

# The journal of Apple technology.

**Volume Number: 19 (2003)**

**Issue Number: 4**

**Column Tag: Reviews**

## Books on Cocoa Programming

### The field is filling in...

*by Rich Morin*

I've spent the last few months learning my way around Cocoa programming, using a combination of Objective-C and Perl. I'm not anything like a wizard yet, but I'm getting by and even starting to feel a bit confident. I have received lots of support from folks on assorted mailing lists, but these books got heavy usage first. "Read the Fine Manual", and all :-).

Whether you're just starting up the Cocoa learning curve or well along the way, I'd suggest that you consider getting some of these books:

Building Cocoa Applications: A Step-by-Step Guide Simson Garfinkel & Michael K. Mahoney

O'Reilly, 2002; ISBN 0-596-00235-1

622 pp.; \$44.99

Cocoa Programming

Scott Anguish, Erik M. Buck, Donald A. Yacktman

SAMS, 2002; ISBN 0-672-32230-7

1245 pp.; \$59.99

Cocoa Programming for Dummies

Erick Tejkowski

Wiley, 2003; ISBN 0-7645-2613-8

366 pp.; \$24.99

Cocoa Programming for Mac OS X

Aaron Hillegass

Addison-Wesley, 2002; ISBN 0-201-72683-1

383 pp.; \$44.99

Cocoa Recipes for Mac OS X: The Vermont Recipes

Bill Cheeseman

Peachpit Press, 2003; ISBN 0-201-87801-1

752 pp.; \$44.99

Learning Cocoa with Objective-C (2nd edition)

James Duncan Davidson & Apple

O'Reilly, 2002; ISBN 0-596-00301-3

360 pp.; \$34.95

## **Executive Summary**

If you're just starting out, I'd recommend "Cocoa Programming for Dummies". Really. It's economical, a fast read, and will give you a quick and painless introduction to the main topics you'll be studying later.

Next, you'll want to skim the two O'Reilly books ("Learning Cocoa with Objective-C" and "Building Cocoa Applications"). You'll also want to keep them close at hand for reference. If you already have some O'Reilly books, you'll know what to expect from these: clean, solid presentation, consistent organization, good production values, etc..

With this background, you'll be ready to tackle any of the last three volumes. I found each of these to contained things that the others did not, but "Cocoa Programming" was by far the most detailed and definitive of the lot. In fact, if you can only afford to buy one Cocoa programming book, "Cocoa Programming" would be the best use of your money.

## **Cocoa Programming for Dummies**

The goals and approach of this book are about as far from "Cocoa Programming" as it is possible to be. It isn't particularly detailed, let alone definitive, and screenshots fill many of the pages. Nonetheless, it is a worthwhile book. In fact, it's the best "jump start" book I've seen, providing a painless way for "newbies" to pick up the key concepts of Cocoa programming. Just be ready to buy some more books before trying to shift into second :-).

Although the book itself contains relatively little code, the supporting web site ([www.dummies.com/extras](http://www.dummies.com/extras)) has quite a bit. In fact, it provides a 16 MB ".sea" archive with source code, pre-built packages, and more. The examples are all simple, but that's quite appropriate to the introductory nature of the book.

## **Learning Cocoa with Objective-C**

Although I like this book, I was surprised by the number of typos and other small errors I encountered while reading it. I shouldn't be able to find dozens of glitches in any O'Reilly book; given that this is a second edition, I should be hard-pressed to find any at all.

Nonetheless, it's a good introductory book. It covers the basics of Cocoa, Objective-C, OOP, etc.. It won't take you very far into Cocoa programming, but it's a fine lead-in and will also be handy to keep around as a reference. Like "Cocoa Programming for Mac OS X", it contains a wealth of useful diagrams, which frequently help to clarify the topic under discussion.

## **Building Cocoa Applications**

Although I found this book useful, it is not my idea of a "Step-by-Step Guide". After short introductions to Aqua and Apple's development tools (~60 pages each), it jumps headlong into three large "Applications":

- "Calculator: Building a Simple Application (152 pp.)
- "MathPaper: A Multiple-Document, Multiprocess Application (154 pp.)
- "GraphPaper: A Multithreaded, Mouse-Tracking Application (150 pp.)

Each application is used as a vehicle for explaining certain aspects of Cocoa programming. For instance, the Calculator app is used to introduce "Nibs and Icons", "Delegation and Resizing", "Events and Responders", and "Darwin and the Window Server".

## **Cocoa Programming for Mac OS X**

This book "splits the difference" between "Cocoa Programming" and "Cocoa Programming for Dummies". It moves a lot faster than one and a lot slower than the other. It has a nice sprinkling of screenshots, many useful diagrams, and a fair amount of detail. I'm particularly happy with the coverage of design techniques, programming pitfalls, and other "incidental" issues.

The writing style is quite informal and relaxed, but the content is very solid. If you can't afford to attend a course at "Big Nerd Ranch" (the author's training facility), you may find that this book provides a similar set of information, delivered in a similar manner.

## **Cocoa Recipes for Mac OS X**

This book is based on the "Vermont Recipes" application, a "sampler" which shows off most of the simpler Cocoa widgets (e.g., buttons, drawers, sliders, tabs, and text items), complete with error-checking code and other niceties. , tab views, and drawers). It walks the reader, in painstaking detail, through the steps needed to reproduce and enhance the application.

The web site ([www.stepwise.com/Articles/VermontRecipes](http://www.stepwise.com/Articles/VermontRecipes)) contains both the app and the project files that were used to build it. If you like to work your way through someone else's examples, this site (and the accompanying book) should be very worthwhile. Alternatively, the app provides a fast way to play around with a lot of Cocoa widgets.

Because my own approach involves blundering my way through my own projects, I haven't used this book (or any of these books, really) as the authors intended. Nonetheless, I was able to use it from time to time as a way to clarify obscure points of Cocoa programming and tool use.

## **Cocoa Programming**

As noted above, this is a very detailed and definitive volume. In fact, I found it rather intimidating at first: lots of text, a fair amount of code, and only a small number of screenshots and tables. With time, however, I realized that this book was answering my programming questions better than any of the other volumes.

I like this book so well, in fact, that I'd like to see it expanded into two volumes. The first could be augmented with more introductory information (e.g., on Apple's developer tools, the Cocoa programming model, etc.) The second could concentrate on some of the more esoteric aspects of the Apple Frameworks.

## **Coming Attractions**

Other Cocoa programming books are on the way. Some of these will fill in holes left by the current crop; others will

attempt to supplant existing books. None of the current books cover Cocoa programming in languages other than Objective-C (e.g., AppleScript, Java, Perl, Python, Ruby, and Visual Basic); I know of at least one such book (on Perl) that is in development.

I would really like to see a detailed reference for the entire range of Apple's developer tools. I have spent far too much time trying to find information on Interface Builder and Project Builder; I'd love to have a single book that covers them, as well as some of the obscure tools in `"/Developer/Tools"` and `"/Developer/Applications"`, in painstaking detail.

---

**Rich Morin**