CDE and Solaris Motif Toolkits 3

The Solaris 2.3, Solaris 2.4, and Solaris 2.5 releases and the CDE 1.0 release provide versions of the Motif toolkit. This chapter discusses these toolkits.

CDE Motif Toolkit	*11*
Widgets Available for CDE Application Development	*15*
Solaris Motif Toolkits	*20*

For a summary of the differences between these toolkits, see Table 1-1 on page 3.

CDE Motif Toolkit

CDE Motif, the Motif toolkit that is provided with the CDE 1.0 release, adds functionality to the Motif 1.2.3 toolkit while maintaining backward binary compatibility. It is source and binary compatible with Motif 1.2 applications. Existing Motif 1.2 applications will compile using CDE Motif. Existing Motif 1.2 binaries will run without modification using CDE Motif.

The CDE Motif Toolkit consists of the Motif 1.2.3 widget library with enhancements to existing functionality and bug fixes, as well as some new features. Motif 1.2.3 is a patch of Motif 1.2. The APIs and look and feel for Motif 1.2 and Motif 1.2.3 are the same.

In addition, the CDE library `libDtWidget` provides control widgets for graphical user interface objects not found in Motif 1.2.3. For more information on these widgets, see "CDE Motif Control Widgets" on page 15 and *CDE Programmer's Guide*.

Features Added to Motif 1.2.3

CDE provides the following added features to Motif 1.2.3 to support desktop applications:

- Complete localization of toolkit error messages
- `XmGetPixmap()` and `XmGetPixmapByDepth()` use the environment variable `XMICONSEARCHPATH` or `XMICONBMSEARCHPATH` as the icon search path. If neither of these variables is set, then they use `XBLANGPATH`, which is the Motif 1.2 behavior. See the corresponding CDE Motif man pages for more information.

Enhancements to Existing Motif 1.2.3 Functionality

The Solaris CDE Motif library contains minor enhancements to Motif 1.2.3 usability to emulate certain OPEN LOOK user interface and Microsoft® Windows features. The usability enhancements include:

- Optionally allowing mouse button 2 on a three-button mouse to be used to extend the current selection. This is equivalent to the OPEN LOOK Adjust function.
- Allowing Tab to be used to move through a group of `PushButton` widgets and gadgets, `ArrowButton` widgets and gadgets, and `DrawnButton` widgets.
- Allowing mouse button 3 to activate a CascadeButton menu (for OPEN LOOK compatibility).
- Providing three new resources (`pathMode`, `fileFilterStyle`, and `dirTextLabelString`) for the `XmFileSelectionBox` widget, which give the File Selection Dialog Box GUI an improved look and feel.
- Ability to re-map keybindings to be consistent with those for OPEN LOOK or Microsoft Windows applications.
- Providing visual enhancements to the standard Motif look (see the next section, "Visual Enhancements").

Each of the preceding enhancements can be controlled by a resource: either a widget resource (for `XmFileSelectionBox`) or an application-wide resource (all other cases). The default values for this resource provide behavior and APIs that are identical to that of Motif 1.2. For information on these enhancements and resources, see the `XmDisplay(3x)` and `XmFileSelectionBox(3x)` man pages.

Visual Enhancements

CDE changes the Motif 1.2.3 look in the following ways:

- RadioBox fill color was changed to show state more clearly
- RadioBox shape was changed from diamond to circular
- A check mark was added to the CheckBox to show its state more clearly
- CascadeButtons and menu items were changed to have an etched-in border when active
- Thumb was removed from the read-only Scale to distinguish it from the interactive Scale
- Default shadow thickness was changed to 1 pixel
- Default highlight thickness was changed to 1 pixel
- Default PushButton visual was added that highlights the button inside its default shadow

For information on these enhancements, see the `XmDisplay(3)`, `XmPushButton(3)`, `XmPushButtonGadget(3)`, `XmToggleButton(3)`, `XmToggleButtonGadget(3)`, and `XmScale(3)` man pages.

CDE Motif Libraries

Use CDE Motif and X11R5 libraries to develop a CDE Motif-compliant application for the X Window System.

CDE Motif Library (libXm)

Solaris CDE provides all the Motif 1.2.3 header files. The CDE Motif libraries are the Motif 1.2.3 libraries with bug fixes and enhancements.

CDE Motif UIL library (libUil)

The Motif user interface language (UIL) is a specification language for describing the initial state of a Motif application's user interface. The CDE version of the Motif UIL library is essentially unchanged from the Motif 1.2.3 version.

Include the `UilDef.h` header file (found in the `/usr/dt/include/uil` directory) to access UIL.

CDE Motif Resource Manager Library (libMrm)

The Motif resource manager (MRM) is responsible for creating widgets based on definitions contained in user interface definition (UID) files created by the UIL compiler. MRM interprets the output of the UIL compiler and generates the appropriate argument lists for widget creation functions. Use `libMrm` to access the Motif resource manager. The CDE version is essentially unchanged from the Motif version.

Include the `Mrm/MrmPublic.h` header files to access `libMrm` in your application.

Demo Programs

You can find Motif 1.2 demos in `/usr/dt/examples/motif`. These demos do not include any of the CDE Motif enhancements or widgets.

Related Documentation

See the *OSF/Motif Programmer's Reference* for information on UIL, the UIL compiler, UID, and Mrm.

Widgets Available for CDE Application Development

This section discusses the widgets available for Solaris CDE application development, as an extension to CDE Motif.

CDE Motif Control Widgets

The CDE Motif control widgets are designed to ease porting OPEN LOOK applications to the Solaris CDE desktop by providing equivalent functionality in CDE Motif. These widgets are not considered to be part of CDE Motif, but rather an extension to CDE Motif. The `libDtWidget` library contains widgets and functions that are used to provide common functionality across all CDE applications. The widgets provided include:

Table 3-1 CDE Control Widgets

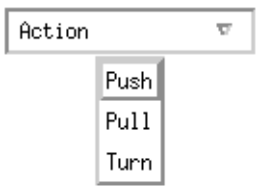
Widget Name	Description
`DtSpinBox`	`TextField` widget with additional controls for incrementing and decrementing numeric values, or browsing through and selecting from a list of text strings. Can be read-only. `DtSpinBox` is functionally similar to the OPEN LOOK numeric text field.
`DtComboBox`	Combination of `TextField` and pop-up list widgets that provides a list of valid choices for the `TextField`. Can be read-only.
`DtMenuButton`	Command widget that provides the menu cascading functionality of an `XmCascadeButton` widget outside of a menu bar, or a menu pane. `DtMenuButton` is functionally equivalent to the OPEN LOOK menu button.

Examples of each type of widget follow:

- Text field and arrow button widget (`DtSpinBox`)

- Text field and list box widget (`DtComboBox`)

- Menu button widget (`DtMenuButton`)



Note – The Solaris CDE software supports all Motif 1.2 widgets.

Compatibility with Motif 2.0

The APIs of the `DtSpinBox` and `DtComboBox` widgets are similar to the Motif 2.0 release of `XmSpinBox` and `XmComboBox` widgets. The APIs are designed so an application can easily switch to the Motif 2.0 version of these widgets. The main thing you need to do to switch is to change the `Dt` names for the class, types, and creation routines to `Xm`.

This information is supplied in case you choose to port your application to Motif 2.0 but this is *not* a recommendation that you do so.

Note – The Solaris CDE software does *not* guarantee strict API or binary compatibility between its widgets and the Motif 2.0 widgets.

Library and Header Files

The library `libDtWidget` provides access to the `DtSpinBox`, `DtComboBox`, and `DtMenuButton` widgets. The `libDtWidget` header files for these widgets are:

- `Dt/SpinBox.h`
- `Dt/ComboBox.h`
- `Dt/MenuButton.h`

Demo Programs

You can find the CDE Motif control widgets demos in `/usr/dt/examples/dtwidget`. Read the `README` file for detailed information on the demos.

Related Documentation

For more information on CDE Motif control widgets, see the relevant man pages and *CDE Programmer's Guide*.

3

CDE Terminal Widget

The `DtTerm` widget is part of the Solaris CDE development environment. It provides the functionality required to emulate an ANSI X3.64-1979-style terminal (specifically a DEC® VT220-like terminal with extensions). The Terminal widget library, `libDtTerm`, provides the `DtTerm` widget for adding a terminal window to a GUI. If you include a terminal in your application, use CDE Motif widgets to add display enhancements to it such as pop-up menus and scrollbars.

The Solaris CDE Terminal application, which is a part of the run-time environment, is a window that behaves as a terminal, enabling access to traditional terminal-based applications from within the desktop. The `DtTerm` widget is the foundation for the desktop run-time terminal.

The `libDtTerm` library includes a set of convenience functions to create, access, and support the `DtTerm` widget.

Library and Header Files

The `libDtTerm` library provides a set of widgets based on Motif for designing a terminal or for adding a terminal window to a GUI.

Include the `Dt/Term.h` header file to access `libDtTerm` APIs in your application.

Demo Programs

You can find the `DtTerm` demos in `/usr/dt/examples/dtterm`. See the `README` file for detailed information on the demos.

Related Documentation

For more information on the `DtTerm` widget, see the relevant man pages.

For more information on the desktop terminal application, see the terminal help volume, the relevant man pages, or *CDE User's Guide*.

Text Editor Widget

The CDE text editing system consists of two components:

- The Text Editor application, which provides editing services through graphical, action, and ToolTalk interfaces
- The editor widget, `DtEditor`, which provides a programmatic interface for the following editing services:
  - Cut and paste
  - Search and replace
  - Simple formatting
  - Spell checking (for 8-bit locales)
  - Undo previous edit
  - Enhanced I/O handling capabilities that support input and output of ASCII text, multibyte text, and buffers of data
  - Support for reading and writing files directly

Although the Motif text widget also provides a programmatic interface, applications that want to assure a system-wide uniform editor should use the `DtEditor` widget. The CDE Text Editor and Mailer applications use the editor widget. Use this widget in the following circumstances:

- You need the functionality, such as spell checking, undo, and find/change, that is provided by the `DtEditor` widget.
- You want users to be able to read and write data to and from a file.
- When your program does not need to edit the text while the widget has control of the text.

Library and Header Files

The `DtEditor` widget is in the `libDtWidget` library. The header file is `Dt/Editor.h`.

Demo Programs

A demo containing an example of the `DtEditor` widget (`editor.c`) is in `/usr/dt/examples/dtwidget` directory. Read the `README` file for detailed information on the demo.

Related Documentation

For more information on the Text Editor widget, see the relevant man pages and *CDE Programmer's Guide*.

Solaris Motif Toolkits

As mentioned previously, the Solaris 2.5 Motif toolkit is the CDE 1.0 Motif toolkit. The Solaris 2.4 Motif toolkit is an early release of the CDE 1.0 Motif toolkit. (It has the same major functionality as CDE Motif, only it includes fewer bug fixes to Motif 1.2.3.) The Solaris 2.3 Motif toolkit is IXI 1.2.2 Motif, which is an extension of Motif 1.2.2.

The directory hierarchy of the Solaris 2.4 and Solaris 2.5 Motif software is different from that of Motif 1.2.2 and of other Motif products. The distribution has a similar structure to the Solaris CDE hierarchy for consistency and to ease migration for future Solaris CDE users.

Note – The CDE Motif control widgets, the CDE Text Editor widget, and the CDE Terminal widget, which are discussed in "Widgets Available for CDE Application Development," are part of the Solaris CDE development environment, not the CDE Motif toolkit. If you develop a Motif application for the OpenWindows 3.5 desktop, you do not have access to these widgets (or any of the other non-Motif Solaris CDE functionality).

IXI Motif 1.2.2 Toolkit

The IXI Motif 1.2.2 toolkit contains some incompatibilities with standard OSF/Motif 1.2.2. These features are not part of the OSF/Motif 1.2 specification, and are not present in the Solaris 2.4 or CDE Motif toolkits.

XmList convenience functions

The following are the non-standard functions in IXI Motif. Remove them from your application code if you have used them:

- `XmListRecolorItem`
- `XmListRecolorPos`
- `XmListSetClientDataPos`
- `XmListSetClientDatasPos`

XmForm widget

The `XmForm` widget implementation in IXI Motif and OSF/Motif 1.2.2 are different, although the APIs are identical. Hence, applications linked with IXI Motif can exhibit minor behavioral or visual differences in the `XmForm` widget when they are re-linked with OSF/Motif.

The Solaris 2.4 and CDE Motif toolkits use the OSF/Motif 1.2.3 `XmForm` widget implementation, which is binary compatible with the OSF/Motif 1.2.2, but not the IXI Motif, `XmForm` widget.

Directory hierarchy

The IXI Motif directory hierarchy is different from that of the CDE and Solaris 2.4 hierarchies. IXI Motif typically installs into `/opt/SUNWmotif`.

Note – IXI 1.2.2 supported conversion of XPM format files for some resources with bitmap semantics. This feature is not available in the Solaris 2.4 release, but is available in the CDE (Solaris 2.5) release of the Motif toolkit.

3