Open Books

Reviews of Recent Titles on Open Systems Subjects

Active Java—Object-Oriented Programming for the World Wide Web

by Adam Freeman
and Darrel Ince
Addison-Wesley
235 pages; $25.95
ISBN# 0-201-40370-6

Web Site Programming with Java

by David Harms,
Barton C. Fiske
and Jeffrey C. Rice
McGraw-Hill
578 pages; $39.95
ISBN# 0-07-912986-2

Active Java—Object-Oriented Programming for the World Wide Web

Java is no longer just another name for coffee. In this age of hyperactivity, it is the well-suited name for a computer language introduced by Sun Microsystems that is supposed to revolutionize the programming world. Java has been characterized as an object-oriented language that looks and feels like C++ but does not have some of its nasty side effects associated with pointers and memory management. Since the world was put on notice about the existence of Java in March 1995, streams of Internet browsers and FTPers have downloaded source code from http://javasoft.com at an incredible rate. Now we are beginning to see the accompanying proliferation of literature that instructs us about Java. One such book is *Active Java—Object-Oriented Programming for the World Wide Web* by two academics from Open University (the largest university in the U.K.).

A Thorough Dissertation

According to the authors, their intention is to provide “a gentle introduction to the Java system which can be read with profit by any reader with programming experience.” They state that the reader will learn 85 percent of the Java language and write stand-alone programs and seamless interfaces, learn more about the object-oriented programs on the Web and learn to access an extensive, growing library of software associated with the Java Development Kit (JDK). Following this plan, the book is divided into two main sections. The first six chapters describe the programming concepts associated with Java; the final six chapters concentrate on the development of Java applications.

Freeman and Ince clearly direct the readers—programmers without C, with C, with C++ or with some experience in object-oriented programming (OOP) from using something like Pascal—to concentrate on chapters appropriate to their backgrounds. If you are not in one of the aforementioned categories, assume that this book would be difficult or unsuitable for you.

First there is a brief description of the Java language and the Internet and its array of browsers. Next come discussions of objects, classes and OOP, with plenty of interesting examples (such as a simple air-traffic control system, robots, an invoicing and employee records system and a printer spooler).

The book proceeds to a study of Java’s classes, data types and control structures. There is also a focus on Java libraries: I/O, Vectors, AWT and the java.net library. The authors examine the JDK and its various tools, including compilers, debuggers and decompilers. Finally, you build a Java applet (the Hello World program) and a Java application using threads, and examine the differences between an applet and a program.

At the end is a review of the Java internals, which are always handy to know about a language when things do not work out as advertised. This kind of knowledge is crucial, as tools can build or destroy based on the proper understanding of the theory and/or paradigm of the focus of a language. This has been one of the key stumbling blocks of C++ and other object-oriented languages that are advertised to be cure-alls for programming ills; sometimes they make the situation worse or are just an even trade.

Active Java—Object-Oriented Programming for the World Wide Web is written in a style similar to that of an all-time favorite, Kernighan and Ritchie’s *The C Programming Language*: dry but with a simple elegance that leaves a profound impression on the reader. Personally, I enjoyed the analytical presentation and style of the book, and recommend it for the reader interested in object-oriented Web programming. However, make sure that you have access to the JDK source first, as the book comes without it. This would be an excellent book for the classroom. For the self-driven, empowered individual of beginner programmer status, however, it might be too advanced.

Web Site Programming with Java

Programmers and software development organizations have always been looking for a software development environment that will enable them to create in a short time brilliant programs that are robust and error-free, and at the same time easy to maintain. For the last several years, object-oriented programming in C++ has been the heir apparent to this “holy grail” of software development. But there have been so many significant problems associated with the C++ language and its associated development environment that this new savior for the common programmer has not taken the masses by storm.

Java is touted by Sun and others as “C++ done right.” In the same breath these promoters have dubbed the Internet programming environment as the focal arena for Java’s strengths. *Web Site Programming with Java* provides a flashy but well-documented introduction to Java.

By Paul Comey

Some technical books are dry, fact-filled documents that are difficult to understand without the assistance of a formal class and an expert teacher. Because Sun has licensed Java for free, a grass-roots initiative has occurred in its acceptance by the programming community. In light of this trend, an instructional book for the empowered individual programmer would be welcome.

With highlighting of key points, significant background explanations and numerous examples, this book is pleasing to the eye as well as full of technical meat. It also does not insult the self-taught student by making him or her feel technically inadequate.

User-Friendly Roadmap

The first part of the book focuses is an introduction and background to Java. A meaningful explanation of the uses, goals and intentions helps to prepare the reader to understand the book. It deals with the fundamental technical cores, such as Java basics (the structure of a simple Java program); Internet basics (how the Internet works with a Web page); the interaction of Java and the previously described Internet architecture; and the JDK development environment. This provides a good base of the primary concepts that become more apparent as the Java programmer-in-training proceeds with this book. This can be the time when a good instructor will emphasize and repeat the key points that will bear fruit later in one's education; the writers have emulated this phase well.

Part 2 investigates the more advanced features in the language, such as the Hello World program; the object-oriented programming paradigm as applied with Java; Java syntax data types and operators, statements and program flow control; I/O; Java classes; and error and exception handling. Other key functional topics include the Java debugger, using threads in your Java application and linking to native C code.

From a real-world perspective, the debugger and C linking subjects are significant, and substantial exploration of these topics is a bonus. However, C++ interface topics are not covered (such as CORBA and OLE), and the debugger and linking topics cover only 10 pages each (out of a total of almost 600). For an introductory text this may be sufficient, but further topic investigation would be needed in a more advanced book.

Part 3 deals with disparate random topics that fill in the gaps of other traditional language introductions. Among them are designing Java programs and client/server programs (with a significant exercise example), the use of animation with Java and a short transitional discussion about using Java on the Web.

Currently, the latest technospeak about client/server involves the use of thick and thin clients. A *thick client* is the traditional single-user PC application, while the *thin client* would be something like a Web browser as a client interface. Although the authors do not significantly delve into the thin client, the reader is at least exposed to some of the technical design aspects of this architectural paradigm.

Part 4 "puts it all together" in one extended example of using Java in a Web application (as promised in the book's title). Aspects of this project include Web site planning; choosing a Web site type and platform; creating Web page content; Web security issues; and a "cookbook"-style guided implementation of a Web site based on chunks of examples that can be found on the companion CD-ROM, which is a helpful feature. Often the most frustrating thing about self-teaching books is the lack of available resources or the significant dependency on other resources not included in the book. The authors and publisher have supplied just about everything except the hardware and operating system that you need to follow the book's instruction.

Web Site Programming with Java is a worthwhile book for a beginner or intermediate-level audience with C or C++ experience. Its strengths lie in its user-friendliness, which comes in handy when trying to teach yourself a language. Plentiful examples highlight the key points of the book. Focus is maintained with a logical flow of topics, most of which are covered in depth.

I would not recommend this book for any investigation of advanced topics, and its references for further investigation are probably not what they should be (this might be due to the newness of the language; Java has already reached level 1.0.1 due to a security patch that had to be developed). Beyond that, since the book focuses on Web development, the reader should bear in mind that Java is a significant general-purpose language with features that transcend the Web. Still, the application of Java on the Web is an appropriate place to start as the application of the language evolves.

***Paul Comey** is a senior software engineer for Sun Microsystems in Mountain View, CA.*