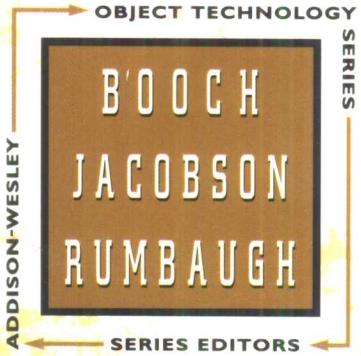
HZ BOOKS

软件工程技术丛书

UML 用户指南

The Unified Modeling Language User Guide



Grady Booch
(美) James Rumbaugh 著
Ivar Jacobson

邵维忠 麻志毅 张文娟 孟祥文 译

机械工业出版社
China Machine Press
Addison-Wesley

软件工程技术丛书

UML用户指南

Grady Booch
（美） James Rumbaugh 著
Ivar Jacobson
邵维忠 麻志毅 张文娟 孟祥文 译

机械工业出版社
China Machine Press

本书介绍了UML的基础知识，包括UML的术语、规则和语言特点，以及如何运用该语言去解决常见的建模问题。书中给出了大量实例，这种基于实际应用的学习方式，有助读者迅速掌握UML的基本概念、独特性质及应用。初学者学习UML最好从阅读本书开始。

本书的三位作者是面向对象方法最早的倡导者，是UML的原创人。对于高级开发人员，本书提供了在高级建模问题中应用UML的一条非常实用的线索。

本书版权登记号：图字：01-1999-3214

图书在版编目（CIP）数据

UML 用户指南/（美）布谢（Booch，G.），（美）朗博（Rumbaugh, J.），（美）雅各布森（Jacobson, J.）著；邵维忠等译. - 北京：机械工业出版社，2001.6
(软件工程技术丛书)
书名原文：The Unified Modeling Language User Guide
ISBN 7-111-07564-1

I. U … II. ①布… ②朗… ③雅… ④邵… III. 统一模型语言，UML-指南 IV.TP312-62

中国版本图书馆CIP 数据核字（2001）第11003 号

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：姚 蕾
北京市密云县印刷厂印刷・新华书店北京发行所发行
2001年6月第1版・2001年8月第2次印刷
787mm × 1092mm 1/16・22.25印张
印数：5 001-10 000册
定价：48.00元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

译者序

随着计算机硬件性能的不断提高和价格的不断下降，其应用领域也在不断扩大。人们在越来越多的领域希望把更多、更难的问题交给计算机去解决。这使得计算机软件的规模和复杂性与日俱增，因而软件技术不断地受到新的挑战。20世纪60年代软件危机的出现就是因为系统的复杂性超出了人们在当时的技术条件下所能驾御的程度。此后在软件领域，从学术界到工业界，人们一直在为寻求更先进的软件方法与技术而奋斗。每当出现一种先进的方法与技术，都会使软件危机得到一定程度的缓和。然而这种进步又立刻促使人们把更多、更复杂的问题交给计算机去解决。于是又需要更先进的方法与技术。

开发一个具有一定规模和复杂性的软件系统和编写一个简单的程序大不一样。其间的差别，借用G. Booch的比喻，如同建造一座大厦和搭一个狗窝的差别。大型的、复杂的软件系统的开发是一项工程，必须按工程学的方法组织软件的生产与管理，必须经过分析、设计、实现、测试、维护等一系列的软件生命周期阶段。这是人们从软件危机中获得的最重要的教益。这一认识促使了软件工程学的诞生。编程仍然是重要的，但是更具有决定意义的是系统建模。只有在分析和设计阶段建立了良好的系统模型，才有可能保证工程的正确实施。正是由于这一原因，许多在编程领域首先出现的新方法和新技术，总是很快地拓展到软件生命周期的分析与设计阶段。

面向对象方法正是经历了这样的发展过程，它首先在编程领域兴起，作为一种崭新的程序设计范型引起世人瞩目。继Smalltalk-80之后，20世纪80年代又有一大批面向对象的编程语言问世，标志着面向对象方法走向成熟和实用。此时，面向对象方法开始向系统设计阶段延伸，出现了如Booch86、GOOD（通用面向对象的开发）、HOOD（层次式面向对象的设计）、OOSD（面向对象的结构设计）等一批OOD（“面向对象的设计”或“面向对象的开发”的缩写）方法。但是这些早期的OOD方法不是以面向对象的分析（OOA）为基础的，而主要是基于结构化分析。到1989年之后，面向对象方法的研究重点开始转向软件生命周期的分析阶段，并将OOA和OOD密切地联系在一起，出现了一大批面向对象的分析与设计（OOA&D）方法；如Booch方法、Coad/Yourdon方法、Firesmith方法、Jacobson的OOSE、Martin/Odell方法、Rumbaugh等人的OMT、Shlaer/Mellor方法等等。截至1994年，公开发表并具有一定影响的OOA&D方法已达50余种。这种繁荣的局面表明面向对象方法已经深入到分析与设计领域，并随着面向对象的测试、集成与演化技术的出现而发展为一套贯穿整个软件生命周期的方法体系。目前，大多数较先进的软件开发组织已经从分析、设计到编程、测试阶段全面地采用面向对象方法，使面向对象无可置疑地成为当前软件领域的主流技术。

各种面向对象的分析与设计方法都为面向对象理论与技术的发展作出了贡献。这些方法各有自己的优点和缺点，同时在各自不同范围内拥有自己的用户群。各种方法的主导思想

以及所采用的主要概念与原则大体上是一致的，但是也存在不少差异。这些差异所带来的问题是，不利于面向对象方法向一致的方向发展，也会给用户的选择带来一些困惑。为此，Rational公司的G. Booch和J. Rumbaugh决定将他们各自的方法结合起来成为一种方法。1995年10月发布了第1个版本，称作“统一方法”（Unified Method 0.8）。此时OOSE的作者I. Jacobson也加入了Rational公司，于是也加入了统一行动。1996年6月发布了第2个版本UML0.9。鉴于统一行动的产物只是一种建模语言，而不是一种建模方法（因为不包含过程指导），所以自0.9版本起，改称“统一建模语言”（Unified Modeling Language）。在此过程中，由Rational公司发起成立了UML伙伴组织。开始时有12家公司加入，共同推出了UML1.0版本，并于1997年1月提交到对象管理组织（OMG）申请作为一种标准建模语言。此后，又把几家分头向OMG提交建模语言提案的公司扩大到UML伙伴组织中，并为反映他们的意见对UML进一步做了修改，产生了UML1.1版本。该版本于1997年11月4日被OMG采纳。此后UML还在继续改进，目前最新的版本是UML1.3。

关于UML的历史、发起的动机、目标、权衡的问题等，这里不想做更多的介绍，因为读者很快会从这套丛书中的《UML用户指南》的前言中看到更详细的叙述。这里想着重指出的是以下三点：第一点是UML的三位发起人G. Booch、J. Rumbaugh和I. Jacobson是从事面向对象研究的著名专家，他们各自的方法和著作在该领域均具有很大的影响。第二点是众多的大公司加入了UML阵营，对UML的制定和推广提供了强有力的支持。第三点是UML经过数年的努力终于被OMG采纳，成为该组织承认的一种标准建模语言。总之，UML是吸收多种方法的成果、凝结了许多组织和个人智慧的产物。

UML是一种用于对软件密集型系统进行可视化、详述、构造和文档化的建模语言，主要适用于分析与设计阶段的系统建模。UML最主要的特点是表达能力丰富。因为它从各种OOA&D方法中吸取了大量的概念，并在“UML语义”、“UML表示法指南”、“对象约束语言规格说明”等UML文献中对这些概念的语义、图形表示法和使用规则作了完整而详细的定义。可以说，UML对系统模型的表达能力超出了以往任何一种OOA&D方法。当然，随之而来的问题是，它的复杂性也超出了以往任何一种方法。

UML的问世受到计算机软件界的广泛重视，因为它代表了一种积极的方向——多种方法相互借鉴、相互融合、趋于一致、走向标准化。建模语言的标准化将为软件开发商及其用户带来诸多便利。因此，在美国等国家已有大量的软件开发组织开始用UML进行系统建模。学习和使用UML已经成为一种潮流。我国软件界对UML也相当关注。许多研究人员和技术人员已在数年前开始学习和研究UML。更有许多人想学习UML，但苦于找不到合适的书籍。由于UML的复杂性，仅通过UML的标准文献来学习和使用它确实不是一件轻松的事。以往国内外也曾发表过一些介绍或评述UML的著作或论文，但是与UML的丰富内容相比，这些介绍远不能满足读者的要求。

值得高兴的是，UML的三位主要设计者G. Booch、J. Rumbaugh和I. Jacobson现在已亲自撰写了这套详细阐述UML的著作，由Addison Wesley于1999年出版。这套著作对UML进行了详细、深入而准确的介绍和论述，而且语言生动、深入浅出、实例丰富、图文并茂。这是一套教会读者掌握和使用UML的教材和指导手册，而不是枯燥的标准文献。对于想学

习和使用UML的广大读者，这是一套难得的好书。为了使中国的读者能够更好地从中受益，我们在机械工业出版社华章公司的恳切建议下，分头翻译了这三本书，即《UML用户指南》、《UML参考手册》和《统一软件开发过程》。

三本原著都是由这三位作者合著，既各自独立，又有很强的内在联系。其中《UML用户指南》介绍了UML的基础知识，包括UML的术语、规则和语言特点，以及如何运用该语言去解决常见的建模问题，初学者学习UML最好从阅读该书开始。《UML参考手册》对UML的组成和概念作了详细的介绍，包括这些概念的语义、语法、表示法和用途，是一本适合软件专业人员使用的方便而全面的参考读物。《统一软件开发过程》给出了一种以UML作为建模语言进行软件开发的过程指导。其内容不是UML固有的组成部分，因为被OMG采纳的UML只是一种建模语言，并不包含过程指导。实际上UML是独立于过程的，可以用于不同的软件过程。但是该书介绍的软件开发过程是三位作者在开发UML时一直在头脑中思考的，因此很切合UML的特点。该书对于如何运用UML的概念进行软件开发提供了详细指导，适合软件专业人员使用。

鉴于UML本身以及这套著作的重要意义，译者在翻译这些著作时采取了特别慎重和严谨的态度。力求准确和通顺。在翻译过程中，一个重要问题是要使这套书中的专业术语的中文译法取得一致。这三本书的译者以往曾分别开展过一些与UML有关的研究和写作，对有些术语的译法互有差异。本次翻译工作中，所有译者在机械工业出版社华章公司的组织下进行了多次讨论、研究和交流，首先对所有专业术语的译法统一意见，达成共识。其中某些术语的译法颇难定夺：既要确切反映英文本意，又要符合中文习惯，还要避免与国内已习惯于与其他英文词对应的中文相混淆。经过反复切磋，大部分问题都得到满意的解决。对个别有争议的问题，在充分讨论的基础上采取放弃己见，服从大局的态度。如此形成了一个译法一致的词汇表。此后在翻译过程中还经常以各种交流方式进行磋商和沟通。最终使这套丛书能以一致的面貌呈献给读者。我们也希望这些工作能为UML术语今后在中文翻译中的统一贡献一份力量。

在科技著作的翻译中，保证准确和通顺的关键因素不仅仅是外文水平，还取决于译者真正了解所涉及的技术内容。这套著作的内容远远超出了UML的标准文献，因为除了介绍UML的语法、语义、使用规则之外，其中还包含许多学术思想、技术策略和实践经验。在翻译中遇到的许多疑难问题，是通过进一步研究UML以及有关的学术和技术问题而得到解决的，从而避免了许多讹误。因此，这套著作的翻译不仅是文字方面的工作，还包含译者在技术上的研究。我们希望这些研究最终通过较准确的翻译文字使读者受益。同时诚恳地希望广大读者对可能存在的疏漏和错误之处给予批评和指正。

译　者

2000年11月于北京

译者简介

邵维忠，男，北京大学计算机科学技术系教授，博士生导师，中国计算机学会理事。1970年毕业于北京大学数学力学系，1979年至1983年在北京大学计算机科学技术系任教并攻读硕士学位。早期从事操作系统和软件工程领域的研究与开发。1987年至1989年在新加坡国立大学参加科技合作，主要从事人工智能研究。回国后在导师杨芙清院士主持的国家“八五”、“九五”重点科技攻关课题（“大型软件开发环境青鸟系统”和“软件工程环境工业化生产技术及系统”）中担任主要技术负责人。自1991年起注重于面向对象的方法与技术研究并开设了研究生课程。写作和翻译过多本关于面向对象方法的学术专著。曾获国家部委级奖励多项，国家科技进步奖和国家“七五”、“八五”科技攻关突出贡献奖荣誉证书。

麻志毅，男，北京大学计算机科学技术系副教授。1999年在东北大学获得博士学位，同年到北京大学计算机科学技术系作博士后，出站后留校工作。现已发表学术论文30余篇，主持或参加政府科研项目十余项。主要研究领域为软件工程、面向对象技术和计算语言学。

张文娟，女，讲师。1991年毕业于北京理工大学计算机科学技术系，获学士学位；1994年在北京机械工业学院获硕士学位；1998年考入北京大学计算机科学技术系攻读博士学位。主要研究领域为面向对象技术、软件复用技术、软件构件技术和软件工程。

孟祥文，男，讲师。1994年在山东矿业学院计算机系获硕士学位并留校任教，1998年开始在北京大学计算机科学技术系攻读博士学位。主持完成煤炭自然科学基金项目一项。主要研究领域为面向对象技术、软件智能化技术、软件工程环境和开放式软件开发技术。

四位译者均在杨芙清院士领导的科研队伍中从事面向对象方法与技术的研究和面向对象开发工具与环境的研制。

前　言

统一建模语言（Unified Modeling Language，UML）是一种用于对软件密集型系统的制品进行可视化、详述、构造和文档化的图形语言。UML给出了一种描绘系统蓝图的标准方法，其中既包括概念性的事物，如业务过程和系统功能，也包括了具体的事物，如用特定的编程语言编写的类、数据库模式和可复用的软件构件。

本书旨在教会读者如何有效地使用UML。

目标

在本书中，读者将获益于以下几点：

- 明白UML是什么，不是什么，以及为什么UML与开发软件密集型系统的过程相关。
- 掌握UML的术语、规则和语言特点，并且一般说来还将学会如何有效地使用这种语言。
- 知道如何应用UML去解决若干普通的建模问题。

这本用户指南为具体使用UML的特征提供了参考，但它不是一本全面的UML参考手册。对于全面的UML描述，请参考《UML参考手册》(Rumbaugh、Jacobson、Booch合著，Addison-Wesley出版公司1999年出版)。（本书的中文版已由机械工业出版社出版。——译者注）

这本用户指南描述了一个使用UML进行开发的过程，但它并不提供对于开发过程的完整参考。并于开发过程的完整参考，请参考《统一软件开发过程》(Jacobson、Booch、Rumbaugh合著，Addison-Wesley出版公司1999年出版)。（本书的中文版即将由机械工业出版社出版。——译者注）

最后，本书提供了如何运用UML去解决若干普通的建模问题的线索和技巧，但没有讲述如何去建模。这类似于一本编程语言的用户指南，它教用户如何使用语言，而不教用户如何编程。

读者对象

进行软件生产、部署和维护的人员均可使用UML。这本用户指南主要针对用UML进行建模的开发组的成员，但它也适用于为了理解、建造、测试和发布一个软件密集型系统而一起工作的读者。虽然这几乎包含了软件开发组织中的所有角色，但本指南特别适合于下述人员：分析员和最终客户（他们要详细说明系统应该具有的结构和行为）、体系结构设计人员（他们设计满足需求的系统）、开发人员（他们把体系结构转换为可执行的代码）、质

量保证人员（他们检验并确认系统的结构和行为）、库管理人员（他们创建构件并对构件进行编目）、项目及程序管理者（他们一般是把握方向的领导者，要进行有序地管理，并合理地分配资源，以保证系统的成功）。

使用本指南的人员应该具有面向对象概念的基本知识。如果读者具有面向对象的编程经验或懂得面向对象的方法，就能更容易地掌握本指南，但这并不是必需的。

怎样使用本书

初次接触UML的开发者最好要按顺序阅读本书。第2章提出了UML的概念模型，读者应要特别予以注意。所有的章节都建立了结构，这种结构标明了与当前章节内容相关的其他章节，使得当前章节成为某一阅读线索中的一部分。

对于寻找用UML建模的一般问题答案的有经验的开发者来说，可以按任意顺序使用本书。读者应该特别注意在各章中提及的普遍建模问题。

本书的组织及特点

这本用户指南主要由7个部分组成：

- 第一部分　入门
- 第二部分　对基本结构建模
- 第三部分　对高级结构建模
- 第四部分　对基本行为建模
- 第五部分　对高级行为建模
- 第六部分　对体系结构建模
- 第七部分　结束语
- 本指南还含有3个附录：UML表示法的概要，UML标准元素的列表，Rational 统一过程的概要。在附录后，本指南还提供了一个术语表。

各章都描述了相应的UML特征的具体用法，其中的大部分按下述方式组织：

1）入门

2）术语和概念

3）普通建模技术

4）提示和技巧

第三部分引入并解决了一组普通建模问题。为了使读者容易地浏览本指南，以便寻找所需要的部分，每一个问题都标有一个明显的标题，如下面的例子。

对体系结构模式建模

每一章都以一个覆盖本章内容的特征概要开始，如下例。

本章内容

- 主动对象、进程和线程
- 对多控制流建模
- 对进程间通信建模
- 建立线程的安全抽象

类似地，把附加的解释和常规的指导作为注释的一部分，如下例所示。

注释 通过使用一个像0..1、3..4、6..*这样的列表，可以描述更为复杂的多重性，该列表的含义是“除了2个或5个对象外，可以有任意数目的对象”。

UML的语义是丰富的，因此对一个特征的描述自然地会涉及到另一个特征。在这种情况下，在自然段的最后部分标注交叉参照，正如本段这样。【在第25章讨论构件。】

在图中使用的蓝字是为了表明这些文字不是模型本身的一部分，只是用于解释模型。（中文版中用汉字代替蓝字表示这种区别。——译者注）程序代码用monospace字体表示以示区别，如this example。（中文版中用英文表示这种区别。——译者注）

UML简史

面向对象的建模语言出现在20世纪70年代中期至80年代后期之间，那时方法论的学者们面对面向对象编程语言的新风格和日趋复杂的应用，开始用分析与设计的新方法进行实践。在1989到1994年间，面向对象的方法从不足10种增加到50种以上。面对这么多的方法，很多用户很难找到一种完全满足他们要求的建模语言，于是就加剧了所谓的方法战争。通过从实践中学习，新一代方法开始出现，一些明显突出的方法脱颖而出，其中最著名的有Booch方法、Jacobson的OOSE（面向对象的软件工程）方法和Rumbaugh的OMT（对象建模技术）方法。其他的重要方法还有Fusion方法、Shlaer-Mellor方法和Coad-Yourdon方法。这些方法中的每一种方法都是完整的，但是每一种方法又都被认为各有优点和缺点。简单来说，Booch方法在项目的设计和构造阶段的表达力特别强；OOSE对以用况作为一种途径来驱动需求捕获、分析和高层设计提供了极好的支持；而OMT-2对于分析和数据密集型信息系统非常有用。

大量的有决定意义的思想开始形成于20世纪90年代中期，当时Grady Booch(Rational软件公司)、Ivar Jacobson(Objectory公司)和James Rumbaugh(通用电气公司)开始彼此从对方的方法中吸纳思想，他们的关于面向对象方法的共同成果，开始被认为是在全球范围内具有领导性的。作为Booch、OOSE和OMT方法的主要作者，促进我们创建统一建模语言的原因有三个。首先，我们的方法已经在朝着相互独立的方向演化，而我们希望它朝着一个方向演化，这样可以消除任何不必要的和不合理的潜在差别，因为这样的差别会加重用户的疑惑。第二，通过统一我们的方法，能够给面向对象的市场带来一定的稳定，能够让人们使用成熟的建模语言去设计项目，使人们能够集中精力开发出更具有使用价值的工具。第三，

希望我们的合作能够对三种早期的方法产生改进，帮助我们吸取教益，以解决我们当前的方法所不能妥善处理的问题。

当我们开始统一工作时，我们就确立了3个工作目标：

1）运用面向对象技术对系统进行从概念到可执行制品的建模。

2）针对复杂系统和关键使命系统中固有的规模问题。

3）创造一种人和机器都可以使用的建模语言。

设计一种用于面向对象分析和设计的语言与设计一种编程语言是不同的。首先，我们必须对问题进行约束：这个语言是否应包含需求描述？这个语言是否应足以支持可视化编程？其次，我们必须在表达能力和表达的简洁性之间作好平衡。太简单的语言将会限制可解决问题的范围，而太复杂的语言将会使开发者无所适从。在统一现有的方法的情况下，我们也必须小心从事。若对语言进行太多的改进，将会给已有用户造成混乱；若不对语言进行改进，则将会失去赢得更广大的用户群和使语言得到简化的时机。UML的定义力争在这些方面做出最好的平衡。

1994年10月，Rumbaugh加入Booch所在的Rational公司，这时正式开始了UML的统一工作。我们的计划最初注重于联合Booch和OMT方法。“统一方法”（当时的名称）0.8版本（草案）在1995年10月发布。大约就在那个时候，Jacobson也加入了Rational公司，于是UML项目的范围又做了扩充，把OOSE也结合进来。经过我们的努力，在1996年6月发布了UML0.9版本。在1996年全年，我们在软件工程界征求和收集反馈意见。在此期间，明显地有很多软件组织把UML作为商业策略来考虑。我们与几个愿意致力于定义一个强大而完善的UML的组织一起成立了一个UML伙伴组织。

对UML1.0版本作出贡献的合作伙伴有：Digital Equipment Corporation、Hewlett-Packard、I-Logix、Intellicorp、IBM、ICON Computing、MCI Systemhouse、Microsoft、Oracle、Rational、Texas Instruments和Unisys。这些合作伙伴协作产生的UML1.0版本是一个定义明确的、富有表现力的、强大的、可应用于广泛的问题域的建模语言。在1997年1月，他们把UML1.0版本提交给OMG（对象管理组织），申请把UML1.0版本作为一种标准建模语言。

在1997年1月至7月之间，合作伙伴的队伍不断扩大，实际上包括了所有对最初OMG的提议作出贡献的公司，它们是Andersen Consulting、Ericsson、ObjecTime Limited、Platinum Technology、PTech、Reich Technologies、Softeam、Sterling Software和Taskon。为了对UML规格说明进行形式化，并把UML与其他的标准化成果结合起来，成立了一支由MCI Systemhouse的Cris Kobryn领导并由Rational的Ed Eykholt管理的语义专门组织。在1997年7月，把UML的修改版（1.1版本）提交给OMG，申请进行标准化审查。1997年9月，OMG的分析与设计专门组织（Analysis and Design Task Force，ADTF）和OMG的体系结构部接受了该版本，并把它提交给OMG的全体成员进行表决。1997年11月14日，UML1.1版本被OMG采纳。

UML的维护工作由OMG的修订专门组织(Revision Task Force，RTF)接管，该组织由Cris Kobryn领导。RTF在1998年6月发布了一个修订版本（UML1.2版本）。1998年秋季，

RTF发布了UML1.3版本，它在技术上更清晰完整。本指南描述的就是UML1.3版本的内容。

致谢

UML工作的发起者是Grady Booch、Ivar Jacobson和 James Rumbaugh，他们自始至终参加了本项目的工作，但最终的产品是所有的UML合作伙伴共同努力的结果。虽然每个合作伙伴都有各自的观点、各自关心的领域和各自获利的方面，但全部的成就都受益于各个成员的贡献和他们各种各样的经验与观点。

UML的核心成员包括：

- Hewlett-Packard:Martin Griss
- I-Logix:Eran Gery,David Harel
- IBM:Steve Cook,Jos Warmer
- ICON Computing:Desmond D'Souza
- Intellicorp and James Martin and Company:James Odell
- MCI Systemhouse:Cris Kobryn,Joaquin Miller
- ObjecTime:John Hogg,Bran Selic
- Oracle: Guus Ramackers
- Platinum Technology:Dilhar DeSilva
- Rational Software:Grady Booch,Ed Eykholt,Ivar Jacobson,Gunnar Overgaard,Karin Palmkvist, James Rumbaugh
- Taskon: Trygve Reenskaugh
- Texas Instruments/Sterling Software:John Cheesman,Keith Short
- Unisys: Sridhar Iyengar,G.K. Khalsa

在UML1.1版本、UML1.2版本和UML1.3版本开发期间，是Cris Kobryn领导和指挥了UML技术队伍的工作，这里要特别感谢他的贡献。

我们也要感谢下列人员的贡献、影响和支持。由于某些原因，这里提到的一些人员虽然没有正式参与UML，但由于他们的影响，这里仍要表示感谢。他们是Jim Amsden、Hernan Astudillo、Colin Atkinson、Dave Bernstein、Philip Bernstein、Michael Blaha、Conrad Bock、Mike Bradley、Ray Buhr、Gary Cernosek、James Cerrato、Michael Jesse Chonoles、Magnus Christerson、Dai Clegg、Geoff Clemm、Peter Coad、Derek Coleman、Ward Cunningham、Raj Datta、Philippe Desfray、Mike Devlin、Bruce Douglass、Staffan Ehnebom、Maria Ericsson、Johannes Ernst、Don Firesmith、Martin Fowler、Adam Frankl、Eric Gamma、Dipayan Gangopadhyay、Garth Gullekson、David Harel（他创造了状态图的最初概念）、Rick Hargrove、Tim Harrison、Richard Helm、Brian Hendersen-Sellers、Michael Hirsch、Bob Hodges、Yves Holvoet、Jon Hopkins、John Hsia、Glenn Hughes、Ralph Johnson、Anneke Kleppe、Philippe Kruchten、Paul Kyzivat、Martin Lang、Grant

Larsen、Reed Letsinger、Mary Loomis、Jeff MacKay、Joe Marasco、Robert Martin、Terri McDaniel、Jim McGee、Mike Meier、Randy Messer、Bertrand Meyer、Greg Meyers、Fred Mol、Luis Montero、Paul Moskowitz、Andy Moss、Jan Pachl、Paul Patrick、Woody Pidcock、Bill Premerlani、Jeff Price、Jerri Pries、Terry Quatrani、Mats Rahm、Rudolf Riess、Rich Reitman、Erick Rivas、Kenny Rubin、Jim Rye、Danny Sabbahr、Tom Schultz、Colin Scott、Ed Seidewitz、Keith Short、Gregson Sui、Jeff Sutherland、Dan Tasker、Andy Trice、Dave Tropeano、Dan Uhlar、John Vlissides、Larry Wall、Paul Ward、Alan Willis、Rebecca Wirfs-Brock、Bryan Wood、Ed Yourdon 和Steve Zeigler。

UML的发展是一个开放的过程，通过对象技术用户组织(Object Technology User's Group，OTUG)，我们从全世界收到了数千条e-mail信息。虽然我们不能提及每个提交者的人名，但是我们确实要感谢他们向我们提出的评论和建议。我们认真地阅读了每条信息，正是由于这样广泛的国际性反馈，UML才得以完善。我们还要对实验室的一小帮整日喧闹的Rational学生予以特别的感谢，早在1997年，他们就参加了由Booch领导的用户指南课程，其间他们提出了一些杰出的思想，并给出了很有建设性的批评意见，这些都有助于调整本书的内容。他们是：Hernan Astudillo、Robin Brown、Robert Bundy、Magnus Christerson、Adam Frankl、Nookiah Kolluru、Ron Krubek、Grant Larsen、Dean Leffingwell、Robert Martin、Mills Ripley、Hugo Sanchez、Geri Schneider、Tom Schultz、Andy Trice、Dan Uhlar和Lloyd Williams。感谢在Number Six Software疯狂工作的人们和对本书提出技术评论的人们，他们是：Jack Carter、Tim Budd、Bruce Douglass、Martin Fowler、Cris Kobryn、Philippe Kruchten、Ron Lusk、Terry Quatrani 和David Rine。

关于更多的信息

有关UML的最新信息（包括它的形式描述）可以在因特网的*www.rational.com*和*www.omg.org*处找到。有关修订专门组织的工作可以在*uml.shl.com*处找到。

有几个适合于进行UML一般性讨论的电子论坛，它们是因特网新闻组*comp.software-eng*和*comp.object*，以及公共邮件列表*otug@rational.com*和*uml-rtf@omg.org*。

Grady Booch
Lakewood, Colorado
1998年9月
egb@rational.com

目　录

第五部分 对高级行为建模

第六部分 对体系结构建模

第七部分 结 束 语