses from this server permanently. In this case, do nothing since this line is entered in `INSTALLFS` file by default:

  ```
  is_network_info_temporary=false
  ```

- The network has a DHCP server, but the user would like to manage a small group of temporary IP addresses, just for use in doing installations, and the clients will get reassigned new addresses when deployed.

Step 1. Set up DHCP on Ignite-UX server.

Step 2. Use a unique *`dhcp_class_id`* in both the `dhcptab` and the 8KB config file. This *`dhpc_class_id`* could include the server's hostname. In this case, enter the following in `INSTALLFS` using `instl_adm`:

```
is_network_info_temporary=true
```

Step 3. Enter your class ID as the following in the dhcptab and `INSTALLFS`:

```
dhcp_class_id
```

If you have a non-HP server that does not recognize `dhcp_class_id`, specify the server using the `dhcp_server` keyword instead.

- The user would like to setup the Ignite-UX server as a "departmental" DHCP server, in which case the IP address leases are permanent, but they will be isolated to the department's DHCP server.

Step 1. Set up DHCP on the Ignite-UX server.

Step 2. Enter this line:

```
is_network_info_temporary=false
```

Step 3. *And* enter your class ID as the following in the dhcptab and `INSTALLFS`:

`dhcp_class_id`

(Or use the `dhcp_server` keyword as explained in the previous Step 3 above.)

Use a unique *`dhcp_class_id`* in both the dhcptab and the `INSTALLFS` file. This *`dhcp_class_id`* could have the server's hostname in it.

- You want to start using DHCP with this server managing the whole network. Refer to the preceding sections, `/usr/sbin/sam`, and the *sam*(1M) manpage for this procedure.

For more information, see the *setup_server* (1M) and *bootpd* (1M) manpages.

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 BOOTP protocol is a subset of 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, then you do not need any additional entries in `/etc/opt/ignite/inst_boottab`. In this case, you would then boot the clients using `boot lan` instead of `boot lan install`. Only clients known in `/etc/bootptab` would be able to boot if you do not use `instl_boottab`.

A minimal example `/etc/bootptab` entry is shown below (use your own hostname, IP address, hardware address, and subnet mask). Other networking information may also be specified here or via `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=15.1.51.82:\
  sm=255.255.248.0:\
  bf=/opt/ignite/boot/boot_lif
```

Background Information on DHCP Design

The DHCP protocol is implemented as extensions to the BOOTP protocol, and in fact the HP-UX DHCP server daemon and the BOOTP daemon 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 the time that it needs to renew the IP address lease, at which time it will re-contact the server where it got the original lease in order to extend it.

The `dhcpclient` command is not intended to be run by users directly, and is called by other tools during system bootup and installation.

For More Information

See the *auto_parms* (1M)and *dhcpdb2conf* (1M) manpages for more information regarding the networking parameters which DHCP can supply.

More general information on DHCP can be found in the following locations:

- Manpages:

  bootpd (1M)
  dhcptools (1M)
  auto_parms (1M)
  dhcpdb2conf (1M)

- Web:

  `www.dhcp.org`

Index

Index

T