Preface

The *Motif Programmer's Guide* describes how to use the Motif® application programming interface to create Motif® applications. The book gives an overview of the architecture of the Motif widget set, explains features of the Motif toolkit, and presents a model and examples for constructing Motif applications.

Audience

This document is written for designers and developers of Motif programs. It does not provide sufficient information to develop new Motif widgets, although widget developers need to be familiar with the facilities and the issues discussed in this book.

This document assumes that the reader is familiar with the American National Standards Institute (ANSI) C programming language. It also assumes that the reader has a general understanding of the X Window System, the Xlib library, and the X Toolkit Intrinsics (Xt).

Applicability

This is Revision 2.0 of this document. It applies to Release 2.0 of the Motif software system.

Purpose

The purpose of this guide is to explain how to write application programs using the Motif toolkit. After reading this book, you should have a general understanding of the Motif toolkit and the Motif widget set and should be able to write applications that use them. This guide is not intended to explain how to develop new classes of widgets.

Document Usage

This document is organized as follows:

- Chapter 1 introduces the book and gives an overview of the X Window System, Xlib, Xt, and Motif.
- Chapter 2 summarizes the structure of the Motif widget hierarchy and discusses general principles for writing Motif programs.
- Chapter 3 explains the structure and general elements of a Motif application. This chapter also discusses the use of multiple threads in application code.
- Chapter 4 describes the structure of a program that uses the User Interface Language (UIL) and Motif Resource Manager (MRM).
- Chapter 5 discusses most of the primitive widgets that form the building blocks of a Motif application.
- Chapter 6 describes how to use the RowColumn widget to build menus, radio boxes, and check boxes.
- Chapter 7 describes the widgets most appopriate for conducting dialogs with the user.
- Chapter 8 discusses composite widgets commonly used to contain other widgets in the application.
- Chapter 9 explains compound strings (XmString), render tables, tab lists, fonts, and other aspects of rich text.
- Chapter 10 explains the Motif Text and TextField widgets, which provide general display and editing of text.
- Chapter 11 is a guide to internationalizing applications and providing text, font, and other information that is specific to particular language environments.
- Chapter 12 explains how Motif uses colors and pixmaps and how an application can provide its own.
- Chapter 13 discusses issues in handling input, including keyboard focus and traversal, translations, and actions.
- Chapter 14 describes the layout–management policies of particular Motif widgets.
- Chapter 15 describes DrawingArea, a general–purpose widget for displaying graphics and handling user input at a low level.
- Chapter 16 describes how an application can transfer data using the Uniform Transfer Method (UTM).
- Chapter 17 is an extensive discussion of the Motif drag and drop interface, with which the user transfers data by manipulating iconic representations with the pointer.
- Chapter 18 discusses communication between an application and the Motif Window Manager (MWM), by means of resources, selections, protocols, and properties.
- Chapter 19 describes printing from a Motif application.
- Appendix A describes the **XmClipboard** calls.
- Appendix B summarizes data transfer support built into the standard Motif widget set.

Related Documents

For additional information about Motif, refer to the following documents:

- The *Motif Programmer's Reference* provides detailed reference information for programmers writing Motif applications.
- The *Application Environment Specification −− User Environment Volume* defines a stable set of routines for creating user interface applications.
- The *Motif Style Guide* explains the principles of user interface design for application developers.
- The *Motif User's Guide* explains how to interact with Motif applications.
- The *Motif Widget Writer's Guide* explains how to create Motif widgets.

For additional information about Xlib and Xt, refer to the following X Window System documents:

- *Xlib−−C Language X Interface* is the specification for Xlib.
- *X Toolkit Intrinsics−−C Language Interface* is the specification for Xt.

What DocBook SGML Markup Means

This book is written in the Structured Generalized Markup Language (SGML) using the DocBook Document Type Definition (DTD). The following table describes the DocBook markup used for various semantic elements.

Table 1 DocBook SGML Markup

Markup Appearance	*Semantic Element(s)*	*Example*
AaBbCc123	The names of commands.	Use the **ls** command to list files.
AaBbCc123	The names of command options.	Use **ls** *-a* to list all files.
AaBbCc123	Command–line placeholder: replace with a real name or value.	To delete a file, type **rm** *filename*.
`AaBbCc123`	The names of files and directories.	Edit your `.login` file.
AaBbCc123	Book titles, new words or terms, or words to be emphasized.	Read Chapter 6 in *User's Guide*. These are called *class* options. You *must* be root to do this.

Keyboard Conventions

Because not all keyboards are the same, it is difficult to specify keys that are correct for every manufacturer's keyboard. To solve this problem, this guide describes keys using a *virtual key* mechanism. The term *virtual* implies that the keys as described do not necessarily correspond to a fixed set of actual keys. Instead, virtual keys are linked to actual keys by means of *virtual bindings*. A given virtual key may be bound to different physical keys for different keyboards.

See Chapter 13 of this book for information on the mechanism for binding virtual keys to actual keys. For details see the **VirtualBindings**(3) reference page in the *Motif Programmer's Reference*.

Mouse Conventions

This guide assumes a 3–button mouse. On a 3–button mouse, the leftmost mouse button is usually defined as `Btn1`, the middle mouse button is usually defined as `Btn2`, and the rightmost mouse button is usually defined as `Btn3`. For details about how mouse buttons are usually defined, see the **VirtualBindings**(3) reference page in the *Motif Programmer's Reference*.

Problem Reporting

If you have any problems with the software or documentation, please contact your software vendor's customer service department.