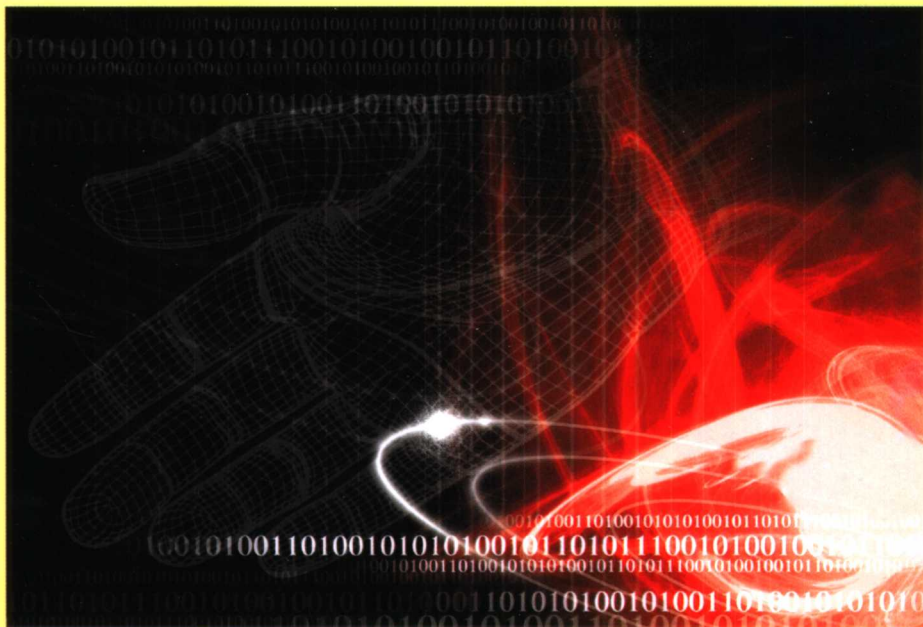


# UML

# 课程设计

董兰芳 刘振安 等编著



机械工业出版社  
China Machine Press

高等院校计算机课程设计指导丛书

# UML

# 课程设计

董兰芳 刘振安 等编著



机械工业出版社  
China Machine Press

本书是为了配合UML建模语言的教学而编写的，目的是通过课程设计的综合训练，培养学生实际分析问题、编程和动手能力，帮助学生系统地掌握该门课程的主要内容，更好地完成教学任务。

本书从用户的需求开始，通过11个课程设计，结合相关概念和设计任务，详细地描述了一个典型的“选课系统”设计实例，并逐步介绍使用Rose实现设计的全过程。课程设计后面均附有评分标准，既可供读者评判自己的设计，也为教师提供方便。

本书既可以作为高等院校相关专业的课程设计教材，也可以作为自学教材或教学参考书。

**版权所有，侵权必究。**

**本书法律顾问 北京市展达律师事务所**

### **图书在版编目(CIP)数据**

UML课程设计/董兰芳等编著. —北京: 机械工业出版社, 2005.1

(高等院校计算机课程设计指导丛书)

ISBN 7-111-15516-5

I. U… II. 董… III. 面向对象语言, UML—程序设计—高等学校—教学参考资料  
IV. TP312

中国版本图书馆CIP数据核字(2004)第112253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)

策划编辑: 温莉芳

责任编辑: 朱起飞

北京昌平奔腾印刷厂印刷·新华书店北京发行所发行

2005年1月第1版第1次印刷

787mm × 1092mm 1/16 · 10.75印张

印数: 0 001-4 000册

定价: 19.00元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换  
本社购书热线: (010) 68326294

# 丛书序言

近年来,我国在计算机应用、计算机软件和电子类相关专业的人才培养方面,取得了长足的进展,每年的毕业生都有数十万人。但是这些毕业生走进企业、公司、政府机构或研究单位之后,往往深刻地感觉到缺乏实际开发设计项目的经验,不善于综合运用所学理论,对知识的把握缺乏融会贯通的能力。

综合考察目前高等院校教学大纲、课程设置以及内容安排等方面的情况,多数学校还是比较重视训练学生的实际设计能力。但是,从安排设计实践的内容上看,基本上是围绕相关课程教学内容而展开的,不能够构成对实际问题的解决方案;从配套程序的规模上看,一般只是几十行到几百行的源代码,或者是一个单独电路的设计,远远小于一个小型项目的规模;从设计的结构上看,由于设计实践是围绕着课程教学内容而进行的,问题已经高度抽象,学生很难得到有关综合运用所学知识的整体训练机会。而且,这些内容相对简单、问题域已经高度抽象、规模较小的设计实践一人基本上就能完成,学生几乎无法通过这些设计实践,去真正获得有关项目管理和团队协作等方面的基本训练和工作经验。

由此可以看出,大多数学校对学生实际设计能力的训练与国外知名大学和国内精品课程相比较,还是存在一些差距的。为此,机械工业出版社华章分社和一批高等院校的教师,针对当前高等院校计算机硬件、软件和电子类相关课程教学中存在的问题,参考国内外知名大学相关课程成功的教学经验,设计编写了这套“高等院校计算机课程设计指导丛书”,其目的就是通过课程设计的一系列训练,把知识获取和项目实践两个方面有机地结合起来。

在这套“高等院校计算机课程设计指导丛书”中的每一门课程设计里,都安排了由多个子项目组成的一个课程设计项目。学生们可以在教师的指导下,逐步设计实现这些子项目,并最终完成一个功能相对完整,可以运行的系统,其代码可以是数千行,甚至上万行。通过这种设计课程,学生一方面可以结合课程的教学内容循序渐进地进行设计方面的实践训练,另一方面,在参与一系列子项目的实践过程中,还能提高如何综合运用所学知识解决实际问题的能力,以及获得有关项目管理和团队合作等等众多方面的具体经验,增强对相关课程具体内容的理解和掌握能力,培养对整体课程知识综合运用和融会贯通能力。

参加丛书编写的各高等院校的教师都有着丰富的教学、科研,以及与企业合作开发项目等多方面的经验。每个课程设计中的子项目和整体项目,都来自教师们具体的科研和设计开发实践,所选设计项目与教学内容配合紧密,项目的难度与规模适宜。

最后,感谢机械工业出版社华章分社编辑们的大力支持,使出版有关这套丛书的计划,从单纯的构想演化成带有油墨芳香的真实。

丛书写作组

2004年7月

# 前 言

UML语言产生于20世纪90年代，UML获得了工业界、科技界和应用界的广泛支持，成为面向对象和可视化建模语言事实上的工业标准。它代表了面向对象的软件开发技术的发展方向，具有巨大的应用前景。高等院校的软件工程相关专业纷纷开设了基于UML语言的软件建模课程。

本书是为了配合UML语言的建模课程教学而编写的，目的是通过课程设计的综合训练，培养学生实际分析问题、编程和动手能力，最终目标是想通过这种形式，帮助学生系统地掌握该门课程的主要内容，更好地完成教学任务。

本书从用户的需求开始，结合相关概念和设计任务，详细地描述了一个选课系统的设计例子，并逐步展开，层层深入，介绍了使用Rose实现设计的全过程。

本书是作者在多年从事面向对象的教学和实践工作的基础上，参阅国内外多种最新教材和资料，根据课程设计的特点编写而成的。其特点如下：

1. 它独立于具体的UML语言教科书，重点放在UML语言的基本特征上，以“不变”应“万变”，涵盖UML语言的重要基础知识。
2. 内容有所取舍。结合实际应用的要求，使课程设计既覆盖知识点，又接近工程实际需要。通过激发学习兴趣，调动学生主动学习的积极性，并引导他们根据实际项目要求，训练自己分析问题的能力和使用UML解决问题的能力，并养成良好的建模习惯。
3. 实用性强。全书没有介绍高深的理论，而是采用平实的语句，对常用的设计方法和流行的实现思想进行阐述，以很多人都熟悉的选课系统为例，循序渐进地介绍了使用UML和Rose进行设计的全过程，最终将组成一个完整的解决方案。
4. 这些课程设计的思想和方法还可以作为学生做毕业论文时的参考资料。

全书共分12章。第1章介绍本书课程设计的目标，从第2章开始，每章首先介绍基本概念，然后介绍设计思想和结果，最后介绍使用Rose完成设计任务。完整的课程设计实例的源代码可从华章网站：[www.hzbook.com](http://www.hzbook.com)上下载。

在本书的编写过程中，承蒙中国科学技术大学教务处和计算机类教材编委会的指导，计算机类教材编委会主任陈国良院士和计算机系陈意云教授的支持，特此表示感谢。

参加本书编写的还有叶静和郜俊涛。

由于水平所限，书中一定存在不当之处，恳请广大读者批评指正。

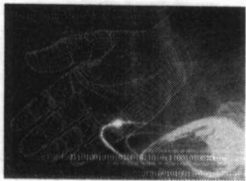
作者

2004年9月于中国科学技术大学

# 目 录

丛书序言	
前言	
第1章 课程设计目标	1
1.1 概述	1
1.2 本课程设计结构	2
1.3 评价标准	4
第2章 UML基本概念和Rose2003	5
2.1 引入可视化建模语言	5
2.2 可视化建模语言UML	6
2.2.1 用例视图	7
2.2.2 逻辑视图	10
2.2.3 组件视图	11
2.2.4 布局视图	11
2.3 Rose简介	11
2.3.1 使用Rose2003设计软件思路	12
2.3.2 使用Rose2003创建模型	12
2.3.3 Rose界面的组成	14
2.3.4 Rose中的基本操作	24
2.4 小结	25
第3章 用例图	27
3.1 用例图的基本概念	27
3.1.1 活动者	28
3.1.2 用例和用例图	28
3.1.3 事件流	30
3.2 用例图内元素间的关系	33
3.3 用例图设计实例	34
3.3.1 需求	34
3.3.2 分析	35
3.3.3 事件流	35
3.3.4 用例图	37
3.4 小结	42
3.5 评价标准	43
第4章 活动图	45
4.1 活动图的基本概念和组成	45
4.2 实例	47
4.2.1 分析	47
4.2.2 创建Add Course活动图	48
4.3 小结	59
4.4 评价标准	59
第5章 序列图和协作图	61
5.1 如何创建交互图	61
5.2 序列图的概念	62
5.3 协作图的概念	63
5.4 实例	65
5.4.1 分析	65
5.4.2 Select Course序列图的创建	67
5.4.3 Add Course序列图的创建	73
5.4.4 Modify Course序列图的创建	75
5.4.5 Delete Course序列图的创建	76
5.4.6 Query Course序列图的创建	77
5.4.7 Registrar Login序列图的创建	77
5.4.8 Select Course协作图的创建	78
5.5 小结	83
5.6 评价标准	83
第6章 类、属性和操作	85
6.1 类的基本概念	85
6.1.1 寻找类	85

6.1.2 类的属性和方法	86	9.1.2 状态图的实例	123
6.1.3 类和类的实例	86	9.2 状态图的创建	123
6.1.4 存取控制	88	9.3 小结	129
6.2 类的创建	89	9.4 评价标准	129
6.2.1 选课系统中的类	89	第10章 组件图	131
6.2.2 类图的创建	90	10.1 组件图的基本概念	131
6.3 小结	95	10.2 实例	131
6.4 评价标准	95	10.2.1 选课系统组件图	132
第7章 类的继承	97	10.2.2 组件图的创建	132
7.1 基本概念	97	10.3 小结	137
7.1.1 概括	97	10.4 评价标准	137
7.1.2 继承	98	第11章 布局图	139
7.1.3 多态性	98	11.1 布局图的概念	139
7.2 继承关系的实例	99	11.2 实例	139
7.3 小结	102	11.2.1 选课系统布局图	139
7.4 评价标准	102	11.2.2 布局图的创建	140
第8章 类的关联	103	11.3 小结	145
8.1 基本知识点	103	11.4 评价标准	145
8.1.1 关联	103	第12章 代码生成	147
8.1.2 连接	104	12.1 基本概念	147
8.1.3 聚合和组合	104	12.2 实例	148
8.2 类的关联的实例	105	12.2.1 模型的检查	148
8.2.1 选课系统的关联关系分析	105	12.2.2 创建组件以及把类映射到组件	149
8.2.2 选课系统中的类、属性和操作	107	12.2.3 设置代码生成属性	149
8.2.3 关联关系的创建	113	12.2.4 选择类、组件或包	150
8.3 小结	118	12.2.5 生成代码	150
8.4 评价标准	119	12.2.6 逆向工程	161
第9章 状态图	121	12.3 评价标准	164
9.1 基本概念	121	参考文献	165
9.1.1 状态图	121		



# 第1章

## 课程设计目标

---

本章简要介绍本课程的设计题目及其预期目标。

### 1.1 概述

一般来讲,课程设计比教学实验复杂一些,涉及的深度广些并更加接近实用。目的是通过课程设计的综合训练,培养学生实际分析问题、编程和动手能力,最终目标是想通过这种形式,帮助学生系统掌握该门课程的主要内容,更好地完成教学任务。

以对象为中心的软件设计和编程过程已经受到软件工程界的广泛关注,在面向对象的分析与设计中,需要采用一种统一的建模语言来记录分析和设计的成果。UML就是一种定义良好、易于表达、功能强大且普遍适用的建模语言,它产生于20世纪90年代,并获得了工业界、科技界和应用界的广泛支持,成为面向对象和可视化建模语言事实上的工业标准,代表了面向对象方法的软件开发技术的发展方向,具有巨大的应用前景。

本书是作者参考国内外知名大学相关课程成功的教学经验,结合自己多年从事面向对象的教学和实践工作的基础上编写而成。通过本课程的学习,希望能够达到如下目标:

- (1) 解释采用UML记录的内容。
- (2) 解释抽象、封装、模块化和层次化。
- (3) 定义类之间的关系。
- (4) 定义多态和继承关系。
- (5) 在模型设计中应用面向对象的概念来开发。
- (6) 使用统一建模语言表示设计的模型。
- (7) 可以使用Rose完成作业中的设计要求。
- (8) 使用Rose中的双向工程的功能。

另外,本课程设计具有如下特点:

(1) 它独立于任何UML语言教科书,重点放在UML语言的基本特征上,以“不变”应“万变”,涵盖UML语言的重要基础知识。

(2) 结合实际应用的要求,使课程设计既覆盖知识点,又接近工程实际需要。通过激发学习兴趣,调动学生主动学习的积极性,并引导他们根据实际项目要求,训练自己分析问题



的能力和使用UML解决问题的能力，并养成良好的建模习惯。

(3) 通过详细的实例，循序渐进地启发学生完成设计。课程设计为学生创造独立思考的条件，学生在吃透要求的前提下，完全可以设计自己的模型。

(4) 有些课程设计提出了一些改进措施或要求，可以供有兴趣的学生扩充自己的设计。

(5) 后续的课程设计尽量引用前面的课程设计内容，以便增加印象并加深理解，最终将组成一个完整的解决方案。

(6) 实用性强。全书没有介绍高深的理论，而是采用平实的语句，对常用的设计方法和流行的实现思想进行阐述，以很多人都熟悉的选课系统为例，循序渐进地介绍了使用UML和Rose进行设计的全过程。

(7) 这些课程设计的思想和方法还可以作为学生做毕业论文时的参考资料。

## 1.2 本课程设计结构

本课程设计共选择11个设计题目，不使用一个独立的例子涵盖所有知识点，而是按层次逐步深入。为了使学生理解它们之间如何相互配合，设计要求使用接近实际需要的方式建模。对每章的课程设计，先介绍知识点，其中重点解析一些难点，然后结合实际工程需要，演示方案的设计方法。这些设计题目及其简要说明如下。

### 1. UML基本概念和Rose2003(第2章)

面向对象的设计方法极大地促进了软件技术的发展，UML语言是可视化面向对象建模语言的标准。该章将介绍UML视图的组成。采用UML进行系统设计的结果和实现方案被记录在4种视图和8种图中，满足不同人员的使用要求。用例视图包括用例图、活动图、序列图和协作图，供最终用户、设计人员、开发人员和测试人员参考。逻辑视图包括类图和状态图，供设计人员和开发人员使用。组件视图供开发人员使用。布局视图供开发人员、集成人员和测试人员使用。除此之外，该章还介绍了使用Rose创建一个模型文件的过程。

### 2. 用例图(第3章)

用例视图能帮助理解和使用系统，这个视图表示活动者和用例之间的交互。该章将介绍选课系统用例图的设计和实现。这个系统描述了某学校的网上选课系统，主要包括管理员通过系统管理界面进入，建立本学期要开的各种课程，将课程信息保存在数据库中并可以对课程进行改动和删除。学生通过客户机浏览器根据学号和密码进入选课界面，这里学生可以进行3种操作：查询已选课程、选课以及付费。同样，通过业务层，这些操作结果存入数据库中。

### 3. 活动图(第4章)

在UML中，活动图用来表示事件的流程。该章介绍与活动图相关的概念。活动图显示与文本事件流相同的信息，是描述工作流的另一种方式，描述采取何种动作、做什么、何时发

生以及在何处发生。

活动图由起始状态、终止状态、状态转移、决策、守护条件、同步棒和泳道组成。本章最后详细介绍了活动图在Rose中的实现。

#### 4. 序列图和协作图(第5章)

序列图和协作图描述了用例的初始实现,在实际设计时,可以考虑只做成其中的一种图。用例的实现用UML中用交互图来指定和说明。交互图通过显示对象之间的关系和对象之间处理的消息来建模系统的动态特性。有两种交互图,它们是序列图和协作图。该章除了介绍序列图和协作图的基本概念和组成,还详细地介绍了这两种图在Rose中的实现。

#### 5. 类、属性和操作(第6章)

类能实现对象的封装,是面向对象的重要特征。该章介绍了类、对象、属性、操作的概念,可以从一类单一的对象中抽象出类的定义,并依据实际要求,设计类的属性和操作。另外,该章还详细介绍了在Rose中定义类、设计类的属性和操作的过程。

#### 6. 类的继承(第7章)

继承模拟了客观世界中存在的特殊和普遍之间的关系。该章介绍了面向对象的重要特征继承的概念。通过继承,可以创建事务的层次分类。当一个类是另外一个类的特殊情形时,可以考虑引入继承关系。父类的属性和操作,只有当存取控制是公有和保护型时,才能被继承。私有属性和操作不能被子类继承。该章还详细介绍了在Rose中建立继承关系的过程。

#### 7. 类的关联(第8章)

现实世界中的对象存在着这样或那样的关系,关联是其中的一种。该章介绍关联关系的概念和实现。关联关系描述了给定类的对象个体之间的语义连接,是两个类或多个类之间的一个关系。连接是关联的一个实例。聚合是表达主体部分关系的关联,组合是关联的更强的形式。聚合和组合是关联的特殊形式。该章最后给出了在Rose中详细定义关联关系的过程。

#### 8. 状态图(第9章)

对象从产生到结束,可以处于一系列不同的状态。状态影响对象的行为,当这些状态的数目有限时,就可以用状态图来建模对象的行为。该章介绍了状态图的概念和组成。状态图显示了单个类的生命周期。在画对象的状态图时,要考虑以下因素:对象有哪些有意义的状态,如何决定对象的可能状态,对象的状态图如何与其他模型之间进行映射。最后,详细地介绍在Rose中做成状态图的过程。

#### 9. 组件图(第10章)

该章介绍了组件图相关的概念。组件视图包含模型代码库、可执行文件、运行库和其他组件的信息。组件是代码的实际物理模块。组件图显示代码本身的逻辑结构,与系统的实现相关。该章的最后详细地介绍在Rose中组件图的实现。

#### 10. 布局图(第11章)

该章介绍了布局图的概念和实现。布局图表示系统的实际部署，与系统的逻辑结构不同，它描述系统在网络上的物理部署。布局图包括的要素有处理器、进程和设备。该章还详细介绍在Rose中布局图的创建。

### 11. 代码生成(第12章)

该章介绍了前向工程和逆向工程的概念。Rose具有的前向工程和逆向工程的能力有助于维护架构的完整性，表现在三个方面：探测和评价架构的改变，在改变的架构之间通信，在每一个迭代中合成代码和模型。该章还详细介绍了前向工程和逆向工程的过程，最后对全书内容进行了总结。

## 1.3 评价标准

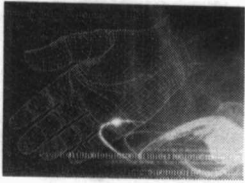
因为已经有参考建模方案，所以一般情况下都能完成预定设计。如果只是按照给定步骤去做，其分数只能在85分以下。为了证明学生已经掌握设计所涵盖的知识点，应该向学生提一些问题，例如如何设计及其原理等问题。本书对全部完成的定义，其实含有这种面试手段。由于各校情况不一，就没有在各章的评价中规定必须提问题才算完成设计任务。

另外，设计的可读性均包含在正确性中记分，不再单列考核标准。

一般遵循如下规律评价：

- (1) 严格控制90分，其标准是有创意。
- (2) 85分以上，必须全部正确，并有一定改进或者能正确回答设计中的问题。
- (3) 有少许失误，可给75~83分。
- (4) 错误不多，给60~73分。
- (5) 没有完成特殊规定的要求，则不予及格。

因为课程设计主要是锻炼学生，所以应该鼓励他们不要作为负担，提高钻研问题的兴趣，放手去做。鉴于以上目的，应该是鼓励他们均能通过，并避免不及格现象。另外，评分标准也可以只设“通过”、“没通过”和“优秀”3档以激发学生的学习兴趣。



## 第2章

# UML基本概念和Rose2003

---

UML(Unified Modeling Language)语言是一种支持对象技术的建模语言,是在计算机系统中表示真实世界的语言,描述真实世界中的对象和它们之间的关系,支持应用的开发。本章主要是介绍UML基本概念,课程设计将围绕这些基本概念展开。实现UML建模需要使用CASE工具,这里简要介绍Rose,以便为课程设计打下基础。

### 2.1 引入可视化建模语言

现实世界是复杂的,以致于用户经常不能精确地理解和描述一些将要开发的系统,建模可以使用户更好地理解需要开发的系统。

模型是真实世界的简化。可视化建模可使要建立的系统可视化,允许用模型表示系统的行为和结构,为建立系统提供一个指导的模板并记录相应的决策。建立的模型要实际反映真正要解决的问题,能够在不同层面上精确地反映系统,与现实的世界连接。通常情况下,单个的模型是不够的。

但是,在过去的软件开发过程中,许多软件团队没能够建立有效和足够的模型来反映现实问题,究其原因主要有如下方面:

- (1) 从项目的需求分析开始编码;
- (2) 编写了大量的结构不好的代码;
- (3) 缺乏有计划的体系结构。

这些问题最终导致了项目失败。建立有效的模型是避免软件项目开发失败的有效途径。

我们周围的世界是由对象(object)组成的,所以在对事物建模(modeling)时,必须先研究事物,再研究过程,必须先知道一个事物是什么,而后才能觉察这个事物中所发生的变化,这就是以对象为中心进行系统的分析、设计和实现的思想的核心。

对象技术(object technology)包括指导软件构造的抽象、封装和多态等原则的集合,以及支持这些原则的语言、数据库和其他工具。

对象技术推动了软件业的快速发展,被广泛地应用于C/S、B/S结构的系统以及实时系统中。使用对象有以下几个优点:

- (1) 反映了单个的范例;
- (2) 促进了架构和代码的重用性;

- (3) 建立的模型更贴近真实的世界;
- (4) 增强了系统的稳定性;
- (5) 更易于修改。

## 2.2 可视化建模语言UML

在计算机发展的早期,程序员对手边的工作经常不做深入的研究,就直接编写代码,从而导致了软件开发中的许多问题。在今天,计算机系统越来越复杂,经常包括很多部分的软件和硬件,通过网络互联,还会使用数据库保存大量的信息。一个软件系统的开发一般要经历如图2-1所示的多个步骤和以图2-1所示为基础的迭代,许多人员参加了软件系统分析、设计和实现的过程,UML提供了用来组织分析设计过程、客户和程序员的手段。

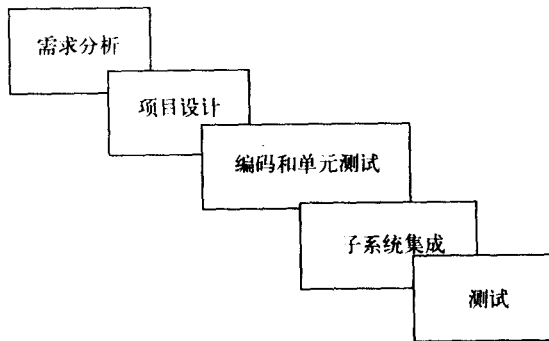


图2-1 软件开发过程

UML语言是可视化建模语言,能用它设计所需要的各种模型图及其相应文档,通过UML建立的模型应该是精确的和完全的。

在UML中,模型是通过视图(view)来描述系统的不同侧面,通过图(diagram)来描述待建立系统的模块。

如图2-2所示,UML由四种视图组成,分别是用例视图(use case view)、逻辑视图

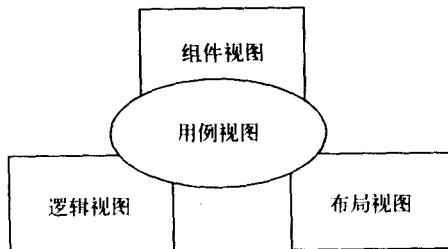


图2-2 UML视图

(logical view)、组件视图 (component view) 和布局视图(deployment view)。其中,用例视图是其他视图的“心脏”,描述了系统应该做什么,在集成其他三种视图的内容中扮演了重要的角色。

支持UML的CASE工具有Rational Rose、Together、MS Visual Modeler等。图2-3所示为Rose2003中的一个订货系统的视图。

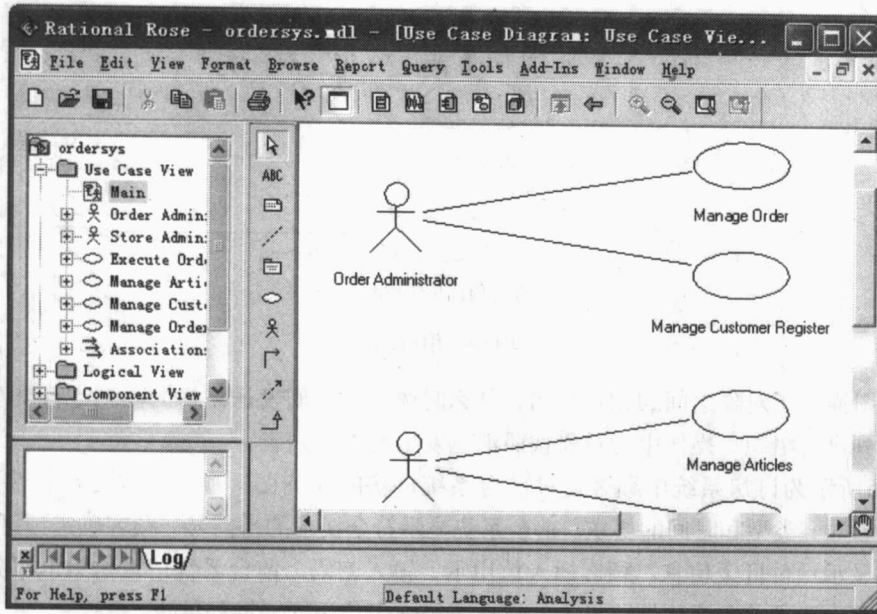


图2-3 Rose2003中的订货系统的例子

### 2.2.1 用例视图

用例视图定义了系统的外部行为,帮助用户理解和使用系统。它包括以下图:用例图 (use case diagram)、序列图 (sequence diagram)、协作图 (collaboration diagram) 和活动图 (activity diagram)。

用例图如图2-4所示,它描述了从外部来看系统在做什么,强调的是系统在做什么而不是怎样做。图2-4是酒店订房的用例图,与订房系统交互的人员和系统有顾客、前台服务员、信用卡服务商(系统)和客房监督员。系统的主要功能有订房间、接受预订、收取订金及客房预订和使用汇总。

序列图(见图2-5)和协作图(见图2-6)都属于交互图(interaction diagram),描述了系统对象之间相互协作的过程。

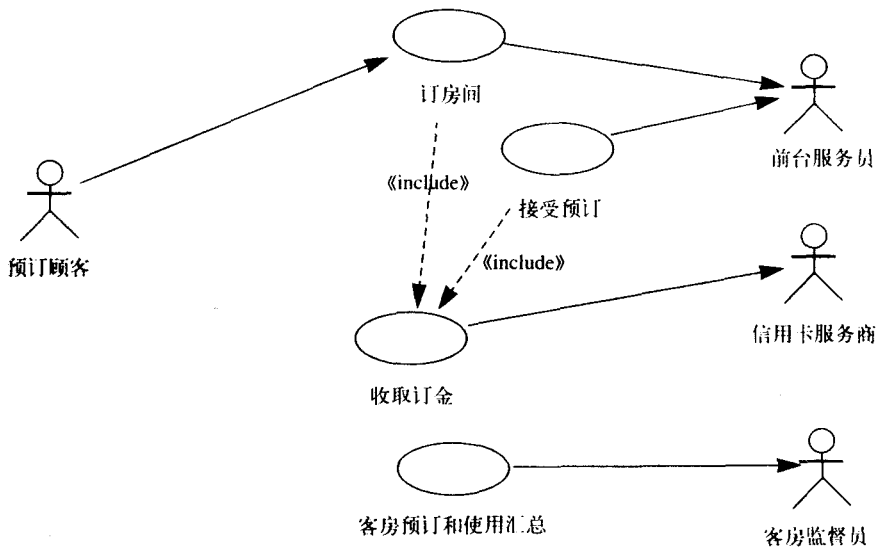


图2-4 用例图

序列图描述了对对象之间的相互作用，什么时候对象之间发送怎样的消息进行交互。它按照时间序列进行组织，操作中的对象按照消息发生的次序列表。

图2-5所示为订房系统中顾客通过前台系统订房的序列图。顾客首先提出订房的要求，输入房间套数、类型和时间的要求。前台系统显示符合要求的房间号。顾客确定预订的房间，前台系统显示预付订金信息。顾客插入信用卡，输入密码。前台系统请求信用卡服务。认证通过后，客户请求前台系统打印报告，然后系统弹出信用卡，过程结束。

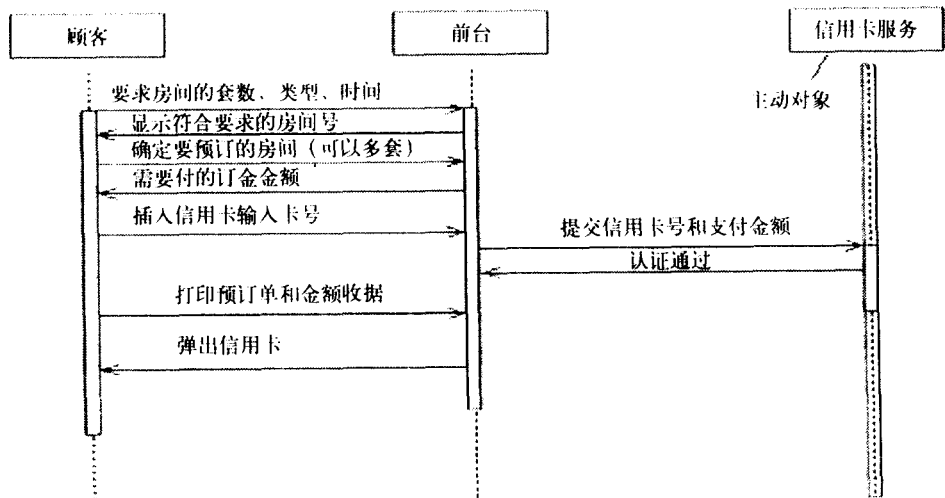


图2-5 序列图

协作图反映的信息和序列图相同，但是它侧重于对象的角色而不是消息发送的时序关系。图2-6所示为支票账户取款的协作图。客户通过柜台的界面向应用系统发出取款请求，界面向应用系统请求取款的服务，应用系统创建账户对象，账户对象从数据库中提取相关信息创建账户对象。应用系统向账户对象发送取款请求，账户对象修改数据库数据。取款过程完成，应用系统请求界面显示相关信息。

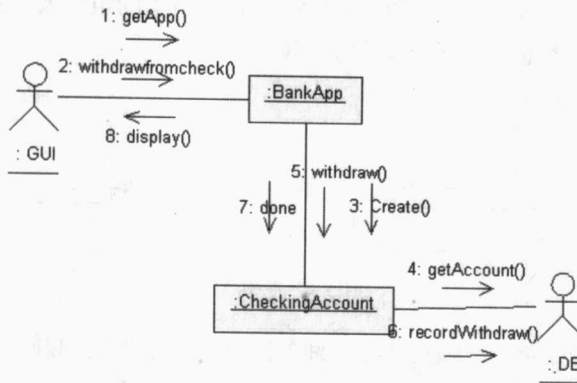


图2-6 协作图

活动图如图2-7所示，它本质上是一种特别的流程图，着重于描述单个过程中活动的流

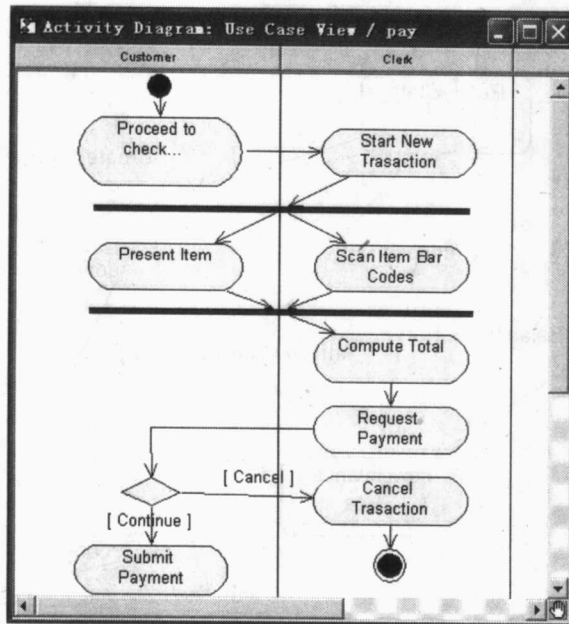


图2-7 活动图



程。活动图显示了这些活动之间如何相互依赖。图2-7所示为一个购物后付款的活动图，反映了付款的流程。

### 2.2.2 逻辑视图

逻辑视图 (logical view) 描述了支持用例图功能的逻辑结构。包括类图 (class diagram) 和状态图 (state diagram)。类图 (见图2-8) 描述了系统中类的组成及它们之间的关系，方框内表示的是类，类与类之间的连线表示类的关联。

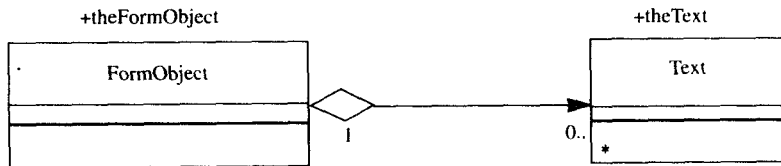


图2-8 类图

状态图显示对象的可能状态以及状态之间的迁移，它回答的问题是对象在某时刻处于什么状态。图2-9显示的是银行账户的状态迁移。在一次交易过程中，账户除了具有初始状态

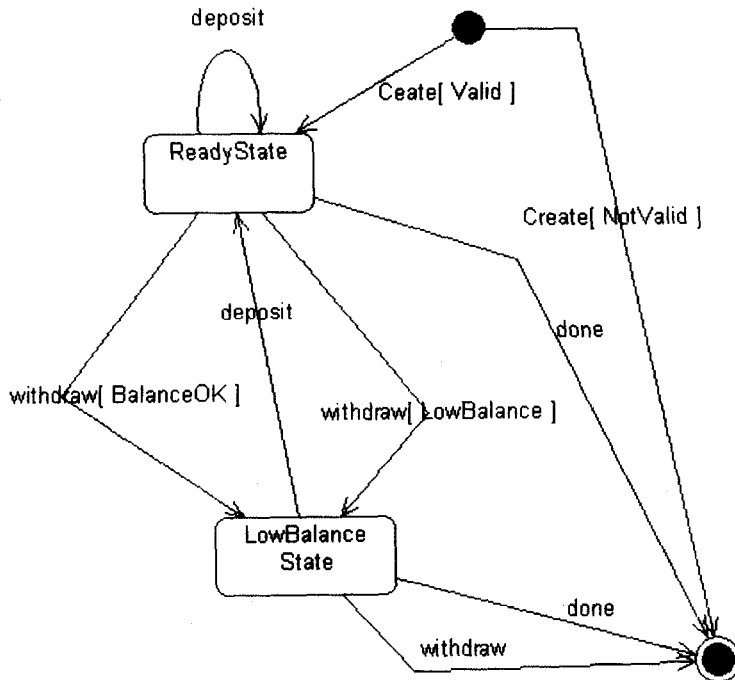


图2-9 状态图