

# **Developer's Toolkit Release Notes for HP-UX 10.30**



**HP Part No. B3393-90006  
Printed in USA July 1997**

**E0797**

---

## Legal Notices

The information contained 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 liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Copyright © 1997 Hewlett-Packard Company.

This document contains information which is protected by copyright. All rights are reserved. Reproduction, adaptation, or translation without prior written permission is prohibited, except as allowed under the copyright laws.

*Hewlett-Packard Company  
Technical Computing Center  
1000 NE Circle Boulevard  
Corvallis, Oregon 97330*

**Restricted Rights Legend.** Use, duplication or disclosure by the U.S. Government Department of Defense is subject to restrictions as set forth in paragraph (b)(3)(ii) of the Rights in Technical Data and Software clause in FAR 52.227-7013.

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Use of this manual and flexible disk(s) or tape cartridge(s) supplied for this pack is restricted to this product only. Additional copies of the programs can be made for security and back-up purposes only. Resale of the programs in their present form or with alterations, is expressly prohibited.

**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.

**Trademark Acknowledgment.** © 1985—86, 1988, 1991, 1994 X Consortium, Inc. *X Window System* is a trademark of X Consortium, Inc.

OSF, OSF/1, OSF/Motif, Motif, and Open Software Foundation are trademarks of the Open Software Foundation in the U.S. and other countries.

UNIX is a trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

---

## Printing History

July 1997 . . . . . Edition 1

The printing date will change when a new edition is printed. Minor changes may be made at reprint without changing the printing date. The manual part number will change when extensive changes are made.



# Contents

---

<b>1. Announcements</b>	
Product . . . . .	1-1
xterm Superseded by dtterm . . . . .	1-1
VueHelp Libraries Removed . . . . .	1-1
Obsolete Functions . . . . .	1-2
Changes in Keyboard Functionality . . . . .	1-2
Rendering International Characters . . . . .	1-3
<b>2. What's In This Version</b>	
Features . . . . .	2-1
Libraries for Motif and the X Window System Development . . . . .	2-1
New Features and Updates . . . . .	2-2
X11 R6.1 Extensions . . . . .	2-2
Thread Safe and Thread-Restricted Libraries . . . . .	2-2
Technical Print Service . . . . .	2-2
X Print Extension Library . . . . .	2-3
Print Dialog GUI . . . . .	2-3
Print Dialog Manager . . . . .	2-3
Localizing Applications that use DtPrintSetupBox . . . . .	2-3
Documentation . . . . .	2-4
CDE Development Environment . . . . .	2-4
Filesets . . . . .	2-5
Documentation . . . . .	2-5
PEX 5.1 Version 4 . . . . .	2-6
Supported Systems . . . . .	2-6
Installation Verification . . . . .	2-6
Checking the Device Files . . . . .	2-7
Read the <b>ReleaseNotes</b> File for Time-Critical Information . . . . .	2-7
Using Example Programs, Demonstrations, and Utilities . . . . .	2-7
PEXlib Reference Pages . . . . .	2-8
X11R6 Athena Widget Set . . . . .	2-8
Imake . . . . .	2-9
Digital Video . . . . .	2-9
Supported Systems . . . . .	2-10
Camera Hardware . . . . .	2-10
VideoOut Hardware . . . . .	2-10
Using Example Programs and Utilities . . . . .	2-10
Filesets . . . . .	2-11
Documentation . . . . .	2-11

<b>3. Known Problems and Bugs</b>	
CDE Threads and Signal Handling . . . . .	3-1
Help System XPM Graphics . . . . .	3-2
Application Conflicts With DtHelp Searches . . . . .	3-2
<b>4. Compatibility Information and Installation Requirements</b>	
Installation . . . . .	4-2
CDE Developer's Toolkit Installation . . . . .	4-2
Linking Mixed Shared and Archived Libraries . . . . .	4-2
Dynamic Loading of Versioned Libraries . . . . .	4-3
X11 Release 6 libXt Dependencies . . . . .	4-3
<b>5. What's Fixed in this Version</b>	
Patch Releases . . . . .	5-1
<b>6. Questions and Answers</b>	
Writing An Internationalized Motif Application . . . . .	6-1
Prerequisites . . . . .	6-1
Sample Application . . . . .	6-2
General Guidelines . . . . .	6-2
<b>7. What Manuals are Available for This Version</b>	
Developer's Toolkit Documentation . . . . .	7-1
Online Documentation . . . . .	7-3
Related Technical Documentation . . . . .	7-3
<b>8. Software Availability in Native Languages</b>	

## Announcements

---

### Product

This is a revision of the Developer's Toolkit for HP-UX 10.30.

---

### xterm Superseded by dtterm

This release of HP-UX includes both the X11 Release 5 and Release 6 versions of xterm, delivered in the following directories as unsupported software:

- R5: /usr/bin/X11/xterm
- R6: /usr/contrib/bin/X11/xterm

The functionality of xterm is superseded by the industry-standard dtterm terminal emulator, making xterm redundant. The Release 5 version of xterm will not be shipped with the next release of HP-UX; it is being provided with this release to facilitate the transition of users from xterm to dtterm.

---

### VueHelp Libraries Removed

The HP VUE Help Developer's Kit has been removed from the HP-UX 10.30 Developer's Toolkit. The VueHelp library (`libXvh`) is supported only for use with the X11 Release 5 development environment.

Application developers using VueHelp will need to port their applications to utilize the industry-standard CDE Help API, `libDtHelp`, which is supported for use with both Release 5 and Release 6 of the X11 development environment.

---

## Obsolete Functions

XHP functions for internationalized keyboard input have been removed. These HP proprietary functions are no longer needed since X11R5 and R6 provide public, standard APIs that provide equivalent functionality. Specifically, the following functions have been removed:

```
XHPConvertLookup()
XHPGetEurasianCvt()
XHPInputChinese_s()
XHPInputChinese_t()
XHPInputJapanese()
XHPInputKorean()
XHPInputRoman8()
XHPInputISO7sub()
XHPNlioctl()
```

Instead, applications should use the following X11R6 routines:

```
XOpenIM()
XCloseIM()
XGetIMValues()
XCreateIC()
XDestroyIC()
XSetICFocus()
XUnsetICFocus()
XmbResetIC()
XIMOfIC()
XSetICValues()
XGetICValues()
```

The new X11R6 functionality gives applications greater control and more flexibility than the HP proprietary functions.

---

## Changes in Keyboard Functionality

`XHPSetKeyboardMapping()` and two related functions, `XHPRefreshKeyboardMapping()` and `XHPSetKbdMapInit()`, are obsolete. These functions were used to change an application's local copy of the keymap in order to emulate a particular national language keyboard. These functions only worked for HP HIL keyboards. Although the functionality has been removed, stub APIs are provided so that older applications may still compile.

In HP-UX 9.\*, many applications (such as `vuepad` and `hpterm`) use `XHPSetKeyboardMapping()` to set the keyboard language by passing the value of the `KBD_LANG` environment variable. As of 10.0, HP VUE no longer sets the `KBD_LANG` environment variable. Clients and libraries have removed the call to `XHPSetKeyboardMapping()` since its behavior is not reliable in a mixed DIN and HIL keyboard environment.



---

## Rendering International Characters

In X11R3 and X11R4, support for rendering Asian characters was done through a proprietary mechanism called the associate font mechanism. In HP-UX 9.\*, rendering of Asian character data requires that two fonts be opened to correctly render all Asian characters. When an Asian font was opened with `XLoadFont()` or `XLoadQueryFont()`, the font was checked for the existence of an associate font property. If this property existed, an additional font was opened and maintained transparent to the application. All calls to calculate metrics or to perform rendering with the font caused the implementation to also use the associate font when present.

With support for 3-byte EUC and 4-byte EUC with HP-UX 10.0, rendering Asian character data often requires that more than two fonts be opened and used when rendering Asian character data. The old associate font mechanism could not be reliably expanded to handle these cases. In addition, X11R5 and Motif 1.2 provide alternate, standard mechanisms for rendering internationalized text data. The associated font mechanism won't be supported on future releases of X11R5 and not at all in X11R6.

Applications directly calling the X library to render international text should replace their calculation of font metrics and their calls to `XDraw*`, `XLoadFont()`, and `XLoadQueryFont()` with the following:

```
XCreateFontSet()
XExtentsOfFontSet()
XmbDrawString(), XwcDrawString()
XmbDrawImageString(), XwcDrawImageString()
XmbDrawText(), XwcDrawText()
XmbTextEscapement(), XwcTextEscapement()
XmbTextExtents(), XwcTextExtents()
XmbTextPerCharExtents(), XwcTextPerCharExtents()
```

Applications and users that set Motif fontList resources to obtain internationalized text data should modify those font resources to specify font sets. As of Motif 1.2, a fontList resource can contain any combination of font structs and font sets. A font set is specified in a fontList resource by adding a ":" to the end of the fontList entry. For a font set, a fontList entry can be one or more ";" separated base names. A base name is either a font name (such as `jpn.8x18`) or an X Logical Font Description (XLFD). A typical fontList setting for Japanese might be:

```
app_name*Text.fontList: *-mincho*-18-*
```

The ":" character at the end of the fontList resource specification is mandatory to use font set technology.

For Motif to use font sets correctly in an application, either `setlocale()` or `XtSetLanguageProc()` must be called prior to initializing the toolkit.



## What's In This Version

---

### Features

The HP-UX Developer's Kit provides information needed to develop applications using the following technologies on HP-UX 10.30 systems.

- OSF/Motif 1.2
- X11 R6 Xlib
- X11 R6 Xt Intrinsics
- X11 R6 Athena Widget Set
- X11 R5 Xlib\*
- X11 R5 Xt Intrinsics\*
- Imake
- Common Desktop Environment 1.0
- Audio
- Imaging
- Digital Video (Video Out, MPEG, and Camera)
- Technical Print Service
- HP PEX 5.1 Version 4
- Starbase

The X11 R5 development libraries have been replaced with the R6 libraries. However, the X11 R5 runtime libraries remain part of the product allowing applications built with earlier versions to execute correctly.

### Libraries for Motif and the X Window System Development

The default X Window System development environment for this release of HP-UX is X11 Release 6.1.

Since the Release 6 environment is the default (with libraries and header files in `/usr/lib/` and `/usr/include/`), developers wanting to use the Release 5 environment must include the following options on the compiler command line:

```
-I/usr/include/Motif1.2 -I/usr/include/X11R5  
-L/usr/lib/Motif1.2 -L/usr/lib/X11R5
```

Notice that the version of the Motif library located in `/usr/lib/Motif1.2` is for linking to X11 Release 5 libraries only (`libX11` and `libXt`). The Motif library associated with `/usr/lib/libXm.sl` and `/usr/lib/Motif1.2_R6` is for linking to X11 Release 6 libraries only.

Also note that the Motif library continues to be based on Release 1.2. This library has been upgraded to support the CDE environment and continues to receive repairs for defects.

---

## New Features and Updates

The Developer's Toolkit includes these new components:

- X11 R6.1 Extensions
- Thread Restricted Libraries
- Technical Print Service

### X11 R6.1 Extensions

New extensions include:

DOUBLE-BUFFER (DBE)	The Double Buffer Extension (DBE) provides a standard way to utilize double-buffering which allows flicker-free animation.
ICE X Rendezvous	The Inter-Client Exchange protocol (ICE) specifies a generic communication framework for data exchange between arbitrary clients. The ICE X Rendezvous protocol is one standard protocol by which two clients who have connections to a common X server can rendezvous.
RECORD	RECORD is an X protocol extension that supports the recording of all core X protocol and arbitrary X extension protocol.
XKEYBOARD (XKB)	XKB provides detailed keyboard descriptions and enhanced keyboard functionality, including support for the ISO 9995 keyboard model.

The Open Group Web site (<http://www.x.org>) provides information about the X Window System. A variety of documentation is available including the X11 R6.1 Release Notes and man pages.

### Thread Safe and Thread-Restricted Libraries

The libX11 library (X11 R6 stack) and libXp library (Technical Print Service stack) are both thread safe.

All other Toolkit libraries are thread-restricted. This means library entry points may be safely called by only a single thread of a multi-threaded application.

### Technical Print Service

The Technical Print Service is a network transparent printing system which allows X applications to generate and send X imaging data to a non-display device such as a printer. It provides an X print server that is capable of generating page description language (PDL) output. Applications integrated with the Technical Print Service can send printed output to a variety of printers regardless of the associated page description language. The Technical Print Service provides print drivers for postscript, PCL, and raster images.

---

**Note**            The Technical Print Service is an implementation of the X Print Service technology developed by Hewlett Packard for the X Consortium.

---

The X Print Service is a collection of components derived from the X 11 Window System, Motif, and Common Desktop Environment (CDE) technologies. For application developers, the main components include:

### **X Print Extension Library**

The X Print Extension library provides protocol extensions to allow an application to generate print “jobs”, “documents”, and “pages”, and to manage a Print Context.

### **Print Dialog GUI**

The Print Dialog provides a graphical interface that allows a user to select printers and their associated X print servers. It also presents generic printer and job setup options and a hook to the Print Dialog Manager. This dialog can be customized to include application-specific information.

### **Print Dialog Manager**

The Print Dialog Manager is a process which provides application print dialogs for setting printer-specific options. In general the Print Dialog Manager is managed by the Print Dialog GUI, however, applications may intervene.

### **Localizing Applications that use DtPrintSetupBox**

For the locales it supports, the DtPrintSetupBox widget obtains localized strings for its GUI labels and messages from its own message catalog. As such, developers do not need to provide localized strings for the DtPrintSetupBox widget. However, when the `DtNprintSetupMode` resource is set to `DtPRINT_SETUP_XP` (the default) the DtPrintSetupBox installs default procedures for various operations. Some of these default procedures (`printerInfoProc`, `selectPrinterProc`, and `selectFileProc`) in turn present dialogs that utilize stock Motif widgets. Motif does not utilize a message catalog, so in order to present localized labels in these widgets, some additional lines need to be added to the application’s `app-defaults` file. An example of the additional resource file lines (with English values) is shown below:

```
! Get XmFileSelectionBox resources via the Dt app-defaults file
#include "/usr/dt/app-defaults/C/Dt"

*XmMessageBox*okLabelString:      OK
*XmMessageBox*cancelLabelString:  Cancel
*XmMessageBox*helpLabelString:    Help

*XmSelectionBox*okLabelString:     OK
*XmSelectionBox*cancelLabelString: Cancel
*XmSelectionBox*helpLabelString:   Help
```

## Documentation

The Technical Print Service Developer's Kit does not include a programmer's manual. The following man pages document the Technical Print Service:

DtCreatePrintSetupBox(3)	XpGetPageDimensions(3)
DtCreatePrintSetupDialog(3)	XpGetPdmStartParams(3)
DtPrintCopySetupData(3)	XpGetPrinterList(3)
DtPrintFillSetupData(3)	XpGetScreenOfContext(3)
DtPrintFreeSetupData(3)	XpPutDocumentData(3)
DtPrintResetConnection(3)	XpQueryExtension(3)
DtPrintSetupBox(3)	XpQueryScreens(3)
DtPrintSetupProc(3)	XpQueryVersion(3)
XmPrintPopupPDM(3)	XpRehashPrinterList(3)
XmPrintSetup(3)	XpSelectInput(3)
XmPrintShell(3)	XpSetAttributes(3)
XmPrintToFile(3)	XpSetContext(3)
XmRedisplayWidget(3)	XpSetLocaleHinter(3)
XpCancelDoc(3)	XpStartJob(3)
XpCancelJob(3)	XpStartPage(3)
XpCancelPage(3)	dtpdm(1M)
XpCreateContext(3)	dtpdmd(1M)
XpDestroyContext(3)	tps(5)
XpEndDoc(3)	tpsconfig(1M)
XpEndJob(3)	tpsconvert(1M)
XpEndPage(3)	tpsedit(1M)
XpFreePrinterList(3)	tpsdocument(4)
XpGetAttributes(3)	tpsjob(4)
XpGetContext(3)	tpsprinter(4)
XpGetDocumentData(3)	tpsshowmodel(1M)
XpGetLocaleHinter(3)	
XpGetOneAttribute(3)	

Technical Print Service Release Notes are located in the `/usr/share/doc/TechPrintServ/C` directory. For other documentation sources refer to Chapter 7.

---

## CDE Development Environment

The Developer's Toolkit provides six libraries that allow you to create applications for the Common Desktop Environment. Each library contains several components:

### ■ Desktop Services Library - libDtSvc

- Desktop Initialization API (DtInitialize)
- Action Invocation API (DtAction)
- Data Typing API (DtDts)
- Drag and Drop API (DtDnd)
- Screen Saver API (DtSaver)
- Session Management API (DtSession)
- Workspace Management API (DtWsm)

### 2-4 What's In This Version

- **Help Widget Library - libDtHelp**
  - Help Utility API (DtHelp)
  - HelpDialog Widget API (DtHelpDialog)
  - HelpQuickDialog Widget API (DtHelpQuickD)
- **Terminal Widget Library - libDtTerm**
  - Terminal Widget API (DtTerm)
- **Desktop Widget Library - libDtWidget**
  - Editor Widget API (DtEditor)
  - ComboBox Widget API (DtComboBox)
  - MenuButton Widget API (DtMenuButton)
  - SpinBox Widget API (DtSpinBox)
- **Calendar Library - libcsa**
  - Calendar API (csa)
- **ToolTalk Messaging Library - libtt**
  - ToolTalk Toolkit API (tttdt, tttk)
  - ToolTalk Messaging API (tt)

## Filesets

The CDE development environment consists of five filesets:

- CDE-DEMOS (example programs)
- CDE-INC (header files)
- CDE-HELP-PRG (example program and sample files)
- CDE-MAN-DEV (man pages)
- CDE-PRG (archive libraries)

## Documentation

The Common Desktop Environment man pages reside in `/usr/dt/man`. Refer to the *CDE Programmer's Overview* for instructions to set the `MANPATH` environment variable.

For a list of CDE application developer's documentation see Chapter 7.

---

## PEX 5.1 Version 4

The Version 4 release of HP PEX 5.1 brings high-performance 3D graphics through the X server to the HP9000 3D Color Workstations. The HP PEX server and HP PEXlib client software extend the capabilities of the X Window System and provide integrated 3D distributed graphics in an open system environment.

The Hewlett-Packard implementation is equipped with a number of features for enhanced performance and interoperability. These features and their operation are explained in the book *HP PEX Implementation and Programming Supplement*. Among other features, the HP PEX product includes an integrated, on-line hypertext tutorial.

This document contains important information about the HP PEX product that is essential for those who program using HP PEX. It provides information on some known behaviors that could affect the outcome of your workstation setup and subsequent operation of HP PEX programs.

If, for any reason, you must reinstall either the X Window System or the HP-UX operating system, you must also reinstall HP PEX. See the section “Information for System Administrators” in Chapter 2 of the *HP PEX Implementation and Programming Supplement*.

### Supported Systems

This version of HP PEX is supported on the HP 9000 3D Color Workstations running HP-UX 10.20 or later and HP VUE 3.0 or HP-CDE 1.0. However, you can choose not to use HP VUE or HP-CDE and use only the X Window System environment instead.

The following workstations and graphics products are supported for PEXlib clients:

- HP 9000 Model 705, 710, 712, 715, 720, 725, 730, 735, 750, and 755 workstations
- On-board graphics in HP 9000 Model 705, 710, 712, 715, 725, B-Class and C-class workstations
- CRX
- Dual CRX
- CRX-24
- CRX-24Z
- CRX-48Z
- HCRX-8
- HCRX-24 with or without the HCRX-8 Z-Buffer and Accelerator
- VISUALIZE-8
- VISUALIZE-24
- VISUALIZE-48/VISUALIZE-48XP
- VISUALIZE-EG

### Installation Verification

If you are setting up a new workstation or server, or have already installed the HP-UX operating system, all HP PEX software may already be preloaded for you. If the HP PEX developer's software is not loaded on your system, you will have to install it before you can use it. Since the HP PEX Developer's Kit is part of the HP-UX Developer's Toolkit, follow the installation instructions for the HP-UX Developer's Toolkit, and include the PEX-related filesets for installation.



To determine whether or not the PEX 3D run-time libraries are installed, run `/opt/graphics/PEX5/demos/verify_install`. If HP PEX resides on your system, `verify_install` will create a window and draw a rotating “PEX cube” in it—a cube with three letter-shaped holes drilled in it, one per face, one in each dimension. Depending upon your hardware and color map settings, you may experience “technicolor”—the distracting effect of displayed objects changing colors as color maps with different contents are installed. (Color map installation is typically under control of a window manager, and is often triggered by moving the window focus from one window to another.) If there are problems running this test program, the error messages will point you to a solution.

To determine whether you have the HP PEX Developer’s software installed, check for the existence of the include file `/opt/graphics/PEX5/include/X11R6/X11/PEX5/PEX.h`.

## Checking the Device Files

Devices specified in X server screen configuration files must correspond to existing device files with appropriate permission. If you don’t already have the appropriate device file, you must create it using SAM (the System Administration Manager).

## Read the ReleaseNotes File for Time-Critical Information

Before you begin using the HP PEX product, it is important to check on-line files. You will find time-critical information in the HP PEX product’s **Release Notes** file, `/opt/graphics/PEX5/10.0_Rel_Notes`. This includes information about utility programs used for setting up workstations or servers properly and which provide interoperability of programs.

Also, look in the **README** files that may be found in the various subdirectories of the HP PEX product’s main directory `/opt/graphics/PEX5`.

## Using Example Programs, Demonstrations, and Utilities

The directory `/opt/graphics/PEX5/` contains various example programs and demonstrations listed below. It also contains important utilities for running these programs with PEXlib extensions supported by Hewlett-Packard. See the **README** file in this directory for more information.

- The program examples from the *PEXlib Programming Manual* by O’Reilly & Associates are installed in the file `/opt/graphics/PEX5/examples/OReilly/`. A **README** file in this directory contains instructions for using these programs. It specifically describes the use of `book_utils.h`, which contains the macros and declarations used by these O’Reilly programs. An important fact to note about the O’Reilly utility procedure source, `book_utils.c`, is that the source shipped with HP PEX has been modified by Hewlett-Packard to provide for improved interoperability.

We recommend that you look at the HP-modified `book_utils.c` file and at the other utility files for ways to improve the interoperability of your programs and of programs you port from other vendor’s platforms.

- HP PEXlib demonstrations are located in the `/opt/graphics/PEX5/demos/` directory. A **README** file in the directory explains how to use the example programs.

- Example programs developed by Hewlett-Packard are installed in the `/opt/graphics/PEX5/examples/hp/` directory. `README` files under this directory explain how to use the example programs.
- Example programs that demonstrate portability and functionality of the CGE extensions are in `/opt/graphics/PEX5/examples/cge`.
- Programming utilities for visual selection and color map creation provided by the CGE PEX vendors are in `/opt/graphics/PEX5/utilities/cge`.
- Other important programming utilities are provided in the directory `/opt/graphics/PEX5/utilities/`.

By listing the files in these directories, you will see utilities designed for improving the interoperability of programs between workstations of different manufacturers, resolving color map issues, selecting visuals and creating windows for HP workstations. These utilities are documented in a `README` in the same directory.

## PEXlib Reference Pages

The reference pages for PEX5.1v4 are presented via HP Help, the hypertext-capable on-line help system. (This on-line help works even if you're only using regular X windows; you don't need to be running HP VUE.) You can start the on-line help system by executing `/opt/graphics/PEX5/help5.1/bin/pexman`.

The Hewlett-Packard PEX product supplements the standard PEXlib reference pages with reference pages on the HP utilities mentioned above. These are also accessed via `pexman`.

---

## X11R6 Athena Widget Set

The Developer's Toolkit includes libraries for the Athena Widget Set (`Xaw`) and Miscellaneous Utilities (`Xmu`) as part of the X11R6-PRG-CTRB fileset. Both libraries (`libXaw.a` and `libXmu.a`) are located in `/usr/contrib/X11R6/lib`. Library header files reside in `/usr/contrib/X11R6/include/X11/Xaw` and `/usr/contrib/X11R6/include/X11/Xmu`.

*The Athena Widget Set - C Language Interface*, an X Consortium document, is available in postscript format and can be found in `/usr/contrib/X11R6/doc`.

---

## **Imake**

The HP-UX 10.30 development environment provides the Imake configuration management tool. This version is based on the X11 R6 implementation released by the X Consortium.

---

## **Digital Video**

The Digital Video Developer's Kit contains libraries, header files, example code, and documentation to enable you to create applications that play MPEG video files, control the HP A4049A Desktop Video Hardware, display or capture video camera images and convert the screen into composite video so it can be captured on a VCR. The kit also includes tools to set up and test the A4049A hardware, as well as sample camera and MPEG viewer applications.

---

**Note**

Some systems may not have the symbolic links in `/opt/video/lib` that point `libyuv2.sl` to `libyuv2.1` and `libv1Video.sl` to `libv1Video.1`. If this is the case on your system, you will need to use the following commands (one time only) as user root:

```
cd /opt/video/lib
ln -s libyuv2.2 libyuv2.sl
ln -s libv1Video.2 libv1Video.sl
```

---

## Supported Systems

The shared libraries included in this kit are designed for use on HP PA-RISC platforms running HP-UX version 10.10 and beyond; however, applications that use these libraries can display to non-HP platforms.

## Camera Hardware

The Digital Video kit can control and process the live video data produced by HP A4049A Desktop Video Hardware. The A4049A is an external SCSI device that converts digital, S-Video, or NTSC/PAL composite video data into a stream of H.261-format video data packets. To connect a camera or other video source to your system using the A4049A interface, refer to the installation instructions that are included with the hardware.

Installation of the Digital Video Developer's Kit does not automatically create a camera device file. Before you can use A4049A for the first time, you must run the `/opt/video/bin/camSetup` script to create the required device file. This provides an easy way to create the device file by answering a few questions during the setup.

## VideoOut Hardware

The Digital Video kit provides control of the A4248A Video Out board. `/bin/graphinfo` tells you if it is installed. See the online manpage `voOpen` for more information.

## Using Example Programs and Utilities

The Digital Video Developer's Kit comes with several sample applications that demonstrate digital video features. Among them are:

- `xcam`, a GUI camera viewer utilizing a Motif camera widget. Build this using the source code in `/opt/video/DVC_SDK/examples/xcam` by running `make` and following the instructions in `/opt/video/DVC_SDK/Readme`.
- `/opt/video/bin/camcheck`, a simple camera test tool
- `player`, a GUI MPEG viewer utilizing MPEG widgets. Build the MPEG widgets and the sample program using the source code in `/opt/video/MPEG` and its subdirectories by running `make` in `/opt/video/MPEG`.
- `gui`, a GUI video out control program. Build it with the Makefile in `/opt/videoout/SDK`.

Sample programs can be found under `/opt/video/DVC_SDK/examples/xcam`, `/opt/video/MPEG/examples` and `/opt/videoout/SDK/examples`. Refer to the Readme files and/or other files in the `/opt/video/DVC_SDK`, `/opt/video/MPEG`, and `/opt/videoout/SDK` directories for details on how to build and use each example.

## 2-10 What's In This Version

## Filesets

The Digital Video Developer's Kit consists of several filesets. Those pertaining to the A4049A Desktop Video Hardware and development of camera applications are:

- DVC-SRV (camera server, configuration files, setup and test tools)
- DVC-SHLIBS (camera libraries)
- DVC-PRG (sample applications)
- DVC-PRGMAN (man pages)

Those pertaining to MPEG playback and application development are:

- DVIDEO-FILES (video clips and miscellaneous notes)
- DVIDEO-PGMAN (man pages)
- DVIDEO-PRG (sample applications)

Those pertaining to Video Out application development are:

- VIDEOOUT-PRG (sample applications)
- VIDEOOUT-PGMAN (man pages)
- VIDEOOUT-SHLIBS (libraries)

## Documentation

The Digital Video Developer's Kit does not include a programmer's manual. For a brief introduction, refer to the Readme files in `/opt/video/MPEG`, `/opt/video/DVC_SDK`, and `/opt/videoout/SDK`.

Installing the UEDK updates `/etc/MANPATH`, but you need to update the MANPATH variable in your environment to read them. You can do this by logging out and back in or using the following command:

```
MANPATH='cat /etc/MANPATH'
```

All of the man pages in the lists that follow are found in `/opt/video/share/man` or `/opt/videoout/share/man`.

The following man pages document MPEG:

MpMpeg	vlGetErrorStatus
MpMpegDispatchEvent	vlGetEventFromServer
MpMpegError	vlGetEventMask
MpMpegErrorLog	vlGetEventPipeFD
MpMpegErrorPtr	vlGetFrame
MpMpegGetVis	vlGetFrameEventRate
MpMpegLoadFile	vlGetFramePosition
MpMpegNextEvent	vlGetFrameRate
MpMpegNextScale	vlGetLibraryVersion
MpMpegPause	vlGetPipeConnectionStatus
MpMpegPending	vlGetScale
MpMpegPlay	vlGetScaleFactors
MpMpegPlayButton	vlGetVFileAttributes
MpMpegPlayPauseButton	vlGrabFrame

MpMpegRewind	vlLoadVFile
MpMpegScaledFrameSize	vlMirrorVideo
MpMpegSeek	vlOpenDisplay
MpMpegSetScale	vlOpenVideoConnection
MpMpegStop	vlPauseVideo
MpMpegSync	vlPreBufferVideo
MpMpegVideoPtr	vlRefreshVideo
MpYuv	vlResumeVideo
MpYuvInitialSize	vlSetAudioQuality
MpYuvNewSize	vlSetAudioVolume
MpYuvResetSomeVars	vlSetColorAdjust
MpYuvSetScale	vlSetDisplayOffset
vlCloseDisplay	vlSetEventMask
vlCloseVideoConnection	vlSetFrameEventRate
vlDisplayFrame	vlSetFramePosition
vlEnableColor	vlSetFrameRate
vlFreeFrame	vlSetLogModeMask
vlGetAudioCapabilities	vlSetScale
vlGetAudioVolume	vlSetWindow
vlGetColorAdjust	vlSync
vlGetDisplayOffset	vlUnLoadVFile

The following man pages document the Motif Camera Widget:

MpCam	MpCamReadFrame
MpCamAdjustImage	MpCamRecord
MpCamDestroySavedFrame	MpCamRefresh
MpCamDisplayFrame	MpCamSaveCurrentFrame
MpCamErrorString	MpCamSeek
MpCamEventPending	MpCamSetFeature
MpCamFlushQueuedFeatures	MpCamSetQueuedFeatures
MpCamGetFeatures	MpCamSource
MpCamGetState	MpCamStartCamera
MpCamNextEvent	MpCamStillCapture
MpCamPausePlayback	MpCamStopCamera
MpCamPauseRecord	MpCreateCam
MpCamPauseSource	MpCreateMpeg
MpCamQueueFeature	MpCreateYuv

The following man pages document the Video Out API:

voClose	voGetFeature	voSetFeature
voFeatureOptions	voGetFeatures	voSetFeatures
voFlush	voOpen	

## Known Problems and Bugs

---

### CDE Threads and Signal Handling

All of the CDE public libraries are thread-restricted. This means that the libraries can be called by only a single thread within a multi-threaded application.

Two CDE libraries, `libDtTerm` and `libtt`, handle signals in a manner that can affect a threaded application's behavior. Both libraries register a signal handler to catch certain signals. In a threaded environment signal handlers are process wide rather than each thread having its own signal handler. In this case, if an application attempts to set a signal handler for these same signals, undefined behavior will result.

#### Terminal Widget Library (`libDtTerm`)

`libDtTerm` currently takes ownership of both the `SIGCHLD` and `SIGPIPE` signals. The `SIGCHLD` signal is used by `libDtTerm` to update internal information regarding the state of the process spawned within the `dtterm` widget. This implies that an application which links with the `libDtTerm` library, and creates `dtterm` widgets, must not attempt to catch the `SIGCHLD` signal, and should block the signal in all other threads. The `SIGPIPE` signal is used by `libDtTerm` only if logging has been enabled. If an application uses the `libDtTerm` library, but does not take advantage of the logging facilities, then `libDtTerm` will not take control of the `SIGPIPE` handler.

#### ToolTalk Messaging Library (`libtt`)

The `libtt` library currently takes ownership of both the `SIGCHLD` and `SIGPIPE` signals; the ownership is only for a very brief period of time and when finished it will restore the signal handlers to their original states. The `libtt` library may also be used by applications which use the `libDtSvc` public functions.

---

## Help System XPM Graphics

When using an `xpm` graphic with the Help System, it is strongly recommended that you fill out the `xpm` specification completely—that is, specify a color value, a gray value, and a monochrome value for each pixel. This allows the Help System to degrade the `xpm` image to your specifications if it cannot allocate all of the colors for a particular graphic. This may occur, for example, when Help is viewed on a display that does not have full color support. It may also occur if Help is viewed on a system that has other applications which make extensive use of the device's color map. In these instances, if you have not completed the `xpm` specification, the Help System will display the “Missing Graphic” icon instead of your `.xpm` image.

---

## Application Conflicts With DtHelp Searches

The following information only applies if you are not using `Xt[App]MainLoop()` to process all events for an application.

There may be an architectural conflict if an application uses `Xt[App]PeekEvent()` and `Xt[App]ProcessEvent()` to process X events. If the application calls `Xt[App]ProcessEvent()` only when `Xt[App]PeekEvent()` returns true, a DtHelp index search will never complete.

A work procedure is registered when the user selects the Index button in a General Help Dialog. In effect, the 'search' occurs only when the application is not busy doing something else. If `Xt[App]ProcessEvent()` is called only when there is an X event, it (the X event) takes precedence over any work procedure registered.

Therefore the Search work procedure is never called/completed.

The best way around this is to use `Xt[App]MainLoop()` to do your event processing.

If you cannot do this, then register your own work procedure before creating any help dialogs. The work procedure can be very simple (a return FALSE) or do the work you want to do if there is no other X event or work procedure pending. Therefore, when a search occurs, its work procedure will take precedence over your work procedure and X events will take precedence over all.

The key is to always call `Xt[App]ProcessEvent()` so that the search work procedure will occur. If no search is in progress, your work procedure will occur, causing the `Xt[App]ProcessEvent()` to not block.



## Compatibility Information and Installation Requirements

---

Read this entire document and any other Release Notes or READMEs you may have before you begin an installation or update.

If you install all the components included in the HP-UX Developer's Toolkit, you will need approximately 100 megabytes of disk space.

The HP-UX Developer's Toolkit consists of these products:

<code>X11MotifDevKit</code>	This product contains the libraries and functions which allow you to use X1 R6 Xlib and Xt Intrinsics, OSF/Motif and Imake.
<code>CDEDevKit</code>	This contains the libraries and functions for the Common Desktop Environment. There are six libraries: Desktop Services, Help Widget, Terminal Widget, Custom Widget, Calendar Services, and ToolTalk Messaging.
<code>AudioDevKit</code>	This contains the libraries and functions to include audio in your applications.
<code>ImagingDevKit</code>	This contains the libraries and functions to include imaging in your applications.
<code>GraphicsPEX5DK</code>	This contains the libraries and functions to allow you to use PEX graphics.
<code>GraphicsSBaseDK</code>	This contains the libraries and functions to allow you to use Starbase graphics.
<code>DigitalVideoDK</code>	This contains the libraries and functions to allow you to include viewing of MPEG video files in your application. It also provides the means for you to develop GUIs to use with the HP Desktop Video hardware and the VideoOut hardware.
<code>TechPrtSvcDK</code>	This contains the libraries and functions to allow you to integrate the Technical Print Service with your application.

---

## Installation

To install your software, run the SD-UX `swinstall` command. It will invoke a user interface that will lead you through the installation.

Perform any necessary tasks to get the system running. The manuals *Managing HP-UX Software with SD-UX (B2355-90089)* and *Installing HP-UX 10.0 (B2355-90086)* have this information. You may also need to use user documentation for other parts of the system, such as networking, system security, and windowing.

---

## CDE Developer's Toolkit Installation

It is recommended that any previous versions of the CDE Developer's Toolkit be removed before installing the current version of the CDE toolkit.

To remove an existing version of the CDE toolkit from the local system, use this command:

```
swremove CDEDevKit
```

---

## Linking Mixed Shared and Archived Libraries

Application developers should be aware of potential problems associated with linking an archived library below a shared library—that is, resolving shared library external reference symbols with an archived library. Minor changes or bug fixes to a shared library may add external references which were not present at the time an application was linked. If the application does not contain a path to the shared library containing the newly referenced symbols, a fatal application error will result.

In general, if an application uses an archived library, be sure that other libraries containing dependencies on that archived library (those stacked “on top” of the archived library) are also linked archived rather than shared.

### Examples of Stacked Libraries

In these examples, assume that `libB` has dependencies on `libA` (since it uses symbols from `libA`). Also, both `libA` and `libB` have dependencies on the standard `libc`.

#### *Appropriately Stacked Libraries*

All Shared Libraries	<code>libB.sl</code>
	<code>libA.sl</code>
	<code>libc.sl</code>

Archived Library Stacked on Shared Library	<code>libB.a</code>
	<code>libA.sl</code>
	<code>libc.sl</code>

#### *Problematic Stacked Libraries*

Shared Libraries Stacked    libB.sl  
on Archived Library        libA.sl  
                                  libc.a

Shared Library Stacked     libB.sl  
on Archived Library        libA.a  
                                  libc.sl

---

## Dynamic Loading of Versioned Libraries

When developing an application that dynamically loads versioned libraries, the name used for loading a library requires special attention. Applications should not dynamically load versioned libraries with names of the form *library\_name.sl*. Names ending in the `.sl` extension are generally associated with symbolic links to one of several possible versions of a library. Although these links are installed in the development environment, they are not typically present in the run-time environment.

To ensure the correct version of a library is loaded, use the name that includes the version extension of the library executable. Use the file name referenced by the `libA.sl` symbolic link for the library in question. For example, if `libA.sl` is a symbolic link to `libA.1`, then use `libA.1` for dynamic loading.

---

## X11 Release 6 libXt Dependencies

Developers should include “-ISM -ICE” in their Makefile because of Release 6 libXt dependencies on libSM and libICE.



## What's Fixed in this Version

---

### Patch Releases

A description of HP-UX patch releases is available on the World Wide Web. The URL is <http://us.external.hp.com>. You should check this site for patches that apply to HP-UX 10.30.

This information can also be obtained from an ftp server. Use anonymous ftp to access the server [hpatlse.atl.hp.com](ftp://hpatlse.atl.hp.com). Change directories to `hp-ux_patches/s700_800/10.*` to browse patch information.



## Questions and Answers

---

### Writing An Internationalized Motif Application

This section describes how to “internationalize” the graphical user interface of a Motif application.

Internationalization (I18N) is a design methodology that allows you to develop a single version of your software that works correctly for your user in their chosen language. I18N is a shorthand convention for “internationalization” (“I” followed by 18 letters followed by “N”).

To create a fully internationalized application, you must handle and process data correctly—whether the data is input by the user or displayed by your application. I18N impacts the design of your application’s graphical user interface. Your application must be able to handle localized data in these ways: obtain coded characters from the keyboard, display local language text, set local language text values and resources, and exchange text data with other internationalized software.

To accomplish this, you will need to use the I18N routines of the C library. These I18N routines are described in a variety of books available in bookstores. Sandra Martin O’Donnell’s book, *Programming for the World*, is a useful reference.

### Prerequisites

Before you start, your software development environment must meet the following prerequisites:

- C or C++ programming language
- X11R5 (or later) and Motif 1.2
- The locales for which you want to test your software must be installed on HP-UX. In some cases, additional software must be purchased. For example, to obtain the input methods and fonts needed to run your software in Japanese, you must purchase a license for Japanese HP-UX.

- Xlib supports the locales you want your software to support

The X library supplied with HP-UX supports all of the locales supported by HP-UX (see `/usr/lib/nls/config`) except: Arabic (`ar_SA.iso88596`, `ar_SA.arabic8`, `ar_DZ.arabic8`), Hebrew (`iw_IL.iso88598`), and Thai (`th_TH.tis620`).

- Fonts for supported languages are accessible to your X server

Fonts can either be placed in a default search path location or the server search path can be modified by using `xset`.

Fonts for most supported locales are supplied with HP-UX. Fonts for Asian locales are supplied with the language-specific version of HP-UX.

- Input methods for supported locales are available

HP-UX delivers input methods for all supported non-Asian locales. Input methods for Asian locales are supplied with the language-specific version of the operating system.

For many Asian input methods, the input method must be started before the client starts. In some instances, Xlib can launch the input method on behalf of the client. For additional information, refer to the documentation provided with the input method.

## Sample Application

A sample application that demonstrates I18N routines is located in `/usr/contrib/Xm1.2/motif.i18n`.

## General Guidelines

To make sure your software's Motif-based graphical user interface is internationalized, follow these guidelines:

1. Call `XtSetLanguageProc(NULL, NULL, NULL)` immediately after the opening main brace of the application. This causes `setlocale()` to eventually be called for you as: `setlocale(LC_ALL, "")`.
2. Call `setlocale(LC_ALL, "")` immediately after calling `XtSetLanguageProc()`.

This ensures that any locale sensitive operations (such as opening a message catalog) that may occur before the toolkit is initialized, are performed according to the rules of the user's chosen locale.

3. Call `Xt[App]MainLoop()` when looping for events.

`Xt[App]MainLoop` causes input methods to be able to examine events (via a registered filter) before they are dispatched to the application. As an application developer, you don't need to be concerned that filtering is occurring.

4. Store text, formats, and bitmaps used by your application externally. For example, place the data in resource files, message catalogs, or bitmap files. Use I18N functions such as `catgets`, `XmGetPixmap()`, and Xt resource routines to load the data at run time.
5. Use a `Text` or `Textfield` widget to obtain text data input by the user. Both widgets are fully internationalized and will automatically connect and interact with the appropriate input method.

The `Text` and `TextField` widgets are fully ICCCM compliant in Motif1.2. They use the new `[Xmb|Xwc]TextListToTextProperty()` routines to generate text for drag-and-drop and cut-and-paste operations. This ensures proper interpretation of text data during interchange.

---

### Note

Motif does not support true on-the-spot preedit/status mode (preeditCallbacks/statusCallbacks); it supports all other preedit/status styles. For information on preedit and status styles, refer to your X11R5 documentation. `VendorShell` and its subclasses, `topLevelShell` and `transientShell`, will handle any geometry management needed for preedit/status area styles.

---

If you cannot use the `Text` or `TextField` widget to obtain keyboard input, you can use the public (but undocumented) Motif 1.2 `XmIm` APIs. You might choose to do this, for



example, to obtain keyboard input in a DrawingArea widget. You can find the syntax of the XmIm routines in the header file `/usr/include/Motif1.2/Xm/Xm.h`.

The public XmIm APIs are documented in Motif 2.0. The book, *X Windows on the World*, also includes documentation and examples of using the XmIm APIs.

6. When creating CompoundStrings (XmStrings), if you don't need a specific tag use the tag `XmFONTLIST_DEFAULT_TAG`; this forces the XmString functions to use X11R5 (or later) font sets and associated routines to render the compound string. This tag replaces `XmSTRING_DEFAULT_CHARSET`.
7. When setting the title or icon name resource, do *not* set the `titleEncoding` property. If your application calls `XtSetLanguageProc()`, the title encoding is set correctly automatically.

Note that the type for title is `String`, not `XmString`. It is a common error to create a compound string and try to pass it as the title. If you encounter difficulties, check that you are using a simple string (`char*`) as the title and use the `titleEncoding` default value.

8. In all cases, specify a font set in the fontlists for your application and widgets. As long as `XmString*()` detects a font set in the fontlist, the new X11R5 (or later) I18N APIs will be used to render the text. Similarly, as long as the Text or TextField widget detects a font set in the fontlist, the new X11R5 (or later) I18N APIs will be used to render text.

Motif's string to fontlist converter automatically generates a font set if your fontlist specification ends with a ":" (colon) character. The colon is the converter's trigger to call `XCreateFontSet` (a new I18N routine for X11R5) instead of `XLoadQueryFont()`.

For example:

```
my_app -xrm "my_app*fontList: *-18-*:"
```

creates an 18 point font set and places it in the fontList.

9. Make sure you have `$LANG` set correctly when running your internationalized application. Values for `LANG` are not standard and vary from vendor to vendor.

If you are using an Asian locale, make sure that `XMODIFIERS` is set according to the instructions provided from your input method supplier.

### Related Documentation

- *Programming for the World* by Sandra Martin O'Donnell  
ISBN 0-13-722190-8, Prentice-Hall
- *X Windows on the World*  
*Developing Internationalized Software with X, Motif and CDE*  
by Thomas C. McFarland  
ISBN 0-13-359787-3, Prentice-Hall



## What Manuals are Available for This Version

---

### Developer's Toolkit Documentation

The following manuals support the HP-UX Developer's Toolkit. These manuals can be ordered in sets, or most can be ordered individually.

#### Series 700 Documentation Product Numbers

- User Environment manuals (B2355EA #OBL)
  - *Motif 1.2 Programmer's Guide, Vol 6A*, O'Reilly & Associates (B1171-90074)
  - *Motif 1.2 Reference, Vol 6B*, O'Reilly & Associates (B1171-90075)
  - *Motif 1.2 Style Guide*, Prentice Hall, (B1171-90058)
  - *Mastering Motif Widgets* (B1171-90057)
  - *X Toolkit Intrinsic Programming Manual, Vol 4*, O'Reilly & Associates (B1171-90053)
  - *X Toolkit Intrinsic Reference Manual, Vol 5*, O'Reilly & Associates (B1171-90054)
  - *Xlib Programming Manual, Vol 1*, O'Reilly & Associates (B1171-90051)
  - *Xlib Reference Manual, Vol 2*, O'Reilly & Associates (B1171-90052)
  - *HP Xlib Extensions* (B1171-90078)
- Common Desktop Environment manuals (B2355EA #AVM)
  - *CDE Programmer's Overview* (B1171-90105)
  - *CDE Programmer's Guide* (B1171-90106)
  - *CDE Desktop Korn Shell User's Guide* (B1171-90108)
  - *CDE Help System Author's and Programmer's Guide* (B1171-90109)
  - *ToolTalk Programmer's Guide* (B1171-90126)
- Multimedia manuals (B2355EA #AVD)
  - *Using the Audio Developer's Kit* (B2355-90069)
  - *Using the Image Developer's Kit* (B2355-90068)
- Technical Print Service
  - The *Technical Print Service System Administrator's Guide* is provided online. It can be viewed using the CDE Help Manager.
  - The online information is also available in a hardcopy version which can be sent directly to a printer. A PCL and Postscript file can be found in `/usr/share/doc/TechPrintServ/C` directory. This manual is provided online only and cannot be ordered as a separate product.
- Starbase manuals (B2355EA #0BM)
  - *Starbase Technical Addendum* (B2355-90117)
  - *HP-UX Starbase Device Drivers Manual* (B2355-90055)
  - *Starbase Reference* (B2355-90056)
  - *Starbase Graphics Techniques* (98592-90081)
  - *Starbase Display List Programmer's Guide* (B2355-90098)

- *Fast Alpha/Font Manager Programmer's Guide* (B2355-90097)
- *Graphics Administration Guide* (B2355-90122)
- PEX Programming manuals (B2355EA #AVE)
  - *HP PEX Implementation and Programming Supplement* (B3176-90118)
  - *PEXlib Programming Manual* O'Reilly & Associates (B3176-90003)
  - *PEXlib Programming Reference* O'Reilly & Associates (B3176-90004)
  - *Portable Programming with CGE PEX 5.1* (B3176-90042)
  - *PEXlib Developer's README FIRST* (B2355-90630)
  - *Graphics Administration Guide* (B2355-90122)

### Series 800 Documentation Product Numbers

- User Environment manuals (B6205AA #OBL)
  - *Motif 1.2 Programmer's Guide, Vol 6A*, O'Reilly & Associates (B1171-90074)
  - *Motif 1.2 Reference, Vol 6B*, O'Reilly & Associates (B1171-90075)
  - *Motif 1.2 Style Guide*, Prentice Hall, (B1171-90058)
  - *Mastering Motif Widgets* (B1171-90057)
  - *X Toolkit Intrinsic Programming Manual, Vol 4*, O'Reilly & Associates (B1171-90053)
  - *X Toolkit Intrinsic Reference Manual, Vol 5*, O'Reilly & Associates (B1171-90054)
  - *Xlib Programming Manual, Vol 1*, O'Reilly & Associates (B1171-90051)
  - *Xlib Reference Manual, Vol 2*, O'Reilly & Associates (B1171-90052)
  - *HP Xlib Extensions* (B1171-90078)
- Common Desktop Environment manuals (B6205AA #AVM)
  - *CDE Programmer's Overview* (B1171-90105)
  - *CDE Programmer's Guide* (B1171-90106)
  - *CDE Desktop Korn Shell User's Guide* (B1171-90108)
  - *CDE Help System Author's and Programmer's Guide* (B1171-90109)
  - *ToolTalk Programmer's Guide* (B1171-90126)
- Multimedia manuals (B6205AA #AVD)
  - *Using the Audio Developer's Kit* (B2355-90069)
  - *Using the Image Developer's Kit* (B2355-90068)
- Technical Print Service
  - The *Technical Print Service System Administrator's Guide* is provided online. It can be viewed using the CDE Help Manager.
  - The online information is also available in a hardcopy version which can be sent directly to a printer. A PCL and Postscript file can be found in `/usr/share/doc/TechPrintServ/C` directory. This manual is provided online only and cannot be ordered as a separate product.
- Starbase manuals (B6205AA #0BM)
  - *Starbase Technical Addendum* (B2355-90117)
  - *HP-UX Starbase Device Drivers Manual* (B2355-90055)
  - *Starbase Reference* (B2355-90056)
  - *Starbase Graphics Techniques* (98592-90081)
  - *Starbase Display List Programmer's Guide* (B2355-90098)
  - *Fast Alpha/Font Manager Programmer's Guide* (B2355-90097)
  - *Graphics Administration Guide* (B2355-90122)

## 7-2 What Manuals are Available for This Version

- PEX Programming manuals (B6205AA #AVE)
  - *HP PEX Implementation and Programming Supplement* (B3176-90118)
  - *PEXlib Programming Manual* O'Reilly & Associates (B3176-90003)
  - *PEXlib Programming Reference* O'Reilly & Associates (B3176-90004)
  - *Portable Programming with CGE PEX 5.1* (B3176-90042)
  - *PEXlib Developer's README FIRST* (B2355-90630)
  - *Graphics Administration Guide* (B2355-90122)

You can order these manuals by contacting your local sales office, Parts Direct Ordering (800) 227-8164, or HP DIRECT at (800) 637-7740. Additionally, you may find *Mastering Motif Widgets*, *Motif Style Guide*, and the O'Reilly & Associates manuals at your local computer bookstore.

You may also order O'Reilly & Associates manuals directly from O'Reilly & Associates.

- US & Canada orders, call (800) 998-9938 (voice) or (707) 829-0104 (fax); or
- From all other locations, call (707) 829-0515.

O'Reilly & Associates, Inc.  
 103 A Morris Street  
 Sebastapol, CA 95472  
 order@ora.com

## Online Documentation

- The CDE desktop provides an online help volume containing HP-UX font information. It can be viewed using the CDE Help Manager.

## Related Technical Documentation

- *X Window System User's Guide Motif Edition* by Valerie Quercia and Tim O'Reilly. ISBN 1-56592-015-5, O'Reilly & Associates, Inc.
- *Thread Time: The Multithreaded Programming Guide* by Scott J. Norton and Mark D. DiPasquale. ISBN 0-13-190067-6, Prentice-Hall
- *Programmer's Supplement for Release R6* edited by Adrain Nye. ISBN 1-56592-089-9, O'Reilly & Associates, Inc.



## **Software Availability in Native Languages**

---

The HP-UX Developer's Toolkit can be ordered either as an English or Japanese product.

