HP-UX 11i Version 1.5 Release Notes

release id B.11.20
for rx9610 and rx4610 hp servers
and i2000 hp workstations

HEWLETT®
PACKARD

Manufacturing Part Number: B9106-90003

May 2001

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 manual and disk(s) or tape cartridge(s) 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.

This software is based in part on the Fourth Berkeley Software Distribution under license from the Regents of the University of California.

Revision History: June 2001, Edition 2.

This edition describes changes between HP-UX 11i and HP-UX 11i

Version 1.5 (B.11.20).

This guide's printing date and part number indicate its current edition. The printing date changes when a new edition is printed. (Minor corrections and updates which are incorporated at reprint do not cause the date to change.) The part number changes when extensive technical changes are incorporated.

New editions of this manual will incorporate all material updated since the previous edition. For the latest version, see the HP-UX 11.x Release Documentation section on the web: docs.hp.com

Typographic Conventions

This manual uses the following typographic conventions:

Boldface Words defined for the first time appear in **boldface**.

`Computer` `Computer font` indicates literal items displayed by the computer. For example: `file not found`

`User input` Bold, computer text indicates literal items that you type. For example, to change to your account's home directory, enter:

cd

Italics Manual titles, variables in commands, and emphasized words appear in *italics*. For example, you would substitute an actual directory name for *`directory_name`* in this command:

cd *directory_name*

[ ] and **|** Brackets **[ ]** enclose optional items in command syntax. The vertical bar **|** separates syntax items in a list of choices. For example, you can enter any one of these three items in this syntax:

`ls [-a | -i | -x]`

Enter Text in **this bold, sans serif font** denotes keyboard keys. A notation like **Ctrl-Q** indicates that you should hold the **Ctrl** key down, then press **Q**.

Contents

1. Overview of the Release Notes

2. Workstation/Server Specific Information

3. HP-UX Features

5. Networking

6. Commands and System Calls

7. Programming

1 Overview of the Release Notes

Welcome to HP-UX

HP-UX 11i Version 1.5 is the release for all Itanium processor family (IPF) based HP-UX systems. The release ID for HP-UX 11i Version 1.5 is B.11.20, and the term **HP-UX 11i Version 1.5 (B.11.20)** will be used throughout the release notes. The HP-UX 11i Version 1.5 (B.11.20) Operating Environment (OE) is an integrated and tested software solution containing the operating system and selected applications.

PA-RISC based systems should continue to run HP-UX 11i (release ID B.11.11) or earlier releases. HP-UX 11i Version 1.5 (B.11.20) will not run on PA-RISC based systems.

This release is not intended for Mission Critical environments, and HA products such as ServiceGuard are not supported on this release. It includes the following major features and enhancements:

- Base VERITAS Volume Manager 3.1 for HP-UX (VxVM), is integrated in HP-UX and is now the default volume manager and root disk
- Support for running most PA-RISC applications on IPF without recompiling
- Enhancements to some commands

The purpose of this chapter is to help you use these release notes along with related HP-UX documentation effectively.

The following topics are covered in this overview:

- What is the Purpose of HP-UX Release Notes
- What's in the Remaining Chapters
- Where Should I Begin
- Release Overview
- Locating Release Notes for Previous Versions of HP-UX
- Other Sources of Information About This Release

What is the Purpose of HP-UX Release Notes

HP-UX Release Notes describe what is new, changed, or obsolete in a release compared to previous releases. HP-UX Release Notes apply only to features that are part of the HP-UX operating system.

Additional product specific release notes files are located in the `/opt` directory, in sub-directories named *`name`*`/newconfig/RelNotes` (where *`name`* represents the name of the product). For example, Distributed Computing Environment (DCE) release notes are loaded in the `/opt/dce/newconfig/RelNotes` directory.

The purpose of the *HP-UX 11i Version 1.5 Release Notes* is to explain the major differences between HP-UX Release 11i and HP-UX 11i Version 1.5 (B.11.20). For full information on changes in previous releases, consult the archival release notes located in: `/usr/share/doc/`*`x.x`*`RelNotes` or at the docs.hp.com web site.

Release notes do not completely document all of the features of a release. Instead, they contain high level information and provide pointers to more detailed operating system and product specific documentation. Where appropriate, release notes also tell you about changes in the support of products.

What's in the Remaining Chapters

The remaining chapters of these release notes are:

- Workstation/Server Specific Information, which presents information on which platforms support the HP-UX 11i Version 1.5 (B.11.20) release and other platform-specific information, including I/O and mass storage cards.
- HP-UX Features, which presents information on changes to the kernel, system administration, and other related subsystems.
- File and Disk Management, which presents information about file systems, including the new EFI file system.
- Networking, which covers changes to networking functionality and networking cards and drivers.
- Commands and System Calls, which includes information of interest to system administrators and users.
- Programming, which covers a wide variety of changes of particular interest to programmers, including changes to libraries.
- Other Functionality, including additional applications in the Operating Environment.

Where Should I Begin

This document contains information about HP-UX 11i Version 1.5 (B.11.20). It includes information on changes between HP-UX 11i and HP-UX 11i Version 1.5 (B.11.20).

Information about the HP-UX 11i release is available in the *HP-UX 11i Release Notes*, available on the web at http://www.docs.hp.com.

Be sure to examine the "*HP-UX 11i Version 1.5 Installation and Configuration Guide*", part number B9106-90001, for details on performing an installation.

Release Overview

HP-UX 11i Version 1.5 (B.11.20) adds software functionality and support for Itanium™-based systems to HP-UX. HP-UX 11i Version 1.5 (B.11.20) is intended for all Itanium processor family (IPF) based HP-UX systems. PA-RISC based systems should continue to run the HP-UX 11i or HP-UX 11.x releases. This release will not run on PA-RISC based systems.

HP-UX 11i Version 1.5 (B.11.20) includes the following major features and enhancements:

- Base VERITAS Volume Manager 3.1 for HP-UX (VxVM), is integrated in HP-UX and is now the default volume manager and root disk
- Support for running most PA-RISC applications on IPF without recompiling
- Enhancements to some commands

Functionality provided in this release will be included in future releases of HP-UX.

The HP-UX Operating Environment (OE)

Beginning with HP-UX 11i, the operating system is delivered as part of the HP-UX Operating Environment (OE). An Operating Environment is an integrated and tested software solution containing the operating system and selected applications. In HP-UX 11i Version 1.5 (B.11.20), the operating system and selected applications are provided on two DVDs, to install the full OE, you must install from both DVDs. For more information, see “*HP-UX 11i Version 1.5 Installation and Configuration Guide*”, part number B9106-90001.

Available Applications

The following applications are available on this release:

- IUX (Cold Install ONLY)
- DCE Remote Procedure Calls (RPC)
- SD-UX
- SAM
- ANSI C++ Compiler

- C Development Bundle
  - ANSI C Compiler
  - WDB
  - Open GL libraries
- Java RunTime, Dev Kit
- JVM
- CxPerf
- OpenView: GlancePlus/Measureware
- Fortran95
- The HP-UX developer’s toolkit bundle (B3394BA)

Unavailable Features

In comparison with HP-UX 11i, HP-UX 11i Version 1.5 (B.11.20) does not provide support for:

- PA-RISC systems
- updates, HP-UX 11i Version 1.5 (B.11.20) must be cold installed
- HFS or LVM root disk, use VxVM or whole disk VxFS instead
- boot from LVM disks
- root, swap or dump on LVM disks
- early dump
- Shared LVM
- PCI OLA/R
- Business Copy/BCV/snapshot features on high-end disk arrays
- Fabric connectivity with Fibre Channel Mass Storage
- High Availability (HA) features

HP-UX 11i Version 1.5 (B.11.20) does not support

- PA cross development on IPF
- HP supplied PA mode archive system libraries.
- HP supplied IPF archived system libraries
- mix & match between IPF and PA binaries and/or 32-bit and 64 bit objects
- network boot
- framebuffer graphics support
- boot from floppy disk

In addition, administrators and developers should make note of the following important changes and limitations:

- the id number returned by uname may no longer be unique

- manual editing of `/stand/system` file is now obsolete

Additional information about the above changes and limitations, and other differences between HP-UX 11i and HP-UX 11i Version 1.5 (B.11.20), are described further in these release notes.

HP-UX Installation Restrictions

Installing HP-UX 11i Version 1.5 (B.11.20) requires a 4 GB minimum root disk. See the *HP-UX 11i Version 1.5 Installation and Configuration Guide* for complete information.

For HP-UX 11i Version 1.5 (B.11.20) the following installation restrictions apply:

- Installing HP-UX 11i Version 1.5 (B.11.20) requires using the DVD media.
- HP-UX 11i Version 1.5 (B.11.20) only supports cold installation of the HP-UX operating system.
- *Updating to HP-UX 11i Version 1.5 (B.11.20) from previous releases is not supported*. For detailed installation information, refer to the Installation and Configuration Guide.
- The HP-UX 11i Version 1.5 (B.11.20) release does not support using Ignite-UX to create and distribute a Golden System or Golden Disk for network installation of HP-UX.

Locating Release Notes for Previous Versions of HP-UX

Release notes are found in the following locations:

- HP Instant Information CD-ROM. See “HP-UX 11i Version 1.5 Instant Information CD-ROM” for more information.
- `/usr/share/doc/11iv1.5RelNotes` on your HP-UX 11i Version 1.5 system. Also included in the `/usr/share/doc` directory are files containing information about previous releases of HP-UX.
- http://docs.hp.com/, the World Wide Web. See The HP Documentation Web Site for more information.

Other Sources of Information About This Release

In addition to these release notes, you have many other sources of information available to you.

README Documents

README documents contain information about the installation process that may not appear in the installation manual. Any product may have a README document, so you may have available several README documents. The README document specific for HP-UX 11i Version 1.5 (B.11.20) is included with your media kit.

White Papers on HP-UX

For HP-UX 11i Version 1.5 (B.11.20), white papers are available on the World Wide Web (see below).

HP-UX 11i Version 1.5 Instant Information CD-ROM

As of HP-UX Release 10.30, Hewlett-Packard introduced a new product, Instant Information, which provides HP-UX documentation on a CD-ROM. This new format replaced the HP LaserROM product as of HP-UX Release 11.0. Instant Information provides improved online presentation, print quality and search capabilities.

Manual Pages

For HP-UX 11i Version 1.5 (B.11.20) systems, the manual pages are available on the HP-UX Welcome Page of your system, from http://docs.hp.com, in Instant Information under the title *HP-UX Reference* volumes 1 through 7, and through the `man` command.

The HP-UX Documentation Set

The HP-UX documentation set describes how to set up and use the basic HP-UX operating system. It includes information on system administration, networking, the X Window System, and so on.

HP-UX Welcome Page

The HP-UX Welcome Page on your HP-UX 11i Version 1.5 system contains pointers to information to help you use your HP-UX system.

The HP Documentation Web Site

Hewlett-Packard provides a web site where the latest HP-UX documentation and updates are available. The site can be accessed through http://docs.hp.com.

2 Workstation/Server Specific Information

This chapter describes workstation and server specific platforms and configurations, including:

- Systems That are Fully Supported
- Hardware Configuration
- Mass Storage Drivers
- SCSI Devices
- Function Key Mappings
- Machine Check

Systems That are Fully Supported

HP-UX 11i Version 1.5 (B.11.20) supports only a 64-bit version of the HP-UX kernel.

The tables below outline the fully supported configurations for rx9610 and rx4610 hp servers, and i2000 hp workstations.

NOTE The information in the following tables is subject to change. For the most up-to-date information, refer to the following web site: http://www.hp.com/products1/itanium/

Table 2-1 **Server Configuration Summary**

Platform	Processor	Cache	Min/Max Memory	# Slots	PCI I/O
rx9610	733Mhz	2MB	256MB to 128GB	64 to 128 PCI	Ultra-2 SCSI, FWD SCSI-2, Gigabit Ethernet, 10/100B-TX, Fibre Channel
	800Mhz	4MB			
rx4610	733Mhz	2MB	1GB to 64GB	10 PCI	Ultra-2 SCSI, FWD SCSI-2, 1000B-SX, 10/100B-TX, Fibre Channel
	733Mhz	4MB			
	800Mhz	4MB			

Table 2-2 **Workstation Configuration Summary**

Platform	Processor	Cache	Min/Max Memory	# Slots	PCI I/O
i2000	733Mhz	2MB	1GB to 4GB	7 PCI	10/100B-TX, USB, external SCSI
	800Mhz	2MB	2GB to 4GB		

Hardware Configuration

rx9610 Server

The rx9610 server can contain up to 4 cells. Each cell includes, among other things, a memory card (16 dimm slots), four Itanium processors, and a connector for an additional memory card (16 additional dimm slots). The configurations for the rx9610 are:

Server	rx9610 - 16-way IPF-based server
Processor	733MHz/2MB 800MHz/4MB
Memory	128GB max memory using 16 SDRAM DIMM sockets per memory expander board per cell
Drives	DVD drive (IDE) Floppy drive LS-120 (IDE) slim line Hard Disk Drives (HDD) 2 hot swap 1" 18GB each, total 36GB
PCI I/O	up to 64 66MHz slots or 128 33MHz slots
Supported Add-On Peripherals	A5272A - Sure Store E Disk System S10 A5236A(z) - Sure Store E Disk System FC10 A5277A - Sure Store E Disk Array FC60 A3311A - High Availability Storage System A5599A - DLT8000 Tape Drive
Supported Add-On Cards	A5150A -Dual Port Ultra2 SCSI A5159A -Dual Port FWD SCSI-2 A4926A - PCI 1000BaseSX Gigabit Ethernet Fiber, single port A4929A - PCI 1000BaseT A6792A - PCI 100BaseT A5158A - Single Port Fibre Channel (Tachlite)

rx4610 Server

The Itanium processor server configurations for the rx4610 are:

Server	rx4610 - a 2-4 way IPF-based server
Processors	733MHz/2MB 733MHz/4MB 800MHz/4MB
Memory	64GB max memory using 32 SDRAM DIMM sockets per memory expander board
Drives	DVD drive (IDE) Floppy drive LS-120 (IDE) slim line 2 hot swap 1" Hard Disk Drives 18GB each, total 36GB
PCI I/O	10 PCI slots - 8 66MHz slots, 2 33MHz slots
Supported Add-On Peripherals	
	A5272A - Sure Store E Disk System SC10 A5236A(z) - Sure Store E Disk System FC10 A5950A - Sure Store E Disk Array XP512 A5700A - Sure Store E Disk Array XP256 A5920A - Sure Store E Disk Array XP48 A5277A - Sure Store E Disk Array FC60 A5616A - Sure Store E Disk System HVD 10 C4318B - Smart Storage Enclosure (DDS4) C6369A/71A - DDS4 Tape Drive A5617A - 10/180 Tape Library A5598A - 9840 Tape Drive A5599A - DLT8000 Tape Drive A5583A(HVD) - 2/20 Tape Library
Supported Add-On Cards	
	A5150A -Dual Port Ultra2 SCSI A5159A -Dual Port FWD SCSI-2 A4926A - PCI 1000BaseSX Gigabit Ethernet Fiber, single port A4929A - PCI 1000BaseT Copper Gigabit Ethernet A6792A - PCI 100BaseT A5158A - Single Port Fibre Channel (Tachlite)

i2000 Workstation

The Itanium-based workstations are available in two configurations, the i2000 733MHZ is a single processor system with 2MB Cache, and 1, 2, or 4GB memory, and the i2000 800MHZ is a dual processor system with 2MB Cache, and 2 or 4GB memory.

Workstation	i2000 - a 1-2 way IPF-based workstation with 2M L2 Cache
Processor	1 733MHz/2MB 2 800 MHz/2MB
Memory	UP - 1GB (add on to 2 or 4GB) MP - 2GB (add on 4 GB)
Drives	DVD drive (IDE) Floppy drive LS-120 or LS-240 (IDE) slim line Hard Disk Drives (HDD) 18GB
Core Built-in I/O	7 PCI slots 10/100BT LAN (10-200B-TX?) USB (4) internal SCSI (Ultra3 Qlogic 12160) PS-2 Keyboard
Graphics Console	NVIDIA Quadro2Pro

Mass Storage Drivers

External mass storage is not supported on the i2000 hp workstations.

New SCSI Drivers

For the rx9610 and rx4610 servers, two new SCSI HBA drivers, qlisp and c8xx, are released in HP-UX 11i Version 1.5 (B.11.20). The c8xx driver runs on all IPF systems and replaces the c720 HBA driver on PA systems. The qlisp HBA driver is a new driver to support the 12160 Qlogic SCSI controller.

The c8xx HBA driver supports both the A5150A, Dual Channel Ultra2 SCSI, and the A5159A, Dual Channel FWD SCSI-2, add-in PCI cards.

The qlisp HBA driver supports the core Qlogic SCSI controller.

There is no early dump support in HP-UX 11i Version 1.5 (B.11.20). Unlike PA systems, on IPF based systems, it is not possible to get a crash dump before the system is booted. For more information, see System Crash Dump Changes in Chapter 3, “HP-UX Features.”

Tachlite Driver

For the rx9610 and rx4610 servers, this release includes td, the Fibre Channel PCI Tachyon TL driver. This version of the driver supports Fibre Channel Arbitrated Loop (FC-AL) only. Fabric is not supported in this release.

The IPF platforms supported by this driver are:

- hp server rx9610
- hp server rx4610

For a complete list of supported platforms and devices as well as other up-to-date information, see the Fibre Channel web site at http://techsolutions.hp.com.

SCSI Devices

External mass storage is not supported on the i2000 hp workstations.

LSI LVD UltraII SCSI Controller

On Itanium-based servers, unlike PA-based systems, the LSI LVD UltraII SCSI controller, part number A5149A, can not boot from targets with SCSI ID greater than 7. These controllers have the Sub System ID Bits (SSID) set such that the card is forced into narrow mode.

Specifying Bootable Devices in the LSI BIOS

The LSI BIOS is used to determine which SCSI HBAs are searched for bootable devices. The LSI BIOS limits this selection to a maximum of four possible choices, determined by the PCI bus slot order. Any boot target must reside on one of those default controllers.

NOTE HP-UX will let you install to any device on the system but if it is not on one of the default controllers you can not boot from it.

QLogic SCSI Adapter Rates

For the rx9610 and rx4610 servers, some JBOD (Just a Bunch Of Disk) enclosures contain internal cabling for a slower SCSI rate than newer disk devices may accept. If an Ultra160 disk accepts the higher rate in an Ultra2 enclosure, a hang will occur in data transfers. To force the autonegotiation to use a slower rate, you must configure the SCSI adapter.

Setting external QLogic SCSI connector for JBODs:

Step 1. Enter **Alt-Q** when prompted on machine book

Step 2. Choose

```
Configuration Settings->Host Adapter Settings
Adapter Configuration:  Manual
```

Changing to `Manual` mode allows access to all the fields in the device settings screen.

Step 3. Choose

`Configuration Settings->SCSI Device Settings->Bus 1`

Press **TAB** to move to the left column (affects all devices in this model)

Jamaica BOX JBOD (Fast settings):

```
                    Sync Offset: 8
                    Sync Period: 25
```

SC10 Megatron (Ultra2 settings):

```
                    Sync Offset: 14
                    Sync Period: 10
```

Step 4. **`Configuration Settings->Host Adapter Settings`**
`Adapter Configuration:  AUTO`

Changing back to `AUTO` mode allows autonegotiation to occur but based on the new device settings. Leaving this at `Manual` may cause the card to complain with warnings as the settings are forced on the SCSI devices without negotiation.

Step 5. Save changes and reboot

Function Key Mappings

Between the time the system powers up and the EFI shell comes up or the kernel is loaded, Alt-key combinations are created by sending the combination ^[} followed by the key. The following key sequence definitions apply:

Table 2-3 **Non-Ascii Key Mappings**

Key	Normal	Shift	Ctrl	Alt
ESC	^[	NS[a]	NS	NS
F1	^[OP	NS	NS	NS
F2	^[OQ	NS	NS	NS
F3	^[OR	NS	NS	NS
F4	^[OS	NS	NS	NS
F5	^[OT	NS	NS	NS
F6	^[OU	NS	NS	NS
F7	^[OV	NS	NS	NS
F8	^[OW	NS	NS	NS
F9	^[OX	NS	NS	NS
F10	^[OY	NS	NS	NS
F11	^[OZ	NS	NS	NS
F12	^[O1	NS	NS	NS
Print Screen	NS	NS	NS	NS
Scroll Lock	NS	NS	NS	NS
Pause	NS	NS	NS	NS
Insert	^[[L	NS	NS	NS
Delete	(7Fh)[b]	NS	NS	NS

Table 2-3 **Non-Ascii Key Mappings**

Key	Normal	Shift	Ctrl	Alt
Home	^[[H	NS	NS	NS
End	^[[K	NS	NS	NS
Pg Up	^[[M	NS	NS	NS
Pg Down	^[[2J	NS	NS	NS
Up Arrow	^[[A	NS	NS	NS
Down Arrow	^[[B	NS	NS	NS
Right Arrow	^[[C	NS	NS	NS
Left Arrow	^[[D	NS	NS	NS
Tab	(09h)	NS	NS	NS

a. NS = Not supported
b. (xxh) = ASCII character xx

Once the EFI shell comes up, the following key sequence definitions apply:

Table 2-4 **Key Sequence Definitions (EFI shell)** [a]

Key	Sequence	Key	Sequence
F1	^[[OP	F2	^[[OQ
F3	^[[Ow	F4	^[[Ox
F5	^[[Ot	F6	^[[Ou
F7	^[[Oq	F8	^[[Or
F9	^[[Op	F10	^[[M

a. from Appendix B Simple input of the EFI spec

Machine Check

Machine check functionality on IPF-based systems is equivalent to PA-based systems, however, the names have changed.

Table 2-5

IPF-based Systems	PA-RISC-based Systems
Machine Check Abort (MCA)	High Priority Machine Check (HPMC)
Corrected Machine Check (CMC)	Low Priority Machine Check (LPMC)

3 HP-UX Features

This chapter describes other new and changed operating-system software functionality supported by the HP-UX 11i Version 1.5 (B.11.20) release including:

- HP-UX IPF Bootloader
- Changes to Machine Identification
- HP-UX Kernel Configuration Commands
- 32-bit DMA Pool - new kernel tunable `dma32_pool_size`
- Software Distributor (SD-UX) Changes
- SAM
- Modifications to /stand/system
- Changing /dev/* permissions
- Maintenance Mode Boot
- System Crash Dump Changes
- High Availability Features
- LSSERV Software Obsolescence

HP-UX IPF Bootloader

HP-UX 11i Version 1.5 (B.11.20) IPF bootloader allows you to locate and boot the desired HP-UX kernel. Because the HP-UX IPF bootloader can boot from the VxFS file system, it is not necessary for `/stand` to be an HFS file system.

When an Itanium-based system boots, you are placed in either the EFI (Extensible Firmware Interface) boot manager or the EFI shell. The HP-UX IPF bootloader is a native EFI application that can be launched from either application. To boot any machine with HP-UX, the HP-UX IPF bootloader must be placed in the EFI partition of the boot disk.

IPF Bootloader Compatibility

The HP-UX IPF bootloader is not compatible with the HP-UX PA bootloader. It can not boot a PA kernel and can not be used in a PA machine.

IPF Bootloader Documentation

For more information, see the *boot* (1M), *hpux* (1M), *isl* (1M) and *pdc* (1M) manual pages.

Changes to Machine Identification

For HP-UX 11i Version 1.5 (B.11.20), the *uname* (1) and *uname* (2) interfaces are replaced by new *confstr* (3C) interfaces as a means to obtain a unique machine identification number. The previous *uname* (1) and *uname* (2) interfaces are supported in HP-UX 11i Version 1.5 (B.11.20), with limitations, and will be obsoleted in a future release.

The model string returned by the *uname* (1), *uname* (2), *getconf* (1), *confstr* (3C), and *model* (1) interfaces on HP's Itanium-based platforms will not begin with `9000/`, as has been the case on previous PA based HP-UX systems. Instead, the model string is, or begins with, `ia64`.

The following command and function return the string `ia64` on all IPF platforms.

- `uname -m`
- *`machine`* field of the `utsname` structure returned by *uname* (2)

The `model` command, per its existing definition, returns the same output as `uname -m`, possibly with additional information. If available, the additional information can be used to distinguish between different systems. The command `getconf MACHINE_MODEL` and the library call `confstr(_CS_MACHINE_MODEL,...)` will return the same output as the `model` command.

Machine ID Compatibility

Applications running on HP-UX 11i Version 1.5 (B.11.20) that require ID numbers guaranteed to be unique across all platforms must be converted (a source code change) to use the new *confstr* (3C) or *getconf* (1) interfaces in place of the *uname* (2) and *uname* (1) interfaces. Applications that do not require completely unique ID numbers may continue to use the *uname* (2) and *uname* (1) interfaces on this release. However, those applications must change to use the new interfaces in a future release.

Applications which recognize HP platforms as a class or HP-UX itself by examining the model string value (e.g., testing for a leading `9000/`) will need to be changed.

Machine ID Documentation

For more information, see Chapter 6, “Commands and System Calls,” on page 79, Chapter 7, “Programming,” on page 93, and the *uname* (1), *getconf* (1), *model* (1), *uname* (2), and *confstr* (3C) manual pages.

HP-UX Kernel Configuration Commands

These are a set of system administration commands to configure, build and manage new kernels with all associated kernel components, and individual DLKM modules.

The public interface for kernel configuration:

- *config* (1M)
- *kmadmin* (1M)
- *kminstall* (1M)
- *kmmodreg* (1M)
- *kmpath* (1M) - new public interface
- *kmsystem* (1M)
- *kmtune* (1M)
- *kmupdate* (1M)
- *mk_kernel* (1M)
- *master* (4)
- *system* (4)

HP-UX Kernel Configuration (KC) commands run native on Itanium-based systems, including DLKM Phase 1 support. In addition, they have been enhanced to support the following new kernel features:

- New Boot Loader based on MFS (Memory File Storage)

  KC commands have been modified to set up the appropriate kernel-specific links for the boot loader.

- Boot-Time-Loading of Kernel Modules (BTL DLKM)

  KC commands are enhanced to support configuring and managing BTL modules. The module-specific configuration attributes `Supported Loading Phases` and `Configured Loading Phase` are added and supported throughout the KC commands.

- Kernel ID String

  config (1M) now generates a kernel ID string (KIS) each time a new kernel is linked. This ID is used to uniquely associate distributed kernel files on disk with a specific kernel image. KRS uses the generated KIS to identify its initialization files associated with a kernel at boot-up. DLKM uses the KIS to identify the location of the `symtab` file at boot-up, and to initialize the module load path associated with the running kernel.

New or Changed Kernel Configuration Command Options or Features

config (1M) and *mk_kernel* (1M)

- No change to the command line.
- Configuration of BTL modules is supported.
- *config* (1M) generates the Kernel ID string (KIS) to associate all kernel components to a specific main kernel file.
- *config* (1M) generates module registration information and kernel-specific persistent information in KRS format.
- The kernel component set goes beyond the main kernel file and the DLKM directory. The KRS file described above is generated.
- The DLKM directory is re-named `/stand/dlkm.KIS` (link to old `/stand/dlkm` is maintained for backward compatibility)

kmadmin (1M)

- New option (`-b`) is added to set the Configured Loading Phase of a registered module.
- Configured Loading Phase is now kept in KRS, and you can get/set it using `kmadmin` to one of the following phases:
  - `BOOT1`
  - `BOOT2`
  - `INIT`
  - `AUTO`

  The `/etc/loadmods` file is supported only for backward compatibility, and can only be used to set Configured Loading Phase to `INIT`. Users should migrate to `kmadmin` interface.

- Output information and format is enhanced to display new attributes:
  - module version (0.0.0 for traditional, 0.1.0 for modular)
  - Configured Loading Phase

kminstall (1M)

- New option (`-s`) is added to silence WARNING messages.

kmmodreg (1M)

- New option (`-a`) is added to schedule asynchronous unregistration of a kernel module (on next system shutdown).
- New option (`-n`) is added to un-schedule the asynchronous unregistration of a kernel module.

kmpath (1M)

- New administration command
- This command can be used to retrieve the name/path of the main kernel file of the currently running kernel, and to retrieve information about the kernel component set, like the KIS, and the DLKM directory.

kmsystem (1M)

- New option (`-b`) is added to set the Planned Loading Phase of a module prior to a kernel or module configuration.
- Output information and format is enhanced to display:
  - module version
  - Planned Loading Phase information

kmtune (1M)

- No change to the command line.
- Output information and format is enhanced to display module version information, and to be forward compatible with Dynamic Tunables feature included in HP-UX 11i and planned for later releases.

kmupdate (1M)

- A new option (-d) is added to remove a specified kernel and its associated component set.
- This command was enhanced to initiate the process of creating and populating the /stand/boot.KIS directory with all the links that are necessary for the MFS boot loader to be able to find all the necessary kernel components at boot time.

NOTE This implies that kmupdate is a *required* step in order to ensure the successful boot of a newly generated kernel.

master (4)

A new section is added to the modular master file (see *master* (4)) to describe the supported loading phases of a kernel module, which now includes:

BOOT1	Module supports loading at phase 1 of boot sequence
BOOT2	Module supports loading at phase 2 of boot sequence
RUN	Module supports loading at run time.

system (4)

A new section is added to the modular system file (see *system* (4)) to tailor the planned loading phase of a module, which includes:

BOOT1	Valid if BOOT1 is a supported loading phase
BOOT2	Valid if BOOT2 is a supported loading phase
AUTO	Auto-load or demand load. Valid if RUN is a supported loading phase
INIT	Load at init time. Valid if RUN is a supported loading phase

Kernel Configuration Documentation

The following new manual pages have been added:

- *kmpath* (1M)

- *system* (4) - this file format used to be embedded in the *config* (1M) manual page

The following existing manual pages have been changed:

- *config* (1M)
- *kmadmin* (1M)
- *kminstall* (1M)
- *kmmodreg* (1M)
- *kmsystem* (1M)
- *kmtune* (1M)
- *kmupdate* (1M)
- *master* (4)

White papers can be found at the docs.hp.com web site.

32-bit DMA Pool

To support I/O devices and drivers that are limited to 32-bit DMA addresses in HP-UX 11i Version 1.5 (B.11.20), a special area of reserved memory accessible by those devices may be allocated. This area is allocated only on systems where the highest physical memory address is greater than 4GB.

A new system tunable, dma32_pool_size, is used to specify the amount of memory to be allocated to support 32-bit I/O cards and devices. The default value of 256MB is expected to handle most configurations. For systems configured with a large number of I/O cards, a larger pool can be reserved, using these memory requirements for each supported 32-bit DMA card:

Table 3-1 32-bit DMA Card Memory Pool Size

32-bit DMA Card	Memory Pool Size
SCSI Adapters	33MB
LAN Adapters	1MB
FC - Tachlite	512KB
Internal IDE	256KB

Although dma32_pool_size is specified in bytes, the actual allocation is done in MB units. The specified size, if not an integral multiple of 1MB, will be rounded up to the next higher MB.

DMA Pool Effect on Performance

While this tunable allows support for I/O cards that do not use 64-bit addressing, the need to copy data to or from the 32-bit DMA pool incurs an obvious performance penalty. For those systems where this is unacceptable, the use of 64-bit cards exclusively is recommended.

Software Distributor (SD-UX) Changes

During `swinstall`, the packaged `machine_type` attribute is matched against output of the `model` command on the target. This matching is part of the compatibility checking done by `swinstall`. For more information see Changes to Machine Identification, and Changed model Command in these release notes, and the `machine_type` attribute in *sd* (4) and *swpackage* (4) manual pages.

HP-UX 11i Version 1.5 (B.11.20) does not support update. Any references to the `update-ux` tool in the *Software Distributor Administration Guide for HP-UX 11i* are not applicable to HP-UX 11i Version 1.5 (B.11.20).

SAM

Kernel parameters on IPF kernels are different from PA kernels. SAM will not display a description or proper help for these new parameters, but customers will be able to modify their values. These new parameters include:

Table 3-2 New Kernel Parameters in SAM

```
dma32_pool_size
dmp_blk_major
dmp_char_major
dmp_rootdev_is_vol
dmp_swapdev_is_vol
maxrsessiz
maxrsessiz_64bit
pa_maxssiz_32bit
pa_maxssiz_64bit
vol_blk_major
vol_char_major
vol_checkpt_default
vol_default_iodelay
vol_fmr_logsz
vol_max_bchain
vol_max_nconfigs
vol_max_nlogs
vol_max_nmpool_sz
vol_max_rdback_sz
vol_max_vol
vol_maxio
vol_maxioctl
vol_maxkiocount
vol_maxparallelio
vol_maxspecialio
vol_maxstablebufsize
vol_mvr_maxround
vol_rootdev_is_vol
vol_subdisk_num
vol_swapdev_is_vol
volcvm_cluster_size
voldrl_max_drtregs
voldrl_min_regionsz
voliomem_chunk_size
voliomem_maxpool_sz
voliot_errbuf_dflt
voliot_iobuf_default
voliot_iobuf_limit
voliot_iobuf_max
voliot_max_open
volraid_rsrtransmax
vols_blk_major
vols_char_major
vxtask_max_monitors
```

In this release, DPS is not supported by SAM or the underlying components of the operating system. You can use the Printers and Plotters area of SAM to manage printers or use HP-UX commands instead.

Dump device configuration is not supported in SAM on IPF. You can use other means from outside of SAM to configure dump devices.

SAM is not compiled natively on IPF. It is still a PA-RISC application and uses a PA emulator for operation. Because of the emulation mode, SAM's performance will be degraded from what customers have seen on previous HP-UX releases. An IPF-based native version will be available in a future release that will improve performance.

SAM Feature Obsolescence

The following features are deprecated or obsolete on HP-UX 11i Version 1.5 (B.11.20):

- `Backup and Recovery` section - deprecated

  `fbackup` and `frecover` are still be available from the command line. If Omniback is present on your system, it's functionality is available through SAM.

- `Distributed Print Service` section - obsolete

  HP Distributed Print Service is obsolete and not available on HP-UX 11i Version 1.5 (B.11.20) or later releases.

- `Instruments` section under the `Peripheral Devices` section - obsolete

  HP-IB instruments are not supported.

- `Kernel Configuration` section - deprecated
- `Run SAM on Remote Systems` section - deprecated

  Use *remsh* (1).

- `Process Management` section - deprecated

  Use the *ps* (1) and *kill* (1) commands.

- `Routine Tasks` section - deprecated

  Use the *shutdown* (1M), *find* (1), and *rm* (1) commands.

- The following sections under the `Performance Monitors` section - deprecated

  — `Disk and Terminal Activity` (*iostat* (1))

  — `Inter-Process Communication Facility Status` (*ipcs* (1))

  — `Process with the Highest CPU Usage` (*top* (1))

  — `System Activity` (*sar* (1M))

  — `Virtual Memory Activity` (*vmstat* (1))

Modifications to /stand/system

Beginning in HP-UX 11.0, direct editing of the`/stand/system` files for kernel configuration purposes (for example, to add or remove drivers, or modify kernel tunables) has been deprecated. In HP-UX 11i Version 1.5 (B.11.20), the `kmsystem` and `kmtune` commands are required in order to make changes to the `/stand/system` file. See the *kmsystem* (1M) and *kmtune* (1M) manual pages for more information.

CAUTION Editing the `/stand/system` file directly may result in the inability to generate a new kernel via the `mk_kernel` or `config` commands (see *mk_kernel* (1M) and *config* (1M)). Use the `kmsystem` and `kmtune` commands to edit the `/stand/system` file.

Changing /dev/* permissions

In the *Administering a System: Managing System Security* chapter of *Managing Systems and Workgroups*, the section Controlling Security on a Network - Link-Level Access states

> *To protect link-level access, make sure that the files /dev/ether*, /dev/ieee*, and /dev/lan* are owned and writable only by root.*

Changing permissions on these files and directories can cause unforeseen and potentially detrimental side-effects due to symbolic links.

CAUTION Check for links and consider the impact of changing permissions on these files and directories before making any changes to their permissions.

Maintenance Mode Boot

Maintenance Mode Boot is useful on systems where a standard boot has failed due to VxVM problems. This would only occur on a system where the boot disk is made up of VxVM volumes that contain the root, stand and possibly other file systems. On such a system, you must resolve the VxVM or related problem before being able to perform a standard boot once again.

Conditions Requiring VxVM Maintenance Mode Boot

Several situations will prevent the system from being booted in a standard way. This section provides descriptions of symptoms that accompany these situations and procedures that should be used to repair the system to once again allow normal operation.

- Missing LABEL file

  During system boot, the LABEL file is used to determine the offset and length of the root, stand, and swap/dump volumes before the VxVM configuration daemon may be started. A missing LIF LABEL file will prevent successful bootup of the system.

- Corrupt or Incorrect LABEL file

  A corrupted LABEL file or a LABEL file containing incorrect data for a VxVM boot disk will prevent successful boot up of the system. An incorrect LABEL file can occur, for example, if the `mkboot` command has been invoked improperly or without a subsequent invocation of the `vxbootsetup` command prior to the latest system reboot.

- Missing or corrupt `/etc/vx/volboot` file

  During system bootup, the VxVM configuration daemon is started. It makes use of the file `/etc/vx/volboot`. If for any reason that file is somehow missing or somehow corrupted, the configuration daemon will fail and abort the boot sequence.

- Booting to VxVM Maintenance Mode

  To boot the system to VxVM Maintenance Mode Boot use the `-vm` option to the boot command. This boots the system to single user mode without starting the VxVM configuration daemon.

- Missing or stale device files and missing or corrupt `/stand/ioconfig` file

  If the I/O hardware configuration is changed and any time later, the `ioconfig` file is removed or corrupted, the mapping between the I/O system and device files will be changed. As a result, some or all device files may no longer map to valid or correct I/O hardware paths. On a system where this situation exists, the VxVM configuration daemon (vxconfigd) will not be able to start up properly and will prevent normal system bootup.

Maintenance Mode Boot Documentation

For more information, see the *HP-UX 11i Version 1.5 Maintenance Mode Boot* white paper.

System Crash Dump Changes

The major changes to system crash dump functionality in HP-UX 11i Version 1.5 (B.11.20) are largely internal and do not affect the user-level interface:

- ❑ The boot loader passes user boot arguments through, unchanged, to the kernel, where they are parsed. Accordingly, the dump/crash path handles user boot arguments affecting dump device configuration, initializes, at boot-time, certain dump-related data structures, allows for user `config`(1M)-time and run-time configuration of dump devices and selection of memory to be dumped, and explicitly passes any information necessary for the taking of a crash dump to the dump application.
- ❑ Dump support is currently available for the Qlogic and LSI SCSI controllers only.
- ❑ There is currently no dump support for:
  - — IDE (you can't dump to the CD read/write device or the LS-120 floppy device).
  - — Fibre Channel
  - — dump-to-tape

System Crash Dump Documentation

For more information, see the *HP-UX 11i Version 1.5 System Crash Dump* white paper.

High Availability Features

High Availability (HA) features such as ServiceGuard, MetroCluster, SG-OPS edition, Continental Clusters and ServiceGuard Manager are not supported on HP-UX 11i Version 1.5 (B.11.20).

LSSERV Software Obsolescence

LicensePower/iFOR is not supported in HP-UX 11i Version 1.5 (B.11.20). The licensing product can be obtained directly from Isogon Corporation, the owner of the product.

If you are using this product, you are encouraged to visit the Isogon Corporation web site for further information about LSSERV support. See http://www.isogon.com for more information.

4 File and Disk Management

This chapter describes other new and changed operating-system software functionality supported by the HP-UX 11i Version 1.5 (B.11.20) release including:

- EFI File System Commands
- VERITAS Volume Manager
- LVM and MirrorDisk/UX

EFI File System Commands

The EFI (Extensible Firmware Interface) file system is new type of file system. It provides support for operating system loaders on Itanium-based systems. The HP-UX OS loader is located on the EFI file system.

An EFI partition of an IPF bootdisk uses the EFI file system to store the HP-UX OS loader and is required for booting an Itanium-based system.The EFI file system commands allow a user to create, display and access the EFI file system on a given device.

There are seven commands for accessing the EFI file system. They are:

`idisk` creates operating system partitions on a disk that will be used for IPF.

`efi_fsinit` creates new EFI file system on a specified partition.

`efi_mkdir` creates new directory on a specified EFI file system.

`efi_cp` copies a file between any supported HP-UX file system and a specified EFI file system.

`efi_ls` lists contents of a specified EFI file system.

`efi_rm` removes an existing file from a specified EFI file system.

`efi_rmdir` removes an existing directory from a specified EFI file system.

EFI File System Compatibility

EFI file system commands are available for both IPF and PA systems.

Change the time (using the `time` command) only in HP-UX, not in the EFI shell. Currently, changing the time in the EFI shell causes problems with time in HP-UX. As long as the time in HP-UX is correct, the time should be consistent. The side effect of this problem is that the time shown in the EFI shell (using the `time` command) may not represent the correct time.

EFI File System Documentation

Additional information about the EFI file system can be found in the following manual pages:

- *idisk* (1M)
- *efi_cp* (1M)
- *efi_fsinit* (1M)
- *efi_ls* (1M)
- *efi_mkdir* (1M)
- *efi_rm* (1M)
- *efi_rmdir* (1M)
- *efi* (4)

VERITAS Volume Manager

Base VERITAS Volume Manager 3.1 for HP-UX (VxVM), is a state-of-the-art online disk management software product. It is the default disk storage mechanism for HP-UX 11i Version 1.5 (B.11.20). VxVM is functionally equivalent to the HP Logical Volume Manager (LVM) and the HP MirrorDisk/UX products.

Base VERITAS Volume Manager 3.1 for HP-UX is bundled with all HP-UX 11i Version 1.5 (B.11.20) systems and includes:

- support for managing the root disk (i.e., rootability)

VERITAS Volume Manager 3.1 for HP-UX (B9116AA) is an optional product available at an extra charge. The add-on VxVM product includes features such as mirroring, RAID-5, and DMP for active/active devices. See *VERITAS Volume Manager 3.1 for HP-UX Release Notes* for more information about the add-on VxVM product.

VxVM Libraries

VxVM adds two new kernel libraries, `libvxvm.a` and `libvxdmp.a`, that get built into the base HP-UX kernel. VxVM also provides tunables defined in `/usr/conf/space.h.d/vxvm_space.h` and `/usr/conf/space.h.d/vxdmp_space.h`, these tunables can be found in `/usr/conf/master.d/vxvm` and `/usr/conf/master.d/vxdmp`. Detailed information of these tunables can be found in the *VERITAS Volume Manager 3.1 Administrator's Guide* on the http://docs.hp.com web site.

VxVM adds four new init scripts:

- `/sbin/rc0.d/K930vxvm-daemon-kill`
- `/sbin/rc1.d/S092vxvm-startup2` starts VxVM I/O daemons and enables the VxVM configuration daemon which rebuilds the `/dev/vx/dsk` and `/dev/vx/rdsk` directories. This script also rebuilds the user level nodes for all the DMP devices in `/dev/vx/dmp` and `/dev/vx/rdmp` that were detected by the kernel.
- `/sbin/rc1.d/S093vxvm-reconfig` VxVM reconfiguration init script
- `/sbin/rc2.d/S095vxvm-recover` handles recovery of stale volumes. Also starts the VxVM watch daemon, *vxrelocd* (1M), which sends email to the system administrator when any problems are found.

VxVM Compatibility

Coexistence with HP Logical Volume Manager (LVM)

The VERITAS Volume Manager for HP-UX coexists with HP Logical Volume Manager (LVM). With HP-UX 11i Version 1.5, LVM cannot be used to control the root/boot disk, so if a volume manager is used for the root disk, it must be VxVM.

Both LVM and VxVM utilities are aware of the other volume manager, and will not overwrite disks that are being managed by the other volume manager. The administrative utilities (SAM and vmsa) recognize and identify all disks on the system.

A conversion utility, *vxvmconvert* (1M), is available to convert LVM volume groups to VxVM disk groups. See the *VERITAS Volume Manager 3.1 Migration Guide* for more information about vxvmconvert and for a comparison of VxVM and LVM.

Storage Administrator Graphical User Interface and SAM

The VERITAS Volume Manager Storage Administrator (vmsa) provides a Java-based graphical user interface for managing VxVM. A command line interface is also available. The Storage Administrator has two parts: a server and a client. The server must run on the system running VxVM. The client can run on the server machine, or the client software can be installed on a different HP-UX 11i Version 1.5 (B.11.20) system to manage VxVM remotely. Note that only HP-UX 11i Version 1.5 clients are supported.

SAM, the HP-UX system administration manager, and Storage Administrator exist as independent entities. The Storage Administrator client can and should be launched from SAM. SAM is used to manage LVM objects and the Storage Administrator is used to manage VxVM objects. However, Storage Administrator recognizes and labels LVM volumes and disks, and similarly, SAM recognizes and labels VxVM volumes and disks. To manage VxVM disks graphically, you must use Storage Administrator. For information about Storage Administrator, see the *VERITAS Volume Manager 3.1 Storage Administrator Administrator's Guide.*

VxVM Documentation

The following VxVM documents are available in HP-UX 11i Version 1.5

(B.11.20) via the Instant Information CD, and on the http://docs.hp.com web site.

- *VERITAS Volume Manager 3.1 for HP-UX Release Notes*
- *VERITAS Volume Manager 3.1 Administrator's Guide*
- *VERITAS Volume Manager 3.1 Reference Guide*
- *VERITAS Volume Manager 3.1 Storage Administrator Administrator's Guide*
- *VERITAS Volume Manager 3.1 Migration Guide*

Manual Pages

The following new manual pages will be installed in `/usr/share/man`:

Table 4-1 New VxVM Manual Pages

autostart (1M)
dgcfgbackup (1M)
dgcfgdaemon (1M)
dgcfgrestore (1M)
pfto (7)
properties (4)
vgrestore (1M)
vmsa (1M)
vmsa_server (1M)
vol_pattern (4)
vx_emerg_start (1M)
vxassist (1M)
vxbootsetup (1M)
vxconfig (7)
vxconfigd (1M)
vxdiskadm (1M)
vxdisksetup (1M)
vxdmp (7)
vxdmpadm (1M)
vxedit (1M)
vxevac (1M)
vximportdg (1M)
vxinfo (1M)
vxinstall (1M)
vxintro (1M)
vxiod (7)
vxiod (1M)
vxmake (1M)
vxmake (4)
vxmend (1M)
vxprint (1M)
vxr5check (1M)
vxreattach (1M)
vxrecover (1M)
vxrelayout (1M)
vxrelocd (1M)
vxresize (1M)
vxrootmir (1M)
vxsd (1M)
vxsparecheck (1M)
vxstat (1M)
vxtask (1M)
vxtrace (1M)
vxunreloc (1M)
vxvmboot (1M)

Table 4-1 New VxVM Manual Pages

vxdctl (1M)

vxdg (1M)

vxdisk (1M)

vxdiskadd (1M)

vxmirror (1M)

vxnotify (1M)

vxpfto (1M)

vxplex (1M)

vxvmconvert (1M)

vxvol (1M)

VxVM Limitations in this Release

VERITAS Volume Manager 3.1 for HP-UX has the following limitations in HP-UX 11i Version 1.5, which will be removed in subsequent releases:

- The VERITAS FastResync option (FR) is not available in this release, although it is described in the *VERITAS Volume Manager 3.1 Administrator's Guide*.
- The VERITAS Cluster Volume Manager (CVM) is not available in this release, although it is described in the *VERITAS Volume Manager 3.1 Administrator's Guide*.
- VxVM does not support the HP Process Resource Manager (PRM). If you use PRM, you will not be able to gather statistics on disks managed by VxVM.
- A disk monitor integrated with the EMS framework is not yet available for disks being managed by VxVM.

VxVM Known Problems

Bad Disks Cause Long Delays in Initialization and Startup

- *Problem:* You may have a bad disk if you notice long periods of delay (several minutes) when doing any of the following:
  - — Installing the system.
  - — Starting vxconfigd at boot time:

    ```
    Starting VxVM restore daemon...
    VxVM starting in boot mode...
    ```

    The following message may appear:

    ```
    DIAGNOSTIC SYSTEM WARNING:
    ```

```
The diagnostic logging facility has started receiving
excessive
errors from the I/O subsystem.  I/O error entries will be
lost
until the cause of the excessive I/O logging is
corrected.
If the diaglogd daemon is not active, use the Daemon
Startup command
in stm to start it.  If the diaglogd daemon is active,
use the
logtool utility in stm to determine which I/O subsystem
is logging
excessive errors.
```

— Starting HP-UX

```
HP-UX Start-up in progress
__________________________
Configure system crash dumps
........................................ OK
VxVM device node check
.............................................. OK
VxVM general startup
................................................
```

NOTE Some delays are expected on machines with a large number of disks, even if no disks are bad.

- *Workaround:* Run `vxdisk list` after VxVM is up to see if you have a bad disk. In the following example, the error status of `c3t11d0` indicates it is a bad disk.

```
# vxdisk list
DEVICE       TYPE     DISK        GROUP       STATUS
c0t3d0       simple   disk01      rootdg      online
c0t4d0        simple   -           -            LVM
c0t5d0       simple   disk02      rootdg      online
c0t6d0        simple   -           -            LVM
c3t11d0       simple   -           -             error
c3t12d0       simple   -           -            LVM
c3t13d0       simple   -           -            LVM
c3t14d0       simple   -           -            LVM
c3t15d0      simple   -           -            LVM
```

VxVM DMP Lists Disabled Paths That Have Been Reused

- *Problem:* When one of multiple paths or cables to a disk array is disconnected, fails, or is swapped with another path, and then that same path or cable is reconnected or replaced, it is possible that HP-UX will recognize the recovered path as a new path, not as the same path that has simply recovered. In this case, DMP will list twice as many paths: the “new” ones in the `ENABLED` state and the “old” ones (that is, from before the paths were swapped, removed or replaced) in the `DISABLED` state. I/O continues to be routed correctly.
- *Workaround:* None necessary. VxVM DMP will not automatically clean up the paths that are no longer in use, or that are in the `DISABLED` state. When the host is rebooted, the DMP database will be rebuilt without the `DISABLED` path definitions.

Run vxdctl enable to Show Status Changes for LVM Disks

- *Problem:* VxVM output will not reflect status changes for LVM disks until *vxdctl* (1M) is run. For example, if you clear an LVM disk with *pvremove* (1M), `vxdiskadm` will still list the status of that disk as “LVM,” until you run `vxdctl enable`. This is also true for `vmsa` output and the output from other VxVM commands.
- *Workaround:* Run `vxdctl enable` after making any changes to LVM disks to update VxVM’s database.

VMSA Continues Running With No rootdg

- *Problem:* If `rootdg` is on an external device which must be shut down, then VxVM commands won’t run. However, VMSA continues to run, even though it cannot complete operations.
- *Workaround:* None.

Duplicate Device Name Creation in rootdg

- *Problem:* When you create new volumes in the `rootdg` disk group, two sets of device nodes are created: under both `/dev/vx/[r]dsk/` and `/dev/vx/[r]dsk/rootdg`. Although either path can be used for *mkfs* (1M) or *mount* (1M), the duplicate sets of device node names can be confusing.
- *Workaround:* We recommend using the full pathname to `rootdg` disk devices in command line arguments. This is consistent with the

naming of device nodes in other disk groups. For example, to mount a `rootdg` volume use:

```
# mount -F vxfs /dev/vx/dsk/rootdg/vol01 /vol01
```

Do not use `/dev/vx/dsk/vol01` as the pathname.

vmsa (1M) will always use the full pathname.

Storage Administrator Issues

The following issues have been identified as VERITAS Storage Administrator problems, and will be fixed in a future release of VxVM.

Splitter Cursor Problem

- *Problem:* The splitter cursor does not always go away. This can prevent the wait cursor from being displayed. This problem is caused by a Java bug.
- *Workaround:* Move the cursor outside the main window and then back into the main window.

Extraneous Scrollbar Warning

- *Problem:* When a dialog box, such as a disk/space allocation dialog box, is brought up, a message similar to the following can display:

  ```
  Warning:
  Name: scrollbar
  Class: XmScrollBar
  The scrollbar page increment is less than 1.
  ```

- *Workaround:* This message can be ignored.

VMSA Failure on a Sun Display

- *Problem:* When the VMSA client is run on an HP system but displayed on a Solaris system with CDE window manager, the VMSA client may fail with a segmentation violation.

  The VMSA client for HP displays properly on an HP console running the default HP CDE window manager.

- *Workaround:* Before running vmsa for HP-UX on a Sun display, type

  ```
  #xrdb -remove
  ```

Cannot Use Storage Administrator Across Firewall

- *Problem:* The Storage Administrator does not support communication between the client and server across a firewall.
- *Workaround:* None

Problem with Multiple Host Names

- *Problem:* The Storage Administrator can have trouble connecting to a host machine (server) if multiple host names are associated with a single IP address. The following message is displayed:

  ```
  Summary:
  There is no such server (host1)

  Detail:
  java.net.UnknownHostException: Unknown host:
  [host2:32839]; nested exception is:
  java.net.UnknownHostException:host2
  ```

  In this example, the administrator specified `host1` for the VMSA connection, but the server host machine was identified as `host2`.

  In some cases, this may be a problem with the way DNS is set up. A DNS reverse lookup (by IP address) may return a host name that differs from the host name provided to VMSA at startup (`host1`). You may have to make appropriate changes to DNS so that the names are consistent. In other cases, this happens because `host1` is not the first host in the list of hosts for the associated IP address in the `/etc/hosts` file.

- *Workaround:* Make sure `host1` shows up as the first host in the list of names for the address of `host1` in `/etc/hosts`.

Cannot Grow gen Volume

- *Problem:* VMSA does not grow a volume of use_type `gen`
- *Workaround:* None

X Window Errors at VMSA Startup

- *Problem:* When the Storage Administrator is started, the following X Window System error can occur:

  ```
  Xlib: connection to "hostname:0.0" refused by
  server

  Xlib: Client is not authorized to connect to Server
  ```

```
java.lang.InternalError: Can’t connect to X11
window
server using hostname:0.0’as the value of the
DISPLAY
variable.

at sun.awt.motif.MToolkit.<init>(MToolkit.java:48)

at
java.awt.Toolkit.getDefaultToolkit(Toolkit.java:24
4)
```

- *Workaround:* Type `xhost + hostname` to explicitly authorize the *hostname* client, or omit *hostname* to allow unlimited X server access.

LVM and MirrorDisk/UX

HP-UX 11i Version 1.5 (B.11.20) includes IPF functionality for basic Logical Volume Manager (LVM) features including non-boot disk support and full mirroring capability on non-boot disks in a stand-alone configuration. LVM boot disks are not supported in this release.

If you are familiar with LVM on previous releases of HP-UX you have access to the same features on IPF-based platforms as you had on PA-RISC platforms on earlier releases. In many cases, the inclusion of LVM will aid your migration from PA to IPF. Instead of being forced to start with clean disks or to migrate all of your data to the VERITAS Volume Manager (VxVM), before moving to an Itanium-based platform, you will be able to use your pre-built data along with the LVM commands that are already familiar.

Note that not all LVM functionality is provided on IPF systems for the HP-UX 11i Version 1.5 (B.11.20) release. High availability features such as Shared LVM (SLVM - for cluster configurations) and Business Copy/BCV/ snapshot features for high-end arrays are not supported. Likewise, support for LVM-formatted boot disks is not supported (including root, swap and dump to LVM disks) in this release.

Disk striping capability is provided with this version of LVM on Itanium-based systems and, when used appropriately, may significantly improve disk-access times. See *Managing Systems and Workgroups* for more information.

LVM Compatibility

LVM data developed on a PA-RISC system may be imported to be used on an IPF system (and vice-versa). Both PA-RISC and IPF platforms, however, carry the same constraint that LVM-formatted devices are not compatible with VxVM.

LVM Documentation

Boot-disk options are not available on IPF-based systems in HP-UX 11i Version 1.5 (B.11.20). See the *lvlnboot* (1M), *lvrmboot* (1M), and *pvcreate* (1M) for more information. Additional information about LVM can be found on the web at http://docs.hp.com.

5 Networking

This chapter describes new and changed networking functionality supported by the HP-UX 11i Version 1.5 (B.11.20) release including:

- Network Drivers
- EISA Interface Cards Are Not 64-bit Compatible
- Secure Internet Services Limitations
- Troubleshooting Commands
- /etc/netconfig File

See Chapter 2, “Workstation/Server Specific Information,” for information on supported hardware and network interface cards.

Network Drivers

New intl100 Driver

intl100 is a new PCI 10/100 Mbps full duplex ethernet driver based on the 82559 chipset. It allows the cpu and the interface card to work in parallel on the memory buffer descriptors and batches transmit commands if the incoming rate is more than the service rate. As a result, this driver offers:

- improved efficiency, compared to older drivers
- improved overall system performance caused by fewer interrupts
- increased network load handling

intl100 supports cards with an RJ45 connector, 100Base-TX card A6792A, and is similar to, btlan. Because the architecture of the 82559 chipset is different from the 21143 chipset, the hardware dependent routines are different.

The intl100 driver comes pre-installed for all applicable PCI 100Base-T cards.

EISA Interface Cards Are Not 64-bit Compatible

In HP-UX 11i Version 1.5 (B.11.20), 64-bit operation does not include support for EISA interface cards; however, they are supported on 32-bit operating systems. System configurations that include the following EISA interfaces cannot be configured to run on HP-UX 11i Version 1.5 (B.11.20). Please delete the EISA driver using `swremove`. This will prevent users from mistakenly thinking that the EISA driver can be used.

Table 5-1 Unsupported EISA Interface Cards

Product Number	Card Name
J2482A	8-port MUX
J2483A	64-port MUX
A2679A	Single-Ended SCSI-2 interface
A3658A	100BaseT adapter
J2165A/B	802.5 Token Ring
A3402A	Combined 10BaseT/100VG adapter
J2815A	Dual-port X.25
J2220A	SNA PlusLink adapter
J2794A	X.25/ACC, SNAPlus/ACC adapter
A3659A	FDDI (Single or Dual-Attach) adapter
25525B	Fast Diff. SCSI
25560A	HP-IB Interface
25567B	LAN/9000
4031A	Fibre Channel
A4308A/B	100BT LAN
B5502AA/BA	FDDI LAN
2159A	X.25 Link

Table 5-1 Unsupported EISA Interface Cards

Product Number	Card Name
52645AA	100 VG Any LAN
52802B	ATM
52730AA/BA	SNA Link

Secure Internet Services Limitations

Secure Internet Services (SIS) and the OSPF (Open Shortest Path First) agent for gated are not available in HP-UX 11i Version 1.5 (B.11.20). This includes SIS options in ftp, rlogin and related commands, and telnet associated with SIS (inetsvcs_sec), the OSPF agent, isofagt (SNMP MIB - Simple Network Management Protocol Management Information Base) binary, and the SNMP developer’s kit.

Troubleshooting Commands

On IPF-based systems, user-space core dump analysis can not be performed by running `arp`, `lanscan` or `netstat` commands. Unlike PA versions, these commands do not support the functionality of reading or analyzing a core file. For more information, see Changed arp, lanscan, and netstat Commands in Chapter 6.

/etc/netconfig File

The `/etc/netconfig` file is a network configuration database used to store information about networks connected to this system and provides information about which translation libraries should be used for the name-to-address translation services.

On IPF-based systems, `/etc/netconfig` points to libraries with the new extensions `.so.1`. The symbolic links are created in `/usr/lib`, where PA-based libraries are located.

On PA-based systems, `/etc/netconfig` is unchanged and points to libraries with PA extensions `.1`.

/etc/netconfig Compatibility

These symbolic links provide backward compatibility with all PA applications which are run through Aries on IA machines.

If you modify the `/etc/netconfig` file you must make sure to provide the appropriate links, as described in the comment section of the file itself or on the manual page.

If an absolute library pathname is specified for a particular service, the applications using the service will work only for that architecture.

/etc/netconfig Documentation

For more information, see the *netconfig* (4) manual page.

6 Commands and System Calls

This chapter describes other new and changed operating-system software functionality supported by the HP-UX 11i Version 1.5 (B.11.20) release including:

- Changed arp, lanscan, and netstat Commands
- Changed uname() function
- ftp and other Secure Internet Services commands
- Enhanced getconf Command
- Changed locale and localedef Commands
- Changed model Command
- Changed mpsched Command
- Changed uname Command
- Obsolete and Deprecated Commands

Changed arp, lanscan, and netstat Commands

The option of specifying a core filename on the command line is not available on IPF-based systems. This affects the commands:

`arp`	display and control address resolution
`lanscan`	display LAN device configurations and status
`netstat`	display network status

If you specify a core file, an error message will be displayed. For IPF-based systems, the default and only acceptable value for the core command line option is `/dev/kmem`. `/dev/kmem` provides access to the currently running system. The tool of choice for user-space core dump analysis is now `q4`.

arp, lanscan, and netstat Compatibility

Any scripts using these commands for core dump analysis should be changed to use either `/dev/kmem` for current configuration or q4 for core dump analysis.

arp, lanscan, and netstat Documentation

For more information, see *arp* (1M), *lanscan* (1M), and *netstat* (1) manual pages. For more information on `q4`, see http://devresource.hp.com.

Changed uname() function

In HP-UX 11i Version 1.5 (B.11.20), the value of the `idnumber` field in the `utsname` structure is not guaranteed to be unique. This field will be obsoleted in a future release. Use the *confstr* (3C) interfaces instead.

For more information, see "Changes to Machine Identification" on page 39, "Changes to confstr()" on page 100, and the *uname* (2) manual page.

ftp and other Secure Internet Services commands

Secure Internet Services (SIS) is not available in HP-UX 11i Version 1.5 (B.11.20). Therefore, SIS options in the following commands are not available:

- ftp
- gated
- rlogin
- remsh
- rcp
- telnet

See the manual pages for the specific SIS related options.

Enhanced getconf Command

The `getconf` command has been enhanced to support three additional parameters.

`MACHINE_IDENT`

ID number for each physical machine. Returned as an opaque string of printable ascii characters not to exceed 64 bytes including the string terminator. This string has the same value for all partitions in a physical machine. For hardware classes first released with HP-UX 11i Version 1.5 (B.11.20) or later this ID number is unique across all hardware classes. For earlier hardware classes the ID number is unique only within the hardware class. A null string is returned if no ID number is available (expected to be the case only for prototype machines or others improperly configured in manufacturing).

`PARTITION_IDENT`

ID number for each partition existing on a machine. Returned as an opaque string of printable ascii characters not to exceed 64 bytes including the string terminator. For any machine not supporting partitions this value will be the same as `MACHINE_IDENT`.

`MACHINE_SERIAL`

machine serial number as found labeled on the external machine chassis. The value will be a printable ascii string not to exceed 64 bytes including the string terminator. This string is not available on all classes of machines; if unavailable, the string will be empty. This string is not a unique identifier of the machine, since machines of different classes can have the same serial number. If a unique identifier is needed, use `MACHINE_IDENT` or `PARTITION_IDENT`.

For more information, see “Changes to Machine Identification” on page 39 and the *getconf* (1) manual page.

Changed locale and localedef Commands

To enable smooth migration from PA to IPF, both PA and IPF locale libraries and iconv methods need to exist on an Itanium-based system.

Directory Structure Changes

The current directory structure on 64-bit PA systems is:

```
/usr/lib/nls/loc/                    # PA32 locale directory
/usr/lib/nls/loc/locales.1           # PA32 versioned 10.20 locale libraries
/usr/lib/nls/loc/locales.2           # PA32 versioned 11.10 locale libraries
/usr/lib/nls/loc/locales             # Link to /usr/lib/nls/loc/locales.2
/usr/lib/nls/loc/methods.1           # PA32 versioned 10.20 locale methods
/usr/lib/nls/loc/methods.2           # PA32 versioned 11.11 locale methods
/usr/lib/nls/loc/methods             # Link to /usr/lib/nls/loc/methods.2
/usr/lib/nls/iconv/methods.1         # PA32 versioned 10.20 iconv method
                                     # libraries
/usr/lib/nls/iconv/methods.2         # PA32 versioned 11.11 iconv method
                                     # libraries
/usr/lib/nls/iconv/methods           # Link to /usr/lib/nls/iconv/methods.2

/usr/lib/nls/loc/pa20_64/            # PA64 locale directory
/usr/lib/nls/loc/pa20_64/locales.2   # PA64 locale libraries
/usr/lib/nls/loc/pa20_64/locales     # Link to
                                     # /usr/lib/nls/loc/pa20_64/locales.2
/usr/lib/nls/loc/pa20_64/methods.2   # PA64 locale methods
/usr/lib/nls/loc/pa20_64/methods     # Link to
                                     # /usr/lib/nls/loc/pa20_64/methods.2
/usr/lib/nls/iconv/pa20_64/methods.2 # PA64 iconv method libraries
/usr/lib/nls/iconv/pa20_64/methods   # Link to
                                     # /usr/lib/nls/iconv/pa20_64/methods.2
```

New directories have been created for 32-bit and 64-bit IPF locale libraries, which co-exist with PA libraries. The new directories are:

```
/usr/lib/nls/loc/hpux32/
/usr/lib/nls/loc/hpux32/locales.1    # IA-64 native 32 bit locales
/usr/lib/nls/loc/hpux32/locales      # Link to /usr/lib/nls/loc/hpux32/locales.1
/usr/lib/nls/loc/hpux32/methods.1    # IA-64 native 32 bit methods
/usr/lib/nls/loc/hpux32/methods      # Link to /usr/lib/nls/loc/hpux32/methods.1
/usr/lib/nls/iconv/hpux32/methods.1  # IA-64 native 32 bit iconv libraries
/usr/lib/nls/iconv/hpux32/methods    # Link to
```

```
                                         # /usr/lib/nls/iconv/hpux32/methods.1

/usr/lib/nls/loc/hpux64/
/usr/lib/nls/loc/hpux64/locales.1        # IA-64 native 64 bit locales
/usr/lib/nls/loc/hpux64/locales          # Link to /usr/lib/nls/loc/hpux64/locales.1
/usr/lib/nls/loc/hpux64/methods.1        # IA-64 native 64 bit methods
/usr/lib/nls/loc/hpux64/methods          # Link to /usr/lib/nls/loc/hpux64/methods.1
/usr/lib/nls/iconv/hpux64/methods.1      # IA-64 native 64 bit iconv libraries
/usr/lib/nls/iconv/hpux64/methods        # Link to
                                         # /usr/lib/nls/iconv/hpux64/methods.1
```

Changes to locale Command

Locales are listed according to the underlying processor type if the `-a` option of the `locale` command is specified. The `-a` option is used as follows:

`-a`	list 32-bit IPF locales
`-a 32`	list 32-bit IPF locales
`-a 64`	list 64-bit IPF locales
`-pa32`	list 32-bit PA locale libraries in `/usr/lib/nls/loc/locales.2`
`-pa64`	list 64-bit IPF locale libraries in `/usr/lib/nls/loc/pa20_64/locales.2`

A new option, `-A` lists all the locale libraries irrespective of the flavors and versions. This list includes locales in the directories

```
/usr/lib/nls/loc/locales.2
/usr/lib/nls/loc/pa20_64/locales.2
/usr/lib/nls/loc/hpux32/locales.1
/usr/lib/nls/loc/hpux64/locales.1
```

Changes to localedef Command

Beginning in 11.0, `localedef` builds both 32-bit and 64-bit locale libraries on 64-bit systems, and only 32-bit libraries on 32-bit systems. On Itanium-based systems, both 32-bit and 64-bit libraries will be built. The libraries are installed as described in Directory Structure Changes.

There are slight differences in method files due to the different paths for methods on PA and IPF systems.

Method file format for PA systems:

```
# method file for univ.utf8 locale
METHODS
mblen      "__mblen_utf8"    "/usr/lib/nls/loc/methods.2" "utf8"
mbtowc     "__mbtowc_utf8"   "/usr/lib/nls/loc/methods.2" "utf8"
mbstowcs   "__mbstowcs_utf8" "/usr/lib/nls/loc/methods.2" "utf8"
wctomb     "__wctomb_utf8"   "/usr/lib/nls/loc/methods.2" "utf8"
wcwidth    "__wcwidth_utf8"  "/usr/lib/nls/loc/methods.2" "utf8"
wcswidth   "__wcswidth_utf8" "/usr/lib/nls/loc/methods.2" "utf8"
wcstombs   "__wcstombs_utf8" "/usr/lib/nls/loc/methods.2" "utf8"
__mbtopc   "__mbtopc_utf8"   "/usr/lib/nls/loc/methods.2" "utf8"
__pctomb   "__pctomb_utf8"   "/usr/lib/nls/loc/methods.2" "utf8"
__mbstopcs "__mbstopcs_utf8" "/usr/lib/nls/loc/methods.2" "utf8"
__pcstombs "__pcstombs_utf8" "/usr/lib/nls/loc/methods.2" "utf8"
END METHODS
```

Method file format for IPF systems:

```
# method file for univ.utf8 locale
METHODS
mblen      "__mblen_utf8"    "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
mbtowc     "__mbtowc_utf8"   "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
mbstowcs   "__mbstowcs_utf8" "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
wctomb     "__wctomb_utf8"   "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
wcwidth    "__wcwidth_utf8"  "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
wcswidth   "__wcswidth_utf8" "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
wcstombs   "__wcstombs_utf8" "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
__mbtopc   "__mbtopc_utf8"   "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
__pctomb   "__pctomb_utf8"   "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
__mbstopcs "__mbstopcs_utf8" "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
__pcstombs "__pcstombs_utf8" "/usr/lib/nls/loc/hpux32/methods.1" "utf8"
END METHODS
```

locale and localedef Compatibility

The default behavior of the `locale` command on IPF systems lists IA locales instead of PA locales. PA applications that list PA locales will see different results on IPF and PA systems.

The `localedef` command will not be able to generate PA locales on an Itanium-based system. If you need PA locales, build them on a PA system and move them to the Itanium-based system.

Changed model Command

In HP-UX 11i Version 1.5 (B.11.20), the `model` command will return a model string that begins with `ia64`.

For more information, see “Changes to Machine Identification” on page 39 and the *model* (1) manual page.

Changed mpsched Command

The `-g` option of the `mpsched` command cannot be used with any other option. The usage message returned by `mpsched` and the manual page *mpsched* (1) have been changed to reflect the same.

Changed uname Command

In HP-UX 11i Version 1.5 (B.11.20), the value from the `-i` option is not guaranteed to be unique. This option will be obsoleted in a future release.

For more information, see “Changes to Machine Identification” on page 39 and the *uname* (1) manual page.

Obsolete and Deprecated Commands

Many commands marked obsolescent in previous releases of HP-UX are no longer available in HP-UX 11i Version 1.5 (B.11.20). Additionally, several commands are obsolescent in HP-UX 11i Version 1.5 (B.11.20) and will not be available in a later release of HP-UX. For obsolete and deprecated libraries, see libHcurses and Related APIs.

Obsoleted Commands

The following commands were previously obsoleted and are not available in HP-UX 11i Version 1.5 (B.11.20):

- `bfs`

  Use `ed`.

- `mkpdf`, `pdfck`, `pdfdiff`

  Use `sd`.

- `revck`

  Use `what`.

- `/usr/old/bin/sh`

  Use a different shell.

- `cue`, `cuegetty`, `cuesession`

  Not supported.

Deprecated Commands

The uucp subsystem and the dos commands are deprecated as of HP-UX 11i Version 1.5 (B.11.20). These commands are still available, but will be removed in a future release of the operating system. Relevant manual pages have been updated to warn against their use. Applications and scripts that use these commands should be changed to use appropriate alternatives.

Deprecated commands previously located in `/usr/bin` have been moved to `/usr/old/usr/bin` and linked to `/usr/bin`. Deprecated commands not located in /usr/bin have not been moved.

Table 6-1 **Commands Deprecated in HP-UX 11i Version 1.5**

Subsystem or Function	Commands	Notes
uucp commands	mkuupath, uucp, uulog, uuls, uuname, uupath, uupick, uusnap, uusnaps, uustat, uuto, Uutry, uux	Moved to /usr/old/usr/bin and linked to /usr/bin. Alternatives such as ftp and telnet should be used.
	clrsvc, getx25, opx25, uucheck, uucico, uuclean, uucleanup, uucpd, uudemon.admin, uudemon.cleanu, uudemon.hour, uudemon.poll, uugetty, uusched, uusched, uusub, uuxqt, clean_uucp, dialit	Not moved. Alternatives such as ftp and telnet should be used.
dos commands	doschmod, doscp, dosdf, dosll, dosls, dosmkdir, dosrm, dosrmdir	Moved to /usr/old/usr/bin and linked to /usr/bin.

7 Programming

This chapter describes new and changed programming libraries and related tools functionality supported by the HP-UX 11i Version 1.5 (B.11.20) release including:

- aC++ Libraries
- Changes to libc
- libcres.a Library
- New libdcekt.sl and libdcecpkt.sl Libraries
- New Linux Compatible API Library
- New locale Libraries
- New Unwind Library
- X11 Libraries
- SNMP
- Machine Dependent Headers
- Threadsafe Documentation
- Obsolescence and Deprecation of APIs
- Linker Toolset
- gprof Multiple Shared Library Profiling
- CXperf Performance Monitoring Support
- HP Caliper
- WDB Debugger Tool
- ADB Debugger Tool
- Fortran
- Packaging for Specific Releases

aC++ Libraries

The HP-UX 11i Version 1.5 (B.11.20) aC++ libraries have been enhanced:

- A new C++ Support Library, libstd_v2, supports the ANSI, ISO/IEC-14882 standard. The old library remains available under its current name, libstd.
- The obsolete Standard Components Library is not part of the IPF version of aC++. Most of its functionality is still available in the existing libraries.

New C++ Support Library

A new library, libstd_v2, contains the C++ Standard Library as defined in the ANSI, ISO/IEC-14882 standard. In particular, this library implements the "std" namespace, simplifying porting applications which use the "std" namespace from other platforms. The Itanium-based support library is source-level compatible with the PA support library.

This library is based on the most recent version of the RogueWave C++ Standard Library, which makes it easier for customers to use recent versions of other RogueWave libraries.

The new library coexists with an "old style" C++ library which does not implement the "std" namespace. The two libraries are binary incompatible. Applications can be built against the new or old library, but not both at the same time.

Removal of the Standard Components Library

HP-UX 11i Version 1.5 (B.11.20) does not include the obsolete Standard Components Library that was originally part of the Cfront C++ runtime. The archive libraries `lib++.a`, `libGA.a`, and `libGraph.a` no longer exist.

Most of the functions in this library are in other libraries provided with the aC++ compiler (libstd or libstd_v2 and librwtools.) Applications that use the Standard Components should be changed to use the current runtime libraries.

Changes to libc

libc.a Obsolesence

libc.a is not available for native IPF development. Applications which explicitly link to libc.a will fail. This change affects makefile compatibility, makefiles which explicitly link to libc.a using

- +A (aCC)
- -aarchive (ld)
- -Wl,-aarchive (compilers)
- -noshared (ld, compilers)
- -l:libc.a (ld)
- -Wl,-l:libc.a (compilers)

should be modified.

Libc Floating-Point Conversion

The libc routines that convert between decimal strings representing floating-point numbers, and internal binary floating-point formats were changed for HP-UX 11i Version 1.5 (B.11.20).

When compiled with the -fpwidetypes option, the HP-UX 11i Version 1.5 (B.11.20) <stdlib.h> header defines two new types:

```
typedef __float128 quad;
typedef __float80 extended;
```

quad is a synonym for long double; it is the 128-bit IEEE-754 floating type supported on PA systems. extended is the 80-bit IEEE-754 double-extended type in the Itanium architecture. The definitions, which also appear in the <math.h>, <float.h>, and <complex.h> headers, are protected by __ia64 and _INCLUDE_HPUX_SOURCE switches. To use these type names for other purposes, you should undefine them (with #undef) after inclusion of the above headers.

The HP-UX 11i Version 1.5 (B.11.20) <stdlib.h> header, under the ia64 and _PROTOTYPE switches, defines the long_double type to be the true "long double" type supported by the compiler. In previous releases, long_double was defined to be a struct. A program which requires the struct definition can be compiled with the -Ac option (the default prior to HP-UX 11.0), or, alternatively, with -D_LONG_DOUBLE_STRUCT added to the compile command.

The HP-UX 11i Version 1.5 (B.11.20) <stdlib.h> header, under the __ia64 and _INCLUDE_HPUX_SOURCE switches, declares the new function:

```
extern float strtof(const char *, char **);
```

and, when compiled with the -fpwidetypes options, also declares these new functions:

```
extern __float128 strtoq(const char *, char **);
extern __float80 strtow(const char *, char **);
extern char *_extecvt(__float80, int, int *, int *);
extern char *_extfcvt(__float80, int, int *, int *);
extern int _extecvt_r(__float80, int, int *, int *, char *,
    int);
extern int _extfcvt_r(__float80, int, int *, int *, char *,
    int);
```

all of which are defined in libc.

- strtoq() is equivalent to strtold()
- strtof() and strtow() are float and extended (80-bit floating-point) versions, respectively, of strtod()
- _extecvt(), _extfcvt(), and _extgcvt() are extended versions of ecvt(), fcvt(), and gcvt() respectively
- _extecvt_r() and _extfcvt_r() are extended versions of the reentrant ecvt_r() and fcvt_r() respectively

The routines which convert between decimal strings representing floating-point numbers and internal binary floating-point formats are correctly rounded according to the specification in ISO/IEC C99, to 36 significant decimal digits. (36 is the smallest number of digits that will distinguish all the numeric values in the 128-bit long double type.) Corresponding HP-UX 11i routines do not round correctly to the last bit or digit in all cases. The HP-UX 11i Version 1.5 (B.11.20) binary-to-decimal conversion (used by printf) produces zeros after the 36th significant digit, for all floating-point precisions; this differs from the HP-UX 11i conversion which produces zeros after the 17th significant digit in double and after the 33rd significant digit in long double.

The long double strtold function for overflow cases returns the value

HUGE_VALL (defined to be +infinity in <math.h>), with the appropriate sign. This follows the ISO/IEC C99 specification, and is analogous to strtod(), which returns HUGE_VAL (also equal to +infinity), with the appropriate sign. On HP-UX 11.0 and 11i, strtold() for overflow cases returns _LDBL_MAX or _LDBL_NMAX (defined in <values.h>), depending on the sign of the value; these are the largest-magnitude finite long double values.

All the HP-UX 11i Version 1.5 (B.11.20) *cvt functions (double, extended, and long double) treat NaNs and infinities the same way, and that way is consistent with the HP-UX 11i double [efg]cvt functions: they return ++ and -- for +infinity and -infinity, and they return [-]? for NaNs. On HP-UX 11i, the long double _ld[efg]cvt functions return +INFINITY and -INFINITY for +infinity and -infinity, and they return [-]NaN for NaNs.

New Environment Variables and Defaults for malloc()

In earlier versions of libc on PA, libc used a single lock in the malloc routines to make them thread-safe. In a multi-threaded application, there could be contention on this single lock if multiple threads are calling malloc and free at the same time. On HP-UX 11i Version 1.5 (B.11.20), libc provides multiple arenas, where malloc can allocate space from, and a lock for each arena. Threads are distributed among the arenas. Two new environment variables are introduced:

```
_M_ARENA_OPTS
_M_SBA_OPTS
```

_M_ARENA_OPTS can be used to tune the number of arenas and the arena expansion factor for threaded applications. In general, the more threads in an application, the more arenas should be used for better performance. Expansion factors control the number of pages to expand each time and assumes the page size is 4096 bytes. The number of arenas can be from 1 to 64 for threaded applications. (For non-threaded applications, only one arena is used regardless of whether this environment variable is set or not. However, you still can use this environment variable to change the expansion factor for non-threaded applications). If the environment variable is not set, or the number of arenas is set to be out of the range, the default number of 8 is used. The expansion factor is from 1 to 4096; the default value is 32. Again, if the factor is out of the range, the default value will be used. For example:

```
$ export _M_ARENA_OPTS=8:32
```

where the number of arenas is 8, and the expansion size is 32*4096 bytes. In general, the more arenas you use, the smaller the expansion factor should be, and vice versa.

Beginning in HP-UX 11i Version 1.5 (B.11.20), for Itanium-based systems only, the small block allocator is active by default with the following configuration: *__maxfast* = 512, *num_smallblocks* = 100, *grain* = 16. The default for *num_smallblocks* is unchanged from PA systems and the SVID 3 default. The default for *grain* is the default alignment for Itanium-based systems (16), which is the smallest *grain* allowable given that small blocks must have at least the default alignment. This requirement in turn comes from the standards requirement that `malloc()` must always return an address suitably aligned for any basic data type, and Itanium-based systems support 16-byte basic data types.

The new default for *__maxfast* is 512, which is changed from 0 on PA systems. This means that the SBA is now active by default, rather than inactive by default, as was and remains the case for PA systems.

Three exported integer variables have been added for IPF:

```
int __hp_malloc_maxfast;
int __hp_malloc_grain;
int __hp_malloc_num_smallblocks;
```

These three variables may be set by applications to override the SBA defaults; e.g. if an application links to a module that defines

```
int __hp_malloc_maxfast = -1;
```

then the value `-1` will be seen for *__hp_malloc_maxfast* when the function `__get_sba_global()` is called. The `-1` value for *__hp_malloc_maxfast* means that the SBA should be turned off, i.e. *__maxfast* will be set to 0. A positive value for *__hp_malloc_maxfast* will replace the default value of *__maxfast* after rounding to a multiple of the *grain*. Positive values for the variables *__hp_malloc_grain* and *__hp_malloc_num_smallblocks* will be used to replace the *grain* and *num_smallblocks* values if the resulting values would be legal.

These global variables override the environment variable _M_SBA_OPTS. It is presumed that any ISV that builds an application so that these variables are set wants that tuning and no other. However, subsequent `mallopt()` calls will revise the values for SBA configuration if the SBA has not yet been allocated. (In the same way that `mallopt()` calls will revise the SBA configuration even when the environment variable _M_SBA_OPTS has been set if the SBA has not yet been allocated at the time of the `mallopt()` calls.)

`_M_SBA_OPTS` turns on the small block allocator, and sets up parameters for the small block allocator, namely, *maxfast*, *grain*, *num_smallblocks*. Refer to `mallopt()` for details about the small block allocator, and its parameters. Applications with a small block allocator turned on usually run faster than with it turned off.

A small block allocator can be turned on through `mallopt()`; however, it is not early enough for C++/Java applications. The environment variable turns it on before the application starts.

`mallopt()` call can still be used the same way. If the environment variable is set, and no small block allocator has been used, the subsequent `mallopt()` calls can still overwrite whatever is set through `_M_SBA_OPTS`. If the environment variable is set, and a small block allocator has been used, then `mallopt()` will have no effect. For example:

```
$ export _M_SBA_OPTS=512:100:16
```

where the *maxfast* size is 512, the number of small blocks is 100, and the *grain* size is 16. You must supply all 3 values, and in that order. If not, the default ones will be used instead.

The `_M_ARENA_OPTS` and `_M_SBA_OPTS` environment variables have the following impact:

- Performance is improved for multi-threaded applications.
- Threaded applications may experience increased heap storage usage but you can adjust the heap usage through `_M_ARENA_OPTS`.

NOTE Threaded applications which are linked with archive `libc` and other shared libraries where those shared libraries have dependencies on shared `libc` may break.

New malloc family API

The new API, `memalign()`, has been added to the malloc family. See the *memalign* (3C) manual page for details.

Changes to confstr()

HP-UX 11i Version 1.5 (B.11.20) includes an enhanced version of confstr() that defines three new parameter values.

_CS_MACHINE_IDENT

ID number for each physical machine. Returned as an opaque string of printable ascii characters not to exceed 64 bytes including the string terminator. This string has the same value for all partitions in a physical machine. For hardware classes first released with HP-UX 11i Version 1.5 (B.11.20) or later this ID number is unique across all hardware classes. For earlier hardware classes the ID number is unique only within the hardware class. A null string is returned if no ID number is available (expected to be the case only for prototype machines or others improperly configured in manufacturing).

_CS_PARTITION_IDENT

ID number for each partition existing on a machine. Returned as an opaque string of printable ascii characters not to exceed 64 bytes including the string terminator. For any machine not supporting partitions this value will be the same as _CS_MACHINE_IDENT.

_CS_MACHINE_SERIAL

machine serial number as found labeled on the external machine chassis. The value will be a printable ascii string not to exceed 64 bytes including the string terminator. This string is not available on all classes of machines; if unavailable, the string will be empty. This string is not a unique identifier of the machine, since machines of different classes can have the same serial number. If a unique identifier is needed, use _CS_MACHINE_IDENT or _CS_PARTITION_IDENT.

The preferred method of calling these functions is defined in the *confstr* (3C) manpage as:

```
bufsize=confstr(_CS_MACHINE_IDENT,NULL,(size_t)0);
buffer=(char *)malloc(bufsize+1);
confstr(_CS_MACHINE_IDENT,buffer,bufsize+1);
```

The first line will return the length of the string to be returned, allocate memory based on this value, then call `confstr()` again to get the actual value.

For more information, see Changes to Machine Identification in HP-UX Features and the *confstr* (3C) manual page.

Locale Names

HP-UX 9.x locale names are no longer supported.

Linux compatibility

The linux compatibility interfaces, `getresgid()` and `getresuid()` have been added to libc. These functions return the real, effective, and saved gids and uids, respectively. See the *getresgid* (3C) and *getresuid* (3C) manual pages for more details.

libc Documentation

For more information about these changes, see

- *libc.a - not available on IPF* for makefile incompatibility at http://devresource.hp.com/STK/impacts/i782.html, http://devresource.hp.com/STK/class_ipf_list.html, and Obsolescence and Deprecation of APIs.
- *The Libm Library and Floating-Point Arithmetic for HP-UX on Itanium* at http://devresource.hp.com/STK/class_ipf_list.html
- the *memalign* (3C) manual page
- Changes to Machine Identification in HP-UX Features and the *confstr* (3C) manual page.

libcres.a Library

The `libcres.a` library is an archive of functions (select functions from `libc`) that are leaf functions that do not use shared data. The intent of this library is to provide performance critical applications with the ability to call certain performance critical functions more efficiently than would be possible if those functions were in a shared library. This library is useful because, from the HP-UX 11i Version 1.5 (B.11.20) release onwards, `libc` is no longer available in archive form.

The libcres usage model has changed significantly, it does not need to be linked with the application explicitly. The `libcres.a` linkage is handled automatically by the tool chain.

The functions in `libcres.a` may change in future releases.

libcres.a Documentation

For more information, see the *libcres.a* (5) manual page.

New libdcekt.sl and libdcecpkt.sl Libraries

In HP-UX 11i Version 1.5 (B.11.20), the `libdcekst.sl` and `libdcecpkt.sl` libraries support DCE applications which use kernel level threads implementation. The `libdcekt.sl` library supports DCE applications which use POSIX Thread Specification P1003.1c. Native IPF `libdcekt.sl` exists in both 32 and 64 bit versions, located in `/usr/lib/hpux32` and `/usr/lib/hpux64`, respectively.

For more information, see Distributed Computing Environment (DCE) in Chapter 8 of these release notes and the *DCE Release Notes* available at `/opt/dce/newconfig/RelNotes`.

New Linux Compatible API Library

HP-UX 11i Version 1.5 (B.11.20) includes the new `libhplx` library. `libhplx` contains the linux C library (`glibc`) APIs that are not available on HP-UX.

This library was previously distributed as part of the Linux Porting Toolkit CD. By making it available in HP-UX 11i Version 1.5 (B.11.20), applications can link dynamically to this library, simplifying porting and packaging Linux applications. The library also includes some APIs expected to be added to Unix standards in the future.

The `libhplx` library takes 1.1 MB disk space in the `/usr/local directory`.

Linux Compatible API Library Documentation

For more information, see the *Linux to HP-UX Porting Guide* at http://devresource.hp.com/LPK

New locale Libraries

New locale libraries and iconv methods are provided on Itanium-based systems to allow applications to move smoothly from PA to IPF. See Directory Structure Changes in Chapter 6 for more information.

New Unwind Library

HP-UX 11i Version 1.5 (B.11.20) includes a procedure call stack unwind support library, `libunwind` (also accessible by linking to `libcl`). With the new library, you need not link to a large library which contains unrelated support such as Pascal Try/Recover, Fortran IO, and Packed Decimal support to use the unwind functions. Changes to the library include:

- interface improvements to encapsulate data structures which are sensitive to changes in the architecture
- removal of COBOL Packed Decimal and Pascal language support features
- separation from Fortran IO support features (Fortran IO support will be delivered in a separate shared library)
- newly added `unwind.h` header file located in `/usr/include/unwind.h`
- fully supports the C++ ABI for Itanium architecture: Exception Handling, a language independent exception handling support specification

Unwind Library Interface Improvements

Value `get` and `put` functions will be used to read and to modify the unwind context data structure. This allows the library to properly encapsulate the data, protecting client code from changes in the implementation of the unwinder.

Function name, signature, and specification changes: Because the unwinder is sensitive to the processor architecture and the run-time architecture, the IPF unwinder will have different entry points than the PA unwinder. Type

```
man U_STACK_TRACE
```

or

```
man 5 unwind
```

for unwinder documentation.

The set of text error messages produced by the unwind library has been replaced with the enumeration `_UNW_ReturnCode` in the `/usr/include/unwind.h` header file. This allows programatic operation of the unwinder.

NOTE The `U_STACK_TRACE()` function is unchanged. It is the most commonly used and least architecturally dependent interface into the unwinder. A large percentage of customers who use Unwind only use `U_STACK_TRACE()`.

Unwind Library Name Changes

HP-UX 11i Version 1.5 (B.11.20) includes both 32 and 64 bit versions of the unwind library. The 32 bit version is at

`/usr/lib/hpux32/libunwind.so.1`

and linked to

`/usr/lib/hpux32/libunwind.so`

The 64 bit version is at

`/usr/lib/hpux64/libunwind.so.1`

and linked to

`/usr/lib/hpux64/libunwind.so`

To prevent link line compatibility issues during a port from PA to IPF, `/usr/lib/hpux32/libcl.so` and `/usr/lib/hpux64/libcl.so` each list the appropriate `libunwind.so` as a dependency. Also, `libcl.so` lists the appropriate Fortran I/O support library as a dependency.

Unwind Library Compatibility

Developers writing stack unwind routines should be aware of the following porting compatibility issues:

- Applications which used the PA unwind library (in libcl) to perform analysis when encountering an exception or a breakpoint (as in a debugger) and which used entry points other than `U_STACK_TRACE()` will need to change calls to the unwinder and interpret the return codes in the enumeration `_UNW_ReturnCode`.

- Functional equivalents to `U_init_frame_record()`, `U_resume_executionx()`, and `U_get_previous_frame_x()` are provided. The new interface does not require (and therefore does not have functional equivalents to) `U_prep_frame_rec_for_unwind()`, `U_update_state_vectorx()`, and `U_copy_frame_info()`.

Unwind Library Documentation

Information on entry points to the unwinder can be found on the *unwind* (5) manual page.

Additional information about the generation and layout of unwind information on Itanium-based systems, and about coding conventions for producing unwindable code, but not about the unwind library itself, is available in the Runtime Architecture documents listed below. These documents can be found on the web at http://devresource.hp.com.

- *Itanium Runtime Architecture and Software Conventions*
- *Runtime Supplement 32-Bit Runtime Architecture for HP-UX*
- *Runtime Supplement Program Startup on HP-UX*
- *Runtime Supplement Statically-Bound Programs on HP-UX*

X11 Libraries

The X11 libraries in HP-UX 11i Version 1.5 (B.11.20)

- do not support local graphics devices in this release. Only remote graphics via network protocols are available.
- include 32 and 64 bit native IPF X11R6 libraries and header files
  - — Motif (Xm)
  - — X Toolkit Intrinsics (Xt)
  - — Session Manager (SM)
  - — Athena Widgets (Xaw)
  - — X Miscellaneous utilities (Xmu)
  - — Motif Resource Manager (Mrm)
- include native IPF runtime and development environments
- include 32 and 64 bit PA X11R5 and X11R6 libraries and header files

Applications are limited to displaying X Windows graphics remotely, over the network, to systems that have graphics capability and an X Server.

Applications which link to X11R5 may need to link to X11R6 instead. The X11 libraries are forward compatible, so application code changes, other than makefiles, should not be required.

X/Motif Developers Toolkit

The IPF-native X/Motif developers toolkit can be used to develop native IPF applications. The developer’s toolkit contains

- X and Motif header files
- X and Motif man pages
- Motif example programs
- shared object soft links (`.so`)
- IPF native user interface language compiler (32 and 64 bit)
- IPF native imake

Native X/Motif Libraries

HP-UX 11i Version 1.5 (B.11.20) includes native Itanium-based

- 32-bit and 64-bit X/Motif libraries
- 32-bit and 64-bit user interface language complier (uil & uil46)
- 32-bit imake

The 32 bit libraries are:

- `/usr/lib/hpux32/Motif2.1/libXm.so.1`
- `/usr/lib/hpux32/libXm.so.1` (a symbolic link to `/usr/lib/hpux32/Motif2.1/libXm.so.1`)
- `/usr/lib/hpux32/X11R6/libXt.so.1`
- `/usr/lib/hpux32/libXt.so.1` (a symbolic link to `/usr/lib/hpux32/X11R6/libXt.so.1`)
- `/usr/lib/hpux32/X11R6/libSM.so.1`
- `/usr/lib/hpux32/libSM.so.1` (a symbolic link to `/usr/lib/hpux32/X11R6/libSM.so.1`)
- `/usr/contrib/X11R6/lib/hpux32/libXaw.so.1`
- `/usr/lib/hpux32/libXaw.so.1` (a symbolic link to `/usr/contrib/X11R6/lib/hpux32/libXaw.so.1`)
- `/usr/contrib/X11R6/lib/hpux32/libXmu.so.1`
- `/usr/lib/hpux32/libXmu.so.1` (a symbolic link to `/usr/contrib/X11R6/lib/hpux32/libXmu.so.1`)
- `/usr/lib/hpux32/Motif2.1/libMrm.so.1`
- `/usr/lib/hpux32/libMrm.so.1` (a symbolic link to `/usr/lib/hpux32/Motif2.1/libMrm.so.1`)
- `/usr/lib/hpux32/X11R6/libX11.so.1`
- `/usr/lib/hpux32/libX11.so.1` (a symbolic link to `/usr/lib/hpux32/X11R6/libX11.so.1`
- `/usr/lib/hpux32/X11R6/libXext.so.1`
- `/usr/lib/hpux32/libXext.so.1` (a symbolic link to `/usr/lib/hpux32/X11R6/libXext.so.1`
- `/usr/lib/hpux32/X11R6/libXi.so.1`

- /usr/lib/hpux32/libXi.so.1 (a symbolic link to /usr/lib/hpux32/X11R6/libXi.so.1
- /usr/lib/hpux32/X11R6/libXtst.so.1
- /usr/lib/hpux32/libXtst.so.1 (a symbolic link to /usr/lib/hpux32/X11R6/libXtst.so.1
- /usr/lib/hpux32/X11R6/libXIE.so.1
- /usr/lib/hpux32/libXIE.so.1 (a symbolic link to /usr/lib/hpux32/X11R6/libXIE.so.1
- /usr/lib/hpux32/X11R6/libXp.so.1
- /usr/lib/hpux32/libXp.so.1 (a symbolic link to /usr/lib/hpux32/X11R6/libXp.so.1
- /usr/lib/hpux32/X11R6/libICE.so.1
- /usr/lib/hpux32/libICE.so.1 (a symbolic link to /usr/lib/hpux32/X11R6/libICE.so.1

The 64 bit libraries are

- /usr/lib/hpux64/Motif2.1/libXm.so.1
- /usr/lib/hpux64/libXm.so.1 (a symbolic link to /usr/lib/hpux64/Motif2.1/libXm.so.1)
- /usr/lib/hpux64/X11R6/libXt.so.1
- /usr/lib/hpux64/libXt.so.1 (a symbolic link to /usr/lib/hpux64/X11R6/libXt.so.1)
- /usr/lib/hpux64/X11R6/libSM.so.1
- /usr/lib/hpux64/libSM.so.1 (a symbolic link to /usr/lib/hpux64/X11R6/libSM.so.1)
- /usr/contrib/X11R6/lib/hpux64/libXaw.so.1
- /usr/lib/hpux64/libXaw.so.1 (a symbolic link to /usr/contrib/X11R6/lib/hpux64/libXaw.so.1)
- /usr/contrib/X11R6/lib/hpux64/libXmu.so.1
- /usr/lib/hpux64/libXmu.so.1 (a symbolic link to /usr/contrib/X11R6/lib/hpux64/libXmu.so.1)
- /usr/lib/hpux64/Motif2.1/libMrm.so.1
- /usr/lib/hpux64/libMrm.so.1 (a symbolic link to

`/usr/lib/hpux64/Motif2.1/libMrm.so.1`)

- `/usr/lib/hpux64/X11R6/libX11.so.1`
- `/usr/lib/hpux64/libX11.so.1` (a symbolic link to `/usr/lib/hpux64/X11R6/libX11.so.1`
- `/usr/lib/hpux64/X11R6/libXext.so.1`
- `/usr/lib/hpux64/libXext.so.1` (a symbolic link to `/usr/lib/hpux64/X11R6/libXext.so.1`
- `/usr/lib/hpux64/X11R6/libXi.so.1`
- `/usr/lib/hpux64/libXi.so.1` (a symbolic link to `/usr/lib/hpux64/X11R6/libXi.so.1`
- `/usr/lib/hpux64/X11R6/libXtst.so.1`
- `/usr/lib/hpux64/libXtst.so.1` (a symbolic link to `/usr/lib/hpux64/X11R6/libXtst.so.1`
- `/usr/lib/hpux64/X11R6/libXIE.so.1`
- `/usr/lib/hpux64/libXIE.so.1` (a symbolic link to `/usr/lib/hpux64/X11R6/libXIE.so.1`
- `/usr/lib/hpux64/X11R6/libXp.so.1`
- `/usr/lib/hpux64/libXp.so.1` (a symbolic link to `/usr/lib/hpux64/X11R6/libXp.so.1`
- `/usr/lib/hpux64/X11R6/libICE.so.1`
- `/usr/lib/hpux64/libICE.so.1` (a symbolic link to `/usr/lib/hpux64/X11R6/libICE.so.1`

X11 Libraries Compatibility

Itanium-based native libraries are not backward compatible with HP-UX 11i and earlier releases.

SNMP

SNMP Agent

SNMP agents (SNMP master agent, MIB-II subagent, Trap Dest subagent, and HP-UX Subagent) are not available in HP-UX 11i Version 1.5 (B.11.20). If you want to install the SNMP Agent on this release you should download the README and agent depot file from the ftp location listed below and follow the instructions in the README file to install the agent on HP-UX 11i Version 1.5 (B.11.20).

- ftp://ovweb.external.hp.com/SNMPAgent/11.20/README
- ftp://ovweb.external.hp.com/SNMPAgent/11.20/OVSNMPAgent11.20.depot.gz

SNMP Developer's Kit and libsis.sl

The SNMP Developer's Kit and libsis.sl are not available in HP-UX 11i Version 1.5 (B.11.20). This affects the availability of SIS functionality and the OSPF agent. `libsis.sl` was obsoleted in release HP-UX 11i. See Secure Internet Services Limitations in Chapter 5 and ftp and other Secure Internet Services commands in Chapter 7.

Machine Dependent Headers

The machine dependent headers in the `/usr/include/machine` directory have been restructured. Applications including the headers from the `/usr/include/machine` directory will need to make minor modifications to `#include` statements accordingly in order to compile.

NOTE Please note that source compatibility is not guaranteed for applications which include any undocumented headers.

Threadsafe Documentation

The new manual page, *thread_safety* (5),documents the list of threadsafe APIs in `libc`, `libpthread`, and system functions. Other manual pages with threadsafe information have been modified to point to the new manual page. This change will make it easier to develop applications which use threads.

Table 7-1 **System Calls Affected by thread_safety(5)**

accept (2)
acl (2)
adjtime (2)
aio_fsync (2)
aio_read (2)
aio_suspend (2)
aio_write (2)
brk (2)
close (2)
creat (2)
creat64 (2)
dup (2)
dup2 (2)
exec (2)
fcntl (2)
getcontext (2)
getmsg (2)
getsid (2)
kill (2)
msem_init (2)
msem_lock (2)
msem_remove (2)
msgop (2)
msync (2)
nanosleep (2)
open (2)
pause (2)
pipe (2)
poll (2)
pstat (2)
putmsg (2)
recv (2)
rename (2)
select (2)
semop (2)
send (2)
sendfile (2)
sighold (2v)
sigsuspend (2)
sigvector (2)
sigwait (2)
socketpair (2)
sysconf (2)
timers (2)
ulimit (2)
wait3 (2)
waitid (2)

Table 7-2 **Functions Affected by thread_safety(5)**

a64l (3C)
abort (3C)
abs (3C)
aclsort (3C)
acltostr (3C)
addch (3X)
getcwd (3C)
getdate (3C)
getdiskbyna (3C)
getgrent (3C)
gethostent (3N)
getlogin (3C)
putpwent (3C)
puts (3S)
putwc (3C)
putws (3C)
qsort (3C)
rand (3C)

Table 7-2 Functions Affected by thread_safety(5)

addsev (3C)
assert (3X)
atexit (3)
basename (3C)
bgets (3G)
bigcrypt (3C)
bindresvpor (3N)
blmode (3C)
bsd_signal (3C)
bsearch (3C)
btowc (3C)
bufsplit (3G)
catgets (3C)
catopen (3C)
cfspeed (3C)
chownacl (3C)
clearenv (3C)
clock (3C)
confstr (3C)
conv (3C)
copylist (3G)
cpacl (3C)
cr_perror (3)
crt0 (3)
ctermid (3S)
ctime (3C)
getmntent (3X)
getnetent (3N)
getnetgrent (3C)
getopt (3C)
getprotoent (3N)
getpwent (3C)
gets (3S)
getservent (3N)
getspent (3C)
getsubopt (3C)
gettimer (3C)
gettxt (3C)
getusershel (3C)
getut (3C)
getutx (3C)
getwc (3C)
getwd (3C)
getwin (3X)
glob (3C)
grantpt (3C)
gss_accept_sec_context (3)
hsearch (3C)
iconv (3C)
inet (3N)
inet6 (3N)
initgroups (3C)
random (3M)
rcmd (3N)
re_comp (3X)
realpath (3X)
regcomp (3C)
regexp (3X)
reltimer (3C)
resolver (3N)
rexec (3N)
rmtimer (3C)
rpc_svc_calls (3N)
scandir (3C)
scanf (3S)
setaclentry (3C)
setbuf (3S)
setcat (3C)
setclock (3C)
setjmp (3C)
setlabel (3C)
setlocale (3C)
sigpause (3C)
sigset (3C)
sigsetops (3C)
sleep (3C)
ssignal (3C)
statfsdev (3C)

Table 7-2 Functions Affected by thread_safety(5)

ctype (3C)
cuserid (3S)
datalock (3C)
dbm (3C)
delscreen (3X)
derwin (3X)
devnm (3)
dial (3C)
directory (3C)
div (3C)
drand48 (3C)
dupwin (3X)
echo_wchar (3X)
ecvt (3C)
elf_getbase (3E)
end (3C)
fattach (3C)
fclose (3S)
fdetach (3C)
fegetenv (3M)
ferror (3S)
fgetpos (3S)
fgetpos64 (3S)
fgetws (3C)
fileno (3S)
flockfile (3S)

insque (3C)
isastream (3C)
isendwin (3X)
l3Tol (3C)
lckpwdf (3C)
ldcvt (3C)
localeconv (3C)
logname (3C)
lsearch (3C)
ltostr (3C)
malloc (3C)
mbrlen (3C)
mbrtowc (3C)
mbsinit (3C)
mbsrtowcs (3C)
memalign (3C)
memory (3C)
mkdirp (3G)
mktemp (3C)
mktimer (3C)
monitor (3C)
multibyte (3C)
ndbm (3X)
net_aton (3C)
nl_langinfo (3C)
nlist (3C)

statvfsdev (3C)
strfmon (3C)
strftime (3C)
string (3C)
strord (3C)
strptime (3C)
strtoacl (3C)
strtod (3C)
strtol (3C)
strtold (3C)
swab (3C)
syslog (3C)
tcattribute (3C)
tccontrol (3C)
tcgetpgrp (3C)
tcgetsid (3C)
tcsetpgrp (3C)
tmpnam (3S)
towctrans (3C)
tsearch (3C)
ttyname (3C)
ttyslot (3C)
ungetc (3S)
ungetwc (3C)
vprintf (3S)
vscanf (3S)

Table 7-2 **Functions Affected by thread_safety(5)**

fmtmsg (3C)
fnmatch (3C)
fopen (3S)
fread (3S)
ftok (3C)
fwide (3C)
fwprintf (3C)
fwscanf (3C)
get_expiration_time (3T)
getc (3S)
getclock (3C)
pam_strerror (3)
pathfind (3G)
pfmt (3C)
popen (3S)
prcmd (3N)
printf (3S)
pthread_cond_signal (3T)
ptsname (3C)
putc (3S)
putenv (3C)
putp (3X)
vwprintf (3C)
wconv (3C)
wcrtomb (3C)
wcsftime (3C)
wcsrtombs (3C)
wcstod (3C)
wcstol (3C)
wcstring (3C)
wctype (3C)
wmemory (3C)
wordexp (3C)

Obsolescence and Deprecation of APIs

This section defines the obsolescence of core system libraries and relocatable objects. Obsolescence of other products are covered in separate sections.

Rationale and Objectives

The HP rational and objectives of obsolesence and deprecation of APIs are:

- provide common, standard APIs across *UNIX* vendors
- facilitate portability for our ISVs
- reduce confusion for the selection of similar APIs
- reduce the size of libc, thus increasing performance of shared libc
- reduce the continued application turbulence for future architecture changes
- remove the compatibility problems for applications which link with shared libraries that have dependencies on archive system libraries
- reduce satisfaction issues with APIs that have specification defects, for example, *compatibility issues*
- reduce support costs for APIs that are not in the strategic direction of standards, the industry, and our customers
- minimize adoption issues for new releases on PA or IPF

The intent is that there will be NO gratuitous changes, and obsolescence of APIs and libraries is acceptable when initiated to avoid application breakage or duplicate functionality.

Terms and Definitions

Deprecated: A *“deprecated”* interface can have the following characteristics:

- functionality is available on the system
- deprecation is a step towards obsolescence

- the specification is in flux
- less value to users
- functionality no longer makes sense
- functionality has been replaced
- support/enhancement expectations have been lowered
- usage is discouraged
- warnings against usage/alternatives are provided
- the provider continues to test functionality
- migration plan/tools are provided

The reasons for marking an interface as “*deprecated*” may include:

- marked "*to be withdrawn*" by standards
- support is available via more standard means
- equivalent, enhanced, more reliable counterparts exist
- also all reasons listed in the "*Obsolete*" section below

Obsolete: An "*obsolete*" interface may have the following characteristics:

- functionality is no longer available on the system
- runtime support is undefined
- cannot develop or build with this interface
- documentation is not provided or recommends against usage
- the final stage of the product life cycle has been reached

The reasons for marking an interface as "*obsolete*" may include:

- underlying infrastructure in either the software or hardware is obsolete or not available
- changes to the system have decreased reliability
- miscellaneous business decisions such as:
  - — third parties solution exists
  - — not strategic
  - — support costs are too high

— not enough ROI

Core System Libraries

With the release of HP-UX on the Itanium-based platform, there are several APIs and libraries that are not available for native (Itanium-based platform) development. The PA equivalent versions will be available on Itanium-based systems to support deployment of PA applications on Itanium-based systems (see Aries - PA-RISC Binary Support).

These libraries and APIs were deprecated in HP-UX 10.30 and later releases, and, excluding `libc.a`, will be obsolete in a future release on PA systems. Applications using these libraries and APIs will fail to compile (missing definition in header file) or will not link (missing library or unsatisfied symbol). The list of libraries that are not available in native form on Itanium-based systems for HP-UX 11i Version 1.5 (B.11.20) is:

- `libc.a`, pa20_64/libc.a (archived libc, 32-bit and 64-bit)
- `libp/libc.a` (archived profile libc)
- `libpicc.a` (position independent library)
- `libdbm.a`, libdbm.1, and `libdbm.sl` (database management library)
- `libndbm`, `libndbm.2`, and `libndbm.sl` (database management library)
- `libPW.a` (AT&T Programmer's Workbench)
- `libBSD.a` (BSD 4.2 library)
- `libcma.1`, `libcma.2`, and `libcma.sl` (CMA thread library)
- `libd4r.1` and `libd4r.sl` (_r thread-safe POSIX draft 4 APIs)
- `/usr/old/libmalloc3x.a` and `/usr/old/malloc3c.o` (old PA malloc object files)
- `libsis.sl` (Secure Internet Services)

The following header files are not provided on Itanium-based systems. You should use alternative APIs, as documented in HP-UX 10.30 and later release notes.

- `/usr/old/user/include/nl_ctype.h` (HP-UX 9.x interfaces)

The following libraries are empty on PA and are not provided on Itanium-based systems. Modify your makefiles for native development.

- `libV3.a`
- `libV7.a`
- `libbsdpic.a`
- `libcrypt.a`
- `libgprof.a`
- `libmalloc.a`
- `libn.a`
- `libndir.a`
- `libnet.a`
- `libnsipc.a`
- `libprof.a`

The following obsolete APIs are not available in the native IPF libc:

- dump state of memory allocator, `memorymap()`
- HP block mode terminal interface

  — `blclose()`
  — `blget()`
  — `blopen()`
  — `blset()`
  — `blread()`
  — `blmodeio.h`

- internal multi-byte APIs

  — `mbscat()`
  — `mbscmp()`
  — `mbslen()`
  — `mbsncat()`
  — `mbsncmp()`
  — `mbsncpy()`
  — `mbspbrk()`
  — `mbsrchr()`
  — `mbswidth()`
  — `pcstombs_wc_cnt()`

  - mbstoint()
  - mbsinvalid()

- process trace
  - ptrace()
  - ptrace64()

- find symbol in object file - moved to libelf
  - nlist()
  - nlist64()

- nl_tools_16 APIs and macros
  - byte_status()
  - firstof2()
  - secof2()
  - c_colwidth()
  - ADVANCE()
  - BYTE_STATUS()
  - C_COLWIDTH()
  - CHARADV()
  - CHARAT()
  - FIRSTof2()
  - PCHAR()
  - PCHARADV()
  - SECof2()
  - WCHAR()
  - WCHARADV()
  - WC_COLWIDTH()

In moving applications which use any of the above libraries or APIs from PA to IPF, you will need to modify the source code to use an alternative interface, or continue to run the application as a PA program.

libHcurses and Related APIs

The previously deprecated `libHcurses` library, along with `captoinfo_colr`, `infocmp_colr`, `terminfo_colr`, `tic_colr`, `tput_colr`, and `libcur_colr`, is not available on HP-UX 11i Version1.5. Color capabilities are include in the `terminfo` database and `libxcurses`.

Some applications may need to be ported as there are minor differences between `libHcurses` and `libxcurses`.

Linker Toolset

The IPF linker toolset provides a set of tools for linking and managing ELF object files. This toolset includes the `ar`, `chatr`, `elfdump`, `ld`, `ldd`, `lorder`, `nm`, `size`, and `strip` utilities.

Linking and Loading Behavior

In HP-UX 11i Version 1.5 (B.11.20) the linker and dynamic loader use a linking and loading model that is consistent with the de facto industry standard. The IPF linker maintains certain 32-bit PA behaviors to make the transition from 32-bit PA to IPF easier. You can enable this compatibility feature by using the `+compat` linker option. The `+compat` option forces the linker and loader to mimic some of the 32-bit PA behaviors.

Linking and Loading Compatibility

There is no incompatibility for applications migrating from 64-bit PA to 64-bit IPF. Incompatibility for applications migrating from 32-bit PA to 32-bit on IPF can be grouped as follows:

- obsoleted options and features

  Beginning with HP-UX 11.00, the linker has issued compatibility warnings when these options or features were used. Obsoleted linker options:

  `-A` *`name`* This option produces an object file suitable for incremental loading into an already executing archive program. IPF applications must use shared libraries instead.

  `-C` *`n`* This option enables function parameter type checking. Parameter type checking and relocation stubs are not supported in the IPF run time architecture.

  `-S` This option is specific to 32-bit PA SOM object file format. It enables generation of SOM Initial Program header (IPL) instead of HP-UX auxiliary header.

Obsoleted features:

Intra-library versioning

In HP-UX 10.0 and later releases, the 32-bit PA linker supported SVR4 library level versioning in addition to intra-library versioning. The IPF linker does not support intra-library versioning. You must use SVR4 library-level versioning instead.

Name space conflict

In 64-bit PA and IPF, the code and data symbol names share the same name space. Applications migrating from 32-bit PA to IPF should rename conflicting symbols.

- locating dependent shared libraries (or dynamic path searching)

  In 64-bit PA and IPF applications, the dynamic loader locates dependent shared libraries using a model similar to other SVR4 systems. You can enforce the conflicting symbols 32-bit PA dynamic loading behavior by using the `+compat` linker option.

- symbol searching

  In 32-bit PA, the dynamic loader searches shared libraries using a depth-first search order. In 64-bit PA and IPF, the dynamic loader searches shared libraries using a breadth-first search order. Breadth-first symbol searching is used on all SVR4 platforms. You can enforce the old 32-bit PA style depth-first symbol searching with the `+compat` linker option.

Linker Toolset Debugging Behavior

On PA systems the default behavior is `+noobjdebug`, on HP-UX 11i Version 1.5 (B.11.20), the default debugging behavior is `+objdebug`. When `+objdebug` is enabled, debug information is not copied over to the executable file, relocatable object files should not be deleted, and they must be present when debugging the program. This results in a significant improvement in link time.

Linker Toolset Debugging Compatibility

Use the +noobjdebug option to force the old style debug model.

PA-RISC Linker Toolset

The linker toolset supports the building of native IPF applications. PA-RISC applications can be run on IPF (through the Aries code emulator) but not compiled and linked on IPF. The PA-RISC linker toolset files (ar, chatr, ldd, nm, odump, size, strip) are provided in a separate directory (/usr/ccs/pa/usr/ccs/bin) and can be used to support existing PA-RISC applications. To use the PA-RISC linker toolset, set the SDKROOT environment variable to /usr/ccs/pa and specify the regular toolset file in the command line. This method invokes a script that starts up the corresponding PA-RISC linker toolset file. For example, with the k-shell:

```
$ export SDKROOT=/usr/ccs/pa
$ chatr a.out
```

To run commands from the IPF linker toolset again, unset SDKROOT. Or, if you are running in a mixed environment and need to run IPF and PA-RISC linker tools on different files, you can set SDKROOT for just one command:

```
$ SDKROOT=/usr/ccs/pa chatr a.out
```

Linker Toolset Documentation

For more information, see http://devresource.hp.com and the *ld* (1) manual page.

gprof Multiple Shared Library Profiling

gprof, in HP-UX 11i Version 1.5 (B.11.20), introduces multiple shared library profiling (MSLP). This allows users to profile multiple shared libraries along with native IPF applications.

To use MSLP, the environment variables LD_PROFILE and LD_PROFILEBUCKET_SIZE must be set appropriately before execution of the program starts. Additionally, appropriate options must be set at compile and link time.

gprof Performance

The accuracy of timing information may degrade slightly compared to earlier version of gprof.

gprof Documentation

For more information, see the *gprof* (1), *cc* (1), *ld* (1), and *smonitor* (3C) manual pages.

CXperf Performance Monitoring Support

HP-UX 11i Version 1.5 (B.11.20) supports the CXperf performance monitoring utility with no graphical user interface. For details refer to the *cxperf* (1) manpage.

CXperf is an optional product.

CXperf Documentation

Additional information and updates are available on the *cxperf* (1) manual page and at

http://devresource.hp.com

HP Caliper

HP Caliper 1.0 is a new general-purpose performance analysis and performance improvement tool for Itanium-based applications bundled with HP ANSI C, HP aC++, and HP Fortran 9x.

HP Caliper helps you analyze and improve the performance of your native Itanium-based programs in three ways:

- A simple and quick way to optimize the performance of your program by providing information for compiler profile-based optimization.
- Commands to measure the overall performance of your program.
- Commands to drill down to identify performance parameters of specific functions in your program.

A significant advantage of HP Caliper is that it requires no special compilation, link options, or libraries for your Itanium-based programs. HP Caliper dynamically measures performance on:

- C, C++, and Fortran 9x binaries
- 32- or 64-bit binaries
- Shared or archive libraries
- Debug or optimized programs

Another advantage is that HP Caliper performs its measurements with low overhead and for well-behaved executables, HP Caliper does not alter the semantic behavior of the target program.

Required disk space is 12 MB. Memory requirements vary with settings used for performance measurement.

HP Caliper Documentation

HP Caliper 1.0 includes the following documentation:

- On-line User Guide in HTML format available with the `+help` option.
- Printable User Guide in PDF format.
- Command-line option summary available with `--help` option.
- On the web at http://www.hp.com/go/caliper
- The *caliper* (1) man page.

WDB Debugger Tool

HP-UX 11i Version 1.5 (B.11.20) includes an Itanium-based version of WDB (`/opt/langtools/bin/gdb`), a debugger for C, C++, Fortran and assembly language. This version will only work on native Itanium-based applications (programs compiled for HP-UX on Itanium-based systems). gdb command functionality on Itanium-based systems is the same as on PA based systems except for:

- the `-tui` mode. On PA based systems this provides a terminal interface similar to xdb. On HP-UX 11i Version 1.5 (B.11.20), gdb will only have a line mode interface. A graphical user interface will be available in a later release.
- run-time heap checking. The features in the 2.1 release of WDB on PA-based systems that detected memory leaks and heap problems are not available in this release.
- fix-and-continue. The fix-and-continue features introduced in the 2.1 release of WDB are not available in this release.

ADB Debugger Tool

Changes to the adb command have been made to enable adb to handle 64-bit architectures in an improved manner. The new version of the ADB debugger tool also provides better overall performance and reliability, including improvements to command line syntax and options, format strings, writing and searching, and expressions. The ADB Debugger also supports the following new features:

- Shared library support
- Threads support
- Multiprocessor dump reading support
- 64-bit DLKM dump reading support
- Support for all types of ELF and SOM object files
- Single binary for 32-bit and 64-bit

The following discussion provides a summary of some of the major changes to adb.

Specifying the New ADB Behavior

To specify the new adb command, use the -n command line option. (For Itanium-based systems, the -n option is the default). For example,

```
$ adb -n /stand/vmunix /dev/kmem

adb> runningprocs/d

         runningprocs:
            4

adb> q

$
```

There are no binary compatibility issues with this change.

Retaining Old ADB Behavior

The only reason you should enable old adb behavior is to ensure that any scripts that were written earlier run unchanged. For all interactive use and for new development the old adb behavior is *strongly discouraged*.

If you wish, you can specify the old adb behavior on Itanium-based systems by giving the -o command line option to adb. Another way to enable old behavior is to add the following line to the file $HOME/.adbrc.

```
$a backcompat 1
```

Changes in ADB Command Line Syntax

The -k and -m command line options are no longer required. If you wish to look at a crash dump, you simply specify the crash dump directory as the sole argument. (In previous versions of adb, the -k option was required to recognize HP-UX crash dumps or /dev/mem. The -m option was required to recognize multiple file HP-UX crash dumps.) In the backward compatibility mode of operation of adb, the -k and -m options are ignored with a warning.

ADB Command Prefix Changes

New adb commands no longer require a $ prefix (as they did in previous versions of adb). The following examples show the old syntax and the equivalent new syntax.

Example 7-1 **To quit adb**

Old command: $q
New command: q

Example 7-2 **To print the value of a variable named q in hexadecimal:**

Old command: q/X
New command: p q/x

ADB Command Processing

The new version of adb ignores the SIGQUIT signal. However, on SIGINT signal adb interrupts the current command and waits for the next command. An interrupted command will not update any state variables. In this case, adb prints a warning message.

adb exits on an EOF. EOL is treated as a command with no command components and interpreted accordingly.

ADB Changes to Format Strings

In the new version of adb, format characters make adb syntax scalable and regular. All format characters are now lower case with a preceding optional length specifier character.

printf Style Format Strings

The new adb also supports printf style format strings, which should be preceded by a comma and enclosed within quotes. In this case, the format characters and length specifiers are the same as above, but flags, field widths and precision can be specified just as for the `printf()` library function.

Changes in ADB Writing and Searching

The old adb uses the format characters `w` and `W` for writing and `l` and `L` for searching. However, the new adb uses characters `w` and `l` as regular format characters. Therefore, in the new adb the following replacements occur:

Table 7-3

Old Format Character	Equivalent New Character
w	=,2
W	=
l	/,2
L	/

Impact of Changes on Pre-Existing ADB Scripts

The features or behaviors that are enabled in the backward compatibility mode of adb (see Retaining Old ADB Behavior) are retained only for supporting macros written for earlier versions of adb. No attempt is made to preserve the interactive behavior as it will not affect macros. Users are discouraged from using these features in new macros and scripts, as these features may be obsoleted in a later version of adb. Most of the new features introduced in this version of adb are not available in the backward compatibility mode of operation of adb.

The features or behaviors described as deprecated (in the *adb* (1) manpage) will be obsoleted over one or two major releases of the HP-UX operating system. These features should not be used in new macros and scripts. Most of these features and behaviors will have only minor impact on existing macros or scripts. If there are any macros which use these features, they should be rewritten before these are obsoleted.

ADB Documentation

The updated *adb* (1) manpage describes the new and changed features of the ADB Debugger Tool in detail. For additional information see the *ADB Release Notes* available at http://docs.hp.com.

Fortran

The HP Fortran for Itanium-based systems is source compatible with the HP-UX 11i PA Fortran compiler. The differences are

- Some command line arguments now conform to the Itanium-based compiler options use by all HP IPF compilers. Old flags work as well.
- Consistent driver flags for some common functionality across all languages.

Packaging for Specific Releases

Software that runs only on certain releases of HP-UX should be packaged to install only on those OS releases.

This can be accomplished by specifying the correct `os_release` product attribute in the Product Specification File (PSF). Software Distributor matches the value assigned to that attribute against the `uname -r` value of a system, such as `B.11.20`. Some wildcards and regular expressions are allowed. For example, to make the software installable on any HP-UX 11.x release, include the following line in the PSF:

```
os_release ?.11.*
```

If the software is compatible with HP-UX 11i Version 1.5 (B.11.20) or HP-UX 11i (B.11.11) but not with HP-UX 11.0 (B.11.00), include the following line:

```
os_release ?.11.[12]*
```

More information about the PSF is available in the *sd* (4) and *swpackage* (4) manual pages.

8 Other Functionality

This chapter describes other new and changed operating-system software functionality including:

- Changes to PAM
- Event Monitoring Service (EMS)
- Distributed Computing Environment (DCE)
- HP Distributed Print Service Obsolete
- Internationalization
- HP-UX Software Transition Kit (STK)
- Aries - PA-RISC Binary Support
- Netscape Communicator
- Java 2
- Apache Web Server
- OpenGL (OGL) Graphics API

Changes to PAM

PAM (Pluggable Authentication Module) gives system administrators the flexibility of choosing any authentication service available on the system to perform authentication. The framework interface is implemented by the library `/usr/lib/libpam.1` and new authentication service modules can be plugged in and made available without modifying the applications. The authentication services are implemented by their own loadable modules whose paths are specified through the `/etc/pam.conf` and `/etc/pam_user.conf` files.

In order for PAM to support multiple instruction set architectures (i.e. 32/64-bit IPF/PA), the Corrigendum U039 of the PAM Standard is implemented in HP-UX 11i Version 1.5 (B.11.20) (see http://www.opengroup.org/pubs/corrigenda/u039f.htm). The standard does not allow mixing of 32/64-bit (for example, a 32-bit application cannot execute 64-bit library code).

The format of the `/etc/pam.conf` and `/etc/pam_user.conf` files and the paths of the pluggable service authentication modules have changed. The paths have not changed for 32-bit PAM libraries for PA architecture. New paths are assigned for new architectures supported starting with the this release. The new paths are for 32-bit IPF PAM libraries, 64-bit IPF PAM libraries, and 64-bit PA libraries.

NOTE 64-bit PAM support is not yet available for IPF and PA.

The new `pam.conf` and `pam_user.conf` files also support the `$ISA` token as defined by Corrigendum U039.

Here is the new `pam.conf` file (note the comment):

```
==============================================================================
#
# PAM configuration
#
# Authentication management
# Note: For PA applications, /usr/lib/security/libpam_unix.so.1 is a
# symbolic link that points to the corresponding PA PAM module.
#
#
```

```
login    auth required  /usr/lib/security/$ISA/libpam_unix.so.1
su       auth required  /usr/lib/security/$ISA/libpam_unix.so.1
dtlogin  auth required  /usr/lib/security/$ISA/libpam_unix.so.1
dtaction auth required  /usr/lib/security/$ISA/libpam_unix.so.1
ftp      auth required  /usr/lib/security/$ISA/libpam_unix.so.1
OTHER    auth required  /usr/lib/security/$ISA/libpam_unix.so.1
#
# Account management
#
login    account required       /usr/lib/security/$ISA/libpam_unix.so.1
su       account required       /usr/lib/security/$ISA/libpam_unix.so.1
dtlogin  account required       /usr/lib/security/$ISA/libpam_unix.so.1
dtaction account required       /usr/lib/security/$ISA/libpam_unix.so.1
ftp      account required       /usr/lib/security/$ISA/libpam_unix.so.1
#
OTHER    account required       /usr/lib/security/$ISA/libpam_unix.so.1
#
# Session management
#
login    session required       /usr/lib/security/$ISA/libpam_unix.so.1
dtlogin  session required       /usr/lib/security/$ISA/libpam_unix.so.1
dtaction session required       /usr/lib/security/$ISA/libpam_unix.so.1
OTHER    session required       /usr/lib/security/$ISA/libpam_unix.so.1
#
# Password management
#
login    password required      /usr/lib/security/$ISA/libpam_unix.so.1
passwd   password required      /usr/lib/security/$ISA/libpam_unix.so.1
dtlogin  password required      /usr/lib/security/$ISA/libpam_unix.so.1
dtaction password required      /usr/lib/security/$ISA/libpam_unix.so.1
OTHER    password required      /usr/lib/security/$ISA/libpam_unix.so.1
============================================================================
```

Module providers can specify their own absolute path in the `/etc/pam.conf` and `/etc/pam_user.conf` files. In addition, the PAM engine library is modified to detect the `$ISA` token and the calling application type. It will then load the correct PAM module that is of the same architecture as the calling application. For example:

- 32-bit IPF applications

  Loads `/usr/lib/security/hpux32/libpam_unix.so.1`.

- 32-bit PA applications

  Loads `/usr/lib/security/libpam_unix.so.1` which is a link to `/usr/lib/security/libpam_unix.1`

NOTE A calling application must be built for the same architecture as the PAM service module.

PAM Compatibility

If the PAM modules install process edits `pam.conf` *or* non-32-bit PA versions of the module are being provided, then this process needs to be adapted to the new `pam.conf` file format. If the install process changes `pam.conf` to the old format, then the PAM modules will not be available to non-32-bit PA applications (that is, 32-bit IPF support will NOT be available for users of the PAM module in question unless `pam.conf` follows the PAM Standard). Also, for PA PAM modules, the install process must create a symbolic link to the actual PAM module (for example, the install process for `libpam_xyz.1` must create a symbolic called `libpam_xyz.so.1` that points to `libpam_xyz.1` in the `/usr/lib/security` directory).

Event Monitoring Service (EMS)

The Event Monitoring Service (EMS) Version A.03.30 is a framework that is used to monitor various system resources. On HP-UX 11i Version 1.5 (B.11.20), EMS provides base-level functionality and support for EMS hardware monitors. It does not provide HP-UX 11i release equivalent functionality. EMS will increase its functionality in the future as more monitors are developed for this operating system.

In this release, EMS is the underlying platform for the EMS hardware monitors. Customers are expected to use EMS only to configure monitoring requests for two EMS Hardware Monitors: the Kernel Resource Monitor (krmond), and the Peripheral Status Monitor (PSM).

The EMS High Availability monitors are not available for HP-UX 11i Version 1.5 (B.11.20). The EMS MIB monitor (mibmond) is also not available.

EMS Size Requirements

- The disk space requirement is 2.75MB. An additional 13MB of disk space should be allocated in `/etc/opt` to support EMS logging facilities.
- The memory requirement is 3MB.

EMS Documentation

For HP-UX 11i Version 1.5 (B.11.20), EMS is released as part of the OnlineDiag bundle (B4708AA). For more information about the EMS Hardware Monitors, see the http://docs.hp.com web site under

`Diagnostics -> EMS Monitors`

Distributed Computing Environment (DCE)

HP Distributed Computing Environment (HP DCE) provides a high-quality, comprehensive, standards-based framework to develop, administer and use distributed applications.

On HP-UX 11i Version 1.5 (B.11.20), the following products are available:

- DCE-Core (HP-UX 11i, PA version)
- DCE-IA64-Core (native on IPF platforms)
- DCE-IA64-Tools (native on IPF platforms)

Other DCE Core Services (Security and Cell Directory Services) are not available in native mode on Itanium-based systems.

DCE-IA64-Core provides DCE-RPC support and includes 32 and 64 bit version of libdcekt. See New libdcekt.sl and libdcecpkt.sl Libraries in Chapter 7 for more information. DCE-IA64-Tools is a subset of HP-UX 11i DCE-Core tools containing RPC related application development.

To take advantage of IPF features, RPC applications should be recompiled. Applications which use DCE-Core service APIs other than RPC (including `rpcd`) are not supported in native mode on Itanium-based systems.

A minimum of 64 MB RAM and 70 MB hard disk space is required for installation and use of the DCE Clients and libraries on HP-UX 11i Version 1.5 (B.11.20) systems.

DCE Compatibility

DCE clients can use either the IPF or the PA RPC APIs. DCE clients on IPF systems can work with DCE servers on a PA system, and DCE clients on PA systems can work with DCE servers on IPF systems. RPC applications may have performance degradation and unexpected behavior in this release.

`rpcd` is only available in PA mode.

DCE Documentation

For more information, see New libdcekt.sl and libdcecpkt.sl Libraries in Chapter 7 of these release notes and the *DCE Release Notes* available at `/opt/dce/newconfig/RelNotes`.

HP Distributed Print Service Obsolete

The HP Distributed Print Service (HPDPS) print environment is obsolete in HP-UX 11i Version 1.5 (B.11.20) and is not available. HPDPS, however, will continue to be supported for HP-UX 11.0 and 10.20 until further notice.

Four options are available for print services in this release:

1. the LP spooler which is part of the HP-UX operating system
2. the HP DirectJet 4000 Printing Appliance
3. the HP Document Router
4. the Dazel Output Server

Each option beyond the first is additive and increases in cost. Option 1 will provide base level print services. Option 2 will spool print jobs from Windows and HP-UX clients and provide Web-accessible print queues, management, and configuration. Option 3 will automate the delivery of documents and provide a Web-based administrative interface for TCP-connected network printers, fax, and email. Option 4 will provide an enterprise wide information delivery infrastructure to enable reliable document delivery and centralized print management across the entire enterprise.

Options 2-4 are available directly from HP and you should contact your local HP office for further information about functionality, platform support, and prices.

Option 3, the HP Document Router, most closely replicates the services that were provided by HPDPS although there is not a one-to-one mapping of commands.

HPDPS commands that are being obsoleted are listed here for quick reference:

Table 8-1 Obsolete HPDPS Commands

pdclean
pdcreate
pddcesetup
pddelete
pddisable
pdenable
pdgwcfg
pdls
pdmod
pdmsg
pdmsghlp
pdpause
pdpr
pdpromote
pdps
pdq
pdresubmit
pdresume
pdrm
pdset
pdshutdown
pdstartclient
pdstartspl
pdstartsuv
pdstopd

Internationalization

Asian System Environment (ASE)

HP-UX provides Asian systems for the Asian countries of the Far East, consisting of the following products:

JSE	Japanese System Environment
KSE	Korean System Environment
SSE	Simplified-Chinese System Environment
TSE	Traditional-Chinese System Environment

HP-UX provides several Asian enhancements as server features, including some new Asian codesets, UDC (User Defined Characters, or Gaiji), printing, and codeset conversions with mainframe codesets.

The new, changed, deleted features as well as some troubleshooting information is described below. For further information, see the following documentation:

- JSE
  - — *Japanese System Environment User's Guide* (B3782-90873)
  - — *HP XJIM Japanese Input Method Guide* (B3782-90869)
  - — *ATOK8 Japanese Input Method Guide* (B3782-90870)
  - — *EGBridge Japanese Input Method Guide* (B3782-90871)
  - — *VJE-gamma Japanese Input Method Guide* (B3782-90872)
- *KSE - Korean System Environment User's Guide* (5969-4454)
- *SSE - Simplified Chinese System Environment User's Guide* (5969-4455)
- *TSE - Traditional Chinese System Environment User's Guide* (5969-4453)

To get release information on earlier versions of ASE, see the following files:

- JSE: `/usr/share/doc/ASX-JPN`
- KSE: `/usr/share/doc/ASX-KOR`
- SSE: `/usr/share/doc/ASX-SCH`
- TSE: `/usr/share/doc/ASX-TCH`

Changed Features

The following features are not available in this release. They will be available in a future release.

- ASE Common
  - — X Print Server related features are not available in this release.
- JSE
  - ❑ ATOK X for HP-UX Preview Edition on HP-UX 11i.
  - ❑ Mainframe code set conversion on HP-UX 11i.
  - ❑ New UDC feature for client/server or distributed environments on HP-UX 11i.
  - ❑ Input of JISX0212 characters with XJIM.
- KSE
  - — Printing support for Euro and registered trademark (R) symbols to PCL printers with model file PCL5.asian.

Deleted Features

- ASE Common
  - ❑ HPDPS related features are not included in this release.

NOTE The Japanese input methods EGBridge and VJE-gamma will be obsoleted in an upcoming release.

Troubleshooting Information

- JSE
  - ❑ XJIM
    - — On a low-resolution display, customize window is cut off by default. Specify 14-dot font with `-fn` option or `XJim*fontList` resource.
    - — If you use ‘KANA’ input (not ‘ROMAJI’ input) as the key input

method at ‘YOMI’ input, and you input a ‘KANA’ character and ‘HANDAKUTEN’ or ‘DAKUTEN’ successively, the input method server does not compose ‘KANA’ with ‘DAKUTEN’ or ‘HANDAKUTEN’ as one character, but displays the ‘KANA’ character and ‘DAKUTEN’ or ‘HANDAKUTEN’ symbol. In this case, you should make the composite character using ‘ZENKAKU-HIRAGANA’ conversion (press **Shift** + **F5** key), or ‘ZENKAKU-KATAKANA’ conversion (press **F6** key).

❑ EGBridge

Closing the EGBridge main window during Kana-Kanji conversion on `hpterm` may also close `hpterm`. You should finish conversion before closing the EGBridge main window.

❑ IMS common (XJIM/ATOK8/EGBridge/VJE-gamma)

— Window focus sometimes cannot be moved by **Meta(Alt)-Tab** key if applications use `XIMStatusNothing` and they overlap each other with KANJI-ON state. To avoid this problem, set `stackChange` resource to `False` as follows:

XJIM	`XJim*stackChange: False`
ATOK8	`Atok8*stackChange: False`
EGBridge	`EGIms*stackChange: False`
VJE	`Vje*stackChange: False`

See the “Resource” section in each Input Method manual for details.

— On Motif 1.2 and Motif 2.1 applications, the **F10** and **Shift-F10** keys cannot be used as the Japanese input function key because those keys are used to switch focus to the menu bar. To assign these keys to certain functions for IMS, set the following:

— for DIN keyboard: `$ xmodmap -e "keycode 25 = F10"`
— for ITF keyboard: `$ xmodmap -e "keycode 38 = F10"`

— Japanese IMS is not available with X11R4 (including Motif 1.1) applications using PS2-DIN-JIS keyboard if `LANG` is “`ja_JP.SJIS`” or “`ja_JP.eucJP`”. To avoid this problem, set `LANG` “`japanese`” or “`japanese.euc`” when invoking X11R4 (Motif 1.1) applications.

  - — Even if you merge UDC in X font after running the input method server, the server cannot display UDC in the pre-edit and the candidate. You should merge UDC in X font server before running the input method server. Re-login makes sure that the input method server displays UDC on CDE.

- ❑ JIS keyboard

  - — Do not set the `KBD_LANG` shell variable or Motif 1.1 applications will not work with a JIS keyboard.

  - — The **Yen** key on JIS keyboard with X terminal does not work correctly. To use the **Yen** key, execute the command.

```
$ xmodmap -e "keysym yen = backslash bar prolongedsound"
```

- ❑ 106/109 Keyboard

  - — You cannot turn off EGBridge (although you can turn on). The solution is to change the key map file `$HOME/.egb/EGBMap` (for personal use) or `/etc/opt/egb/config/EGBMap` (for system use). You open the key map file with an editor and change the following entry:

    old: `LKONOFF = XK_Henkan XK_Meta_L`
    new: `LKONOFF = XK_Henkan XK_Meta_L XK_Alt_L`

    Then save the updated key map file and restart EGBridge. You can turn EGBridge on/off with the left “**Alt**” key.

- ❑ `udcload`

  - — When UDCs are not arranged in the code order in the UDC file, `udcload` cannot load UDC. Therefore. you should arrange UDCs in the code order. UDCs generated by `xudced` have no problem because `xudced` generates UDSs arranged in code order.

- • KSE

  - ❑ `xk0input`

    Xkim is not available with X11R4 (including Motif 1.1) applications using PS2-DIN keyboard if `LANG` is `ko_KR.eucKR`. To avoid this problem, set `LANG` to `korean` when invoking X11R4 (Motif 1.1) applications.

- • ASE Common

- ❑ xudced (UDC editor)

  When you select “Search...” in the main menu “Edit”, you cannot specify the character directly. Only the Index number can be specified to search a character.

Software Availability in Native Languages

The commands used with this product are the ones supported by the Native Language Support Catalog of HP-UX.

Command and Library Support

To ensure smooth migration from PA to IPF systems, both PA and IPF locale libraries and iconv methods need to be present on the IPF system. See Changed locale and localedef Commands in Chapter 6.

HP-UX Software Transition Kit (STK)

The HP-UX Software Transition Kit (STK) includes tools and documentation to help you successfully transition to the Itanium architecture of HP-UX 11i Version 1.5. To transition your software and scripts, you may have to resolve issues such as data model and API changes. Many tools are available to help you resolve these issues. API file scanners are provided in the HP-UX 11.x STK, while other tools are part of the HP-UX operating system, are included in HP-UX language products, or are supplied by third parties.

The HP-UX 11i Version 1.5 (B.11.20) STK provides step-by-step instructions for performing transitions, a complete set of background and technical documents, and file scanners to help you identify and resolve any required API changes in your source files.

In these types of source files:

- C and C++ programs
- Fortran programs
- COBOL programs
- scripts
- Makefiles

The HP-UX 11i Version 1.5 (B.11.20) STK file scanners can help you locate and fix any changed or obsolete:

- functions
- commands
- path names
- macros
- structures and structure members
- header files
- language keywords
- libraries
- variables

One of the HP-UX 11i Version 1.5 (B.11.20) STK file scanners, `scansummary`, helps you plan your transition by summarizing the number and type of API impacts in your source files. The other tool, `scandetail`, helps you resolve those impacts by identifying the file name and line number in your source where each impact occurs. Both tools provide links to more detailed information about each impact. The file scanners can also identify opportunities for using some enhanced features of HP-UX 11.x.

To use the HP-UX 11i Version 1.5 (B.11.20) STK file scanners, you must install the STK-TOOLS fileset. The HP-UX 11i Version 1.5 (B.11.20) STK is available free of charge via the web:

http://devresource.hp.com/STK/

Check this web site often for updated content.

Aries - PA-RISC Binary Support

HP-UX 11i Version 1.5 (B.11.20) includes Aries: HP's transparent binary compatibility solution for HP-UX/PA-RISC applications on HP-UX/Itanium platforms. Aries allows PA-RISC applications written for HP-UX releases up to (and including) HP-UX 11i (B.11.11) to run on Itanium-based systems running HP-UX 11i Version 1.5 (B.11.20). Aries supports both 32-bit and 64-bit PA-RISC applications.

Aries is composed of 4 shared libraries.

- `aries32.so`
- `pa_boot32.so`
- `aries64.so`
- `pa_boot64.so`

The following applications and/or behaviors are not supported:

- emulation of debuggers
- emulation of privileged PA-RISC instructions
- HP-UX/IPF applications that attempt to use HP-UX/PA-RISC shared libraries or vice-versa
- correct operation of timing-dependent applications

Aries Documentation

Further information about Aries can be found on the web at http://devresource.hp.com/STK/Aries.html and on the *aries* (5) manual page.

Netscape Communicator

HP-UX 11i Version 1.5 (B.11.20) includes Netscape Communicator Version 4.7x, which includes Netscape's popular web browser Navigator, as well as Messenger and Composer. Communicator offers the complete set of tools for browsing dynamic web content plus complete email capability.

Netscape 4.77 has been observed to occassionally experience problems running on HP-UX 11i Version 1.5 (B.11.20) where the browser stops responding, waiting on the `dns_lookup` subprocess. If you experience DNS lookup problems in Netscape Communicator, please use this workaround - set the `MOZILLA_NO_ASYNC_DNS` environment variable as defined below:

Table 8-2

Shell	**Command**
csh, tch	**`setenv MOZILLA_NO_ASYNC_DNS True`**

sh, bash, ksh	**`MOZILLA_NO_ASYNC_DNS=True`**
	`export MOZILLA_NO_ASYNC_DNS`

Java 2

HP-UX 11i Version 1.5 (B.11.20) includes the HP-UX Software Developer's Kit (SDK), for the Java™ 2 Platform Version 1.3. It provides the tools for developing and deploying Java applications on HP-UX 11i Version 1.5 systems.

Java 2 Documentation

Java documentation is provided on the HP-UX 11i Version 1.5 Instant Information CD and on the web at:

http://hp.com/go/java.

The new features of the standard version 1.3 of the Java 2 platform and performance, tool support, and tool enhancements can be found at

http://java.sun.com/products/jdk/1.3/docs/relnotes/features.html

The complete list of features of the standard version 1.3 can be found at

http://java.sun.com/products/jdk/1.3/docs/index.html#guide

The packages included in the APIs core to the Java 2 platform are described at

http://java.sun.com/products/jdk/1.3/docs/api

Apache Web Server

The Apache Web Server for HP-UX Version 1.3.12 is an HTTP/1.1 compliant server which implements the latest protocols, including HTTP/1.1 (RFC2616). The server includes software developed by the Apache Group for use in the Apache HTTP server project. It can be customized by writing software *modules* using the Apache module API.

This release of the Apache Web Server includes pre-compiled binaries that have been preconfigured to run on HP-UX 11i Version 1.5 (B.11.20) and later releases. It is a 32-bit product with 128-bit strong encryption. Apache SSL connections use a default dummy certificate provided by HP.

NOTE You must obtain certificates from HP authorized agencies for commercial purposes, and make the necessary changes in `httpd.conf` to use the desired certificates.

Apache Documentation

Further information on the Apache Web Server can be found at http://www.hp.com/go/webservers. The online Apache release notes that come with the Apache software are available at `/opt/apache/apache.release.notes`. For updated information on the Apache Web Server for HP-UX, see:

http://software.hp.com/

For more information on the Apache Software Foundation, see:

http://apache.org.

OpenGL (OGL) Graphics API

OGL provides both a runtime and a development environment for applications using the OpenGL 3D graphical interface. In HP-UX 11i Version 1.5 (B.11.20),

- there is no support for local graphics devices, only remote graphics via network protocols will be available.
- native code Itanium-based libraries and executables will be provided
- no support for threads

No 3D graphics hardware will be available on any HP-UX Itanium-based systems in the near future. Therefore device drivers that are normally included in the OGL product will be omitted. Applications will be limited to displaying 3D graphics remotely, over the network, to systems that have graphics capability.

OGL Compatibility

No changes to source code will be necessary in applications. IPF native code applications compiled on HP-UX 11i Version 1.5 (B.11.20) will need to be recompiled in a subsequent release. When PA libraries and executables are released, OGL applications currently running on HP-UX PA systems will run on HP-UX Itanium-based systems without any modifications.

Index