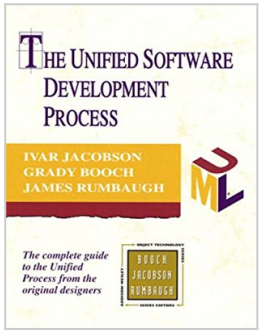

The Unified Software Development Process

By Ivar Jacobson, Grady Booch, James Rumbaugh

The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh

This landmark book provides a thorough overview of the Unified Process for software development, with a practical focus on modeling using the Unified Modeling Language. The Unified Process goes beyond mere object-oriented analysis and design to spell out a proven family of techniques that supports the complete software development life cycle. The result is a component-based process that is use-case driven, architecture-centric, iterative, and incremental. The Unified Process takes full advantage of the industry-standard Unified Modeling Language. This book demonstrates how the notation and process complement one another, using UML models to illustrate the new process in action. The authors clearly describe the semantics and notation of the different higher-level constructs used in the models. Constructs such as use cases, actors, subsystems, classes, interfaces, active classes, processes, threads, nodes, and most relations are described in the context of a model. Object technology practitioners and software engineers familiar with the authors' past work will appreciate The Unified Software Development Process as a useful means of learning the current best practices in software development.

The Unified Software Development Process

By Ivar Jacobson, Grady Booch, James Rumbaugh

The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh

This landmark book provides a thorough overview of the Unified Process for software development, with a practical focus on modeling using the Unified Modeling Language. The Unified Process goes beyond mere object-oriented analysis and design to spell out a proven family of techniques that supports the complete software development life cycle. The result is a component-based process that is use-case driven, architecture-centric, iterative, and incremental. The Unified Process takes full advantage of the industry-standard Unified Modeling Language. This book demonstrates how the notation and process complement one another, using UML models to illustrate the new process in action. The authors clearly describe the semantics and notation of the different higher-level constructs used in the models. Constructs such as use cases, actors, subsystems, classes, interfaces, active classes, processes, threads, nodes, and most relations are described in the context of a model. Object technology practitioners and software engineers familiar with the authors' past work will appreciate The Unified Software Development Process as a useful means of learning the current best practices in software development.

The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh Bibliography

- Sales Rank: #1329222 in Books
- Published on: 1999-02-14
- Original language: English
- Number of items: 1
- Dimensions: 9.55" h x 1.25" w x 7.72" l, 2.21 pounds
- Binding: Hardcover
- 512 pages

Editorial Review

Amazon.com Review
A *software process* defines the steps required to create software successfully. Written by the same authors who brought you the Unified Modeling Language (UML), *The Unified Software Development Process* introduces a new standard for creating today's software that will certainly be useful for any software developer or manager who is acquainted with UML.

Early sections introduce four basic principles of the unified process: that software should stress *use cases* (which show how it interacts with users), that the process is *architecture-centric*, and that it is *iterative* and *incremental*. The authors then apply these principles to their software process, which involves everything from gathering system requirements to analysis, design, implementation, and testing. The use-case examples are excellent and include concrete examples drawn from such areas as banking and inventory control.

The authors point out the connection between UML document types (like use cases, class diagrams, and state transition diagrams) with various models used throughout the software process. They provide very short, real-world examples that illustrate how their ideas have been successfully applied. The straightforward tour of the new unified software process gets extra elaboration--along with some advice--in later chapters that further describe the author's ideas on design. With the weight of these three expert authors behind it, readers can expect *The Unified Software Development Process* to be an important book and one that will be valuable to any working designer or manager. --*Richard Dragan*

From the Back Cover

This landmark book provides a thorough overview of the Unified Process for software development, with a practical focus on modeling using the Unified Modeling Language (UML). The Unified Process goes beyond mere object-oriented analysis and design to spell out a proven family of techniques that supports the complete software development life cycle. The result is a component-based process that is use-case driven, architecture-centric, iterative, and incremental.

The Unified Process takes full advantage of the industry-standard Unified Modeling Language. This book demonstrates how the notation and process complement one another, using UML models to illustrate the new process in action. The authors clearly describe the semantics and notation of the different higher-level constructs used in the models. Constructs such as use cases, actors, subsystems, classes, interfaces, active classes, processes, threads, nodes, and most relations are described in the context of a model. Object technology practitioners and software engineers familiar with the authors' past work will appreciate ***The Unified Software Development Process*** as a useful means of learning the current best practices in software development.

0201571692B04062001

About the Author

Ivar Jacobson, Ph.D., is “the father” of many technologies, including components and component architecture, use cases, modern business engineering, and the Rational Unified Process. He was one of the three amigos who originally developed the Unified Modeling Language. He is the principal author of five best-selling books on these methods and technologies, in addition to being the coauthor of the two leading

books on the Unified Modeling Language. Ivar is a founder of Jaczone AB, where he and his daughter and cofounder, Agneta Jacobson, are developing a ground-breaking new product that includes intelligent agents to support software development. Ivar also founded Ivar Jacobson Consulting (IJC) with the goal of promoting good software development practices throughout teams worldwide.

Grady Booch, is the Chief Scientist at Rational Software Corporation and developer of the Booch Method of object-oriented analysis and design. He is also co-developer of the Unified Modeling Language (UML). Widely recognized for these and many contributions in the field, he is a popular speaker at technology conferences around the world. Booch has twice received Software Development magazine's coveted Jolt-Cola Product Excellence Award for his seminal text, *Object-Oriented Analysis and Design with Applications*. Dr. James Rumbaugh is one of the leading object-oriented methodologists. He is the chief developer of the Object Modeling Technique (OMT) and the lead author of the best-selling book Object-Oriented Modeling and Design. Before joining Rational Software Corporation in October 1994, he worked for more than 25 years at General Electric Research and Development Center in Schenectady, New York.

He has been working on object-oriented methodology and tools for many years. He developed the DSM object-oriented programming language, the state tree model of control, the OMT object modeling notation, and the Object Modeling Tool graphic editor. The foundations for the OMT notation were developed more than 10 years ago with Mary Loomis and Ashwin Shah of Calma Corporation. The OMT methodology was developed at GE R&D Center with coauthors Mike Blaha, Bill Premerlani, Fred Eddy, and Bill Lorensen.

Dr. Rumbaugh received his Ph.D. in computer science from MIT. During his Ph.D. research under Professor Jack Dennis, Dr. Rumbaugh was one of the inventors of data flow computer architecture. His career has dealt with semantics of computation, tools for programming productivity, and applications using complex algorithms and data structures. Dr. Rumbaugh has published journal articles on his work and has spoken at leading object-oriented conferences. He writes a regular column for the Journal of Object-Oriented Programming.

Dr. Rumbaugh is the lead author of the recent best-selling book Object-Oriented Modeling and Design, published by Prentice Hall. His latest book, *OMT Insights: Perspectives on Modeling from the Journal of Object-Oriented Programming*, was released in October 1996. He and his colleagues developed the OMT methodology described in the book based on real-world applications at GE, and they have worked to extend the original methodology. He has taught courses based on the methodology to different audiences around the world, ranging from one-hour seminars to intensive several-day training courses.

He has a B.S. in physics from MIT, an M.S. in astronomy from Caltech, and a Ph.D. in computer science from MIT.

During his career at GE, he worked on a variety of problems, including the design of one of the first time-sharing operating systems, early work in interactive graphics, algorithms for computed tomography, use of parallel machines for fast image generation, VLSI chip design, and finally, object-oriented technology.

Jim developed OMTool, an interactive graphical editor for manipulation of object model diagrams. The editor is commercially available. In addition, he led a five-year programming effort producing production-quality software.

In addition, Jim was the manager of the Software Engineering Program at GE, where he led a team of eight to ten Ph.D. and M.S. scientists performing research in software engineering in the areas of algorithm development, programming languages, program proving, and VLSI computer-aided design. In addition, he performed personal research.

Jim developed Chipwright, an interactive graphical CAD system for VLSI layout with incremental design rule checking. He also led a team of four programmers in implementation.

Jim developed and implemented the object-oriented language DSM, combining object-oriented concepts with database concepts and distributed it within GE for use on production applications. The language was heavily used at Calma Corporation and was extensively extended based on user feedback with a preliminary version.

Jim also developed Vista, a hierarchical interactive standard graphics system (similar to the PHIGS system) written in the object-oriented DSM language. He implemented user-interface applications based on this system, including a configuration-management tool and a user-interface generation tool.

Jim developed the concept of state trees, a structured extension of finite state machines incorporating a new model of object-oriented control. He applied it to the design of user interfaces, and the technique was used as a main aspect of the CHIDE user-interface system developed by colleagues at GE-CRD. Later, it was used in the OMTool object editor.

Jim also developed the Flow Graph System, a generic interactive graphic system for controlling a network of design engineering jobs, including management of multiple versions of data and coordination of information flow among applications. He received a patent on the underlying concepts.

In addition, Jim developed algorithms for the reconstruction of images for computerized tomography using fewer input points and with reduced noise in the reconstructed images. He also developed algorithms for display of three-dimensional images in real time using array processors, and he developed Parallax, a language for programming pipelined array processors.

Jim has served on various committees, including the OOPSLA Program Committee and the TOOLS Program Committee.

Users Review

From reader reviews:

Georgetta Watson:

In other case, little folks like to read book The Unified Software Development Process. You can choose the best book if you want reading a book. Providing we know about how is important the book The Unified Software Development Process. You can add expertise and of course you can around the world by just a book. Absolutely right, mainly because from book you can learn everything! From your country until eventually foreign or abroad you will find yourself known. About simple factor until wonderful thing you can know that. In this era, you can open a book or perhaps searching by internet gadget. It is called e-book. You may use it when you feel fed up to go to the library. Let's read.

Ilene Venne:

Spent a free a chance to be fun activity to accomplish! A lot of people spent their leisure time with their

family, or all their friends. Usually they doing activity like watching television, likely to beach, or picnic inside the park. They actually doing same every week. Do you feel it? Will you something different to fill your own free time/ holiday? Could possibly be reading a book might be option to fill your free of charge time/ holiday. The first thing that you'll ask may be what kinds of publication that you should read. If you want to try look for book, may be the book untitled The Unified Software Development Process can be fine book to read. May be it could be best activity to you.

Rosalva Nichols:

The Unified Software Development Process can be one of your beginning books that are good idea. We recommend that straight away because this guide has good vocabulary that could increase your knowledge in language, easy to understand, bit entertaining but still delivering the information. The article author giving his/her effort to place every word into enjoyment arrangement in writing The Unified Software Development Process nevertheless doesn't forget the main stage, giving the reader the hottest along with based confirm resource data that maybe you can be one among it. This great information could drawn you into completely new stage of crucial thinking.

Lydia Donaldson:

Your reading 6th sense will not betray anyone, why because this The Unified Software Development Process reserve written by well-known writer who knows well how to make book that could be understand by anyone who else read the book. Written with good manner for you, still dripping wet every ideas and producing skill only for eliminate your own personal hunger then you still uncertainty The Unified Software Development Process as good book not only by the cover but also through the content. This is one book that can break don't determine book by its include, so do you still needing yet another sixth sense to pick this!? Oh come on your studying sixth sense already told you so why you have to listening to a different sixth sense.

Read The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh for online ebook

The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh Free PDF d0wnl0ad, audio books, books to read, good books to read, cheap books, good books, online books, books online, book reviews epub, read books online, books to read online, online library, greatbooks to read, PDF best books to read, top books to read The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh books to read online.

Online The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh ebook PDF download

The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh Doc

The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh Mobipocket

The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh EPub