CONVEX
CXwindows V3.0
Release Notice

Document No. 710-005930-007

January 1992

CONVEX Press
Richardson, Texas USA

CONVEX
CXwindows V3.0
Release Notice

Document No. 710-005930-007

Printed in the United States of America

Contents

3 Frequently Asked Questions

Overview of CONVEX CXwindows V3.0

1

About this package

CONVEX CXwindows allows users to access CONVEX supercomputers through familiar X windowing techniques. With CONVEX CXwindows, workstation users can transfer data from windows on a CONVEX supercomputer into other application windows on remote machines. The CONVEX machine can serve as an X or PEX applications engine serving many workstations in the network.

CONVEX CXwindows V3.0 software includes:

- The X programming libraries:
  - The X Protocol library.
  - The X Toolkit Intrinsics library.
  - The MIT Athena widget set.
- OSF/Motif widgets and programming libraries.
- Common MIT X clients, CONVEX-specific X clients, window managers and application development utilities including:
  - The OSF/Motif Window Manager, `mwm`.
  - The OSF/Motif User Interface Language compiler, `uil`.
  - `editres`, an interactive client for editing X resource files.
  - WCL, the Widget Creation Language library.
- The X font server, `fs`.
- PEX, the PHIGS Extension to X:
  - A PHIGS/PHIGS PLUS application programming interface for C and FORTRAN.

Documentation for CONVEX CXwindows V3.0 includes:

- Man pages for all programming library routines and for all X clients.

All software in the CONVEX CXwindows V3.0 release is based on X Version 11 Release 5, OSF/Motif V1.1.1 and ANSI C.

Prerequisites

Before you can install this package, your system must already be running these software packages:

- ConvexOS V9.1 or a later release of the operating system.
- ConvexOS Utilities V9.1 or a later release of the system utilities.
- ConvexOS Internet Services V9.1 or a later release of the networking utilities.
- CONVEX C Compiler V4.1 or a later release of the compiler.

If your system is not running these four software packages, you must install them before continuing. If you need additional information on new releases, contact the CONVEX Technical Assistance Center (TAC).

X Display Server Requirements

In addition to the CONVEX software mentioned above, you must be running an R4 or R5-compliant X server. CONVEX CXwindows has been qualified on these servers:

- Network Computing Devices (NCD) NCD15b, NCD16, NCD17c, NCD19 X Display Stations running the NCD V2.2 server.
- Network Computing Devices V2.1 Xremote server.
- MIT X Version 11 Release 4 Sun 3 sample server running on Sun Microsystems 3/50, 3/60 and 3/80 workstations.
- MIT X Version 11 Release 4 Sun 4 sample server running on a Sun Microsystems Sparcstation 4/330 workstation.
- MIT X Version 11 Release 4 Sun 4 sample server running on a Sun Microsystems Sparcstation 2 workstation.
- MIT X Version 11 Release 5 Sun 3 sample server running on Sun Microsystems 3/50, 3/60 and 3/80 workstations.
- MIT X Version 11 Release 5 Sun 4 sample server running on a Sun Microsystems Sparcstation 2 workstation.
- Apple Computer MacX V1.0 and V1.1 servers running on all MacII platforms.
- Tektronix, Inc. X server version number 11.0, vendor release number 2, running on an XP29 X terminal.

In order to run the CONVEX CXwindows PEX software, you must be running an R5-compliant PEX server. CONVEX CXwindows PEX has been qualified on these servers:

- Tektronix XP29P PEX terminal.
- All Evans and Sutherland PEX servers.

CONVEX does not directly support any of these X servers.

Disk Space Requirements

CONVEX CXwindows V3.0 requires 70 megabytes of disk space. By default, CONVEX CXwindows is installed in the directory /usr/X11 in the /usr partition.

If your /usr partition does not have adequate free space or if you choose to do so, you can install CONVEX CXwindows into an alternate partition. The CONVEX CXwindows install script creates the necessary links from /usr to the alternate install directory.

Refer to the *CONVEX CXwindows V3.0 Installation Procedures* for more details.

Corequisites

CONVEX CXwindows will operate with CONVEX COVUEnet V2.1 or later releases of CONVEX COVUEnet. CONVEX CXwindows V3.0 will not work with COVUEnet releases prior to COVUEnet V2.1.

If you are running an older versions of CONVEX COVUEnet, please contact your CONVEX sales representative for information on upgrading your software.

Installing this Release

Refer to the *CONVEX CXwindows V3.0 Installation Procedures* for special instructions on how to install the software.

What is new in CONVEX CXwindows V3.0?

CONVEX CXwindows V3.0 adds many features to make X easier to use and administrate. Some of the new features include a centralized font server, support for scalable fonts, interactive X resource file editing, several new features in the standard X clients and many, many bug fixes.

CONVEX CXwindows V3.0 also includes PEX, the three-dimensional graphics extension to X. CONVEX CXwindows V3.0 includes a client-side Application Programming Interface (API) based on ISO IS PHIGS and PHIGS PLUS that generates PEX protocol. Refer to the section "PHIGS Extension to X" below for more information.

New clients

CONVEX CXwindows V3.0 includes several new clients. Many of these clients were specifically created to make X easier to use.

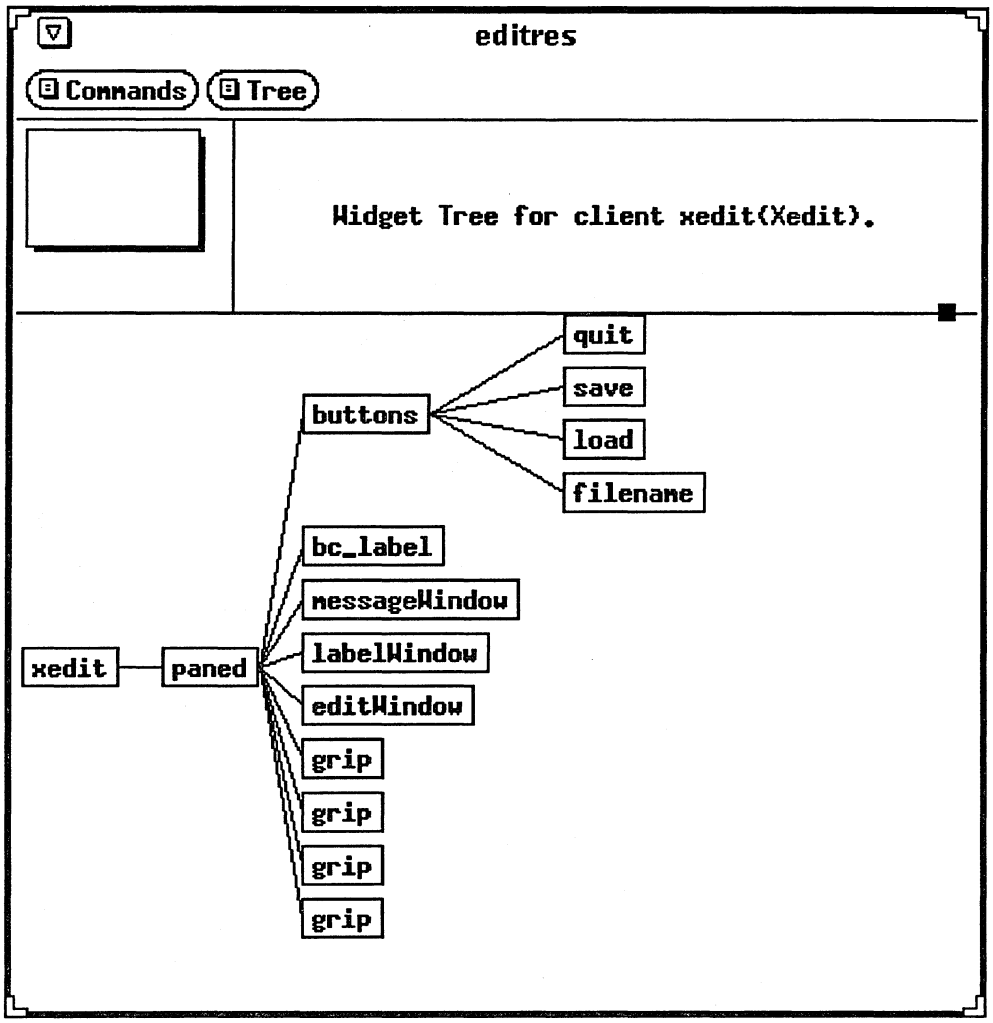
editres

`editres` is a tool that allows you to view the full widget hierarchy of any X Toolkit-based client that speaks the Editres protocol. In addition, `editres` will help you construct resource specifications and allow you to apply the resource to the application and view the results dynamically. Once you are happy with a resource specification, you can append the resource string to your X resources file.

To use `editres`, start a CONVEX CXwindows V3.0 application like `xedit`; then start `editres`. Select "Get Widget Tree" from the `editres` "Commands" menu. When the pointer turns into a crosshair, move the pointer into the `xedit` window and press the mouse button.

`editres` will display the widget hierarchy of the `xedit` application.

Your window should look something like this:



After capturing the widget tree, you can then select widgets within the tree display and edit their resources directly. If the widget tree is too large for the display window, enlarge the `editres` window or use the panner (the rectangle in the top-left of the window) to pan around within the display.

To edit resources, select "Show Resource Box" from the "Commands" menu. Choose a resource, edit its value and then click "Apply". Your change will appear immediately.

`editres` will work with any CONVEX CXwindows V3.0 client, clients you write based on the R5 toolkit Intrinsics and with any client based on CONVEX OSF/Motif V1.1.1.

See the `editres` man page for more details on how to use editres.

Note

You do not need an R5-compliant server to run editres. editres uses the standard X selection mechanism for interclient communication.

viewres

If you are developing application with the Athena widgets, you can use `viewres` to browse the Athena widget hierarchy.

The font server

One of the most difficult problems for X system administrators is font management. Each X server has a slightly different method (tftp, NFS, local disk, ROM) for accessing fonts and each server type requires its own binary font format. An administrator maintaining a network of heterogenous X servers must understand the configuration parameters of all of the servers and must maintain several versions of the same font.

The font server was created to address the complex problem of font management. Instead of forcing each X server to read all fonts from its local (or NFS mounted) file system, the font server manages fonts separately from the X server.

The font server provides a central, server-independent repository for fonts. In addition, for fonts that take a long time to open, the X server can continue processing other clients' requests while the font server services the font requests.

You do not use the font server directly; instead, your X server contacts the font server for a list of available fonts. Your X server then makes these fonts available to clients.

Note

You do not need an R5-compliant server to run most CONVEX CXwindows V3.0 X-based applications. However, in order to use the font server and access scalable fonts, you must be running an R5-compliant X server; R3 or R4-based X servers cannot use scalable fonts or the font server.

How to use the font server

To use a font server, you set the FONTSERVER environment variable to specify the host name of the font server and the port to connect to. This is very similar to setting the DISPLAY variable.

For example, to specify the font server running on the host named pixel, enter the command:

```
csh% setenv FONTSERVER pixel:7000
```

Any X client you invoke from now on can access fonts from the font server on pixel.

To find out information about the font server, use the fsinfo client. If FONTSERVER is set correctly, fsinfo will output something like:

```
name of server: pixel:7000
version number: 1
vendor string: MIT X Consortium
vendor release number: 5000
maximum request size: 16384 longwords (65536 bytes)
number of catalogues: 1
 all
Number of alternate servers: 0
number of extensions: 0
```

To list the fonts that are available on the font server, use the `fslsfonts` utility; this is similar to `xlsfonts`, but contacts the font server rather than the X server.

The font server also supports scalable fonts. For some fonts, you can now specify any point size; the fonts will be scaled and created on the fly and sent to your X server.

If you would like to experiment with scalable fonts, do the following:

```
csh% fslsfonts | grep p-0
```

Any fonts with zeroes as its average width are scalable fonts. Your list should contain font names like this:

```
-adobe-helvetica-bold-o-normal--0-0-100-100-p-0-iso8859-1
-adobe-helvetica-bold-o-normal--0-0-75-75-p-0-iso8859-1
-adobe-helvetica-bold-r-normal--0-0-100-100-p-0-iso8859-1
-adobe-helvetica-bold-r-normal--0-0-75-75-p-0-iso8859-1
-adobe-helvetica-medium-o-normal--0-0-75-75-p-0-iso8859-1
-adobe-helvetica-medium-r-normal--0-0-75-75-p-0-iso8859-1
```

To display the font, use the `xfd(1)` font display utility:

```
csh% xfd -fn '-adobe-helvetica-*-*-*-*--*-230-*-*-*-0-*-*'

csh% xfd -fn '-adobe-helvetica-*-*-*-*--*-400-*-*-*-0-*-*'
```

You can also use the xfontsel `(1)` utility to display the scalable fonts that are available. Start xfontsel and set *avgWidth* to 0; this will limit the set of selected fonts to all the scalable fonts. Then set the other fields to choose a font.

For example, set *avgWidth* to 0, set *fndry* to adobe and then choose a *ptSz*. In addition to the point sizes that are recommended, you can also specify point size. The font server will scale the font to the appropriate size.

If the font server does not respond

If the font server does not respond, make sure the font server daemon is installed and running on the font server host. Refer to the *CONVEX CXwindows V3.0 Installation Procedures* for instructions on how to install and initiate the font server.

Changes to clients

All of the popular X clients are still available in CONVEX CXwindows V3.0. Many have been enhanced. The following table presents some new features of interest.

Client	Change
`oclock`	oclock can be transparent (try `oclock -transparent`).
`xterm`	Cuts of wrapped lines (like on the command line) are now treated as single lines.
	Pasting large amounts of data into an `xterm` now works.
	You can now change the font in an `xterm` window dynamically. (Try this: put a font name in the X cut buffer and then choose "Selection" from the `xterm` font menu.)
`xlswins`	Is no longer supported; use **`xwininfo -root -tree`**.
`xmh`	Now uses the MH environment variable, if set.
	Better recovery from inconsistencies in the file system.
	Now supports checking for mail in multiple mail drops.
	You can now specify an editor to use for composing messages.

Client	Change
bitmap	Greatly enhanced.
xmag	Greatly enhanced.
twm	Client-specific colormap specifications are now honored according to the rules of the ICCCM.

CONVEX OSF/Motif

CONVEX OSF/Motif has been fully integrated with R5 and all CONVEX OSF/Motif widget resources can be edited with editres.

No incompatible changes have been made to our version of OSF/Motif; the widgets and window manager are exactly the same as in CONVEX CXwindows V2.1.

Other changes

Several enhancements have been made to X resource databases:

- A "#include" syntax is now supported in resource files. This is useful for deriving custom app-defaults files (for Xt applications) from base app-defaults files.
- A new reserved component name, '?,' has been defined that matches a single level in the resource hierarchy ('*' matches any number of levels). This name makes it easier to override resources in app-defaults files.
- Screen-specific resources can be set. For example, you can set resources for a color screen and not effect a monochrome screen on the same server.
- Resource databases are loaded and parsed more quickly.

Function prototypes in header files are now complete. R5 is also ANSI C and POSIX-compliant.

List of clients

These are the clients supported in CONVEX CXwindows V3.0:

appres	atobm	bdftopcf	bitmap
bmtoa	constype	editres	execqt
fs	fsinfo	fslsfonts	fstobdf
ico	imake	listres	makedepend
maze	mkdirhier	mkfontdir	mwm
oclock	puzzle	resize	showfont
showrgb	twm	uil	viewres
xauth	xbiff	xcalc	xclipboard
xclock	xcmsdb	xcmstest	xconsole
xcutsel	xditview	xdm	xdpr
xdpyinfo	xedit	xev	xeyes
xfd	xfontsel	xgas	xgc
xhost	xkill	xload	xlogo

xlsatoms	xlsclients	xlsfonts	xmag
xman	xmh	xmkmf	xmodmap
xpr	xprop	xrdb	xrefresh
xset	xsetroot	xstdcmap	xterm
xwd	xwininfo	xwud	

Accessing CONVEX CXwindows V3.0 man pages

To access the new X man pages, add /usr/X11/manX to your MANPATH variable. To access the new PEX man pages, add /usr/X11/manPEX to MANPATH.

Example X configuration files

CONVEX CXwindows V3.0 includes many example X configuration files:

- A sample xdm configuration can be found in /usr/X11/lib/xdm-sample. System managers can tailor this sample configuration to meet the needs of their site.
- An elaborate twm configuration file can be found in /usr/X11/lib/twm/system.twmrc. This configuration will be used by default if a user does not have a .twmrc file.
- A sample font server configuration file can be found in /usr/lib/X11/fs/config.

Contributed software

This release of CONVEX CXwindows includes a large set of contributed X clients and other utilities such as the MH mail system. The contributed software was collected from sources available from MIT, the news group comp.sources.x and from a number of anonymous **ftp** sites around the world.

The contributed software has not been fully tested and is not supported by CONVEX. This software is provided to you as a convenience. Many of the applications are already compiled and ready to use; other applications have already been modified for CONVEX systems and only require recompilation. If the contributed software has been installed on your system, it can be found in the directory /usr/X11/unsupported.

The following sections describe some of the X clients and applications that are available.

/usr/X11/unsupported/bin

This directory contains executables of some contributed clients including:

- PEX demonstration programs.
- OSF/Motif V1.1 demonstration clients.

  These clients were built with OSF/Motif V1.1 and X Version 11 Release 5 and may behave differently on the CONVEX than on other platforms.
- Image display and manipulation utilities including xfig (an X drawing application), xloadimage (an X-based image viewing application) and the Portable Bitmap (PBM) utilities.

- Clients to facilitate reading and sending electronic mail and Internet news.

/usr/X11/unsupported/lib, /usr/X11/unsupported/include

This directories mirror the standard CONVEX CXwindows directories /usr/X11/lib and /usr/X11/include. They include:

- Contributed clients' application default files.

  Clients in /usr/X11/unsupported/bin automatically search the alternate app-defaults directory for their default configurations.

- All bitmaps needed by the contributed clients in unsupported/bin.
- All specialized directories needed by the contributed clients.
- A FORTRAN API for PEX based on the ISO IS PHIGS Binding and the soon-to- be standardized PHIGS PLUS Binding. You can use this FORTRAN API with CONVEX FORTRAN V7.0 or later releases of the compiler.

  See below for more information on CONVEX PEX.

/usr/X11/unsupported/src

This directory contains source for all of the clients and libraries. The source directory is divided into three subdirectories:

- The Motif subdirectory contains source for clients that use the OSF/Motif widget set.
- The clients subdirectory contains the source for all other clients and utilities. This subdirectory contains the source for the MH mail system, the PBM utilities, Xrn (an X-based News reader), tvtwm (a virtual desktop version of twm) and many other useful clients.
- The demos subdirectory contains the source for PEX demos, general X demonstration clients and OSF/Motif (provided by the Open Software Foundation) demonstration clients.

All of these sources (except Motif) are also available via anonymous ftp from a number of sites around the world (including export.lcs.mit.edu and uunet.uu.net).

CONVEX PEX

New in CONVEX CXwindows V3.0 is PEX, the PHIGS Extension to X, a new X protocol extension that adds 3D graphics to the X Window System. Also new to this release is a PHIGS programming library that interfaces to the PEX protocol.

CONVEX PEX is based on the PEX Sample Implementation (SI) and includes a C Application Protocol Interface (API) for clients to generate PEX protocol. The API provided with CONVEX PEX is the ISO IS PHIGS Binding and the soon-to-be standardized PHIGS PLUS Binding.

CONVEX PEX also includes PHIGS API man pages and the PHIGS Monitor (PM), a separate process started at client run time to handle PHIGS Input functionality.

Building X applications

The programming libraries provided in CONVEX CXwindows V3.0 are based on ANSI C.

Building ANSI C applications

To build X applications written in ANSI C with the new C compiler, use a command like:

```
cc -o Xapp Xapp.c -lXmu -lXext -lX11
```

If you are building an application using the MIT Athena widget set, use a command like:

```
cc -o xaw xaw.c -lXaw -lXt -lXmu -lXext -lX11
```

Building non-ANSI C applications

If your application is not based on ANSI C, you have to compile your application source code with the -pcc option and link using a command like the one above.

To build non-ANSI C applications with the new C compiler, use a sequence of commands like:

```
cc -c Xapp.c
cc -o Xapp Xapp.o -lXmu -lXext -lX11
```

If you are building an X application using the MIT Athena widget set, use a sequence of commands like:

```
cc -c Xapp.c
cc -o Xapp xaw.o -lXaw -lXt -lXmu -lXext -lX11
```

Building OSF/Motif V1.1 applications

To build Motif applications, you must use the Motif libraries and the CONVEX CXwindows Xlib and X Toolkit Intrinsics libraries. The OSF/Motif-specific toolkit intrinsics have been replaced with the standard R5 X Toolkit Intrinsics.

The order of the libraries in your compilation is important. To build applications that use the OSF/Motif widget set, use the command:

```
cc -o app app.c -lXm -lXt -lXext -lX11
```

Known software problems and fixes in this release

This section contains a list of software bug reports. The list is divided into known problems and problems that have been fixed since the last release.

Known problems

This section lists the known problems with CONVEX CXwindows V3.0 as of October 17, 1991. Problems reported after this date are not reflected in this document.

Please refer to this list before reporting a problem. If a workaround is known it is included in the description of the bug.

- xdm does not check the /etc/ttys file to determine if a pseudo-terminal is secure or not.

xdm does not use pseudo-terminals; it is not appropriate or possible for xdm to restrict access to certain ttys.

To work around the problem, the system administrator can edit the system's Xsession script and prevent certain users from logging in via xdm. The file /usr/X11/lib/xdm/Xaccess can also be used to prevent certain displays from connecting to xdm.

- If you compile X applications using the Release 5 header files and link with Release 4 (CONVEX CXwindows V2.1) X libraries, you will get an error message "XtStrings is undefined."

  To avoid this problem, be sure to compile and link your X application with corresponding header files and libraries.

Fixed problems

This section lists fixes that have been made to CONVEX CXwindows since the last release.

- `xman` honors the ordering of the $MANPATH environment variable.
- `xterm` sets the $LOGNAME environment variable.
- `xdm` now enforces password aging. If a user's password has expired, the user can change his or her password via xdm and then continue with the X session. If a user does not successfuly change his or her password, xdm prohibits login.
- In all cases, `xterm` correctly writes and erases login entries in /etc/utmp and /etc/wtmp.

Reporting problems

To report bugs on CONVEX CXwindows V3.0 software, please use the `contact` utility. When submitting bugs, please be as explicit as you can; if the problem is reproducible, include instructions on how to reproduce the bug.

When reporting bugs please specify:

- The name of the client (e.g., `xterm, oclock, editres`) if reporting bugs against any of the X utilities.
- The name of the library (e.g., `libX11.a, libXm.a`) if reporting bugs against any of the programming libraries.
- The name of the man page (e.g. `XOpenDisplay(3)`) if reporting bugs against any of the X manual pages.

Also, please specify the following information about your X environment:

- The type of your CONVEX system (C3200, C3400, C3800) and the version of the operating system.
- The type of X display you are using (e.g., NCD16, NCD19, Sun 3, etc.)
- The version number (R3, R4, R5) of your server (easily determined with the `xdpyinfo(1)` utility).
- The name of your window manager (`mwm, twm`).

Converting to OSF/Motif V1.1 Widgets

2

Introduction

The OSF/Motif V1.1 widget set has changed significantly since the previous release (CX/Motif V1.0). New widgets have been added and many of the existing widgets have been rewritten to add features and to be more robust. All OSF/Motif V1.1 widgets are based on the X Version 11 Release 5 Toolkit Intrinsics.

The OSF/Motif V1.1 widget set is compatible with earlier versions. However, some changes in the widget set will affect applications written against the CX/Motif V1.0 widgets. This chapter describes how to convert your CX/Motif V1.0 applications to the new widget set, as well as how to convert your code to use the new features of OSF/Motif V1.1.

Developers are urged to migrate their code to the new widgets wherever possible.

Changes to the widget set

This section specifies changes that have been made to each widget. Wherever possible, information on how to port existing code to use the new widgets is provided.

Application Contexts

The Release 4 and Release 5 X Toolkit Intrinsics requires you to use new routines when manually creating application shells. Rather than use XtCreateShell, you must use XtAppCreateShell. For example, when you do not use XtInitialize, you must do the following:

```
XtToolkitInitialize();

app_context = XtCreateApplicationContext();

display = XtOpenDisplay(app_context, ...);

shell = XtAppCreateShell(...);
```

Each subsequent shell can be created by repeating:

```
shell = XtAppCreateShell(...);
```

In addition, XtAppMainLoop must be used in placed of XtMainLoop, XtAppPending must replace XtPending and XtAppNextEvent must replace XtNextEvent. Release 4 also introduced XtAppInitialize and XtVaAppInitialize; these routines should replace XtInitialize.

XtGetValues

If XtGetValues returns incorrect values for a resource, check the value that was assigned to the resource. If the resource was set to an integer, it might have to be set to a value of type Dimension. Changing the type of the value will clear the problem.

Values of type Position and Dimension are now 16-bit unsigned quantities. Applications that declared values of type short or int should change the declaration to Dimension or Position.

Widget Traversal

All references in your source to the function XmAddTabGroup should be removed. A new resource, XmNnavigationType is used to set the tab group policy in both manager and primitive widgets.

NULL Widget Translations

A widget cannot specify a NULL for its translations. A widget can specify an empty translation table.

Label and Button Widgets

If a label or button widget is a direct child of the shell, the border of those widgets will not be visible.

Text Widgets

Single line text widgets no longer display multiple lines. Prior to this release, single line text widgets allowed you to browse through a large buffer; in this release if you have multiple lines, only the last line is shown.

The resource XmTextGetTopPosition has been changed to XmTextGetTopCharacter and XmTextSetTopPosition has been changed to XmTextSetTopCharacter.

The resource XmNtopPosition has been changed to XmNtopCharacter.

Retrieving A Label from A Widget

If you want to retrieve the value of a label from a widget, use the following code fragment:

```
XmString label;

XtSetArg(args[0], XmlabelString, &label);

XtGetValues(w, args, 1);
```

If you want to retrieve the text part of "label," and "label" was created by XmStringCreateLtoR with the default character set XmSTRING_DEFAULT_CHARSET, use this code fragment:

```
char *s;

XmStringGetLtoR(label,XmSTRING_DEFAULT_CHARSET, &s);
```

The variable s will point to the text part of "label."

Paned Window XmNminimum and XmNmaximum Resources

Applications that use or set the resource XmNminimum or XmNmaximum in a SetArg, UIL code or in a X resource file should be changed. These resources have changed names to avoid conflicts with other resources. The new names are XmNpaneMinimum and XmNpaneMaximum.

Frequently Asked Questions

3

This chapter presents answers to commonly asked questions about CONVEX CXwindows.

What version of X is CONVEX CXwindows V3.0?

CONVEX CXwindows V3.0 is based on X Version 11 Release 5 and includes MIT patches 1-4. CONVEX CXwindows also includes the OSF/Motif V1.1.2 Motif Window Manager and Motif user interface development software.

Is there an X server included with CONVEX CXwindows?

CONVEX CXwindows does not include an X server; CONVEX CXwindows only includes client-side software only.

What is the DISPLAY environment variable?

The DISPLAY environment variable specifies the server your X clients should connect to. The DISPLAY variable specifies the name of the machine to connect to, and what screen on that machine to use.

For example, if the DISPLAY variable was set to mysun:0, your X client would connect to the machine named "mysun" and be displayed on screen 0 of that workstation. To set the display variable in csh, enter

```
% setenv DISPLAY displayname:0
```

where *displayname* is the name of your X server. *displayname* is usually the host name of your workstation or X terminal.

To set the DISPLAY in the Bourne shell, use the sequence

```
$ DISPLAY=displayname:0
$ export DISPLAY
```

For a complete description of the DISPLAY variable, refer to the *O'Reilly and Associates X Window System User's Guide, Motif Edition.*

Can I run CONVEX CXwindows V3.0 with X Version 11 Release 3 (R3) servers?

CONVEX CXwindows V3.0 is based on X Version 11 Release 5 (R5) and includes new features not supported by R3 servers.

For example, R3 servers cannot display round windows; clients such as oclock that use round windows will be displayed in rectangular windows. R5 also supports better inter-client communication paradigms that R3-based window managers will not understand. Most CONVEX CXwindows clients will work with most R3 servers.

In order to use theR5 font server and access scalable fonts, you must be running an R5-compliant X server; R3 or R4-based X servers cannot use scalable fonts or the font server.

It is recommended that CONVEX CXwindows V3.0 be used with R4 or R5 servers. You can refer to Chapter 1 for a list of servers that have been used with CONVEX CXwindows V3.0.

What are X resources?

X resources allow you to customize X applications. With X resources you can specify the fonts and colors of an application, control the size and placement of windows, and change the bindings of your keyboard. X resources allow you to customize the behavior of applications to suit your preferences.

Refer to Chapter 10 of the *O'Reilly and Associates X Window System User's Guide* for a complete discussion of X resources.

Why are there C preprocessor directives in the resources files?

`xrdb` defines a number of symbols based on the capabilities of the server you are using. For example, `xrdb` will define the symbol COLOR if the display you are using supports color.

You can use `xrdb` and C preprocessor directives to create a flexible X resources file that will work with a variety of X servers. Refer to the `xrdb` man page for more information.

What is imake?

`imake` is a portable front-end to the `make` utility. `imake` is used to generate Makefiles from a template, a set of C preprocessor macros, and a simplified file of build rules called an Imakefile.

Imakefiles are preferred over Makefiles because different operating systems have different versions of `make`. With `imake`, machine dependencies (such as compiler options, alternate command names, and special make rules) can be kept separate from the descriptions of the items to be built. Imakefiles can be ported to any system that has an `imake` configuration.

To use `imake`, refer to the `imake` man page. The directory /usr/lib/X11/config includes `imake` templates and the rules for building on ConvexOS.

Refer to the file /usr/lib/X11/config/Imake.rules and /usr/lib/X11/config/Convex.rules for a list of `imake` macros that are available. The file /usr/lib/X11/config/convex.cf declares machine-dependent information for ConvexOS.

Can I preview a font to see what it looks like?

To view what a font family looks like, use the CONVEX CXwindows client `xfontsel`. If you want to look at an individual font, you can use `xfd`. Refer to the `xfontsel` and `xfd` man pages for more details.

What widget libraries does CONVEX support?

CONVEX supports the MIT Athena widgets and the OSF/Motif V1.1.2 widgets. Both widget libraries are included as part of CONVEX CXwindows V3.0.

Are the OSF/Motif widgets provided with CONVEX CXwindows V3.0 compatible with previous releases of OSF/Motif?

There have been many changes made to the OSF/Motif widgets to make the widgets compliant with X Version 11 Release 5. In some cases, your source will not need modification to use the new widgets. However, they are many cases where changes are required.

Refer to Chapter 2 of this release notice for more information about the new OSF/Motif widgets.