



UML 基础



Rose 建模教程

| 蔡敏 徐慧慧 黄炳强 | 编著 |



人民邮电出版社
POSTS & TELECOM PRESS

UML基础



Rose建模教程

| 蔡敏 徐慧慧 黄炳强 | 编著 |

人民邮电出版社

图书在版编目 (CIP) 数据

UML 基础与 Rose 建模教程 / 蔡敏, 徐慧慧, 黄炳强编著.

—北京: 人民邮电出版社, 2006.1

ISBN 7-115-14229-7

I. U... II. ①蔡...②徐...③黄... III. 面向对象语言, UML—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2005) 第 150676 号

内 容 提 要

本书全面、详细地介绍了 UML 的基础知识和 Rational Rose 的使用方法, 并通过 4 个综合性的案例, 展示了使用 UML 和 Rose 进行软件建模的具体方法和步骤。全书共分 20 章, 前 4 章是基础部分, 介绍了 UML 和 Rose 的基础知识; 第 5 章到第 13 章是本书的重点, 介绍了 UML 的 9 种图和组成元素, 以及相关的建模技术, 并在每章的最后介绍了各种图在 Rose 中的绘制方法; 第 14 章到第 17 章是高级应用部分, 介绍了 UML 的扩展机制、双向工程、RUP 以及 OCL 的相关内容; 第 18 章到第 21 章是综合案例部分, 包括图书馆管理系统、汽车租赁系统、网络教学系统和网上图书销售系统, 以帮助读者加深对前面所学内容的理解与掌握, 达到边学习、边实践的目的。

本书不仅适合初学者学习 UML 建模的相关知识, 也适用于从事软件开发的工程人员学习和参考, 或者作为高等院校计算机和软件相关专业的教学用书或参考书。

UML 基础与 Rose 建模教程

- ◆ 编 著 蔡 敏 徐慧慧 黄炳强
责任编辑 汤 倩
- ◆ 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>
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 22.5
字数: 540 千字 2006 年 1 月第 1 版
印数: 1—6 000 册 2006 年 1 月北京第 1 次印刷

ISBN 7-115-14229-7/TP · 5108

定价: 36.00 元

读者服务热线: (010)67132692 印装质量热线: (010)67129223

前 言

20 世纪 50 年代, 软件诞生。20 世纪 60 年代, 软件工程的观念被提出。20 世纪 70 年代, 面向对象的建模语言出现。软件工程领域在 1995~1997 年取得了前所未有的进展, 其成果超过过去 15 年来的成就总和。其中最重要的、具有划时代意义的成果之一就是统一建模语言 (UML, Unified Modeling Language)。1997 年, UML1.1 被 OMG (对象管理组织) 采纳, 成为了软件工业界事实上的标准。

UML 是用来对软件系统进行描述、构造、可视化和文档编制的一种语言, 它统一了 Booch、Rumbaugh 和 Jacobson 的表示方法, 而且做了进一步的发展, 并最终成为标准的建模语言。

Rational Rose 是由美国的 Rational 公司开发的、一种面向对象的可视化建模工具。利用这个工具, 可以建立用 UML 描述的软件系统模型, 而且可以自动生成和维护 C++、Java、Visual Basic 和 Oracle 等语言和系统的代码。

为什么写本书

目前, 国内介绍 UML 和 Rational Rose 的资料较少, 内容不够全面, 特别是缺少综合介绍两者的中文资料。为此我们编写了本书, 旨在帮助读者全面了解 UML 的相关知识, 学习并实践使用 Rational Rose 建模的具体方法和步骤。

本书的特点

● 内容全面。书中详细地介绍了 UML 的基础知识, 如视图、图、模型元素和通用机制等, 同时, 结合具体的案例, 给出了相关理论在 Rational Rose 中的建模实践。特别值得一提的是, 书中介绍了国内 UML 相关书籍中很少提及的正向工程和逆向工程在 Rational Rose 中的具体实现方法。

● 案例丰富。全书共提供了 4 个综合性 Rational Rose 建模案例, 即图书馆管理系统、汽车租赁系统、网络教学系统和网上图书销售系统。其中, 图书馆管理系统的建模案例贯穿于 UML 基础知识的相关章节, 有助于读者边学习、边思考、边实践。

● 图文并茂。书中的各个章节都配有大量的设计流程图和 Rose 建模图, 有助于读者更加直观地理解 UML 的理论知识, 并在实际的学习和工作中学以致用。

读者对象

本书适用于从事面向对象软件开发的软件工程人员, 如系统和分析员、系统设计员、项目经理和实际的开发者等进行学习和参考, 也适合高等院校计算机软件工程相关专业、作为教学用书或参考书。

技术支持

为了方便高等院校相关专业的教学，本书配备了教学大纲和课件，如果需要，可以访问人民邮电出版社的网站：www.ptpress.com.cn 获取，或者与本书的责任编辑联系，责任编辑的 E-mail 是 tangqian@ptpress.cn。

本书主要由蔡敏、徐慧慧和黄炳强编写，负责图片绘制和资料整理的有茅健、陈晓和、万星新和赵玉玺等。在编写过程中，我们力求精益求精，但难免存在一些不足之处，如果读者使用本书时，发现差错或遇到问题，可以发 E-mail 到 tangqian@ptpress.cn 与我们联系。

编 者

目 录

第 1 章 基础知识	1
1.1 软件工程概述	1
1.1.1 软件工程的发展历史	1
1.1.2 软件工程的生命周期	1
1.1.3 现代软件工程	2
1.2 建模的目的	2
1.2.1 建模的重要性	3
1.2.2 建模四原则	4
1.2.3 面向对象建模	5
1.3 习题	5
第 2 章 面向对象技术与 UML	6
2.1 面向对象技术概述	6
2.1.1 面向对象的基本概念	6
2.1.2 面向对象的开发语言	10
2.1.3 面向对象的设计工具与开发工具	12
2.2 面向对象的分析	13
2.2.1 面向对象分析的层次	13
2.2.2 面向对象分析的基本过程	14
2.2.3 面向对象分析的基本原则	15
2.2.4 类图	16
2.2.5 发现对象和定义类	16
2.2.6 对象之间的通信	17
2.3 面向对象的设计	19
2.3.1 面向对象设计的模型	19
2.3.2 面向对象的高层设计	20
2.3.3 面向对象的类设计	22
2.4 基于 UML 的面向对象分析、设计过程	23
2.5 习题	23
第 3 章 Rational Rose 简介	24
3.1 初识 Rational Rose	24
3.1.1 Rational Rose 的特点	24
3.1.2 Rational Rose 的运行环境	25
3.1.3 Rational Rose 的获取	25
3.2 Rational Rose 的安装	25
3.3 Rational Rose 使用介绍	30

3.3.1	Rational Rose 的主界面	30
3.3.2	Rational Rose 的视图	49
3.3.3	使用 Rational Rose 建模	53
3.3.4	设置全局选项	61
3.4	习题	62
第 4 章	UML 语言纵览	63
4.1	UML 的构成	63
4.2	视图	63
4.2.1	用例视图	65
4.2.2	逻辑视图	65
4.2.3	并发视图	65
4.2.4	组件视图	65
4.2.5	配置视图	65
4.3	图	66
4.4	UML1.4 与 UML2.0 的比较	67
4.5	模型元素	68
4.5.1	事物	68
4.5.2	UML 中的关系	70
4.6	通用机制	71
4.6.1	修饰	71
4.6.2	注释	71
4.6.3	规格说明	72
4.6.4	通用划分	72
4.6.5	扩展机制	72
4.7	UML 的扩展组件	73
4.8	UML 建模的简单流程	73
4.9	习题	75
第 5 章	用例图	76
5.1	用例图的概念	76
5.1.1	概述	76
5.1.2	参与者 (Actor)	76
5.1.3	用例 (Use Case)	78
5.1.4	用例间的关系	80
5.2	用例图建模技术	82
5.2.1	对语境建模	82
5.2.2	对需求建模	82
5.3	实例——图书馆管理系统中的用例图	83
5.3.1	确定系统涉及的总体信息	83
5.3.2	确定系统的参与者	83

5.3.3	确定系统用例	83
5.3.4	使用 Rose 绘制用例图的步骤	84
5.3.5	图书馆管理系统中的用例图	90
5.4	习题	92
第 6 章	类图 and 对象图	93
6.1	类图的概念	93
6.1.1	概述	93
6.1.2	类 (Class)	94
6.1.3	接口 (Interface)	98
6.1.4	类之间的关系	98
6.2	类图建模技术	104
6.2.1	对简单协作建模	104
6.2.2	对逻辑数据库模式建模	104
6.2.3	正向工程和逆向工程	104
6.3	对象图的概念	105
6.3.1	概述	105
6.3.2	类图和对象图的区别	105
6.4	对象图建模技术	106
6.4.1	对对象结构建模	106
6.4.2	正向工程和逆向工程	106
6.5	实例——图书馆管理系统的类图	107
6.5.1	使用 Rose 绘制类图的步骤	107
6.5.2	图书馆管理系统中的类图	113
6.6	习题	114
第 7 章	包图	115
7.1	概述	115
7.2	包 (Package)	115
7.2.1	名称	115
7.2.2	拥有的元素	116
7.2.3	可见性	116
7.2.4	引入与输出	117
7.2.5	标准元素	117
7.3	包之间的关系	118
7.4	包图建模技术	118
7.4.1	对成组的元素建模	118
7.4.2	对体系结构视图建模	118
7.5	实例——图书馆管理系统的包图	119
7.5.1	使用 Rose 绘制包图的步骤	119
7.5.2	图书馆管理系统中的包图	122

7.6 习题	123
第 8 章 状态图	124
8.1 状态机	124
8.2 状态图	124
8.2.1 状态	125
8.2.2 转换	125
8.2.3 初始状态	126
8.2.4 终止状态	126
8.2.5 判定	126
8.3 状态	127
8.3.1 概述	127
8.3.2 状态名	127
8.3.3 内部转换	127
8.3.4 入口动作与出口动作	128
8.3.5 简单状态	128
8.3.6 组成状态	128
8.3.7 历史状态	130
8.4 事件	131
8.4.1 入口事件	131
8.4.2 出口事件	131
8.4.3 动作事件	131
8.4.4 信号事件	131
8.4.5 调用事件	131
8.4.6 修改事件	131
8.4.7 时间事件	132
8.4.8 延迟事件	132
8.5 转换	132
8.5.1 外部转换	132
8.5.2 内部转换	133
8.5.3 完成转换	133
8.5.4 复合转换	133
8.5.5 触发事件	134
8.5.6 监护条件	134
8.5.7 动作	134
8.6 状态图建模技术	135
8.7 实例——图书馆管理系统的状态图	135
8.7.1 使用 Rose 绘制状态图的步骤	135
8.7.2 图书馆管理系统中的状态图	141
8.8 习题	143

第 9 章 活动图	144
9.1 概述	144
9.1.1 活动图的图形表示	144
9.1.2 活动图与流程图的区别	145
9.2 活动图的组成元素	145
9.2.1 动作状态	145
9.2.2 活动状态	145
9.2.3 动作流	146
9.2.4 分支与合并	146
9.2.5 分叉与汇合	147
9.2.6 泳道	148
9.2.7 对象流	149
9.3 活动的分解	150
9.4 活动图建模技术	151
9.5 实例——图书馆管理系统的活动图	151
9.5.1 使用 Rose 绘制活动图的步骤	151
9.5.2 图书馆管理系统中的活动图	159
9.6 习题	162
第 10 章 时序图	163
10.1 概述	163
10.2 时序图的组成	164
10.2.1 对象	164
10.2.2 生命线	164
10.2.3 消息	164
10.2.4 激活	165
10.3 对象的创建和撤销	165
10.4 时序图建模	166
10.5 实例——图书馆管理系统的时序图	167
10.5.1 使用 Rose 绘制时序图的步骤	167
10.5.2 图书馆管理系统中的时序图	170
10.6 习题	173
第 11 章 协作图	174
11.1 概述	174
11.2 对象、链和消息	174
11.2.1 对象	174
11.2.2 链	175
11.2.3 消息	175
11.3 时序图与协作图的比较	175
11.4 时序图与协作图的互换	176

11.5	协作图的建模	177
11.6	实例——图书馆管理系统的协作图	178
11.6.1	使用 Rose 绘制协作图的步骤	178
11.6.2	图书馆管理系统的协作图	182
11.7	习题	183
第 12 章	组件图	184
12.1	概述	184
12.2	组件	185
12.2.1	名称	185
12.2.2	组件的种类	185
12.2.3	Rose 中不同类型组件的图标表示	186
12.3	接口	188
12.4	依赖关系	188
12.5	组件图的建模技术	189
12.6	实例——图书馆管理系统的组件图	189
12.6.1	使用 Rose 绘制组件图的步骤	189
12.6.2	图书馆管理系统中的组件图	194
12.7	习题	195
第 13 章	配置图	196
13.1	概述	196
13.2	节点	196
13.2.1	名称	197
13.2.2	节点的种类	197
13.2.3	节点中的配置	197
13.3	关联关系	198
13.4	配置图的建模	199
13.5	实例——图书馆管理系统的配置图	199
13.5.1	使用 Rose 绘制配置图的步骤	199
13.5.2	图书馆管理系统中的配置图	205
13.6	习题	205
第 14 章	UML 的扩展机制	206
14.1	UML 的体系结构	206
14.1.1	四层元模型体系结构	206
14.1.2	元元模型层	207
14.1.3	元模型层	208
14.2	构造型	209
14.2.1	构造型的表示法	210
14.2.2	UML 中预定义的标准构造型	210
14.3	标记值	213

14.3.1	标记值的表示法	213
14.3.2	UML 中预定义的标准标记值	214
14.4	约束	214
14.4.1	约束的表示法	214
14.4.2	UML 中预定义的标准约束	215
14.5	用于业务建模的 UML 扩展	216
14.5.1	业务模型建模的构造型	217
14.5.2	业务建模的关联规则	217
14.5.3	业务建模的构造型图标	218
14.6	习题	218
第 15 章	Rose 的双向工程	219
15.1	双向工程简介	219
15.2	Java 代码生成	220
15.3	逆向工程	226
15.4	实例——类图的代码生成与逆向工程	228
15.5	习题	233
第 16 章	UML 与统一开发过程	234
16.1	软件开发过程	234
16.1.1	软件开发过程简介	234
16.1.2	当前流行的软件过程	235
16.2	RUP 简介	235
16.2.1	RUP 的产生背景	235
16.2.2	传统的软件开发模型	235
16.3	RUP 的二维开发模型	237
16.3.1	RUP 的核心 workflow	238
16.3.2	RUP 的 4 个阶段	239
16.3.3	RUP 的迭代开发模型	242
16.4	RUP 的核心 workflow	243
16.4.1	需求捕获 workflow	243
16.4.2	分析 workflow	247
16.4.3	设计 workflow	249
16.4.4	实现 workflow	252
16.4.5	测试 workflow	255
16.5	习题	260
第 17 章	对象约束语言	261
17.1	对象约束语言概述	261
17.2	标准 OCL 类型	261
17.2.1	OclType 类型	261
17.2.2	OclAny 类型	262

17.2.3	基本类型	263
17.2.4	集合类型	263
17.2.5	OCL 中的关键字	266
17.3	OCL 表达式	266
17.3.1	OCL 表达式的语法	266
17.3.2	操作符的优先级	267
17.4	用 OCL 表达对象性质约束	267
17.4.1	不变量	268
17.4.2	前置条件和后置条件	268
17.4.3	let 表达式与约束	268
17.5	习题	268
第 18 章	图书馆管理系统	269
18.1	软件需求分析	269
18.1.1	软件需求的定义	269
18.1.2	软件需求的层次	269
18.2	图书馆管理系统的需求分析	270
18.2.1	系统功能需求	270
18.2.2	基本数据维护模块	271
18.2.3	基本业务模块	272
18.2.4	数据库模块	272
18.2.5	信息查询模块	273
18.3	系统的 UML 基本模型	273
18.3.1	UML 初始模型	273
18.3.2	系统的用例图	274
18.3.3	系统的时序图	277
18.3.4	系统的协作图	281
18.3.5	系统的状态图	284
18.3.6	系统的活动图	285
18.4	系统中的类	289
18.4.1	类图的生成	289
18.4.2	各类之间的关系	291
18.5	系统的配置与实现	292
18.5.1	系统的组件图	292
18.5.2	系统的配置图	292
第 19 章	汽车租赁系统	294
19.1	汽车租赁系统的需求分析	294
19.1.1	系统功能需求	294
19.1.2	基本数据维护模块	295
19.1.3	基本业务模块	295

19.1.4	数据库模块	296
19.1.5	信息查询模块	296
19.2	系统的 UML 基本模型	297
19.2.1	UML 模型框架	297
19.2.2	系统的用例图	298
19.2.3	系统的时序图	300
19.2.4	系统的协作图	303
19.2.5	系统的状态图	305
19.2.6	系统的活动图	306
19.3	系统中的类	307
19.3.1	类图的生成	307
19.3.2	各个类之间的关系	309
19.4	系统的配置与实现	310
19.4.1	系统的组件图	310
19.4.2	系统的配置图	310
第 20 章	网络教学系统	312
20.1	网络教学系统的需求分析	312
20.1.1	系统功能需求	312
20.1.2	数据库管理模块	312
20.1.3	基本业务模块	313
20.1.4	信息浏览、查询模块	313
20.2	系统的 UML 基本模型	314
20.2.1	UML 初始模型	314
20.2.2	系统的用例图	315
20.2.3	系统的时序图	317
20.2.4	系统的协作图	320
20.2.5	系统的状态图	321
20.2.6	系统的活动图	322
20.3	系统中的类	324
20.3.1	类图的生成	324
20.3.2	各类之间的关系	326
20.4	系统的配置与实现	327
20.4.1	系统的组件图	327
20.4.2	系统的配置图	327
第 21 章	网上图书销售系统	329
21.1	网上图书销售系统的需求分析	329
21.1.1	系统总体的功能需求	329
21.1.2	用户接口模块	330
21.1.3	管理员接口模块	331

21.2	系统的 UML 建模	333
21.2.1	建立初始模型	333
21.2.2	系统的用例图	334
21.2.3	系统的时序图与协作图	336
21.2.4	系统的活动图	339
21.3	系统中的类	340
21.3.1	类图的生成	340
21.3.2	各类之间的关系	342
21.4	系统的配置与实现	342
21.4.1	系统的组件图	343
21.4.2	系统的配置图	343
参考文献		344

第1章 基础知识

本章将对软件工程、建模的目的和 UML 进行简要的介绍。通过本章的阅读，读者可以对什么是软件工程、为什么要建模以及什么是 UML 有一个清楚地认识。

1.1 软件工程概述

从软件工程概念的提出至今已经近 40 年了，但是客观地说，软件工程还处于摸索发展的阶段。本节将主要介绍软件工程的 development 历史、软件工程的生命周期及其发展趋势。

1.1.1 软件工程的发展历史

1946 年，世界上第一台电子计算机在美国研制成功。20 世纪 50 年代软件诞生。伴随计算机和软件出现的还有软件从业人员，这些人多数是数学家和电子工程师。

在计算机系统发展的初期，计算机通常只执行一个单一的、为某个特定目的所编写的程序，这就使得早期软件的通用性十分有限。早期大多数的软件都是由使用者自己编写的，往往带有很强的个人色彩。早期的软件开发也没有系统的方法可以遵循，开发者只是在大脑中构思一个大体的流程，除了源代码以外没有任何文档。

从 20 世纪 60 年代中期到 70 年代中期，软件业进入了一个发展的时期。这一时期软件作为一种产品开始被广泛使用，同时出现了所谓的软件公司。这一时期的软件开发方法仍然沿用早期的自由的软件开发方式。但是随着软件规模的急剧膨胀，软件的需求日趋复杂，软件维护的难度越来越大，开发成本以指数级的速度增长，失败的软件项目也比比皆是，这就是“软件危机”。

概括说来，软件危机包含了两个方面：

- (1) 如何开发软件，以满足不断增长的、日趋复杂的要求；
- (2) 如何维护规模不断庞大的软件产品。

“软件危机”的出现使得人们开始对软件开发的方法进行重新审视。人们意识到，优秀的程序除了功能正确、性能优良以外，还应该易读、易用、易维护。而早期所谓的优秀程序常常通篇充满了程序员的编程技巧，很难被别人看懂。

1968 年，北大西洋公约组织的科技委员会召集了近 50 名一流的程序员、计算机科学家，以及工业界人士在德国召开了一次以讨论和制定摆脱“软件危机”为主题的国际学术会议，会议上第一次提出了软件工程（Software Engineering）概念。

软件工程是一门建立在系统化、规范化、数量化等工程原则和方法上的，关于软件开发各阶段的定义、任务和作用的工程学科。软件工程包括两方面内容：软件开发技术和软件项目管理。软件开发技术包括软件开发方法学、软件工具和软件工程环境；软件项目管理包括软件度量、项目估算、进度控制、人员组织、配置管理和项目计划等。

1.1.2 软件工程的生命周期

经典的软件工程思想将软件开发分成 5 个阶段：需求分析阶段（Requirements Capture）、

系统分析与设计 (System Analysis and Design) 阶段、系统实现 (Implementation) 阶段、测试 (Testing) 阶段和维护 (Maintenance) 阶段。

(1) 需求分析阶段。软件需求是指用户对目标软件系统在功能、行为、性能和设计约束等方面的期望。通过对应用问题及其环境的理解与分析,建立系统模型,将用户的需求精确化、完全化,最终形成需求规格说明,这一系列的活动就构成了需求分析阶段。

需求分析是整个系统开发过程的基础,对后面几个阶段的进展有直接的影响。纵观软件技术的发展过程,不难发现,需求分析的工作量在不断增加,其地位也在不断提高。

(2) 系统分析与设计阶段。系统分析与设计阶段包括紧密联系的分析和设计两个阶段。分析与设计阶段通常需要在系统分析员的指导下进行,系统分析员不仅要对计算机的专业知识有深刻的理解,而且还要对相关的业务有一定的了解。一般情况下,系统分析是与需求分析同时进行的,而系统设计一般是在系统需求之后进行的。

(3) 系统实现阶段。系统实现阶段也就是通常所说的编码阶段。在软件工程思想出现以前,编码几乎是软件开发的全部。而在现代的软件工程中,编码阶段所占的比重正在逐渐地缩小。

(4) 测试阶段。软件测试是软件工程过程的一个重要阶段,此阶段的主要任务是对软件需求分析、设计和编码各阶段的产品进行检查,保证软件开发产品的正确性、完全性和一致性,检测并修正软件中出现的错误。软件测试以检查软件产品内容和功能特性为核心,不仅是软件质量保证的关键步骤,也是软件开发目标实现的重要保障。

(5) 维护阶段。软件维护是在软件产品交付之后进行的修改工作。在软件工程思想出现之前,由于软件浓烈的个人色彩,使得软件的修改成为一个令所有人头疼的问题。即使在软件工程思想迅速发展的今天,维护问题仍然是软件开发面临的最困难的问题。

1.1.3 现代软件工程

软件开发中包含了物和人的因素,存在着很大的不确定性,这就使得软件工程不可能像理想的、可以基于物理学等的原理来做的物质生产过程。最初的软件开发只考虑到了人的因素,而传统的软件工程又过分强调物的因素,现代软件工程则最重视人与物的关系,即人和机器(工具、自动化)在不同层次上、不断循环发展的关系。

基于面向对象的分析、设计方法的出现使得软件的开发方法发生了翻天覆地的变化。随之而来的是面向对象建模语言(以 UML 为代表)、软件复用、基于组件的软件开发等新的方法和领域。软件工程进入了一个新的发展阶段。

1.2 建模的目的

在软件界有这么一条真理:一个开发团队首要关注的不应是漂亮的文档、世界级的会议、响亮的口号或者华丽的源码,而是如何满足用户和项目的需要。

为了保证软件满足要求,开发组织必须深入到使用者中间了解系统的真实需求;为了开发具有持久质量保证的软件,开发组织必须建立一个富有弹性的、稳固的结构基础;为了快速、高效地开发软件并使无用和重复开发最小化,开发组织必须具有精干的开发人员、正确的开发工具和合适的开发重点。为了实现以上的要求,在对系统生存周期正确估计的基础上,