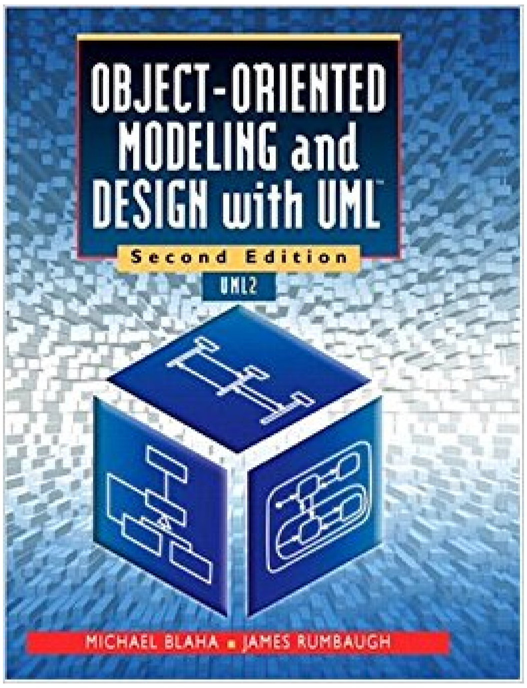
Object-Oriented Modeling And Design With UML (2nd Edition)


OBJECT-ORIENTED MODELING and DESIGN with UML
Second Edition
UML2
MICHAEL BLAHA • JAMES RUMBAUGH

Synopsis

This revision offers a crisp, clear explanation of the basics of object-oriented thinking via UML models, then presents a process for applying these principles to software development, including C++, Java, and relational databases. An integrated case study threads throughout the book, illustrating key ideas as well as their application.Ã Â

Book Information

Paperback: 496 pages

Publisher: Pearson; 2 edition (December 9, 2004)

Language: English

ISBN-10: 0130159204

ISBN-13: 978-0130159205

Product Dimensions: 6.9 x 1.1 x 8.9 inches

Shipping Weight: 2 pounds (View shipping rates and policies)

Average Customer Review: 3.3 out of 5 stars 9 customer reviews

Best Sellers Rank: #135,722 in Books (See Top 100 in Books) #10 inÃ Â Books > Computers & Technology > Programming > Software Design, Testing & Engineering > UML #64 inÃ Â Books > Textbooks > Computer Science > Object-Oriented Software Design #219 inÃ Â Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Object-Oriented Design

Customer Reviews

In a sense, Jim Rumbaugh and Mike Blaha are two of the "founding fathers" of UML. They invented UML, the basis for UML, along with Booch notation. Their first edition was crucial to the development and adoption of Object-Oriented methods when they were in their infancy. Blaha is a worldwide consultant and is a partner with Modelsoft Consulting, and SentientPoint Corporation. He is active in the IEEE Computer Society. Rumbaugh is a Distinguished Engineer with the Rational brand of IBM and is one of the original co-designers of UML. He is a highly influential author.

I had to use this book for school. I bought the E-book but can only use it on the Kindle for PC. The manufacture limits how this book can be read and this is a PAIN. The content is not overly helpful either and most of the content can be found on via google much faster and with better explanation.

Had some good stuff, but didn’t help me as much as I’d hoped with my problems at work.

buy for class

Good book

A tough read but then again computer science is always a tough read.

Good Product. Received in a timely manner.

UML is still a pretty new notation language which is developing as we speak. An experienced programmer may run into some small ambiguities, where there is no concrete way to do something.There are various names and notations for several things in UML. The capitalizations/small case for state diagrams may not always match up with other books, and some things can be drawn in more than one way. For instance, a pentagon to describe the entire diagram may exist or not be used at all in some cases.The discrepencies not only show the language's slow progression because of many differing opinions of language developers, but it is also a good indication of how difficult it may be to notice and understand how functions and drawings are to be interpretted between books and software usage. What's also interesting is that Rumbaugh has written various UML books, so he is aware of many of these differences. The question is whether authors as a team could only agree on a certain way to write the book, or if Rumbaugh was more in a hurry with the others to just get books done and printed to make money. If Rumbaugh comes out with a book on his own, it may be worth considering strongly! The same could probably be true for Blaha.Like many UML books probably, the book doesn't help you become aware of something like if an interface cannot be pointing to another interface. That is a more simpler example, but there are much more complex case whereas if you do not have a good grip on computer languages, it probably won't make much sense. Even though I cannot expect a book to cover most of these type of things, I think a book can attempt to cover some of these things so that a beginner could learn more easily. Part of the problem is because the language is still developing, and people just want to get books printed. If you know your UML well, here's your chance!You can still pass a course not knowing any of the idiosyncracies, but at a graduate level, it will not be a fulfilling way to learn.The way the book goes over the content it does cover, it does it very well. There are some things missing such as differing ways to show the same things, or it does not elaborate on topics enough that one would probably need to know for a graduate level course.

One striking feature about this book is the sheer mass of exercises given for each chapter. OO modelling and designing using UML can be treated as a very high level process. Indeed it is certainly possible to do this, after reading the book. But the authors have striven to make as concrete as possible to anyone new to UML.Just as one example, consider Chapter 3, on class modelling. It has some of the ambiance of the designing of relational tables, if perhaps you are more familiar with the latter. Books on relational design often have many examples and problems, to drive home the concepts to students. Likewise here. Granted, if you do hail from a relational background, the style of object oriented modelling, and the usage of UML with which to express this, will be different from what you are used to.Later in the book, another useful feature is how you are challenged to convert pseudocode UML examples into actual code, C++ or Java. A vital skill. Since often a top down approach with UML is where UML encodes a large picture. But you then have to manually instantiate this as code. So the book gives you lots of practice.By the way - the book also has useful tips on good object oriented practices for both C++ and Java. These arose out of the discussion about UML. But you should consider adopting these practices irrespective of whether you use UML or not.

Download to continue reading...

Object-Oriented Modeling and Design with UML (2nd Edition) Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3rd Edition) Systems Analysis and Design: An Object-Oriented Approach with UML UML Distilled: A Brief Guide to the Standard Object Modeling Language (3rd Edition) The Object Primer: Agile Model-Driven Development with UML 2.0 Introduction to Programming with Greenfoot: Object-Oriented Programming in Java with Games and Simulations (2nd Edition) Tools For Structured and Object-Oriented Design (7th Edition) Object-Oriented Analysis and Design with Applications (3rd Edition) Object Oriented Software Development Using Java (2nd Edition) An Object-Oriented Approach to Programming Logic and Design Head First Object-Oriented Analysis and Design Object Oriented Systems Analysis and Design Object-Oriented Design and Patterns Object-Oriented Analysis and Design with the Unified Process (Available Titles CengageNOW) Design Patterns: Elements of Reusable Object-Oriented Software Practical Object-Oriented Design in Ruby: An Agile Primer (Addison-Wesley Professional Ruby) The Object-Oriented Thought Process (4th Edition) (Developer’s Library) Object-Oriented Programming in C++ (4th Edition) Object-Oriented Programming in Java: A Graphical Approach, Preliminary Edition An Introduction to Object-Oriented Programming (3rd Edition)