

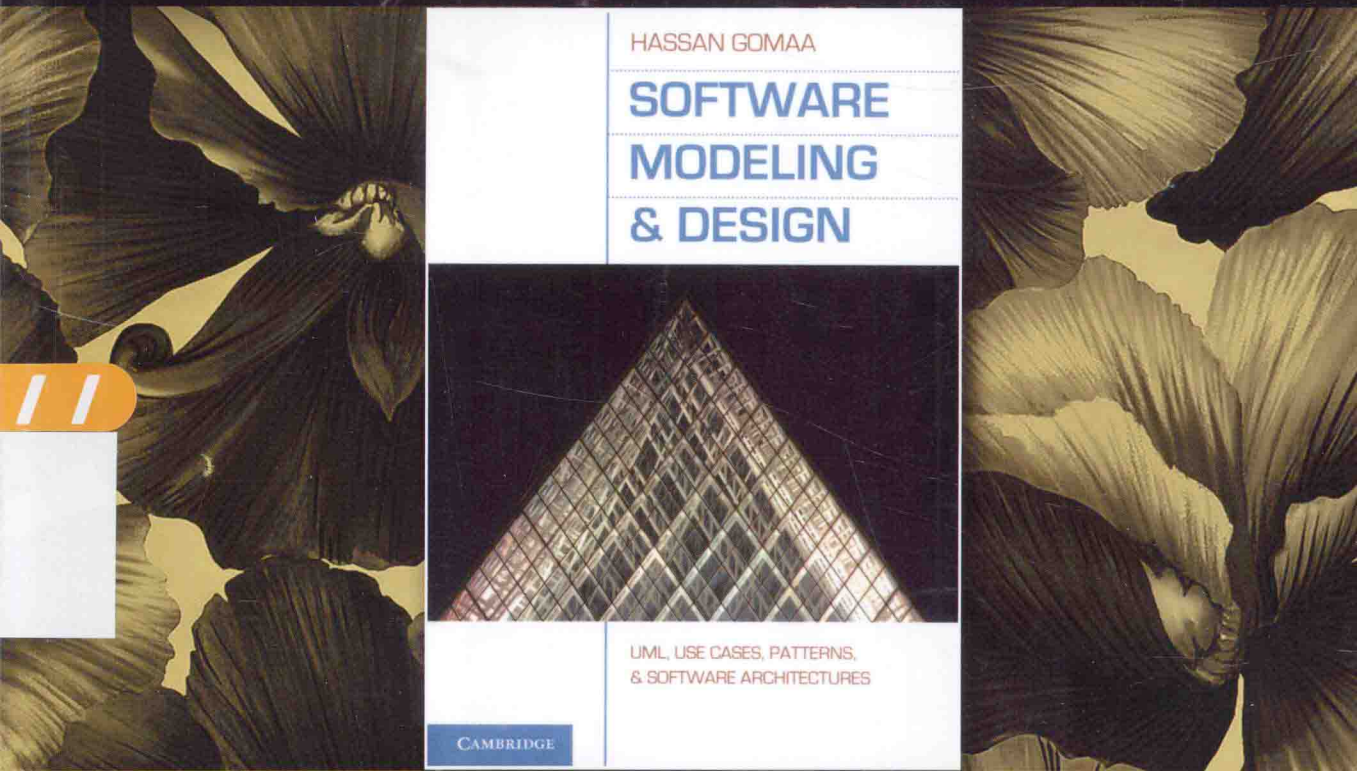
软件建模与设计

UML、用例、模式和软件体系结构

(美) Hassan Gomaa 著 彭鑫 吴毅坚 赵文耘 等译

Software Modeling & Design

UML, Use Cases, Patterns, & Software Architectures



014057448

计 算 机 科 学 丛

TP312
966

软件建模与设计

UML、用例、模式和软件体系结构

(美) Hassan Gomaa 著 彭鑫 吴毅坚 赵文耘 等译

Software Modeling & Design

UML, Use Cases, Patterns, & Software Architectures

HASSAN GOMAA

SOFTWARE MODELING & DESIGN

UML, USE CASES, PATTERNS,
& SOFTWARE ARCHITECTURES

CAMBRIDGE



北航

C1742243



机械工业出版社
China Machine Press

TP312
966

01407448

图书在版编目 (CIP) 数据

软件建模与设计 : UML、用例、模式和软件体系结构 / (美) 戈马 (Gomaa, H.) 著 ; 彭鑫等译 . —北京 : 机械工业出版社 , 2014.6

(计算机科学丛书)

书名原文 : Software Modeling & Design: UML, Use Cases, Patterns, & Software Architectures

ISBN 978-7-111-46759-5

I. 软… II. ①戈… ②彭… III. 面向对象语言 - 程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2014) 第 101580 号

本书版权登记号 : 图字 : 01-2011-3361

This is a simplified Chinese of the following title published by Cambridge University Press: Hassan Gomaa, Software Modeling & Design: UML, Use Cases, Patterns, & Software Architectures (978-0-521-76414-8).

Copyright © 2011 by Hassan Gomaa.

This simplified Chinese for the People's Republic of China (excluding Hong Kong, Macau and Taiwan) is published by arrangement with the Press Syndicate of the University of Cambridge, Cambridge, United Kingdom.

© Cambridge University Press and China Machine Press in 2014.

This simplified Chinese is authorized for sale in the People's Republic of China (excluding Hong Kong, Macau and Taiwan) only. Unauthorized export of this simplified Chinese is a violation of the Copyright Act. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of Cambridge University Press and China Machine Press.

此版本仅限在中华人民共和国境内 (不包括香港、澳门特别行政区及台湾地区) 销售。

本书介绍了关于软件应用建模和设计的知识。从 UML 中的用例到软件体系结构, 本书展示了在解决现实世界问题的过程中如何应用 COMET, 介绍了针对各种体系结构的模式 (包括客户端 / 服务器以及基于构件的软件体系结构中的客户端 / 服务模式, 面向服务的体系结构中的代理、发现和事务模式, 实时软件体系结构中的实时控制模式, 软件产品线体系结构中的分层模式) 和软件质量属性 (包括可维护性、可修改性、可测试性、可追踪性、可伸缩性、可复用性、性能、可用性和安全性)。此外, 还包含了四个案例研究 (包括银行系统、在线购物系统、应急监控系统和自动引导车辆系统)。

本书非常适合作为高等院校相关专业高年级本科生或研究生软件工程课程的教材, 同时适合对大规模软件系统的分析、设计和开发感兴趣的软件工程师阅读参考。

出版发行 : 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 : 100037)

责任编辑 : 迟振春

责任校对 : 殷虹

印刷 : 藁城市京瑞印刷有限公司

版次 : 2014 年 8 月第 1 版第 1 次印刷

开本 : 185mm × 260mm 1/16

印张 : 27.75

书号 : ISBN 978-7-111-46759-5

定价 : 85.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线 : (010) 88378991 88361066

投稿热线 : (010) 88379604

购书热线 : (010) 68326294 88379649 68995259

读者信箱 : hzsj@hzbook.com

版权所有 · 侵权必究

封底无防伪标均为盗版

本书法律顾问 : 北京大成律师事务所 韩光 / 邹晓东

文艺复兴以降，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的传统，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅肇划了研究的范畴，还揭示了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短的现状下，美国等发达国家在其计算机科学发展的几十年间积淀和发展的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起到积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章公司较早意识到“出版要为教育服务”。自1998年开始，我们就将工作重点放在了遴选、移译国外优秀教材上。经过多年的不懈努力，我们与 Pearson, McGraw-Hill, Elsevier, MIT, John Wiley & Sons, Cengage 等世界著名出版公司建立了良好的合作关系，从他们现有的数百种教材中甄选出 Andrew S. Tanenbaum, Bjarne Stroustrup, Brian W. Kernighan, Dennis Ritchie, Jim Gray, Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, Abraham Silberschatz, William Stallings, Donald E. Knuth, John L. Hennessy, Larry L. Peterson 等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及珍藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力襄助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专程为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近两百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍。其影印版“经典原版书库”作为姊妹篇也被越来越多实施双语教学的学校所采用。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证。随着计算机科学与技术专业学科建设的不断完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都将步入一个新的阶段，我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方法如下：

华章网站：www.hzbook.com

电子邮件：hzjsj@hzbook.com

联系电话：(010) 88379604

联系地址：北京市西城区百万庄南街1号

邮政编码：100037



华章教育

华章科技图书出版中心

软件开发的主要困难在于概念化层次上的问题理解和方案规划, 主要包括需求理解、分析和软件设计。其中, 软件体系结构设计是一项最具挑战性的任务。一个高质量的软件体系结构应当完整、准确地实现特定软件系统所要求的各项功能需求, 同时实现性能、可用性、可伸缩性、可维护性等软件质量属性。为此, 设计人员需要全面、准确地理解软件需求和软件质量属性, 掌握各种常见的软件体系结构模式并具有在特定问题环境中综合应用多种模式的能力。此外, 为了准确、规范地描述需求模型、分析模型、体系结构模式以及特定的体系结构设计模型, 统一、规范的可视化建模语言及相应的建模指导也是必不可少的。

本书面向软件体系结构设计, 以统一建模语言(UML)为基础, 从基于用例的需求建模、基于类图的静态建模、基于对象交互分析的动态建模、状态机建模等基本的需求和分析建模手段开始, 逐步介绍了多种软件体系结构模式以及基于模式的软件体系结构设计方法。这些体系结构模式覆盖了当前最为流行的多种软件体系结构类型, 包括面向对象的软件体系结构、客户端/服务器软件体系结构、面向服务的软件体系结构、基于构件的软件体系结构、并发和实时软件体系结构以及软件产品线体系结构。本书还提供了四个详细的案例研究以及一系列模型实例, 所有例子都使用UML 2表示法进行描述。本书不仅适用于高年级本科生及研究生软件分析与设计相关课程教学, 而且还可以作为软件开发和管理相关领域专业人员学习、应用软件分析与设计方法和建模规范的参考。

彭鑫、吴毅坚、赵文耘主要组织并参加了本书的翻译和审校工作。其中, 彭鑫负责第1章、第12~19章、第21~24章、附录及索引的翻译工作, 吴毅坚负责第2~11章、第20章及术语表的翻译工作, 赵文耘审阅了全文。参加本书翻译工作的还包括陈碧欢、林云、钱文亿、赵欣等。

由于本书内容丰富、覆盖面广, 同时译者水平有限, 译文中难免有疏漏或错误, 恳请各位读者批评指正。

概述

本书介绍了一种用例驱动、基于 UML 的软件体系结构建模和设计方法，包括面向对象的软件体系结构、客户端 / 服务器软件体系结构、面向服务的体系结构、基于构件的软件体系结构、并发和实时软件体系结构，以及软件产品线体系结构。本书为软件体系结构设计提供了一套统一的方法，同时针对每种类型的软件体系结构进行了特殊考虑。此外，本书还包含了四个案例研究：一个客户端 / 服务器银行系统、一个面向服务的体系结构的在线购物系统、一个基于构件的分布式应急监控系统和一个实时自动引导车辆系统。

本书介绍了一种基于 UML 的软件建模和设计方法——COMET (Collaborative Object Modeling and Architectural Design Method)。COMET 是一种高度迭代的面向对象的软件开发方法，覆盖了面向对象开发生存周期的需求、分析和设计建模阶段。

本书针对希望使用一种系统化的基于 UML 的方法（从基于用例的需求建模开始，通过静态和动态建模，直至基于体系结构设计模式的软件设计）来设计软件体系结构的读者。

本书内容

目前市场上有很多介绍面向对象分析和设计概念及方法的书。本书针对设计软件体系结构的特定需要，介绍了基于 UML 的软件体系结构设计，由基于用例的需求建模、基于类图的静态建模、基于对象交互分析的动态建模和状态机建模开始，直至基于体系结构设计模式的软件设计。书中所有的例子都使用 UML 2 表示法进行描述。具体而言，本书包括以下内容：

- 全面介绍了应用基于 UML 的面向对象概念进行需求建模、分析建模和设计建模。需求建模包括用例建模，以描述功能性需求，同时通过扩展的方式来描述非功能性需求。分析建模包括静态建模和动态建模（交互建模和状态机建模）。设计建模涉及重要的体系结构问题，包括一种将基于用例的交互图集成到初始软件体系结构以及应用体系结构和设计模式来设计软件体系结构的系统化的方法。
- 提供了一种通用的需求和分析建模方法，并且分析了设计不同类型的软件体系结构所面对的特定问题（每种类型的软件体系结构对应一章），包括面向对象的软件系统、客户端 / 服务器系统、面向服务的系统、基于构件的系统、实时系统和软件产品线。
- 介绍了如何通过首先考虑与相应的软件体系结构类型相关的模式来设计软件体系结构，包括：针对客户端 / 服务器以及基于构件的软件体系结构的客户端 / 服务模式；针对面向服务的体系结构的代理、发现和事务模式；针对实时软件体系结构的实时控制模式；针对软件产品线体系结构的分层模式。
- 介绍了对软件产品质量有重要影响的软件质量属性。其中很多属性都可以在软件体系结构设计时进行考虑和评价。所涉及的软件质量属性包括可维护性、可修改性、可测试性、可追踪性、可伸缩性、可复用性、性能、可用性和安全性。

- 介绍了四个详细的案例研究。这些案例研究按照软件体系结构类型进行组织，包括针对客户端/服务器体系结构的银行系统、针对面向服务的体系结构的在线购物系统、针对基于构件的软件体系结构的应急监控系统，以及针对实时软件体系结构的自动引导车辆系统。
- 本书最后提供了软件体系结构模式分类、针对基于本书进行学术性课程和工业课程教学的教学建议以及术语表、参考文献、索引。此外，本书的大多数章节后都包含相应的练习。

目标读者

本书的目标读者包括学术界以及计算机相关专业的从业人员。学术界的目标读者包括计算机科学及软件工程专业的高年级本科生、研究生以及本领域的研究者。相关从业人员包括参与大规模工业和政府软件系统的分析、设计和开发的分析师、软件架构师、软件设计人员、程序员、项目经理、技术经理、程序经理和质量保障专家。

阅读本书的方式

本书可以通过多种不同的方式进行阅读。可以按章节顺序进行阅读：第1章至第4章提供了介绍性的概念；第5章对COMET/UML软件建模和设计方法进行了概述；第6章至第20章对软件建模和设计进行了深入介绍；第21章至第24章提供了详细的案例研究。

除此之外，有些读者可能希望跳过某些章节，这取决于读者对相关内容的熟悉程度。第1章至第4章是介绍性的内容，有经验的读者可以跳过。熟悉软件设计思想的读者可以跳过第4章。对软件建模和设计特别感兴趣的读者可以直接从第5章开始阅读关于COMET/UML的介绍。不熟悉UML或者对UML 2的变化感兴趣的读者可以阅读第2章以及第5章至第20章。

有经验的软件设计者还可以将本书作为参考书，将各章作为项目到达需求、分析或设计过程中特定阶段时的参考。每一章的内容都相对独立。例如，可以通过第6章了解用例，通过第7章了解静态建模，通过第9章了解动态交互建模。第10章可以作为设计状态机的参考；第12章和附录A可以作为软件体系结构模式的参考；第14章可以作为面向对象软件体系结构的参考；第15章可以作为基于静态模型设计关系数据库的参考。可以通过第16章了解面向服务的体系结构；通过第17章了解基于构件的分布式软件设计；通过第18章了解实时设计；通过第19章了解软件产品线设计。也可以通过阅读案例研究来加强对如何使用COMET/UML方法的理解，因为每个案例研究都对实际应用的设计中需求、分析和设计建模过程所做出的各种决策进行了解释。

Hassan Gomaa

邮件地址：hgomaa@gmu.edu

个人网站：<http://mason.gmu.edu/~hgomaa>

第一部分：概览

第 1 章：引言

本章对软件建模和设计进行了简要介绍，讨论了一些软件设计问题，介绍了软件体系结构，此外还概述了基于 UML 的面向对象分析和设计。

第 2 章：UML 表示法概述

本章介绍了 UML 表示法，包括用例图、类图、交互图、状态图、包图、并发通信图和部署图。本章还介绍了 UML 扩展机制以及 UML 逐渐标准化的发展过程。

第 3 章：软件生存周期模型和过程

本章介绍了用于开发软件的软件生存周期模型，包括瀑布模型、原型过程、迭代过程、螺旋模型和统一过程。本章还对这些过程和模型进行了比较。

第 4 章：软件设计和体系结构概念

本章对一些关键的软件设计概念进行了介绍和讨论，包括类、对象、信息隐藏和继承等面向对象的设计概念，以及使用并发对象的并发处理。此外，本章还介绍了软件体系结构和构件、软件设计模式以及软件质量属性。

第 5 章：软件建模和设计方法概览

本章对软件建模和设计方法进行了概述，包括需求建模、分析建模和设计建模。此外，本章还对本书中所涉及的各种不同类型的软件体系结构进行了概述。

第二部分：软件建模

第 6 章：用例建模

本章首先对需求分析和需求规约进行了概述，然后介绍了用于需求开发的用例建模方法，接下来介绍了一种用例开发方法。本章涉及用例、参与者、识别用例、描述用例以及用例关系。本章还介绍了用于单个用例细化建模的活动图。此外，用例还被扩展用于描述非功能性需求。

第 7 章：静态建模

本章介绍了静态建模的相关概念，包括关联、整体 / 部分关系（组合和聚合）以及泛化 / 特化关系。此外，本章还介绍了一些特殊问题，包括系统边界的建模、信息密集的实体类建模。

第 8 章：对象和类组织

本章介绍了应用类的分类，即类在应用中所扮演的不同角色。主要的分类包括边界对象、实体对象、控制对象和应用逻辑对象。本章还介绍了各种对象相应的行为模式。

第 9 章：动态交互建模

本章介绍了动态交互建模概念。针对每个用例所开发的交互（顺序或通信）图包括主场景和可替换场景。本章还介绍了如何从用例开始开发交互模型。

第 10 章：有限状态机

本章介绍了有限状态机建模的概念。具体而言，一个状态相关的控制类需要使用有限状态机建模并描述为状态图。本章还介绍了事件、状态、条件、动作、进入和退出动作、复合状态、顺序和正交状态。

第 11 章：状态相关的动态交互建模

本章介绍了状态相关的对象交互的动态交互建模，包括状态机和交互图如何相互关联以及二者如何保持一致。

第三部分：软件体系结构设计

第 12 章：软件体系结构概览

本章介绍了软件体系结构的概念，包括软件体系结构的多视图、软件体系结构模式（结构模式和通信模式），提供了一种软件体系结构模式的描述模板，此外还对接口设计进行了介绍和讨论。

第 13 章：软件子系统体系结构设计

本章介绍了软件体系结构设计各种问题，包括从分析到体系结构设计的转换、子系统设计中的关注点分离、子系统组织准则，以及子系统消息通信接口的设计。

第 14 章：设计面向对象的软件体系结构

本章介绍了顺序性软件体系结构的面向对象设计，特别是使用信息隐藏、类和继承等思想的设计。在类接口设计中，类的设计者需要决定哪些信息应该隐藏而哪些信息应该通过类接口（由类所提供的操作组成）暴露出来。本章讨论了契约式设计和顺序性类设计，包括数据抽象类、状态机类、图形用户界面类和业务逻辑类的设计，此外还讨论了类的详细设计。

第 15 章：设计客户端 / 服务器软件体系结构

本章介绍了客户端和服务器的设计，并对客户端 / 服务模式（结构以及行为）、顺序性和并发服务、从静态模型到关系数据库的映射（包括数据库包装器的设计以及逻辑关系数据库的设计）进行了讨论。

第 16 章：设计面向服务的体系结构

本章介绍了面向服务的体系结构的特点，讨论了 Web 服务和模式，包括注册、代理和发现模式，然后介绍了事务模式和事务设计，包括原子事务、两阶段提交协议、复合事务和长事务。本章还介绍了如何设计可复用的服务、如何基于服务复用构建应用、服务协调。

第 17 章：设计基于构件的软件体系结构

本章介绍了基于构件的分布式软件体系结构设计，包括构件接口（供给接口及请求接口）设计。本章还讨论了基于构件的软件体系结构如何使用结构化类和 UML 2 中引入的复合结构图表示法进行描述，该表示法可以描述构件、端口、连接器、供给和请求接口。

第 18 章：设计并发和实时软件体系结构

本章分析了嵌入式实时系统的特点，讨论了并发与控制、实时系统的控制模式、并发任务组织（包括事件驱动的任务、周期性任务和按需驱动的任务）以及任务接口的设计（包括消息通信、事件同步和通过被动对象的通信）。

第 19 章：设计软件产品线体系结构

本章介绍了软件产品线的特点——面向产品族进行共性和可变性建模，讨论了特征建模、可变性建模、软件产品线体系结构和应用工程。本章还介绍了用例、静态和动态模型、软件体系结构中的可变性建模。

第 20 章：软件质量属性

本章介绍了软件质量属性以及如何使用软件质量属性评价软件体系结构的质量。软件质量属性包括可维护性、可修改性、可追踪性、可用性、可复用性、可测试性、性能和安全性。本章还讨论了体系结构设计方法如何支持软件质量属性。

第四部分：案例研究

每个案例研究都详细描述了如何应用此前所介绍的概念和方法设计各种不同的软件体系结构：客户端/服务器软件体系结构、面向服务的体系结构、基于构件的软件体系结构，以及实时软件体系结构。每个案例研究都对建模和设计决策的原理进行了讨论。

第 21 章：客户端/服务器软件体系结构案例研究：银行系统

本章介绍了如何应用软件建模和设计方法来设计一个由一个银行服务器和多个 ATM 客户端组成的客户端/服务器系统——银行系统。ATM 客户端的设计同时也是并发软件设计的例子。银行服务的设计是顺序性面向对象设计的例子。

第 22 章：面向服务的体系结构案例研究：在线购物系统

本章介绍了如何应用软件建模和设计方法来设计一个面向服务的软件体系结构——在线购物系统，其中包含多个由多个客户端调用的服务并且需要服务代理、服务发现和服务协调。

第 23 章：基于构件的软件体系结构案例研究：应急监控系统

本章介绍了如何应用软件建模和设计方法来设计一个基于构件的软件体系结构——应急监控系统，其中软件构件可以在部署时被分配到硬件配置上。

第 24 章：实时软件体系结构案例研究：自动引导车辆系统

本章介绍了如何应用软件建模和设计方法来设计一个实时自动引导车辆系统（由多个并发任务组成）。该系统是一个工厂自动化系统之系统（system of system）的一部分。

附录 A：软件体系结构模式分类

为便于参照，本附录按照通用模板对本书中所用到的软件体系结构的结构模式、通信模式和事务模式进行了描述。

附录 B：教学考虑

本附录描述了基于本书进行学术性教学和工业课程教学的方法。

致谢

Software Modeling & Design: UML, Use Cases, Patterns, & Software Architectures

衷心感谢各位评审者对本书初稿所提出的建设性意见和建议，包括 Rob Pettit、Kevin Mills、Bran Selic 以及其他未留名的评审者。十分感谢我在乔治梅森大学所开设的软件设计和可复用的软件体系结构课程的学生，感谢他们的热情、奉献和宝贵的反馈。同样十分感谢 Koji Hashimoto、Erika Olimpiew、Mohammad Abu-Matar、Upsorn Praphamontripong 和 Sylvia Henshaw 对本书的插图所做出的辛勤和细致的工作。感谢剑桥大学出版社的编辑和出版工作人员，包括 Heather Bergman、Lauren Cowles、David Jou、Diane Lamsback，以及 Aptara 的出版工作人员。没有他们，这本书将无法面世。

衷心感谢培生教育出版集团允许我使用此前出版的书籍中的一些材料，包括《Designing Concurrent, Distributed, and Real-Time Applications with UML》以及《Designing Software Product Lines with UML》。

最后但并非不重要，谢谢我的妻子 Gill 对我的鼓励、理解和支持。

出版者的话	1
译者序	1
前言	1
本书导读	1
致谢	1

第一部分 概览

第 1 章 引言	2
1.1 软件建模	2
1.2 面向对象方法与统一建模语言	2
1.3 软件体系结构设计	3
1.4 方法和表示法	3
1.5 COMET: 一种基于 UML 的软件应用 建模和设计方法	4
1.6 UML 标准	4
1.7 软件体系结构的多视图	5
1.8 软件建模和设计方法的发展	5
1.9 面向对象分析和设计方法的发展	7
1.10 并发、分布式和实时设计方法	8
1.11 总结	8
练习	8
第 2 章 UML 表示法概述	10
2.1 UML 图	10
2.2 用例图	10
2.3 类和对象	11
2.4 类图	12
2.4.1 关联	12
2.4.2 聚合和组合层次	12
2.4.3 泛化和特化层次	13
2.4.4 可见性	13
2.5 交互图	13
2.5.1 通信图	13
2.5.2 顺序图	13
2.6 状态机图	14
2.7 包	15

2.8 并发通信图	16
2.9 部署图	17
2.10 UML 扩展机制	17
2.10.1 构造型	18
2.10.2 标记值	18
2.10.3 约束	18
2.11 本书的约定	19
2.11.1 需求建模	19
2.11.2 分析建模	19
2.11.3 设计建模	20
2.12 总结	20
练习	20
第 3 章 软件生存周期模型和过程	22
3.1 软件生存周期模型	22
3.1.1 瀑布生存周期模型	22
3.1.2 瀑布模型的局限性	22
3.1.3 抛弃型原型	24
3.1.4 通过增量开发的演化式原型	24
3.1.5 抛弃型原型和增量开发的结合	24
3.1.6 螺旋模型	27
3.1.7 统一软件开发过程	27
3.2 设计验证和确认	28
3.2.1 软件质量保证	29
3.2.2 软件设计的性能分析	29
3.3 软件生存周期的活动	29
3.3.1 需求分析和规约	29
3.3.2 体系结构设计	29
3.3.3 详细设计	30
3.3.4 编码	30
3.4 软件测试	30
3.4.1 单元测试	30
3.4.2 集成测试	30
3.4.3 系统测试	30
3.4.4 验收测试	30
3.5 总结	31
练习	31

第4章 软件设计和体系结构概念	32
4.1 面向对象概念	32
4.2 信息隐藏	33
4.2.1 面向对象设计中的信息隐藏	34
4.2.2 信息隐藏示例	34
4.2.3 设计信息隐藏对象	36
4.3 继承和泛化/特化	36
4.4 并发处理	37
4.4.1 顺序应用与并发应用	38
4.4.2 并发对象	38
4.4.3 并发对象间的协作	38
4.4.4 同步问题	38
4.4.5 生产者/消费者问题	40
4.4.6 异步消息通信	40
4.4.7 带回复的同步消息通信	40
4.5 设计模式	41
4.6 软件体系结构和构件	41
4.6.1 构件和构件接口	42
4.6.2 连接器	42
4.7 软件质量属性	42
4.8 总结	42
练习	43
第5章 软件建模和设计方法概览	44
5.1 COMET 基于用例的软件生存周期	44
5.1.1 需求建模	44
5.1.2 分析建模	44
5.1.3 设计建模	44
5.1.4 增量软件构建	46
5.1.5 增量软件集成	46
5.1.6 系统测试	46
5.2 COMET 生存周期与其他软件 过程的比较	46
5.2.1 COMET 生存周期与统一软件 开发过程对比	46
5.2.2 COMET 生存周期与螺旋模型对比	47
5.3 需求、分析和设计建模	47
5.3.1 需求建模中的活动	47
5.3.2 分析建模中的活动	47
5.3.3 设计建模中的活动	48
5.4 设计软件体系结构	48

5.5 总结	49
练习	49

第二部分 软件建模

第6章 用例建模	52
6.1 需求建模	52
6.1.1 需求分析	52
6.1.2 需求规约	53
6.1.3 软件需求规约的质量属性	53
6.2 用例	54
6.3 参与者	55
6.3.1 参与者、角色和用户	55
6.3.2 主要和次要参与者	55
6.3.3 对参与者建模	55
6.3.4 谁是参与者	57
6.4 标识用例	57
6.5 用例模型中文档化用例	57
6.6 用例描述示例	58
6.7 用例关系	59
6.8 包含关系	59
6.8.1 包含关系和包含用例示例	60
6.8.2 结构化冗长的用例	61
6.9 扩展关系	61
6.9.1 扩展点	62
6.9.2 扩展点和扩展用例示例	62
6.10 用例组织指南	64
6.11 规定非功能性需求	64
6.12 用例包	65
6.13 活动图	65
6.14 总结	66
练习	67
第7章 静态建模	68
7.1 类之间的关联	68
7.1.1 类图描述关联	68
7.1.2 关联的多重性	69
7.1.3 三元关联	71
7.1.4 一元关联	71
7.1.5 关联类	71
7.2 组合和聚合层次	72
7.3 泛化/特化层次	73

7.4 约束	73	9.1.5 用例和场景	98
7.5 静态建模和 UML	74	9.1.6 通用和实例形式的交互图	99
7.6 系统上下文的静态建模	75	9.2 交互图上的消息序列编号	99
7.7 使用 UML 构造型对类分类	76	9.2.1 交互图上的消息标签	99
7.8 外部类建模	77	9.2.2 交互图上的消息序列编号	100
7.8.1 从外部类开发软件系统上下 文类图示例	78	9.2.3 并发和可替换的消息序列	100
7.8.2 参与者和外部类	78	9.2.4 消息序列描述	101
7.8.3 从参与者开发软件系统上下 文类图示例	79	9.3 动态交互建模	101
7.9 实体类的静态建模	80	9.4 无状态动态交互建模	101
7.10 总结	81	9.5 无状态动态交互建模示例	102
练习	82	9.5.1 查看警报示例	102
第 8 章 对象和类组织	83	9.5.2 下单请求示例	103
8.1 对象和类的组织准则	83	9.6 总结	110
8.2 对应用类和对象建模	83	练习	110
8.3 对象和类的组织分类	84	第 10 章 有限状态机	112
8.4 外部类与软件边界类	85	10.1 有限状态机和状态转换	112
8.5 边界类和对象	85	10.1.1 事件	112
8.5.1 用户交互对象	85	10.1.2 状态	113
8.5.2 代理对象	86	10.2 状态图示例	113
8.5.3 设备 I/O 边界对象	87	10.2.1 ATM 状态图示例	113
8.5.4 描述外部类和边界类	88	10.2.2 微波炉状态图示例	115
8.6 实体类和对象	89	10.3 事件和警戒条件	116
8.7 控制类和对象	90	10.4 动作	117
8.7.1 协调者对象	90	10.4.1 状态转换中的动作	117
8.7.2 状态相关的控制对象	91	10.4.2 进入动作	119
8.7.3 计时器对象	91	10.4.3 退出动作	119
8.8 应用逻辑类和对象	92	10.5 层次化状态图	120
8.8.1 业务逻辑对象	92	10.5.1 层次化状态分解	121
8.8.2 算法对象	92	10.5.2 复合状态	121
8.8.3 服务对象	94	10.5.3 状态转换的聚合	121
8.9 总结	94	10.5.4 正交状态图	122
练习	94	10.6 开发状态图的指导原则	123
第 9 章 动态交互建模	96	10.7 从用例开发状态图	124
9.1 对象交互建模	96	10.8 从用例开发状态图示例	125
9.1.1 通信图	96	10.8.1 为每个用例开发状态图	125
9.1.2 顺序图	97	10.8.2 考虑可替换序列	126
9.1.3 对象交互建模的分析和设计决策	97	10.8.3 开发集成的状态图	126
9.1.4 顺序图和通信图对比	98	10.8.4 开发层次化状态图	128
		10.9 总结	129
		练习	130

第 11 章 状态相关的动态交互建模	132	第 13 章 软件子系统体系结构设计	159
11.1 状态相关的动态交互建模中的步骤	132	13.1 软件体系结构设计中的问题	159
11.2 使用交互图和状态图对交互场景 建模	132	13.2 集成通信图	160
11.3 状态相关的动态交互建模示例: 银行系统	133	13.3 子系统设计中的关注点分离	162
11.3.1 确定主序列	133	13.3.1 复合对象	162
11.3.2 确定可替换序列	135	13.3.2 地理位置	164
11.3.3 主序列: 有效 PIN 码	136	13.3.3 客户端和服务	164
11.3.4 可替换序列: 无效 PIN 码	136	13.3.4 用户交互	164
11.3.5 可替换序列: 第三次无效 PIN 码	137	13.3.5 外部对象的接口	164
11.3.6 可替换序列: 被盗的或过期的 卡片	137	13.3.6 控制范围	165
11.3.7 所有场景通用的交互图	138	13.4 子系统组织准则	166
11.3.8 控制对象和状态图的编序	138	13.4.1 客户端子系统	166
11.4 总结	140	13.4.2 用户交互子系统	166
练习	141	13.4.3 服务子系统	168
		13.4.4 控制子系统	168
		13.4.5 协调者子系统	169
		13.4.6 输入/输出子系统	170
		13.5 子系统间消息通信的决策	170
		13.6 总结	172
		练习	172
第三部分 软件体系结构设计		第 14 章 设计面向对象的软件体系结构	174
第 12 章 软件体系结构概览	144	14.1 面向对象的软件体系结构的概念、 体系结构和模式	174
12.1 软件体系结构以及基于构件的 软件体系结构	144	14.2 设计信息隐藏类	174
12.1.1 基于构件的软件体系结构	144	14.3 设计类接口和操作	175
12.1.2 体系结构构造型	145	14.3.1 基于交互模型设计类操作	175
12.2 软件体系结构的多视图	145	14.3.2 基于交互模型设计类操作示例	176
12.2.1 软件体系结构的结构视图	145	14.3.3 基于静态模型设计类操作	177
12.2.2 软件体系结构的动态视图	145	14.4 数据抽象类	177
12.2.3 软件体系结构的部署视图	147	14.5 状态机类	178
12.3 软件体系结构模式	147	14.6 图形用户交互类	179
12.3.1 抽象分层体系结构模式	148	14.7 业务逻辑类	180
12.3.2 调用/返回模式	150	14.8 设计中的继承	180
12.3.3 异步消息通信模式	150	14.8.1 类继承	180
12.3.4 带回复的同步消息通信模式	153	14.8.2 抽象类	182
12.4 描述软件体系结构模式	153	14.8.3 抽象类和子类示例	182
12.5 接口设计	154	14.8.4 抽象超类和子类设计	184
12.6 设计软件体系结构	155	14.9 类接口规约	185
12.7 总结	157	14.10 信息隐藏类的详细设计	186
练习	157	14.10.1 账户抽象超类的详细设计	187

14.10.2 支票账户子类的详细设计	187
14.10.3 储蓄账户子类的详细设计	188
14.11 多态和动态绑定	188
14.12 Java 中类的实现	189
14.13 总结	190
练习	190
第 15 章 设计客户端 / 服务器软件	
体系结构	192
15.1 客户端 / 服务器体系结构的概念、 体系结构和模式	192
15.2 客户端 / 服务软件体系结构的 结构模式	192
15.2.1 多客户端 / 单服务体系结构 模式	192
15.2.2 多客户端 / 多服务体系结构 模式	194
15.2.3 多层客户端 / 服务体系结构 模式	195
15.3 客户端 / 服务器体系结构的通信 模式	195
15.3.1 带回复的同步消息通信模式	196
15.3.2 带回调的异步消息通信模式	196
15.4 客户端 / 服务器系统的中间件	197
15.4.1 客户端 / 服务器系统平台	197
15.4.2 Java 远程方法调用	198
15.5 服务子系统的设计	199
15.5.1 顺序性服务的设计	199
15.5.2 顺序性服务设计示例	199
15.5.3 并发服务设计	200
15.6 包装器类的设计	202
15.6.1 数据库包装器类的设计	202
15.6.2 数据库包装器类示例	202
15.7 从静态模型到关系数据库的设计	203
15.7.1 关系数据库概念	203
15.7.2 确定主键	204
15.7.3 将关联映射到外键	204
15.7.4 将关联类映射到关联表	205
15.7.5 将整体 / 部分关系映射到关系 数据库	206
15.7.6 将泛化 / 特化关系映射到关系 数据库	207
15.8 总结	208
练习	208
第 16 章 设计面向服务的体系结构	210
16.1 面向服务的体系结构的概念、体系 结构和模式	210
16.2 软件体系结构代理者模式	211
16.2.1 服务注册模式	211
16.2.2 代理者转发模式	212
16.2.3 代理者句柄模式	212
16.2.4 服务发现模式	213
16.3 面向服务的体系结构的技术支持	214
16.3.1 Web 服务协议	214
16.3.2 Web 服务	214
16.3.3 注册服务	214
16.3.4 代理和发现服务	215
16.4 软件体系结构事务模式	215
16.4.1 两阶段提交协议模式	216
16.4.2 复合事务模式	218
16.4.3 长事务模式	218
16.5 协商模式	219
16.6 面向服务体系结构中的服务接口 设计	221
16.7 面向服务体系结构中的服务协调	222
16.8 设计面向服务的体系结构	223
16.9 服务复用	225
16.10 总结	225
练习	225
第 17 章 设计基于构件的软件体系 结构	227
17.1 基于构件的软件体系结构的概念、 体系结构和模式	227
17.2 设计基于构件的分布式软件体系 结构	228
17.3 复合子系统和构件	228
17.4 使用 UML 建模构件	229
17.4.1 构件接口设计	229
17.4.2 供给和请求接口	230
17.4.3 连接器和交互构件	230

17.4.4 设计复合构件	231	18.8.3 不带回复的同步(紧耦合)消息 通信	255
17.5 构件组织准则	232	18.8.4 事件同步	255
17.5.1 与物理数据源的邻近性	232	18.8.5 信息隐藏对象上的任务交互	256
17.5.2 局部自治性	232	18.8.6 修正的并发通信图	257
17.5.3 性能	233	18.9 任务接口和任务行为规约	257
17.5.4 特定硬件	233	18.9.1 针对银行服务任务的 TIS 示例	259
17.5.5 I/O 构件	234	18.9.2 针对读卡器接口任务的 TIS 示例	259
17.6 组消息通信模式	234	18.10 Java 中并发任务的实现	260
17.6.1 广播消息通信模式	234	18.11 总结	260
17.6.2 订阅/通知消息通信模式	234	练习	261
17.6.3 使用订阅和通知的并发服务 设计	236	第 19 章 设计软件产品线体系结构	262
17.7 应用部署	237	19.1 演化软件产品线工程	262
17.7.1 应用部署事务	237	19.2 软件产品线的的需求建模	263
17.7.2 应用部署示例	239	19.2.1 软件产品线的用例建模	263
17.8 总结	239	19.2.2 特征建模	264
练习	239	19.3 软件产品线的分析建模	266
第 18 章 设计并发和实时软件体系 结构	241	19.3.1 软件产品线的静态建模	266
18.1 并发和实时软件体系结构的概念、 体系结构及模式	241	19.3.2 软件产品线的动态交互建模	266
18.2 实时系统的特点	241	19.4 软件产品线的动态状态机建模	268
18.3 实时软件体系结构中的控制模式	242	19.5 软件产品线的设计建模	269
18.3.1 集中式控制体系结构模式	242	19.5.1 建模基于构件的软件体系结构	269
18.3.2 分布式控制体系结构模式	242	19.5.2 软件体系结构模式	270
18.3.3 层次化控制体系结构模式	243	19.6 总结	270
18.4 并发任务组织	244	练习	271
18.5 I/O 任务组织准则	244	第 20 章 软件质量属性	272
18.5.1 事件驱动 I/O 任务	244	20.1 可维护性	272
18.5.2 周期性 I/O 任务	245	20.2 可修改性	273
18.5.3 按需驱动 I/O 任务	247	20.3 可测试性	274
18.6 内部任务组织准则	248	20.4 可追踪性	274
18.6.1 周期性任务	248	20.5 可扩展性	276
18.6.2 按需驱动任务	248	20.6 可复用性	276
18.6.3 控制任务	249	20.7 性能	277
18.6.4 用户交互任务	250	20.8 安全性	278
18.7 开发并发任务体系结构	252	20.9 可用性	278
18.8 任务通信和同步	254	20.10 总结	279
18.8.1 异步(松耦合)消息通信	254	练习	279
18.8.2 带回复的同步(紧耦合)消息 通信	254	第四部分 案例研究	
		第 21 章 客户端/服务器软件体系结构 案例研究: 银行系统	282