

课件下载: www.tup.com.cn

主 编 王先国
副主编 方 鹏
曾碧卿
刘 焯

UML

统一建模

实用教程



清华大学出版社



UML

统一建模

实用教程

主 编 王先国
 副主编 方 鹏 曾碧卿 刘 焯

清华大学出版社
北京

内 容 简 介

本书是一本关于 UML 统一建模的实用教程,对 UML 体系结构、UML 元素语义、语法和 UML 应用进行了详细的讲解,表述通俗易懂,同时配备大量的实例加以说明,便于自学。

全书共分 3 篇 15 章。第 1 篇介绍了 UML 语言的特点、UML 语言的体系结构、UML 组成元素、UML 包含的图形种类、UML 图的表示方法、UML 图的作用、UML 的应用领域和建模方法等;第 2 篇介绍了 RUP 统一过程及 UML 的应用、UML 工具 Rational Rose 的使用;第 3 篇以图书管理系统为例,介绍了系统开发的各个阶段,以及如何应用 UML 对系统进行建模。通过对本书的学习,读者可以系统地掌握 UML 语言的阅读方法和建模方法。

本书重点突出了 UML 语言的表示方法和建模方法,内容精练,表达简明,实例丰富,适合作为高等院校计算机专业及相关专业的教材,也可以作为培训机构相关专业的培训教材。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

UML 统一建模实用教程/王先国主编. —北京:清华大学出版社,2009.4

ISBN 978-7-302-19560-3

I. U… II. 王… III. 面向对象语言,UML-程序设计-教材 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 019937 号

责任编辑:陈仕云 王 飞

封面设计:唐韵设计

版式设计:刘 娟

责任校对:焦章英

责任印制:李红英

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

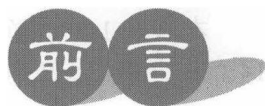
开 本:185×260 印 张:15 字 数:347 千字

版 次:2009 年 4 月第 1 版 印 次:2009 年 4 月第 1 次印刷

印 数:1~4000

定 价:28.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:032194-01



感谢您阅读本书，为了能更好地帮助您理解和掌握本书的知识，请仔细阅读下面内容。

读者对象

本书既适合 UML 语言初学者阅读，也适合系统分析员、设计者和系统测试者阅读。在写作上，本书以 UML 设计元素为主线，以系统建模为目标，运用实例系统地阐明了 UML 语言体系结构、组成、语义、语法和用 UML 元素表示系统的方法。本书理论与应用配合紧密，知识表达通俗易懂，既可作为高等院校计算机专业及相关专业的教材，也可以作为培训机构相关专业的培训教材。

本书特色

市场上有许多关于 UML 统一建模的图书，它们各有特点，本书与这些图书的不同之处在于强调了领域的表示方法和建模方法。本书体系结构安排合理，知识组织风格统一，知识表达通俗易懂，知识讲解由易到难，具体特点如下：

1. 在体系结构的安排上强调内容的系统性、连贯性、逻辑性和实用性。对 UML 元素的语义、语法和建模方法的讲解由易到难逐层展开，便于读者学习和理解。

2. 在对 UML 元素的讲解上充分体现文字描述和图形描述的结合。通过文字描述，详细地定义了 UML 元素的语义、语法；通过图形将 UML 元素可视化、规范化；对每个 UML 元素进行讲解，采用实例对 UML 元素的语义和使用方法进行演示，帮助读者学以致用。

3. 知识组织风格统一，对每个 UML 元素的讲解采用的格式为：UML 元素定义→UML 元素表示法→图形的使用方法。这种对知识进行描述的方法符合人们对问题的理解和运用习惯，使读者可以快速上手。

编者情况

本书主编曾经在大型软件公司从事应用系统的分析和设计工作多年，在开发系统过程中积累了丰富的系统建模方法及经验，能熟练地运用 UML 语言把系统需求分析和系统设



计形式化为标准的需求分析文档和设计文档。近几年来，编者一直从事高校计算机教学工作，既有丰富的系统开发经验，又有丰富的教学经验，是主讲软件工程、UML 统一建模、Java 技术和 Jsp 技术的一线教师。

本书第 1~12 章由王先国编写，第 13 章由刘焯（黄冈职业技术学院教师）编写，第 14 章由方鹏（长江职业学院教师）编写，第 15 章由曾碧卿编写。全书由王先国统稿。

致谢

感谢李利强副教授、杜作阳副教授、肖应旺博士、张承忠老师、杨桂芝老师、刘刚博士、冼广铭博士、邓虹老师、邹竞辉老师、吴莘菀老师、蔡妍老师、彭丰平老师为本书提出了许多建设性的意见和指导。同时感谢我的学生韩超、凌彩欣、张富宇、李妙珍、王娥英为本书的文字编辑所付出的心血和努力。

联系方式

书中实例虽然经过多次检查，但难免会存在疏漏和错误，恳请读者批评指正。如有好的建议或在学习中遇到疑难问题，欢迎大家发电子邮件与本书作者联系（wangxg288@tom.com）。本书配备了教学大纲和课件，如有需要，请到清华大学出版社网站（www.tup.com.cn）下载。

编者
2008 年 12 月



第 1 篇 UML 语言基础

| | |
|----------------------------|----|
| 第 1 章 UML 概述..... | 2 |
| 1.1 什么是 UML..... | 2 |
| 1.1.1 UML 的发展历史..... | 2 |
| 1.1.2 UML 的定义..... | 4 |
| 1.1.3 UML 中的图..... | 4 |
| 1.1.4 UML 的主要特点..... | 6 |
| 1.1.5 UML 的应用领域..... | 6 |
| 1.2 什么是模型..... | 7 |
| 1.2.1 为什么建模..... | 8 |
| 1.2.2 建模的目标和原则..... | 8 |
| 1.2.3 为什么要用 UML 建模..... | 9 |
| 1.2.4 用 UML 可以建立的模型种类..... | 9 |
| 1.3 UML 工具..... | 10 |
| 1.3.1 主要 UML 工具介绍..... | 10 |
| 1.3.2 如何选择 UML 工具..... | 10 |
| 1.4 小结..... | 11 |
| 1.5 习题..... | 12 |
| 第 2 章 UML 语言体系..... | 13 |
| 2.1 UML 语言组成..... | 13 |
| 2.2 基本元素..... | 14 |
| 2.2.1 结构元素..... | 14 |
| 2.2.2 行为元素..... | 17 |
| 2.2.3 分组元素..... | 18 |
| 2.2.4 注释元素..... | 18 |
| 2.3 关系元素..... | 18 |
| 2.4 图和视图..... | 21 |
| 2.5 规则..... | 22 |
| 2.6 公共机制..... | 23 |



| | | |
|--------------|------------------|-----------|
| 2.6.1 | 规格描述..... | 23 |
| 2.6.2 | 修饰..... | 24 |
| 2.6.3 | 通用划分..... | 24 |
| 2.6.4 | 扩展机制..... | 24 |
| 2.7 | 用视图表示系统体系结构..... | 25 |
| 2.8 | 小结..... | 26 |
| 2.9 | 习题..... | 26 |
| 第 3 章 | 类图..... | 28 |
| 3.1 | 类图的概念..... | 28 |
| 3.2 | UML 中的类..... | 29 |
| 3.2.1 | 类的表示..... | 29 |
| 3.2.2 | 类的种类..... | 31 |
| 3.3 | 类图中的关系..... | 34 |
| 3.3.1 | 关系分类..... | 34 |
| 3.3.2 | 关联的属性..... | 36 |
| 3.4 | 阅读类图..... | 38 |
| 3.4.1 | 电子商务网站业务..... | 38 |
| 3.4.2 | 阅读类图..... | 39 |
| 3.5 | 如何建立对象模型..... | 43 |
| 3.5.1 | 问题陈述..... | 43 |
| 3.5.2 | 寻找分析类..... | 43 |
| 3.5.3 | 确定类关系..... | 44 |
| 3.6 | 小结..... | 46 |
| 3.7 | 习题..... | 47 |
| 第 4 章 | 对象图..... | 49 |
| 4.1 | 什么是对象..... | 49 |
| 4.2 | 对象图的概念..... | 50 |
| 4.3 | 对象的表示..... | 50 |
| 4.4 | 链接的表示..... | 51 |
| 4.5 | 阅读对象图的方法..... | 52 |
| 4.6 | 小结..... | 52 |
| 4.7 | 习题..... | 52 |
| 第 5 章 | 包图..... | 54 |
| 5.1 | 包图的概念..... | 54 |
| 5.2 | 包的表示..... | 55 |
| 5.2.1 | 包的名称..... | 55 |
| 5.2.2 | 包的元素..... | 56 |
| 5.2.3 | 包的可见性..... | 56 |



| | | |
|--------------|------------------|-----------|
| 5.2.4 | 包的构造型..... | 57 |
| 5.3 | 包图中的关系..... | 57 |
| 5.3.1 | 依赖关系..... | 57 |
| 5.3.2 | 泛化关系..... | 58 |
| 5.4 | 阅读包图..... | 58 |
| 5.5 | 创建包图..... | 59 |
| 5.5.1 | 寻找包..... | 60 |
| 5.5.2 | 消除循环包依赖..... | 60 |
| 5.6 | 包图建模..... | 61 |
| 5.6.1 | 对成组元素建模..... | 61 |
| 5.6.2 | 对体系结构建模..... | 62 |
| 5.7 | 小结..... | 62 |
| 5.8 | 习题..... | 63 |
| 第 6 章 | 用例图 | 64 |
| 6.1 | 需求技术..... | 64 |
| 6.2 | RUP 开发过程..... | 64 |
| 6.3 | 用例图的概念..... | 65 |
| 6.4 | 用例图的表示..... | 66 |
| 6.4.1 | 参与者的表示..... | 66 |
| 6.4.2 | 用例的表示..... | 67 |
| 6.5 | 参与者之间的关系..... | 68 |
| 6.5.1 | 识别参与者..... | 68 |
| 6.5.2 | 参与者间的泛化关系..... | 68 |
| 6.6 | 用例之间的关系..... | 69 |
| 6.6.1 | 识别用例..... | 69 |
| 6.6.2 | 标识用例间的关系..... | 70 |
| 6.7 | 参与者与用例之间的关系..... | 73 |
| 6.8 | 阅读用例图..... | 73 |
| 6.8.1 | 用例描述模板..... | 74 |
| 6.8.2 | 用例描述举例..... | 75 |
| 6.8.3 | 用例粒度..... | 77 |
| 6.9 | 用例图应用..... | 78 |
| 6.9.1 | 问题描述..... | 78 |
| 6.9.2 | 定义术语表..... | 79 |
| 6.9.3 | 标识参与者..... | 80 |
| 6.9.4 | 标识系统用例..... | 80 |
| 6.10 | 建模要点..... | 82 |
| 6.11 | 小结..... | 82 |



| | | |
|--------------|--------------|------------|
| 6.12 | 习题 | 82 |
| 第 7 章 | 交互图 | 84 |
| 7.1 | 顺序图 | 84 |
| 7.1.1 | 顺序图的概念 | 84 |
| 7.1.2 | 顺序图的表示 | 85 |
| 7.1.3 | 表示循环和分支 | 88 |
| 7.1.4 | 绘制顺序图 | 93 |
| 7.2 | 通信图 | 94 |
| 7.2.1 | 通信图的概念 | 95 |
| 7.2.2 | 通信图的表示 | 95 |
| 7.2.3 | 表示循环和分支 | 97 |
| 7.2.4 | 阅读通信图 | 98 |
| 7.3 | 绘制交互图 | 98 |
| 7.3.1 | 寻找分析类 | 99 |
| 7.3.2 | 构建交互图 | 101 |
| 7.4 | 顺序图与通信图的关系 | 102 |
| 7.5 | 定时图 | 103 |
| 7.6 | 小结 | 104 |
| 7.7 | 习题 | 104 |
| 第 8 章 | 活动图 | 107 |
| 8.1 | 活动图的概念 | 107 |
| 8.2 | 活动图的表示 | 108 |
| 8.3 | 活动图分类 | 110 |
| 8.3.1 | 简单活动图 | 110 |
| 8.3.2 | 标识泳道的活动图 | 110 |
| 8.3.3 | 标识对象流的活动图 | 111 |
| 8.3.4 | 标识信号的活动图 | 112 |
| 8.3.5 | 标识参数的活动图 | 113 |
| 8.3.6 | 标识扩展区的活动图 | 114 |
| 8.3.7 | 嵌套活动图 | 115 |
| 8.4 | 构建活动图 | 115 |
| 8.4.1 | 对工作流程建模 | 116 |
| 8.4.2 | 对操作流程建模 | 116 |
| 8.5 | 小结 | 117 |
| 8.6 | 习题 | 117 |
| 第 9 章 | 交互概述图 | 118 |
| 9.1 | 交互概述图的概念 | 118 |
| 9.2 | 如何绘制交互概述图 | 118 |



| | | |
|---------------|---------------------|------------|
| 9.2.1 | 策略选择..... | 119 |
| 9.2.2 | 选择绘制主线..... | 120 |
| 9.2.3 | 细化重要节点..... | 120 |
| 9.3 | 小结..... | 121 |
| 9.4 | 习题..... | 121 |
| 第 10 章 | 状态机图 | 122 |
| 10.1 | 状态机图的概念..... | 122 |
| 10.1.1 | 状态机图..... | 122 |
| 10.1.2 | 状态机图的作用..... | 123 |
| 10.1.3 | 状态机图的组成元素..... | 123 |
| 10.1.4 | 状态机..... | 123 |
| 10.2 | 状态图的表示..... | 123 |
| 10.2.1 | 状态的表示..... | 123 |
| 10.2.2 | 转换的表示..... | 124 |
| 10.2.3 | 分支的表示..... | 126 |
| 10.3 | 转换的分类..... | 127 |
| 10.4 | 状态的分类..... | 128 |
| 10.4.1 | 简单状态..... | 128 |
| 10.4.2 | 复合状态..... | 129 |
| 10.4.3 | 历史状态..... | 131 |
| 10.4.4 | 子状态机间实现通信..... | 132 |
| 10.5 | 建立状态机图..... | 132 |
| 10.5.1 | 寻找主要状态..... | 133 |
| 10.5.2 | 确定状态间的转换..... | 133 |
| 10.5.3 | 详细描述每个状态和转换..... | 134 |
| 10.5.4 | 把简单状态图转换为复合状态图..... | 135 |
| 10.6 | 状态机图应用范围..... | 135 |
| 10.7 | 小结..... | 136 |
| 10.8 | 习题..... | 136 |
| 第 11 章 | 构件图 | 138 |
| 11.1 | 构件的概念..... | 138 |
| 11.1.1 | 构件..... | 138 |
| 11.1.2 | 构件与类..... | 138 |
| 11.1.3 | 构件分类..... | 139 |
| 11.2 | 构件图的概念..... | 139 |
| 11.2.1 | 构件图的作用..... | 140 |
| 11.2.2 | 构件图的组成元素..... | 140 |
| 11.3 | 构件的表示..... | 140 |



| | | |
|---------------|-----------------|------------|
| 11.3.1 | 没有标识接口的构件表示法 | 140 |
| 11.3.2 | 标识接口的构件表示法 | 141 |
| 11.4 | 构件间的关系 | 141 |
| 11.5 | 构件图分类 | 142 |
| 11.5.1 | 简单构件图 | 142 |
| 11.5.2 | 嵌套的构件图 | 143 |
| 11.6 | 构件图应用 | 143 |
| 11.6.1 | 对可执行程序建模 | 144 |
| 11.6.2 | 对源代码进行建模 | 144 |
| 11.7 | 小结 | 145 |
| 11.8 | 习题 | 145 |
| 第 12 章 | 部署图 | 146 |
| 12.1 | 部署图的概念 | 146 |
| 12.1.1 | 部署图的作用 | 147 |
| 12.1.2 | 部署图的组成元素 | 147 |
| 12.2 | 部署图的表示 | 147 |
| 12.2.1 | 节点 | 147 |
| 12.2.2 | 连接 | 149 |
| 12.3 | 部署图应用 | 150 |
| 12.3.1 | 设计阶段 | 150 |
| 12.3.2 | 实现阶段 | 150 |
| 12.3.3 | 嵌入式系统建模 | 151 |
| 12.3.4 | 客户机/服务器和分布式系统建模 | 151 |
| 12.4 | 小结 | 151 |
| 12.5 | 习题 | 152 |

第 2 篇 RUP 软件过程及双向工程

| | | |
|---------------|---------------------|------------|
| 第 13 章 | RUP (统一软件过程) | 154 |
| 13.1 | 当前流行的软件过程 | 154 |
| 13.2 | 统一软件过程 (RUP) 概述 | 155 |
| 13.2.1 | RUP 的核心 workflow | 156 |
| 13.2.2 | RUP 的 4 个阶段 | 158 |
| 13.2.3 | RUP 中的迭代模型 | 159 |
| 13.3 | RUP 中的核心 workflow | 160 |
| 13.3.1 | 需求 workflow | 160 |
| 13.3.2 | 分析 workflow | 163 |



| | | |
|---------------|------------------------|------------|
| 13.3.3 | 设计工作流..... | 166 |
| 13.3.4 | 实现工作流..... | 169 |
| 13.3.5 | 测试工作流..... | 172 |
| 13.4 | RUP 裁剪..... | 177 |
| 13.5 | RUP 的十大要素..... | 177 |
| 13.6 | 小结..... | 180 |
| 13.7 | 习题..... | 180 |
| 第 14 章 | Rose 的双向工程..... | 181 |
| 14.1 | 双向工程简介..... | 181 |
| 14.2 | 正向工程..... | 181 |
| 14.3 | 逆向工程..... | 184 |
| 14.4 | 实例应用..... | 185 |
| 14.5 | 小结..... | 190 |
| 14.6 | 习题..... | 190 |

第 3 篇 UML 应用实例

| | | |
|---------------|---------------------|------------|
| 第 15 章 | 图书管理系统..... | 192 |
| 15.1 | 软件需求分析..... | 192 |
| 15.1.1 | 需求捕获..... | 193 |
| 15.1.2 | 软件需求工程..... | 194 |
| 15.2 | 图书馆管理系统的需求分析..... | 195 |
| 15.2.1 | 标识参与者..... | 195 |
| 15.2.2 | 标识用例..... | 195 |
| 15.2.3 | 实现用例的时序图..... | 198 |
| 15.2.4 | 实现用例的协作图..... | 203 |
| 15.2.5 | 对象的状态图..... | 205 |
| 15.2.6 | 参与者的活动图..... | 207 |
| 15.3 | 系统中的类..... | 211 |
| 15.3.1 | 系统中主要的类..... | 211 |
| 15.3.2 | 各个类之间的关系..... | 213 |
| 15.4 | 系统的配置与实现..... | 214 |
| 15.4.1 | 系统的组件图..... | 214 |
| 15.4.2 | 系统的配置图..... | 215 |
| 15.5 | 小结..... | 216 |
| 15.6 | 习题..... | 216 |
| 附录 A | UML 图总结..... | 217 |
| 附录 B | OCL 概述..... | 223 |



UML 语言基础

本篇包括第 1~12 章。

第 1 章介绍了 UML 的发展历程、UML 语言的定义、UML 特点、UML 工具、UML 应用领域和 UML 模型种类。

第 2 章介绍了 UML 语言体系结构，即 UML 语言的组成元素、语法规则、图形种类、公共机制和 UML 视图。

第 3~12 章介绍了 UML 语言的 13 种图的定义、图形符号表示方法、建模方法。

学习本篇后，就能理解和把握 UML 语言的基本元素、关系元素、13 种图元素及 UML 视图的语义、语法，并在软件开发过程中使用 UML 模型元素对软件进行建模。

UML 概述

UML 是一种通用的标准建模语言，可以对任何具有静态结构和动态行为的系统进行建模。

1.1 什么是 UML

统一建模语言（Unified Modeling Language, UML）是用来对软件密集系统进行可视化建模的一种语言，是为面向对象开发系统的产品进行说明、可视化和文档编制的一种标准语言，其具体特点如下：

- UML 是非专利的第三代建模和规约语言。在开发阶段，UML 是用于说明、可视化、构建和书写面向对象软件制品的设计语言。UML 展现了一系列最佳工程实践，这些最佳工程实践在对大规模、复杂系统进行建模方面非常有效。
- UML 可以贯穿软件开发周期中的每一个阶段，已被对象管理组织（Object Management Group, OMG）采纳作为业界的标准。
- UML 最适用于数据建模、业务建模、对象建模和组件建模。
- UML 作为一种模型语言，它可以使开发人员专注于建立产品的模型和结构。当模型建立之后，模型可以被 UML 工具转化成指定的程序语言代码。

另外，IBM 的 Rational Rose 和 Microsoft 的 Visio 都是 UML 工具。

1.1.1 UML 的发展历史

公认的面向对象建模语言出现于 20 世纪 70 年代中期，1989—1994 年，其数量从不到十种增加到了五十多种。在众多的建模语言中，语言的创造者努力推崇自己的产品，并在实践中不断完善。但是，面向对象的方法（Object-Oriented Method, 简称 OO 方法）的用户并不了解不同建模语言的优缺点及相互之间的差异，因而很难根据应用特点选择合适的建模语言，于是爆发了一场“方法大战”。20 世纪 90 年代，一批新方法出现了，其中最引人注目是 Booch 1993、OMT-2 和 OOSE 等。

Booch 是面向对象方法最早的倡导者之一，他提出了面向对象软件工程的概念。1991



年，他将之前面向 Ada 的工作扩展到面向整个对象设计领域。Booch 1993 较适用于系统的设计和构造。

Rumbaugh 等人提出了面向对象的建模技术 (OMT) 方法，该方法采用了面向对象的概念，并引入各种独立于语言的表示符，同时使用对象模型、动态模型、功能模型和用例模型共同完成对整个系统的建模。该方法所定义的概念和符号可用于软件开发的分析、设计和实现的全过程，但软件开发人员不必在开发过程的不同阶段进行概念和符号的转换。OMT-2 特别适用于分析和描述以数据为中心的信息系统。

Jacobson 于 1994 年提出了 OOSE 方法，该方法最大的特点是面向用例 (Use-Case)，并在用例的描述中引入了外部角色的概念。用例的概念是精确描述需求的“重要武器”，同时用例贯穿于整个开发过程，包括对系统的测试和验证。OOSE 较适用于商业工程和需求分析。

此外，还有 Coad/Yourdon 方法，即著名的 OOA/OOD，它是最早的面向对象的分析和设计方法之一。该方法简单、易学，适合于面向对象技术的初学者使用，但由于该方法在处理能力方面的局限性，目前已很少使用。

面对众多的建模语言，首先，用户没有能力区别不同语言之间的差别，因此很难找到一种比较适合其应用特点的语言；其次，众多的建模语言实际上各有千秋；第三，虽然不同的建模语言大多雷同，但仍存在某些细微的差别，极大地妨碍了用户之间的交流。因此，在客观上，很有必要在精心比较不同建模语言的优缺点及总结面向对象技术应用实践的基础上，组织联合设计小组，并根据应用需求，取其精华，去其糟粕，求同存异，统一建模语言。

1994 年 10 月，Grady Booch 和 Jim Rumbaugh 开始致力于这一工作。他们首先将 Booch 1993 和 OMT-2 统一起来，并于 1995 年 10 月发布了第一个公开版本，称之为统一方法 UM 0.8 (Unified Method)。1995 年秋，OOSE 的创始人 Jacobson 加盟到这一工作。经过 Booch、Rumbaugh 和 Jacobson 三人的共同努力，于 1996 年 6 月和 10 月分别发布了两个新的版本，即 UML 0.9 和 UML 0.91，并将 UM 重新命名为 UML (Unified Modeling Language)。

1996 年，一些机构将 UML 作为其商业策略已日趋明显。UML 的开发者得到了来自公众的正面反应，并倡议成立了 UML 成员协会，以完善、加强和促进 UML 的定义工作。当时的成员有 DEC、HP、I-Logix、Itellicorp、IBM、ICON Computing、MCI Systemhouse、Microsoft、Oracle、Rational Software、TI 以及 Unisys。这一机构对 UML 1.0 (1997 年 1 月) 及 UML 1.1 (1997 年 11 月 17 日) 的定义和发布起了重要的促进作用。

UML 是一种定义良好、易于表达、功能强大且普遍适用的建模语言，它融入了软件工程领域的新思想、新方法和新技术。它的作用域不仅支持面向对象的分析与设计，还支持从需求分析开始的软件开发的全过程。

面向对象技术和 UML 的发展过程可用图形来表示，标准建模语言的出现是其重要成果。在美国，截止 1996 年 10 月，UML 获得了工业界、科技界和应用界的广泛支持，已有 700 多家公司表示支持采用 UML 作为建模语言。1996 年底，UML 已稳占面向对象技术市场的 85%，成为可视化建模语言事实上的工业标准。1997 年 11 月 17 日，OMG 采纳 UML 1.1 作为基于面向对象技术的标准建模语言。可以说，UML 代表了面向对象方法的软件开发技术的发展方向，具有巨大的市场前景，同时也具有重大的经济价值和国防价值。



1.1.2 UML 的定义

首先，UML 融合了 Booch、OMT 和 OOSE 方法中的基本概念，而且这些基本概念与其他面向对象技术中的基本概念大多相同，因而，UML 必然成为这些方法以及其他方法的使用者乐于采用的一种简单一致的建模语言；其次，UML 不是上述方法的简单汇合，而是在这些方法的基础上广泛征求意见，集众家之长，几经修改而完成的，它扩展了现有方法的应用范围；第三，UML 是标准的建模语言，而不是标准的开发过程。尽管 UML 的应用必然以系统的开发过程为背景，但由于不同的组织和不同的应用领域，需要采取不同的开发过程。

作为一种建模语言，UML 是一个标准的图形表示法，仅仅是一组符号。它的定义包括 UML 语义和 UML 表示法两个部分。

- UML 语义：描述基于 UML 的精确元模型定义。元模型为 UML 的所有元素在语法和语义上提供了简单、一致、通用的定义性说明，使开发者能在语义上取得一致，消除了因人而异的最佳表达方法所造成的影响。此外，UML 还支持对元模型的扩展定义。
- UML 表示法：定义 UML 符号的表示法，为开发者或开发工具使用这些图形符号和文本语法，以及为系统建模均提供了标准。这些图形符号和文字所表达的是应用级的模型，在语义上它是 UML 元模型的实例。

1.1.3 UML 中的图

UML 中的图可分为两大类：结构图和行为图。结构图描绘系统组成元素之间的静态结构，行为图描绘系统元素的动态行为。

1. 结构图的类型

(1) 类图

类图是使用 UML 建模时最常用的图，它展示了系统中的静态事物、它们的结构以及它们之间的相互关系。这种图的典型用法是描述系统的逻辑设计和物理设计。

(2) 构件图

构件图可以展示一组构件的组织和彼此间的依赖关系，它用于说明系统如何实现，以及软件系统内构件如何协同工作等。

(3) 对象图

对象图可以展示系统中的一组对象，它是系统在某一时刻的快照，也可以说对象图是类图在某一时刻的快照。

(4) 部署图

部署图可以展示物理系统运行时的架构，同时可以描述系统中的硬件和硬件上驻留的软件。

(5) 组合结构图

组合结构图可以展示模型元素的内部结构。

(6) 包图

包图用于描绘包之间的依赖关系（包是一个用于组织其他模型元素的通用模型元素）。

(7) 用例图

用例描述了系统的工作方式，以及系统能提供的服务。用例图描述了系统外部参与者如何使用系统提供的服务。

注意：组合结构图、包图及用例图是 UML 2.0 中新增的结构图。

2. 行为图的类型

(1) 活动图

活动图显示系统内的活动流。通常需要使用活动图描述不同的业务过程。

(2) 状态图

状态图显示一个对象的状态和状态之间的转换。状态图中包括状态、转换、事件和活动。状态图是一个动态视图，对事件驱动的行为建模尤其重要，例如可以利用状态图描述一个电话路由系统中交换机的状态，不同的事件可以令交换机转移至不同的状态，用状态图对交换机建模有助于理解交换机的动态行为。在 UML 2.0 中，状态图被称为状态机图（State Machine Diagram）。

(3) 合作图

合作图是交互图的一种，交互图还包括顺序图（以及 UML 2.0 中新定义的其他几种图，稍后将介绍）。合作图突出对象之间的合作与交互。在 UML 2.0 中，合作图被通信图（Communication Diagram）所取代。

(4) 顺序图

顺序图是另一种交互图，它强调一个系统中不同元素间传递消息的时间顺序。

UML 2.0 中又增加了下列几种行为图：

(1) 时间图

时间图也是一种交互图，它描绘与交互元素的状态转换或条件变化有关的详细时间信息。

(2) 交互概观图

交互概观图是一种高层视图，用于从总体上显示交互序列之间的控制流。

注意：在实际进行系统建模时，几乎没有人会使用到 UML 标准中定义的所有图。

从应用的角度看，当采用面向对象技术设计系统时，第一步是描述需求；第二步根据需求建立系统的静态模型，以构造系统的结构；第三步是描述系统的行为。其中在第一步与第二步中所建立的模型都是静态的，包括用例图、类图（包含包）、对象图、构件图和部署图等 5 种图形，是标准建模语言 UML 的静态建模机制，第三步中所建立的模型或者可以执行，或者表示执行时的时序状态或交互关系，它包括状态图、活动图、顺序图和合作图等 4 种图形，是标准建模语言 UML 的动态建模机制。因此，标准建模语言 UML 的主要内容也可以归纳为静态建模机制和动态建模机制两大类。