

## <<UML与Rational Rose 20>>

### 图书基本信息

书名：<<UML与Rational Rose 2003从入门到精通>>

13位ISBN编号：9787121099519

10位ISBN编号：7121099519

出版时间：2010-1

出版时间：电子工业

作者：解本巨//李晓娜//宫生文

页数：317

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## &lt;&lt;UML与Rational Rose 20&gt;&gt;

## 前言

UML是一种具有规范定义、易于表达、功能强大且适用于各种应用领域的建模语言。

1997年11月，UML被国际对象管理组织OMG采纳为面向对象建模语言的国际标准。

目前UML已经成为面向对象技术领域内占主导地位的标准建模语言。

掌握UML语言，不仅有助于理解面向对象的分析与设计方法，也有助于对软件开发全过程的理解。

UML自1995年出现，到1997年，Rational将UML语言（1.1版）的权利授给OMG组织，以使UML成为一个公共标准。

从那时起，OMG协会开始了对UML标准的修改，并在1998年发布了UML 1.3版，在2000年发布了UML 1.4版，在2004年又发布了当前使用的2.0版。

目前UML最新的3.0版本正在制订中。

在众多支持UML的软件设计开发环境中，Rational Rose是业界领先的UML工具，成为到目前为止最畅销的UML产品。

多年以来，开发人员成功地为Rational Rose增加了新特性。

例如，与诸如Visual Basic的编程语言的集成，及最近与IBM Rational XDE产品的集成。

目前，IBM Rational的最新UML产品版本是IBM Rational Software Architect V7.5。

本书采用的是目前应用最为广泛的Rational Rose 2003版本。

本书以UML应用和实例为主，由浅入深地向读者介绍了UML的基础知识，以及如何在实际项目中应用UML技术。

全书共分为18章。

其中第一部分包括第1章～第11章，主要介绍UML的基础知识。

第1章和第2章简要介绍UML的基本概念和发展过程，以及面向对象技术。

第3章主要介绍Rational公司Rose工具的基本使用。

第4章～第11章详细介绍UML的用例图、类图、对象图、状态图、时序图、协作图、活动图、组件图和配置图，并以读者熟悉的“图书管理系统”、“学生成绩管理系统”、“ATM自动取款机系统”，以及“网络教学系统”为例，讲解了UML中这9种图的具体应用。

第二部分包括第12章～第15章，主要介绍UML的扩展机制、双向工程、统一开发过程，以及对象约束语言等高级应用。

第三部分包括第16章～第18章，主要介绍如何在实际的项目中应用UML，分别以工资管理系统、学籍管理系统和在线购物系统这三个实际的项目讲述UML在软件项目的需求分析、系统设计、系统实现、集成和交付这几个阶段的应用。

本书在编写过程中得到了青岛大学孙更新老师在技术上的大力支持，此外宾晟、孙海伦等老师为本书的编写提出过宝贵意见并参与了本书部分资料的搜集工作，感谢北京美迪亚电子信息有限公司各位老师，谢谢你们的帮助和指导。

尽管我们尽了最大努力，但由于时间仓促，加之水平有限，本书难免有不妥之处，欢迎各界专家和读者朋友批评指正。

## <<UML与Rational Rose 20>>

### 内容概要

详细介绍了UML语言的基础知识，以及UML在面向对象的软件系统分析和设计中的应用，并通过丰富的实例讲解了面向对象的分析与设计过程，启发读者如何用UML语言将所学到的面向对象技术应用于软件系统的分析、设计与开发中。

《UML与Rational Rose 2003从入门到精通》从理论的高度阐述了面向对象分析和设计的思想，使读者能够真正地掌握系统架构设计的精髓。

《UML与Rational Rose 2003从入门到精通》将理论与实例结合，内容繁简得当，由浅入深，使读者能十分容易入门并逐步精通。

《UML与Rational Rose 2003从入门到精通》可供正在学习编程、软件工程等知识，准备将来从事IT行业的读者、正努力向设计师或系统分析员发展的技术人员及期望对软件分析设计的运用更上一层楼的设计人员学习和提高之用，也可以作为计算机相关专业软件工程等课程的教材。

## 书籍目录

第1章 软件工程与UML概述 11.1 软件工程概述 11.1.1 软件工程的产生 11.1.2 软件工程的内容及基本原理 31.1.3 软件生命周期 51.1.4 软件开发模型 81.1.5 面向对象方法学 141.2 建模概述 161.2.1 什么是模型 161.2.2 建模的原理 161.2.3 建模的意义与误区 181.2.4 面向对象建模 211.3 UML概述 221.3.1 UML的起源与发展 221.3.2 UML的特点 231.3.3 UML的定义 241.3.4 UML的应用领域 261.3.5 UML 2.0的新特性 271.4 本章小结 28第2章 面向对象技术 292.1 面向对象的技术基础 292.1.1 面向对象技术的起源和发展 292.1.2 面向对象的基本概念 312.1.3 面向对象技术的特点 352.1.4 支持面向对象的语言 362.2 面向对象的分析 402.2.1 面向对象分析模型的层次 412.2.2 面向对象分析的基本过程 412.2.3 面向对象分析的原则 452.3 面向对象的设计 462.3.1 面向对象设计的准则 472.3.2 面向对象设计的模型 482.3.3 面向对象设计的步骤 512.3.4 面向对象的类设计原则 522.3.5 面向对象设计时需要注意的问题 552.4 基于UML的面向对象分析和设计过程 572.4.1 系统需求 572.4.2 图书管理系统用例分析 592.4.3 系统静态建模 592.4.4 系统动态建模 602.5 面向对象实现 602.5.1 程序设计 602.5.2 面向对象测试 622.6 本章小结 63第3章 Rational Rose概述 643.1 初识Rational Rose 643.2 Rational Rose 2003的安装 653.2.1 Rational Rose 2003安装前的准备 653.2.2 Rational Rose 2003安装步骤 663.3 使用Rational Rose 2003 683.3.1 启动Rational Rose 683.3.2 使用Rational Rose建模 713.3.3 设置全局选项 743.3.4 定制工具栏 743.3.5 设置组件的属性 753.3.6 框图设计 753.4 本章小结 78第4章 UML初览 794.1 UML中的视图 794.1.1 用例视图 804.1.2 逻辑视图 804.1.3 并发视图 814.1.4 组件视图 814.1.5 配置视图 814.2 UML中的图 814.2.1 用例图 ( Use Case Diagram ) 824.2.2 类图 ( Class Diagram ) 824.2.3 对象图 ( Object Diagram ) 824.2.4 状态图 ( State Diagram ) 834.2.5 活动图 ( Activity Diagram ) 834.2.6 时序图 ( Sequence Diagram ) 834.2.7 协作图 ( Collaboration Diagram ) 844.2.8 组件图 ( Component Diagram ) 844.2.9 配置图 ( Deployment Diagram ) 844.3 模型元素 844.3.1 UML中的事物 854.3.2 UML中的关系 864.4 通用机制 874.4.1 修饰 874.4.2 规格说明 874.4.3 通用划分 884.4.4 扩展机制 884.5 UML建模的基本流程 894.6 本章小结 90第5章 用户模型视图——用例图 915.1 用例图的概念 915.1.1 用例图的元素——用例 ( use case ) 915.1.2 用例图的元素——参与者 ( Actor ) 925.2 用例和用例之间的关系 935.2.1 识别、分析与创建用例 935.2.2 使用用例的误区 945.2.3 用例之间的关系 955.3 用例图建模技术 985.4 实例——ATM自动取款机系统中的用例图 995.5 实例——学生成绩管理系统中的用例图 1015.6 实例——图书管理系统中的用例图 1025.6.1 系统需求 1025.6.2 图书管理系统中的用例图 1055.7 实例——网络教学系统中的用例图 1065.7.1 系统需求 1065.7.2 网络教学系统中的用例图 1075.8 本章小结 109第6章 逻辑视图——类图和对象图 1106.1 类图 1106.1.1 类图的概念 1106.1.2 类 1106.1.3 接口 1176.1.4 关系 1196.1.5 建立类图 1206.1.6 类图建模技术 1226.1.7 类图小结 1236.2 对象图 1246.2.1 对象 1246.2.2 对象图的概念 1256.3 包与包图 1266.3.1 包 ( Package ) 1276.3.2 包图 1296.4 逻辑视图实例 1316.4.1 实例——图书管理系统的类图 1316.4.2 其他类图实例 1336.5 本章小结 137第7章 并发视图——状态图 1387.1 状态机与状态图 1387.1.1 状态机 1387.1.2 状态图 1397.2 状态 1407.2.1 概述 1407.2.2 简单状态 1407.2.3 组成状态 1417.2.4 初始状态与终止状态 1427.2.5 历史状态 1427.2.6 判定 1437.3 转换 1437.3.1 源状态与目标状态 1447.3.2 触发事件 1447.3.3 警戒条件 1457.3.4 监护条件 1457.3.5 动作 1467.3.6 转换的类型 1477.4 事件 1487.5 状态图建模技术 1497.5.1 状态图建模步骤 1497.5.2 创建状态图通用准则 1537.6 状态图的图标 1547.7 状态图实例 1547.7.1 图书管理系统中的状态图 1547.7.2 其他系统中的状态图 1577.8 本章小结 159第8章 并发视图——活动图 1608.1 活动图概述 1608.1.1 活动图的基本概念 1608.1.2 活动图的用途 1618.1.3 活动图的优点和缺点 1628.2 活动图的组成元素 1638.2.1 动作状态 1638.2.2 活动状态 1638.2.3 判定 1648.2.4 转换与动作流 1648.2.5 分支与合并 1658.2.6 分叉与汇合 1658.2.7 泳道 1658.2.8 对象流 1678.3 活动图建模技术 1678.4 活动图的图标 1688.5 活动图实例 1698.5.1 图书管理系统中的活动图 1698.5.2 其他系统中的活动图 1728.6 本章小结 175第9章 并发视图——时序图 1769.1 时序图概述 1769.2 时序图的组成 1779.3 时序图建模技术 1799.4 时序图的图标 1799.5 时序图实例 1809.5.1 图书管理系统中的时序图 1809.5.2 其他系统中的时序图 1849.6 本章小结 187第10章 并发视图——协作图 18810.1 协作图概述 18810.1.1 相关概念 18810.1.2 协作图的概念 18910.1.3 协作图的必要性 18910.1.4 协作图示例 19010.1.5 协作图的建模技术 19010.2 协作图的组成元素 19110.2.1 对象、链和消息 19110.2.2 协作图在UML中的表示方法 19210.3 时序图与协作图的比较与互换 19410.3.1 时序图与协作图的比较 19410.3.2 时序图与协作图

的互换 19510.3.3 协作图常用的可视化图符 19510.4 协作图实例 19610.4.1 图书管理系统中的协作图  
19610.4.2 其他系统中的协作图 19810.5 本章小结 199第11章 组件视图与配置视图——组件图与配置图  
20011.1 组件图 20011.1.1 组件图概述 20011.1.2 组件图的组成元素 20111.1.3 组件图建模技术 20311.1.4 组  
件图实例 20311.2 配置图 20611.2.1 配置图概述 20611.2.2 节点 20711.2.3 关系 20811.2.4 配置图建模技术  
20811.2.5 配置图实例 20911.3 本章小结 211第12章 UML的扩展机制 21212.1 UML的体系结构 21212.1.1 四  
层体系结构 21212.1.2 元元模型层 21312.1.3 元模型层 21412.2 构造型 21512.2.1 构造型的表示法 21512.2.2  
UML中预定义的标准构造型 21612.3 标记值 21812.3.1 表示标记值 21812.3.2 UML中预定义的标准标记值  
21812.3.3 自定义标记值 21912.4 约束 21912.4.1 表示约束 22012.4.2 UML中预定义标准约束 22212.5 用于业  
务建模的UML扩展 22312.6 本章小结 225第13章 Rose中的双向工程 22613.1 双向工程简介 22613.1.1 Rose  
对编程语言的支持 22613.1.2 RTE举例 22613.2 用Rational Rose生成代码 22813.2.1 代码生成步骤 22813.2.2  
ANSI C++代码生成 23313.2.3 Java代码生成 23513.2.4 Visual Basic代码生成 23813.3 逆向工程 24113.3.1 Java  
逆向工程 24213.3.2 ANSI C++逆向工程 24513.3.3 Visual Basic逆向工程 24613.4 本章小结 247第14章 UML  
与统一开发过程 24814.1 软件开发过程 24814.1.1 软件开发过程简介 24814.1.2 当前流行的软件过程  
24914.2 RUP简介 24914.2.1 什么是RUP过程 24914.2.2 传统的软件开发模型 25014.3 RUP的二维开发模型  
25214.3.1 RUP二维模型的静态结构 25214.3.2 RUP二维模型的时间维 25414.3.3 RUP的迭代开发模型  
25614.4 RUP的核心 workflow 25714.4.1 需求 workflow 25714.4.2 分析 workflow 26014.4.3 设计 workflow 26214.4.4 实  
现 workflow 26514.4.5 测试 workflow 26714.5 小结 270第15章 对象约束语言 27115.1 对象约束语言概述 27115.2  
OCL结构 27115.2.1 抽象语法 27215.2.2 具体语法 27215.3 使用集合 27215.3.1 创建集合 27315.3.2 操作集合  
27315.4 标准OCL类型 27415.4.1 OclVoid类型和OclAny类型 27415.4.2 OclMessage类型 27515.4.3 基本类型  
27515.4.4 集合类型 27715.4.5 模型元素类型 28115.5 OCL表达式 28215.6 用OCL表达对象性质约束  
28315.6.1 不变量 28315.6.2 前置条件与后置条件 28315.6.3 let表达式与约束 28315.7 本章小结 284第16章 工  
资管理系统 28516.1 需求分析 28516.2 UML系统建模 28616.2.1 工资管理系统的用例图 28616.2.2 工资管理  
系统的类图 28716.2.3 工资管理系统的时序图 28916.2.4 工资管理系统的协作图 29416.2.5 工资管理系统的  
组件图 29716.2.6 工资管理系统的配置图 298第17章 学籍管理系统 29917.1 需求分析 29917.2 UML系统建  
模 30017.2.1 学籍管理系统的用例图 30017.2.2 学籍管理系统的活动图 30017.2.3 学籍管理系统的时序图  
30317.2.4 学籍管理系统的协作图 30517.2.5 学籍管理系统的类图 30817.2.6 学籍管理系统的组件图  
30917.2.7 学籍管理系统的配置图 310第18章 在线购物系统 31118.1 需求分析 31118.2 UML系统建模  
31218.2.1 在线购物系统的用例图 31218.2.2 在线购物系统的类图 31318.2.3 在线购物系统的活动图  
31418.2.4 在线购物系统的时序图 31418.2.5 在线购物系统的协作图 31618.2.6 在线购物系统的组件图  
31618.2.7 在线购物系统的配置图 317

## 章节摘录

(1) 计划(需求定义) 第一周期开始利用需求分析技术理解应用领域, 获取初步用户需求, 制订项目开发计划(即整个软件生命周期计划)和需求分析计划。经过一个周期后, 根据用户和开发人员对上一周期工作成果的评价和评审来修改和完善需求, 明确下一周期软件开发的目标和约束条件, 并据此制订新一轮的软件开发计划。

(2) 风险分析 根据本轮制订的开发计划, 进行风险分析, 评估可选方案, 并构造原型进一步分析风险, 给出消除或减少风险的途径。

此时根据风险分析的结果决策项目是否继续。

所以, 螺旋模型是一个风险驱动模型。

(3) 工程实现 利用构造的原型进行需求建模或系统模拟, 直至实现软件系统。

(4) 用户评价与阶段评审 将原型提交用户使用并征求改进意见。

开发人员应在用户的密切配合下进一步完善用户需求, 直到用户认为原型可满足需求, 或对软件产品设计进行评价或确认等。

螺旋模型从第一个周期的计划开始, 一个周期一个周期地不断迭代, 直到整个软件系统开发完成。

螺旋模型的优点如下: (1) 支持用户需求的动态变化, 这就要求构造的原型的总体结构、算法、程序及测试方案应具有良好的可扩充性和可修改性。

也支持软件系统的可维护性, 每次维护过程只是沿螺旋模型继续多走一两个周期。

(2) 原型可看做形式的可执行的需求规格说明, 易于被用户和开发人员共同理解, 还可作为继续开发的基础, 并为用户参与所有关键决策提供了方便。

(3) 螺旋模型特别强调原型的可扩充性和可修改性, 原型的进化贯穿整个软件生存周期, 这将有助于提高目标软件的适应能力。

(4) 螺旋模型为项目管理人员及时调整管理决策提供了方便, 进而可降低开发风险。

螺旋模型的缺点有: (1) 如果每次迭代的效率不高, 致使迭代次数过多, 将会增加成本并推迟提交时间。

(2) 使用该模型需要有相当丰富的风险评估经验和专门知识, 要求开发队伍水平较高。

螺旋模型的适应场合是: 支持需求不明确、特别是大型软件系统的开发, 并支持面向规格说明、面向过程、面向对象等多种软件开发方法, 是一种具有广阔前景的模型。

4. 喷泉模型 喷泉模型是近几年提出来的软件生存周期模型。

它是面向对象的软件开发方法为基础, 以用户需求为动力, 以对象来驱动模型。

## <<UML与Rational Rose 20>>

### 编辑推荐

理论与实例结合、内容繁简得当，使读者能十分容易入门并逐步精通。

详细讲解了UML中所涉及的关键概念、术语和技术，并对UML的基础概念重新组织和归纳整理。

从理论的高度阐述面向对象设计思想，使读者真正掌握系统架构设计的精髓。通过多个实际案例对使用Rational Rose进行UML建模的全过程进行了深入的剖，并对在现实中经常遇到并且较难掌握的问题进行了深入的讲解。

## <<UML与Rational Rose 20>>

### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>