

# About This Book

---

QuickDraw GX is an integrated, object-based approach to graphics programming on Macintosh computers. This book, *Inside Macintosh: QuickDraw GX Typography*, describes the QuickDraw GX typographic shapes that display text and shows you how to create and manipulate those shapes.

For application programming purposes, QuickDraw GX augments the capabilities of some of the Macintosh system software managers documented in other parts of *Inside Macintosh*. It supplements the chapters “QuickDraw Text” and “Font Manager,” as well as parts of the chapter “Script Manager” and the appendix “International Resources” in *Inside Macintosh: Text*. QuickDraw GX and other Macintosh managers coexist without conflict, however, and you can use both in the same program. Furthermore, for tasks outside the scope of QuickDraw GX you need to use other parts of the system software. For example, for multilingual word breaks and line breaks, you need to use the Macintosh Text Utilities.

Before you read this book, you should already be familiar with *Inside Macintosh: QuickDraw GX Objects*. Figure P-1 on page xxii shows the suggested reading order for the QuickDraw GX books. A pictorial overview of *Inside Macintosh*, including the QuickDraw GX suite of books, appears at the back of this book.

## What to Read

---

This book is for all QuickDraw GX programmers. You can read the chapters in any order, except that the first chapter introduces concepts that the others build on:

Chapter 1, “Introduction to QuickDraw GX Typography,” provides an overview of typography and QuickDraw GX and describes how text is stored, measured, and displayed. Read this chapter first.

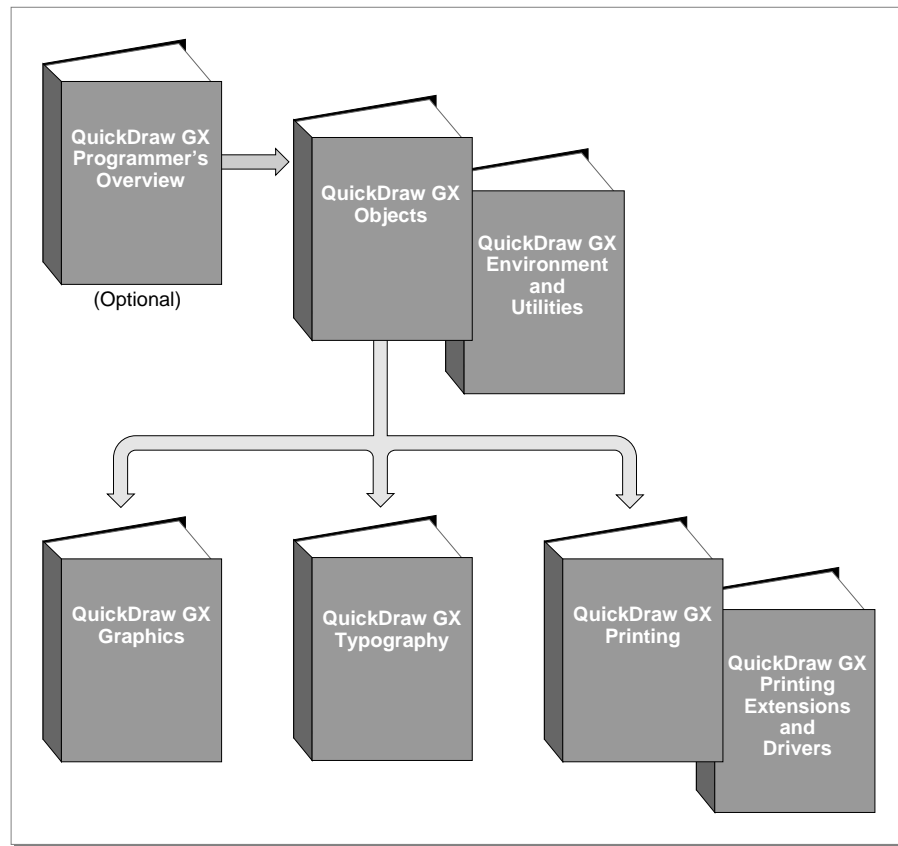
Chapter 2, “Typographic Shapes,” describes how to create and use QuickDraw GX shapes for the text you draw.

Chapter 3, “Text Shapes,” describes how to create and use QuickDraw GX text shapes.

Chapter 4, “Glyph Shapes,” describes how to create and use QuickDraw GX glyph shapes.

Chapter 5, “Layout Shapes,” describes how to create and use QuickDraw GX layout shapes.

Chapter 6, “Typographic Styles,” describes how to use the typographic properties of QuickDraw GX style objects.

**Figure P-1** Roadmap to the QuickDraw GX suite of books

Chapter 7, “Font Objects,” describes the properties of QuickDraw GX font objects and how to use them.

Chapter 8, “Layout Styles,” describes how to use the layout shape-related properties of QuickDraw GX style objects.

Chapter 9, “Layout Line Control,” describes how to measure and control lines for layout shapes.

Chapter 10, “Layout Carets, Highlighting, and Hit-Testing,” describes how to use carets, how to highlight, and how to hit-test layout shapes.

## Chapter Organization

---

Most chapters in this book follow a standard general structure. For example, the chapter “Layout Shapes” contains these major sections:

- “About Layout Shapes.” This section provides an overview of the properties of layout shapes.
- “Using Layout Shapes.” This section describes how you can create and manipulate layout shape objects using QuickDraw GX. It describes how to use the most common functions, gives related user interface information, provides code samples, and supplies additional information.
- “Layout Shapes Reference.” This section provides a complete reference for layout shape objects by describing the constants, data types, and functions that you use with layout shapes. Each function description follows a standard format, which gives the function declaration; a description of every parameter; the function result, if any; and a list of errors, warnings, and notices. Most function descriptions give additional information about using the function and include cross-references to related information elsewhere.
- “Summary of Layout Shapes.” This section shows the C interface for the constants, data types, and functions associated with layout shapes.

## Conventions Used in This Book

---

This book uses various conventions to present certain types of information.

### Special Fonts

---

All code listings, reserved words, and names of data structures, constants, fields, parameters, and functions are shown in `Courier` (`this is Courier`).

When new terms are introduced, they are in **boldface**. These terms are also defined in the glossary.

## Types of Notes

---

There are several types of notes used in this book.

### Note

A note formatted like this contains information that is interesting but possibly not essential to an understanding of the main text. The wording in the title may say something more descriptive than just “Note”; for example, “Terminology Note.” ♦

### IMPORTANT

A note like this contains information that is especially important. (An example appears on page 7-13.) ▲

## Numerical Formats

---

Hexadecimal numbers are shown in this format: 0x0008.

The numerical values of constants are shown in decimal, unless the constants are flag or mask elements that can be summed, in which case they are shown in hexadecimal.

## Type Definitions for Enumerations

---

Enumeration declarations in this book are commonly followed by a type definition that is not strictly part of the enumeration. You can use the type to specify one of the enumerated values for a parameter or field. The type name is usually the singular of the enumeration name, as in the following example:

```
enum gxFontPlatforms {
    gxGlyphPlatform = -1,
    gxNoPlatform,
    gxUnicodePlatform,
    gxMacintoshPlatform,
    gxReservedPlatform,
    gxMicrosoftPlatform,
    gxCustomPlatform,
};
typedef long gxFontPlatform;
```

## Illustrations

---

This book uses several conventions in its illustrations.

In illustrations that show object properties, properties that are object references are in *italics*.

In order to focus attention on the key part of some drawings, other parts are printed in gray, rather than in black.

Objects in diagrams, whether shown with their properties or without, are represented by distinctive icons, such as:



See, for example, Figure 2-4 in Chapter 2.

## Development Environment

---

The QuickDraw GX functions described in this book are available using C interfaces. How you access these functions depends on the development environment you are using.

Code listings in this book are shown in ANSI C. They suggest methods of using various functions and illustrate techniques for accomplishing particular tasks. Although most code listings have been compiled and tested, Apple Computer, Inc., does not intend for you to use these code samples in your applications.

## Developer Products and Support

---

APDA is Apple's worldwide source for over three hundred development tools, technical resources, training products, and information for anyone interested in developing applications on Apple platforms. Customers receive the quarterly *APDA Tools Catalog* featuring all current versions of Apple development tools and the most popular third-party development tools. Ordering is easy; there are no membership fees, and application forms are not required for most of our products. APDA offers convenient payment and shipping options, including site licensing.

## P R E F A C E

To order products or to request a complimentary copy of the *APDA Tools Catalog*, contact

APDA  
Apple Computer, Inc.  
P.O. Box 319  
Buffalo, NY 14207-0319

Telephone	1-800-282-2732 (United States) 1-800-637-0029 (Canada) 1-716-871-6555 (International)
-----------	---

Fax	1-716-871-6511
-----	----------------

AppleLink	APDA
-----------	------

America Online	APDAorder
----------------	-----------

CompuServe	76666,2405
------------	------------

Internet	APDA@applelink.apple.com
----------	--------------------------

If you provide commercial products and services, call 408-974-4897 for information on the developer support programs available from Apple.