



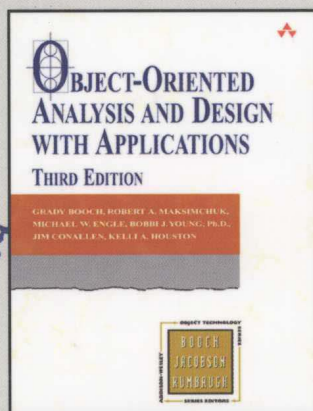
Object-Oriented Analysis and Design with Applications

Third Edition

[美] Grady Booch Robert A. Maksimchuk Michael W. Engle 著
Bobbi J. Young, Ph.D. Jim Conallen Kelli A. Houston

面向对象分析与设计

(第3版)(英文版)



人民邮电出版社
POSTS & TELECOM PRESS



典藏原版书苑

面向对象分析与设计

(第3版)(英文版)

Grady Booch Robert A. Maksimchuk

[美] Michael W. Engle Bobbi J. Young, Ph.D. 著

Jim Conallen Kelli A. Houston

人民邮电出版社

北京

图书在版编目 (CIP) 数据

面向对象分析与设计: 第3版: 英文 / (美) 布奇(Booch, G.)
等著. —北京: 人民邮电出版社, 2008.4
(典藏原版书苑)
ISBN 978-7-115-17306-5

I. 面… II. 布… III. 面向对象语言—程序设计—英文
IV. TP312

中国版本图书馆 CIP 数据核字 (2007) 第 205835 号

版 权 声 明

Original edition, entitled Object-Oriented Analysis and Design with Applications (Third Edition), 9780201895513 by Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Ph.D., Jim Conallen, Kelli A. Houston, published by Pearson Education, Inc, publishing as Addison Wesley Professional, Copyright © 2007 by Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

China edition published by PEARSON EDUCATION ASIA LTD., and POSTS & TELECOMMUNICATIONS PRESS Copyright © 2008.

This edition is manufactured in the People's Republic of China, and is authorized for sale only in People's Republic of China excluding Hong Kong, Macau and Taiwan.

仅限于中华人民共和国境内 (不包括中国香港、澳门特别行政区和中国台湾地区) 销售。

本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签。无标签者不得销售。

典藏原版书苑

面向对象分析与设计 (第3版) (英文版)

◆ 著 [美] Grady Booch Robert A. Maksimchuk
Michael W. Engle Bobbi J. Young, Ph.D.
Jim Conallen Kelli A. Houston

责任编辑 刘映欣

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京铭成印刷有限公司印刷
新华书店总店北京发行所经销

◆ 开本: 800×1000 1/16
印张: 44.5
字数: 959 千字 2008 年 4 月第 1 版
印数: 1—3 000 册 2008 年 4 月北京第 1 次印刷

著作权合同登记号 图字: 01-2008-0491 号

ISBN 978-7-115-17306-5/TP

定价: 89.00 元

读者服务热线: (010) 67132692 印装质量热线: (010) 67129223
反盗版热线: (010) 67171154



内容提要

本书是 UML 创始人 Grady Booch 的代表作之一，书中介绍的概念都基于牢固的理论基础。同时，本书又是一本注重实效的书，面向架构师和软件开发者等软件工程实践者的实际需要。本书通过大量例子说明了基本概念，解释了方法，并展示了在不同领域的成功应用。全书分为理论和应用两部分。理论部分深刻剖析了面向对象分析与设计（OOAD）的概念和方法。应用部分连续列出了 5 个不同类型、不同领域的应用，描述如何从初始阶段到移交阶段将 OOAD 理论和方法应用到项目中。应用部分所涉及的领域包括系统架构、数据获取、密码分析、控制系统和 Web 开发，还给出了一些关于重要问题的有效建议，包括分类、实现策略和高性价比的项目管理。书中的表示法采用最新的 UML 2.0，因此本书是学习 UML 2.0 不可多得的参考书。

本书作者基于长期丰富的经验，提出了改进的对象开发方法，用于解决系统和软件开发者面临的复杂问题，非常适合实际系统和软件的开发、系统分析师或构架师、项目经理阅读。本书主要阐述了软件开发的方法，也可以作为高等院校软件工程和高级编程课程的教材使用。

To Jan
my friend, my lover, my wife
—Grady

前言

人类渴望得到精神上的宁静、美学上的成就、家庭的安全、正义以及自由。这一切都不能通过工业化的生产效率来直接满足。但是，生产效率让人们享有充足的物品，而不至于与匮乏苦苦斗争。这为精神、美学和家庭事务赢得了时间，也使得社会能够将一些特殊的技能赋予司法机构以及维护权利的机构。

Harlan Mills

DPMA and Human Productivity

作为计算机专业人员，我们努力地去构建能工作而且有用的系统。作为软件工程师，我们面临着在计算资源和人力资源有限的条件下创建复杂系统的任务。面向对象（OO）技术已经发展为管理许多不同种类的系统中内在复杂性的手段。对象模型已被证明是非常有力和统一的概念。

对第2版的改动

在本书第2版出版以后，我们看到了一些重要的技术进步。下面列出了其中一些突出的进步：

- 与因特网的高带宽、无线连接已经非常普遍；
- 纳米技术已经出现，并开始提供有价值的产品；
- 我们的机器人在火星表面漫步；
- 计算机生成的特效让电影业能够完全逼真地再现任何想象中的世界；
- 出现了个人气垫船；
- 手机已无处不在，使用非常方便；
- 我们获得了人类基因图谱；
- 面向对象技术已经在工业软件开发中成为主流技术。

在世界各地都能见到面向对象技术被使用。但是，我们仍然遇到许多人，他们还没有采用面向对象的开发方式。对于这两类人，本书的新版本都很有价值。

对于面向对象分析与设计（OOAD）的新手，本书提供了下列信息：

- 面向对象的概念支持和演进式的观点；
- 如何在系统开发生命周期中应用 OOAD 的例子；
- 对系统和软件开发中使用的标准表示法统一建模语言（UML2.0）的介绍。

对于有经验的 OOAD 实践者，本书从不同的角度提供了价值：

- 即使对于有经验的实践者，UML2.0 也是新的。可以看到表示法方面的重要区别。
- 根据前一版本所收到的反馈，更加关注建模。
- 通过本书的概念部分，可以了解在面向对象的世界中，“为什么事情是像它们现在的样子”。许多人可能从没研究过面向对象（OO）概念本身的发展，即使有所了解，在初次学习 OO 方法时，也许未能理解其重要性。

这一版本和以前的版本有 4 项主要区别：

- UML2.0 已经正式通过了，第 5 章“表示法”将介绍 UML2.0。为了加强读者对这种表示法的理解，特别区分了它的基本元素和高级元素。
- 这一版本在应用程序的章节中引入了一些新的领域和背景。例如，应用程序的领域范围很广，包括从高级系统架构到基于 Web 的系统的设计细节等各种不同层次的抽象。
- 在前一版本出版时，作为 OO 编程的概念来说，C++ 相对还是比较新的。读者告诉我们，这种强调不再是主要的考虑。现在有大量的 OO 编程和技术书籍以及培训，还有许多为 OO 开发而设计的编程语言。因此，大部分关于编码的讨论被删除了。
- 最后，响应读者的要求，这一版本更关注 OOAD 建模的方面。应用程序章节将展示如何利用 UML，其中每一章强调了整个开发生命周期中的一个阶段。

本书的目标

本书在面向对象系统构建方面提供了实用指导。它的具体目标是：

- 提供对对象模型的基础概念及其发展变化的正确理解；
- 帮助读者掌握面向对象分析和设计的表示法和过程；
- 教授在不同的问题域中面向对象分析和设计的实际应用。

本书介绍的概念都基于牢固的理论基础，但本书首先是一本注重实效的书，面向架构师和软件开发等软件工程实践者的实际需要。

读者对象

本书既是为计算机专业人员也是为学生编写的。

- 对于实际系统和软件的开发者，本书将展示如何高效地利用面向对象技术来解决

实际问题。

- 对于系统分析师或构架师，本书将利用面向对象的分析与设计，提供一条从需求到实现的途径。我们帮助分析人员或构架师提高识别能力，区分不好的面向对象的结构与好的面向对象的结构，并在现实情况反常时权衡可选的设计方案。也许最重要的就是，我们提供了一些让复杂系统变得有条理的新方法。
- 对于项目经理，本书可以帮助他们更好地理解开发团队的资源分配、软件品质、测量指标以及管理与复杂软件系统相关的风险。
- 对于学生，本书提供了一些必要的指导，使得学生能够开始掌握复杂系统开发的科学与艺术中的一些重要技巧。

本书不仅适合于专业研讨班和个人学习使用，也适合作为高等院校本科生和研究生课程的教材。因为它主要阐述了软件开发的方法，所以非常适合软件工程和高级编程等课程，也可以作为涉及具体面向对象编程语言的课程的补充阅读材料。

本书的组织结构

本书分成 3 个主要部分：概念、方法和应用，其中穿插了大量的补充材料。

概念

第 1 部分研究软件的内在复杂性，以及复杂性的表现方式。本书将对象模型作为一种手段来帮助我们管理这种复杂性；详细地研究了对象模型的基本元素：抽象、封装、模块化、层次结构；讨论了“什么是类？”以及“什么是对象？”等基本问题。由于确定有意义的类和对象是面向对象开发中的关键任务，因此我们花了相当多的时间来研究分类的本质。具体来说，我们研究了生物学、语言学和心理学等其他学科中的分类方法，然后将这些经验应用到发现软件系统中类和对象的问题上。

方法

第 2 部分基于对象模型提出了复杂系统开发的一种方法。针对面向对象的分析与设计，首先提出了一套图形表示法（即 UML），然后是一个通用的过程框架。还研究了面向对象开发的实践，具体来说，就是它在软件开发生命周期中的位置以及它对于项目管理意味着什么。

应用程序

第 3 部分提供了一组 5 个不简单的例子，涉及不同问题域：系统架构、控制系统、密码分析、数据获取和 Web 开发。之所以选择这些问题域，是因为它们是软件工程师实践过程中遇到的复杂问题的代表。展示某些原则如何应用于简单的问题是很容易的，但是我们关注的是为现实世界构建有用的系统，所以我们对如何将对象模型应用于复杂应用程序更加感兴趣。软件系统的开发不同于按菜谱做菜，因此我们强调应用程序的增量式开发，这种开发以一些正确的原则和良好的模型作为指导。

补充材料

本书中穿插了大量的补充资料。多数章节中都有补充材料，这些材料对重要的主题提供了相关的信息。本书包括了一个关于面向对象编程语言的附录，其中总结了一些常见语言的特征。还提供了常用术语的词汇表，以及一个扩展的分类参考书目，列出了关于对象模型的参考资料。

工具说明

读者总是会问创建这本书中的图使用了什么工具。我们主要使用两个很好的工具来画图：IBM Rational Software Architect 和 Sparx Systems Enterprise Architect。为什么不只用一个？市场的实际情况是没有哪一种工具可以做所有的事情。实践 OOAD 的时间越长，最后就会发现有些特别的情况是所有工具都不支持的。（在这种情况下，可能会寻求基本的绘图工具来展示你的想法。）但是不要让这些很少的情况阻止你使用健壮的 OOAD 工具，如我们提到的这两种工具。

本书的阅读方法

本书可以一页一页地读，也可以不按现有的组织形式阅读。如果想对对象模型中基本概念或者面向对象开发的动机有较深的理解，那么就应该从第 1 章开始依次读下去。如果只对面向对象开发分析与设计中的表示法和过程感兴趣，就从第 5 章和第 6 章开始阅读。第 7 章对使用这种方法管理项目的管理者来说特别有用。如果对针对特定问题域的面向对象技术的应用程序更感兴趣，则可以从第 8 章～第 12 章中任选一章或全部阅读。



Acknowledgments

This book is dedicated to my wife, Jan, for her loving support.

Through both the first and second editions, a number of individuals have shaped my ideas on object-oriented development. For their contributions, I especially thank Sam Adams, Mike Akroid, Glenn Andert, Sid Bailin, Kent Beck, Dave Bernstein, Daniel Bobrow, Dick Bolz, Dave Bulman, Kayvan Carun, Dave Collins, Damian Conway, Steve Cook, Jim Coplien, Brad Cox, Ward Cunningham, Tom DeMarco, Mike Devlin, Richard Gabriel, William Genemaras, Adele Goldberg, Ian Graham, Tony Hoare, Jon Hopkins, Michael Jackson, Ralph Johnson, James Kempf, Norm Kerth, Jordan Kreindler, Doug Lea, Phil Levy, Barbara Liskov, Cliff Longman, James MacFarlane, Masoud Milani, Harlan Mills, Robert Murray, Steve Neis, Gene Ouye, Dave Parnas, Bill Riddel, Mary Beth Rosson, Kenny Rubin, Jim Rumbaugh, Kurt Schmucker, Ed Seidewitz, Dan Shiffman, Dave Stevenson, Bjarne Stroustrup, Dave Thomas, Mike Vilot, Tony Wasserman, Peter Wegner, Iseult White, John Williams, Lloyd Williams, Niklaus Wirth, Mario Wolczko, and Ed Yourdon.

A good part of the pragmatics of this book derives from my involvement with complex software systems being developed around the world at companies such as Alcatel, Andersen Consulting, Apple, AT&T, Autotrol, Bell Northern Research, Boeing, Borland, Computer Sciences Corporation, Contel, Ericsson, Ferranti, General Electric, GTE, Holland Signaal, Hughes Aircraft Company, IBM, Lockheed, Martin Marietta, Motorola, NTT, Philips, Rockwell International, Shell Oil, Symantec, Taligent, and TRW. I have had the opportunity to interact with literally hundreds of professional software engineers and their managers, and I thank them all for their help in making this book relevant to real-world problems.

A special acknowledgment goes to Rational for its support of my work. Thanks also to Tony Hall, whose cartoons brighten what would otherwise be just another stuffy technical book. Finally, thanks to my three cats, Camy, Annie, and Shadow, who kept me company on many a late night of writing.

—Grady Booch

I want to thank my family, who, once again, had to deal with those long hours of my absence while working on this project. Thanks to my parents, who gave me their strong work ethic. Thanks to Mary T. O'Brien, who started it all by offering me this opportunity, and thanks to Chris Guzikowski for helping drive this to completion. To my fellow writers, thank you for agreeing to join me on this journey and for all your hard work and contributions toward this project. Last, but absolutely not least, my heartfelt thanks to Grady for all his work those many years ago, creating one of the original, foundational books on object-oriented analysis and design.

—Bob Maksimchuk

I want to express my gratitude to my family for their love and support, which provide the foundation for all my endeavors. I wish to thank Grady for giving me the opportunity to contribute to the third edition of his classic book. Finally, I want to thank Bob Maksimchuk for guiding me in the process of becoming a writer.

—Mike Engle

I would like to dedicate my work to the memory of my mother, Jean Smith, who encouraged me to take on this project. I also want to express my love and appreciation to my family, Russell, Alyssa, and Logan, for their support and encouragement. Thank you, Bob Maksimchuk and Mike Engle, for giving me the opportunity to share in this endeavor.

—Bobbi J. Young

I would like to extend a very special thank you to my husband, Bob, and to my two children, Katherine and Ryan, whose love and support are my true inspiration.

—Kelli A. Houston

Thank you to our reviewers, especially Davyd Norris and Brian Lyons, and to the many other people at Addison-Wesley who worked on this book, especially to Chris Zahn, not only for his development work but also for providing continuity on this long project.



About the Authors

Grady Booch is recognized internationally for his innovative work on software architecture, software engineering, and modeling. He has been with IBM Rational as its Chief Scientist since Rational's founding in 1981. Grady was named an IBM Fellow in March 2003.

Grady is one of the original developers of the Unified Modeling Language (UML) and was also one of the original developers of several of Rational's products. Grady has served as architect and architectural mentor for numerous complex software-intensive projects around the world.

Grady is the author of six best-selling books, including the *UML Users Guide* and the seminal *Object-Oriented Analysis with Applications*. Grady has published several hundred technical articles on software engineering, including papers published in the early 1980s that originated the term and practice of object-oriented design. He has lectured and consulted worldwide.

Grady is a member of the Association for Computing Machinery (ACM), the Institute of Electrical and Electronics Engineers (IEEE), the American Association for the Advancement of Science (AAAS), and Computer Professionals for Social Responsibility (CPSR). He is an IBM Fellow, an ACM Fellow, a World Technology Network Fellow, and a Software Development Forum Visionary. Grady was a founding board member of the Agile Alliance, the Hillside Group, and the Worldwide Institute of Software Architects. He also serves on the advisory board of Northface University.

Grady received his bachelor of science from the United States Air Force Academy in 1977 and his master of science in electrical engineering from the University of California at Santa Barbara in 1979.

Grady lives with his wife and cats in Colorado. His interests include reading, traveling, singing, and playing the harp.

Robert A. Maksimchuk is a Research Director in the Unisys Chief Technology Office. He focuses on emerging modeling technologies to advance the strategic direction of the Unisys 3D-Visual Enterprise modeling framework. Bob brings an abundance of systems engineering, modeling, and object-oriented analysis and design expertise, in numerous industries, to this mission. He is the coauthor of the books *UML for Mere Mortals* and *UML for Database Design* and has also written various articles. He has traveled worldwide as a featured speaker in numerous technology forums and led workshops and seminars on UML and object-oriented development. Bob is a member of the Institute of Electrical and Electronics Engineers (IEEE) and the International Council on Systems Engineering (INCOSSE).

Michael W. Engle is a Principal Engineer with the Lockheed Martin Corporation. He has over 26 years of technical and management experience across the complete system development lifecycle, from project initiation through operations and support. Using his background as a systems engineer, software engineer, and systems architect, Mike employs object-oriented techniques to develop innovative approaches to complex systems development.

Bobbi J. Young, Ph.D., is a Director of Research for the Unisys Chief Technology Office. She has many years of experience in the IT industry working with commercial companies and Department of Defense contractors. Dr. Young has been a consultant mentoring in program management, enterprise architecture, systems engineering, and object-oriented analysis and design. Throughout her career, she has focused on system lifecycle processes and methodologies, as well as enterprise architecture. Dr. Young holds degrees in biology, computer science, and artificial intelligence, and she earned a Ph.D. in management information systems. She is also a Commander (retired) in the United States Naval Reserves.

Jim Conallen is a software engineer in IBM Rational's Model Driven Development Strategy team, where he is actively involved in applying the Object Management Group's (OMG) Model Driven Architecture (MDA) initiative to IBM Rational's model tooling. Jim is also active in the area of asset-based development and the Reusable Asset Specification (RAS). Jim is a frequent conference speaker and article writer. His areas of expertise include Web application development.

He developed the Web Application Extension for UML (WAE), an extension to the UML that lets developers model Web-centric architectures with the UML at appropriate levels of abstraction and detail. This work served as the basis for IBM Rational Rose and Rational XDE Web Modeling functionality.

Jim has authored two editions of the book *Building Web Applications with UML*, the first focusing on Microsoft's Active Server Pages and the latest on J2EE technologies.

Jim's experiences are also drawn from his years prior to Rational, when he was an independent consultant, Peace Corps volunteer, and college instructor, and from his life as a father of three boys. Jim has undergraduate and graduate degrees from Widener University in computer and software engineering.

Kelli Houston is a Consulting IT Specialist at IBM Rational. She is the method architect for IBM's internal method authoring method and is part of the team responsible for integrating IBM's methods. In addition to her method architect role, Kelli also leads the Rational Method Composer (RMC) Special Interest Group (SIG) within IBM and provides consulting and mentoring services to customers and internal IBM consultants on the effective use of RMC.



Contents

Section I Concepts 1

Chapter 1 Complexity 3

- 1.1 The Structure of Complex Systems 4
- 1.2 The Inherent Complexity of Software 7
- 1.3 The Five Attributes of a Complex System 12
- 1.4 Organized and Disorganized Complexity 14
- 1.5 Bringing Order to Chaos 18
- 1.6 On Designing Complex Systems 24

Chapter 2 The Object Model 29

- 2.1 The Evolution of the Object Model 29
- 2.2 Foundations of the Object Model 37
- 2.3 Elements of the Object Model 43
- 2.4 Applying the Object Model 71

Chapter 3 Classes and Objects 75

- 3.1 The Nature of an Object 75
- 3.2 Relationships among Objects 88
- 3.3 The Nature of a Class 92
- 3.4 Relationships among Classes 96
- 3.5 The Interplay of Classes and Objects 111
- 3.6 On Building Quality Classes and Objects 112

Chapter 4 Classification 121

- 4.1 The Importance of Proper Classification 121
- 4.2 Identifying Classes and Objects 126
- 4.3 Key Abstractions and Mechanisms 138

Section II Method 145**Chapter 5 Notation 147**

- 5.1 The Unified Modeling Language 147
- 5.2 Package Diagrams 155
- 5.3 Component Diagrams 163
- 5.4 Deployment Diagrams 171
- 5.5 Use Case Diagrams 175
- 5.6 Activity Diagrams 185
- 5.7 Class Diagrams 192
- 5.8 Sequence Diagrams 206
- 5.9 Interaction Overview Diagrams 213
- 5.10 Composite Structure Diagrams 215
- 5.11 State Machine Diagrams 218
- 5.12 Timing Diagrams 231
- 5.13 Object Diagrams 235
- 5.14 Communication Diagrams 238

Chapter 6 Process 247

- 6.1 First Principles 248
- 6.2 The Macro Process: The Software Development Lifecycle 256
- 6.3 The Micro Process: The Analysis and Design Process 272

Chapter 7 Pragmatics 303

- 7.1 Management and Planning 304
- 7.2 Staffing 308
- 7.3 Release Management 312
- 7.4 Reuse 314
- 7.5 Quality Assurance and Metrics 316

- 7.6 Documentation 320
- 7.7 Tools 322
- 7.8 Special Topics 324
- 7.9 The Benefits and Risks of Object-Oriented Development 326

Section III Applications 331

Chapter 8 System Architecture: Satellite-Based Navigation 333

- 8.1 Inception 334
- 8.2 Elaboration 347
- 8.3 Construction 370
- 8.4 Post-Transition 371

Chapter 9 Control System: Traffic Management 375

- 9.1 Inception 376
- 9.2 Elaboration 385
- 9.3 Construction 396
- 9.4 Post-Transition 411

Chapter 10 Artificial Intelligence: Cryptanalysis 413

- 10.1 Inception 414
- 10.2 Elaboration 421
- 10.3 Construction 427
- 10.4 Post-Transition 446

Chapter 11 Data Acquisition: Weather Monitoring Station 449

- 11.1 Inception 450
- 11.2 Elaboration 463
- 11.3 Construction 474
- 11.4 Post-Transition 487

Chapter 12 Web Application: Vacation Tracking System 489

- 12.1 Inception 490
- 12.2 Elaboration 494
- 12.3 Construction 506
- 12.4 Transition and Post-Transition 534