

国家社会科学基金(教育学学科)
“十一五”规划课题研究成果

全国高等职业院校计算机教育规划教材

软件工程与UML案例解析

SOFTWARE ENGINEERING AND UML CASE ANALYSIS

何晓蓉 主编 李洛 主审

SOFTWARE ENGINEERING AND

UML CASE ANALYSIS

SOFTWARE ENGINEERING AND

UML CASE ANALYSIS

ENGINEERING AND

ANALYSIS OF SOFTWARE ENGINEERING



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

国家社会科学基金（教育学科）“十一五”规划课题研究成果
全国高等职业院校计算机教育规划教材

软件工程与 UML 案例解析

	何晓蓉	主 编
车 书	罗 佳	副主编
张 婵	陈建潮	
	李 洛	主 审

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

为了满足广大读者对软件工程应用技术的学习需求,特别是为了提高高职高专软件技术专业学生的 UML 建模能力,本书在介绍软件开发各阶段所涉及的基本理论的基础上,以实际的开发项目重点介绍了用例模型、分析模型、设计模型等 UML 全程建模过程,并介绍了应用项目管理工具 Microsoft Project 对开发项目进行时间、资源和成本计划的编制与监控管理,使用 PowerDesigner 建模工具构建数据模型,以及使用 LoadRunner 等测试工具对软件进行自动或手工测试的过程。

本书注重理论与实际相结合,内容选取适当,按“问题引入—解答问题—分析问题”的方式设计情景,符合人们的认知过程。全书结构严谨、布局合理、重点突出,具有很强的实用性。

本书适合作为高职高专院校软件技术专业“软件工程”课程的教材或教学参考书,也可作为软件开发人员的自学用书。

图书在版编目(CIP)数据

软件工程与 UML 案例解析 / 何晓蓉主编. —北京:
中国铁道出版社, 2009. 12
全国高等职业院校计算机教育规划教材
ISBN 978-7-113-10880-9

I. ①软… II. ①何… III. ①软件工程—高等学校:
技术学校—教材②面向对象语言—程序设计—高等学校:
技术学校—教材 IV. ①TP311.5②TP312

中国版本图书馆 CIP 数据核字(2009)第 229407 号

书 名: 软件工程与 UML 案例解析
作 者: 何晓蓉 主编

策划编辑: 翟玉峰 王春霞
责任编辑: 翟玉峰
特邀助理: 李红玉
封面设计: 付 巍
版式设计: 郑少云

编辑部电话: (010) 63560056

封面制作: 李 路
责任印制: 李 佳

出版发行: 中国铁道出版社(北京市宣武区右安门西街 8 号 邮政编码: 100054)
印 刷: 北京海淀五色花印刷厂
版 次: 2010 年 4 月第 1 版 2010 年 4 月第 1 次印刷
开 本: 787mm×1092mm 1/16 印张: 17 字数: 409 千
印 数: 4 000 册
书 号: ISBN 978-7-113-10880-9/TP·3704
定 价: 29.00 元

版权所有 侵权必究

本书封面贴有中国铁道出版社激光防伪标签,无标签者不得销售

凡购买铁道版图书,如有印制质量问题,请与本社计算机图书批销部联系调换

国家社会科学基金(教育学科)“十一五”规划课题研究成果 全国高等职业院校计算机教育规划教材

编 审 委 员 会

主 任: 邓泽民

副主任: (按姓氏笔画排序)

吕一中	严晓舟	李 雪	汪燮华	张洪星
张晓云	武马群	赵凤芝	段银田	宣仲良
姚卿达	聂承启	徐 红	彭 勇	蒋川群

委 员: (按姓氏笔画排序)

王浩轩	邓安远	邓璐娟	白延丽	包 锋
朱 立	任益夫	刘志成	刘晓川	孙街亭
延 静	李 洪	李 洛	李 新	李学相
李洪燕	杨立峰	杨永娟	杨志茹	杨俊清
连卫民	吴晓葵	沈大林	宋海军	张 伦
张世正	张晓蕾	张新成	欧阳广	周国征
赵传慧	赵轶群	段智毅	贺 平	秦绪好
袁春雨	徐人凤	徐布克	黄丽民	梅创社
崔永红	梁国浚	蒋腾旭	蔡泽光	翟玉峰

序

PREFACE

国家社会科学基金(教育学科)“十一五”规划课题“以就业为导向的职业教育教学理论与实践研究”(课题批准号BJA060049)在取得理论研究成果的基础上,选取了高等职业教育十个专业类开展实践研究,高职高专计算机类专业就是其中之一。

本课题研究发现,高等职业教育在专业教育上担负着帮助学生构建专业理论知识体系、专业技术框架体系和职业活动逻辑体系的任务,而这三个体系的构建需要通过专业教材体系和专业教材内部结构得以实现,即学生的心理结构来自于教材的体系和结构。为此,这套高职高专计算机类专业系列教材,依据不同教材在其构建知识、技术、活动三个体系中的作用,采用了不同的教材内部结构设计和编写体例。

承担专业理论知识体系构建任务的教材,强调专业理论知识体系的完整性与系统性,不强调专业理论知识的深度和难度;追求的是学生对专业理论知识整体框架的把握,不追求学生只掌握某些局部内容,而求其深度和难度。

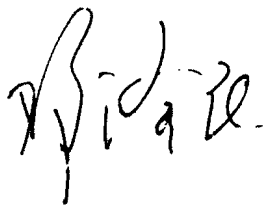
承担专业技术框架体系构建任务的教材,注重让学生了解这种技术的产生与演变过程,培养学生的技术创新意识;注重让学生把握这种技术的整体框架,培养学生对新技术的学习能力;注重让学生在技术应用过程中掌握这种技术的操作,培养学生的技术应用能力;注重让学生区别同种用途的其他技术的特点,培养学生职业活动过程中的技术比较与选择能力。

承担职业活动体系构建任务的教材,依据不同职业活动对所从事人特质的要求,分别采用了过程驱动、情景驱动、效果驱动的方式,形成了“做学”合一的教材结构与体例,诸如项目结构、案例结构等。过程驱动培养所从事人的程序逻辑思维;情景驱动培养所从事人的情景敏感特质;效果驱动培养所从事人的发散思维。

本套教材无论从课程标准的开发、教材体系的建立、教材内容的筛选、教材结构的设计还是教材素材的选择,都得到了信息技术产业专家的大力支持,他们在信息技术行业职业资格标准和各类技术在我国广泛应用的过程中,提出了十分有益的建议。另外,国内知名职业教育专家和一百多所高职高专院校参与了本课题的研究,他们对高职高专信息技术类的人才培养提出了宝贵意见,对高职高专计算机类专业教学提供了丰富的素材和鲜活的教学经验。

这套教材是我国高职高专教育近年来从只注重学生单一职业活动逻辑体系构建,向专业理论知识体系、技术框架体系和职业活动逻辑体系三个体系构建的转变的有益尝试,也是国家社会科学基金课题“以就业为导向的职业教育教学理论与实践研究”研究成果的具体应用之一。

如本套教材有不足之处,敬请各位专家、老师和广大同学不吝赐教。希望通过本套教材的出版,为我国高等职业教育和信息技术产业的发展作出贡献。



2009年8月

前言

FOREWORD

随着计算机科学与网络技术的不断发展,计算机的应用范围越来越广,软件的规模及社会需求量在迅猛增大,软件工程化方法的重要性也日益突出,这从客观上迫切需要众多既具有软件工程理论知识,又掌握软件工程实际应用技能的人才。特别是对于高职高专院校的软件技术专业学生,更需要一本注重软件工程实际应用技术的教材。本书正是在“理论够用、实战为本”的思想指导下,特为高职高专院校软件技术专业学生学习和掌握软件工程应用技术而编写的。

为培养软件开发实用型专门人才,本书在介绍软件工程理论知识的基础上,以实际的软件项目“客户服务系统”的建模过程为主线,按照软件开发的实际工作过程及流程,以任务驱动的方式,重点阐述面向对象的软件开发技术以及 UML 全程建模。注重对学生实际应用技能和动手能力的培养。书中内容按“问题引入—解答问题—分析问题”的方式设计情境,打破了传统的“提出问题—分析问题—解决问题”的问题解决模式,更符合人们的认知过程。

本书正文共分 9 章,主要内容包括:第 1 章概述,主要介绍软件工程及软件工程的目标、过程和原则、UML 等方面的问题和基本概念,并对作为全书案例的软件项目作了简要描述;第 2 章软件项目管理,主要介绍项目管理的概念、组成部分以及工期的计算公式、WBS 等基本知识,项目的时间、资源和成本计划的编制与监控管理,以及用 Microsoft Project 项目管理软件来管理与跟踪这些计划;第 3 章软件系统开发方法,主要介绍软件生命周期及传统的生命周期模型、传统软件开发方法与面向对象软件开发技术、RUP 统一软件开发过程、敏捷软件开发技术等方面的问题及基本概念;第 4 章建立用例模型,主要介绍建立用例模型的过程及相关知识,包括需求获取、分析需求、用例在需求分析中的使用、识别参与者、确定用例、用例的粒度、用例间的关系、用例描述和用例建模等方面的内容;第 5 章建立分析模型,主要介绍建立分析模型的过程及相关知识,包括对象、类和对象、类的 UML 表示,确定关键抽象,类之间的关系及其 UML 表示,建立领域模型,分布模式的选择与应用,构建分析类,职责分配,定义类属性,分析模型等方面的知识;第 6 章数据库建模,主要介绍建立数据库模型的过程及相关知识,包括从业务需求创建数据模型的流程,定义数据需求,定义概念模型,设计逻辑数据模型,设计物理数据模型,数据模型的优化与发布等方面的内容;第 7 章建立设计模型,主要介绍建立系统设计模型的过程及相关知识,包括设计模式的选择与应用,定义设计类,设计类,设计类间的关系,活动图,状态图,设计模型顺序图,设计模型的分包,逻辑视图到构件视图的映射等方面的内容;第 8 章系统架构设计,主要介绍系统架构的设计过程及相关知识,包括业务架构及业务架构的分析,软件架构及软件架构的设计,软件架构与框架的区别,软件架构的“4+1”视图模型,组件图,部署图等方面的内容;第 9 章软件测试,主要介绍软件测试的基本概念、测试方法和测试过程等方面的知识。

本书注重理论与实际应用相结合,具有很强的实用性。

本书由何晓蓉主编,参加编写的有何晓蓉(第 1 章、第 3 章、第 5 章的第 9 节,第 7 章、第 8 章的第 6 节、附录 A、附录 C),张婵和罗佳(第 4 章、第 5 章的第 1~8 节及第 10 节、第 6 章、

附录 B)，车书（第 2 章、第 8 章的第 1~5 节及第 7~8 节），陈建潮（第 9 章）。李洛教授担任本书的主审。本书由何晓蓉拟定大纲，并进行统稿和定稿。在编写过程中，与软件企业通力合作，得到了软件企业开发人员余颖的大力支持与帮助，给我们提供了真实的软件开发案例，在此表示衷心的感谢！

由于编者水平有限，书中难免会有不足之处，敬请广大读者不吝赐教。电子邮箱：xrhe@163.com。

编者

2010 年 2 月

第 1 章 概述	1
1.1 软件工程	1
1.1.1 软件工程的定义	1
1.1.2 软件工程的目标	2
1.1.3 软件工程过程	3
1.1.4 软件工程的原则	3
1.2 UML 简介	4
1.3 案例描述	6
总结	7
思考与练习	8
第 2 章 软件项目管理	9
2.1 项目管理概述	9
2.1.1 项目管理定义	9
2.1.2 项目管理的组成部分	10
2.1.3 工期的计算公式	11
2.1.4 WBS	11
2.2 Microsoft Project 项目管理软件的使用	12
2.2.1 编制项目计划	12
2.2.2 编制项目的时间计划	12
2.2.3 编制项目的资源计划	21
2.2.4 编制项目的成本计划	24
2.2.5 项目监控管理	26
总结	32
思考与练习	33
第 3 章 软件系统开发方法	36
3.1 软件开发生命周期	36
3.2 软件开发模型	37
3.2.1 瀑布模型	38
3.2.2 演化模型	39
3.2.3 螺旋模型	41
3.2.4 增量模型	42

3.3	传统软件开发方法	44
3.4	面向对象软件开发技术	45
3.4.1	面向对象的基本概念	45
3.4.2	面向对象的开发	50
3.5	RUP 统一软件开发过程	51
3.5.1	RUP 的生命周期	51
3.5.2	RUP 的核心工作流程	54
3.6	敏捷软件开发技术	56
3.6.1	敏捷开发技术基本概念	57
3.6.2	极限编程	59
	总结	64
	思考与练习	64
第 4 章	建立用例模型	65
4.1	需求获取	65
4.2	分析需求	67
4.3	用例在需求分析中的使用	68
4.4	识别参与者	69
4.5	确定用例	70
4.6	用例的粒度	71
4.7	用例间的关系	72
4.8	用例描述	74
4.9	用例建模	78
	总结	81
	思考与练习	81
第 5 章	建立分析模型	82
5.1	对象、类和对象、类的 UML 表示	82
5.2	确定关键抽象	83
5.3	类之间的关系及其 UML 表示	85
5.3.1	关联	85
5.3.2	依赖	88
5.3.3	泛化	89
5.3.4	聚合	90
5.3.5	组合	91
5.4	建立领域模型	92
5.5	分布模式的选择与应用	93
5.6	构建分析类	94
5.7	交互图	97
5.7.1	顺序图	97

5.7.2 协作图	99
5.8 职责分配	100
5.9 定义类的属性	102
5.10 分析模型	103
总结	104
思考与练习	104
第 6 章 数据库建模	106
6.1 从业务需求创建数据模型的流程	106
6.2 定义数据需求	107
6.3 定义概念模型	109
6.4 设计逻辑数据模型	110
6.5 设计物理数据模型	112
6.6 数据模型的优化与发布	114
总结	115
思考与练习	115
第 7 章 建立设计模型	117
7.1 设计模式的选择与应用	117
7.1.1 Facade (门面) 模式	119
7.1.2 Adapter (适配器) 模式	120
7.1.3 Factory (工厂) 模式	123
7.2 构建设计类	129
7.2.1 从分析类生成设计类	129
7.2.2 确定类的大小	131
7.3 详细设计类	132
7.3.1 设计公用类	132
7.3.2 设计类接口	132
7.3.3 设计属性和操作	133
7.4 设计类间关系	140
7.4.1 设计继承	140
7.4.2 设计聚合/组合	141
7.4.3 设计关联	142
7.5 活动图	143
7.6 状态图	146
7.7 设计模型顺序图	148
7.8 设计模型的分包	149
7.9 逻辑视图到构件视图的映射	152
总结	160
思考与练习	161

第 8 章 系统架构设计	162
8.1 业务架构	162
8.2 业务架构分析	163
8.2.1 客户服务系统业务架构分析	164
8.2.2 客户服务系统子模块划分	165
8.3 软件架构	168
8.4 软件架构设计	169
8.5 软件架构与框架	174
8.6 软件架构的“4+1”视图模型	175
8.7 组件图	176
8.8 部署图	178
总结	180
思考与练习	180
第 9 章 软件测试	181
9.1 基本概念	181
9.1.1 软件测试的必要性	181
9.1.2 软件测试的目的	183
9.1.3 软件缺陷定义	183
9.1.4 测试用例	184
9.1.5 软件测试的原则	185
9.2 测试方法	189
9.2.1 白盒测试	189
9.2.2 黑盒测试	200
9.3 测试过程	210
9.3.1 软件测试过程模型介绍	210
9.3.2 单元测试	213
9.3.3 集成测试	221
9.3.4 确认测试	224
9.3.5 系统测试	227
9.3.6 验收测试	229
总结	231
思考与练习	232
附录 A 用户界面设计	234
A.1 UI 设计师的工作	234
A.2 用户界面的设计原则	235
A.3 用户界面的设计流程	236
A.4 客户服务系统界面设计	237

附录 B	Java 语言的编码规范	239
B.1	导言	239
B.2	格式	239
B.3	命名规则	242
B.4	变量	243
B.5	注释	243
B.6	编码	246
附录 C	软件工程文档	248
C.1	用户需求说明书模板	248
C.2	软件需求规格说明书模板	249
C.3	概要设计说明书模板	251
C.4	软件设计说明书模板	254
参考文献		256

第1章

概 述

在近代科学技术发展的历史中,工程学科的发展一直是产业发展的极大动力。传统的工程学科,如建筑工程、机械工程、水利工程、电力工程等对各行各业的发展都有非常深远的影响。近年来,人们对环境工程、生物工程、软件工程等给予了极大的关注,许多高等院校都增设了这些专业,以培养研究这些新兴学科的专门人才。然而,由于人们对这些新兴工程学科认识不足,还处在一个艰难的探索阶段,如软件工程。国内许多软件企业对软件工程还不够重视,事实上,软件工程在计算机的发展和应用中的地位非常显著,它对软件产业的形成和发展起着决定性的推动作用,是现代信息产业的巨大支柱。

本章将对软件工程的目标、过程和原则,以及 UML 等方面的问题和基本概念进行简要介绍,以便读者对软件工程的一些基本知识有一定的理解。书中各个部分以一个项目为主线,按任务驱动的方式进行衔接。本章还给出了贯穿全书的案例项目描述,以便读者在进入软件工程专题之前,对本书所要解析的软件工程项目有一个全面了解。

1.1 软 件 工 程

1.1.1 软件工程的定义

问题引入

我们知道,计算机软件是指计算机程序、数据以及解释和指导使用程序和数据的文档的总和。当计算机软件上升到工程学的高度后,又如何来定义它,即什么是软件工程呢?

解答问题

软件工程 (Software Engineering, SE) 是一门研究应用工程化方法构建和维护有效的、实用的和高质量的软件的学科。

分析问题

工程不仅仅是一个学科或一个知识体系,它还是解决问题的方法。这里的方法包括管理、过程和技术三个方面,其中,“过程”是指软件的开发、维护过程以及管理过程。采用工程的概念、原理、技术和方法来开发与维护软件,把经过时间考验而证明正确的管理技术和当前能够得到的最好的技术方法相结合,这就是软件工程。它涉及程序设计语言、数据库、软件开发工具、系统平台、标准、设计模式等方面的内容。

1.1.2 软件工程的目标

问题引入

软件工程的主要目标是采用工程化方法，提高软件产品质量和软件生产率，降低软件开发成本，成功地构建一个满足用户需求的软件系统。那么，一个成功的软件项目需要达到哪些主要目标呢？

解答问题

一个成功的软件工程项目需要达到的主要目标如下：

- ① 达到要求的软件功能。
- ② 取得较好的软件性能。
- ③ 付出较低的开发成本。
- ④ 开发的软件易于移植。
- ⑤ 开发的软件易于维护，需要较低的维护费用。
- ⑥ 能按时完成开发任务，并交付使用。

分析问题

软件的功能是指在一般条件下软件系统能够为用户“做什么”，能够满足用户什么样的需求。用户的需要就是软件开发人员的目标，开发的软件必须实现用户要求的功能；另一方面，一个软件项目“做得怎样”，比如开发的软件的运行速度、易用性、可靠性、适应性等，是否达到了用户的需求。这些都是一个软件工程项目最基本的目标，是必须努力实现。

软件的研制工作需要投入大量复杂的、高强度的脑力劳动，它的成本往往比较高。但随着软件技术的飞速发展，软件开发工具、开发方法等不断出现，应用先进的软件过程管理手段，使得降低开发成本成为可能。所以，不断探索新方法、新技术，努力减少软件开发成本，是软件工程项目所追求的主要目标。

软件的开发和运行常常受到计算机系统的限制，对计算机系统有着不同程度的依赖性。软件不能完全摆脱硬件单独存在。为了提高软件的可移植性，在软件开发过程中要尽量使用不依赖于计算机硬件和操作系统的计算机语言和方法编写程序，这也是软件工程项目所追求的目标。

所谓软件维护就是在软件交付使用之后，为了改正错误或者满足新的需要而修改软件的过程。其目的是保证系统能持续地与用户环境、数据处理操作、政府或其他有关部门的请求取得协调。在现代软件开发过程中，不重视文字资料工作，使分析、设计、实现和支持过程的资料很不完整；兼之，人们常常忽视人与人之间的沟通，发现问题只知道修修补补，这样的软件很难维护。维护费用的支出是不可避免的，但怎样才能降低维护费用呢？主要有三点：

- ① 设计软件时，充分考虑软件的可修改性、可扩展性。
- ② 软件开发文档齐备。
- ③ 加强团队合作精神。

提高软件的易维护性、降低其维护成本也是软件工程项目所追求的最基本的目标。

软件项目极大地复杂性与用户需求高度的不确定性,是软件项目按时完成的困难所在。加之,软件开发人员对项目往往按照最乐观的情况估计,对于任务的复杂性和难度,对于自己能支配的时间,对于可能的突发事件的干扰等没有清楚的认识和估计,在软件企业内部项目管理混乱的情况之下,软件项目常常严重超期或超出预算。提高软件项目管理能力,按时完成软件开发任务,并交付使用,是许多软件企业力争实现的重要目标。

在实际的软件开发项目中,要同时实现所有这些目标往往是比较困难的,甚至有些目标之间会相互冲突。比如,若一味降低开发成本,势必会降低软件的性能、减少软件的功能。因此,往往需要在这几大目标之间进行一些取舍。

1.1.3 软件工程过程

问题引入

在软件工程的定义中,我们强调了过程的概念,例如开发过程、维护过程、管理过程。过程是一组将输入转化为输出的相互关联或相互作用的活动。然而,什么是软件工程过程?

解答问题

软件工程过程是指软件生命周期(软件生命周期的相关知识将在第3章的3.1节详述)所涉及的一系列相关过程,是生产一个最终能满足需求且达到工程目标的软件产品所需要的步骤。

分析问题

软件工程过程主要包括开发过程、运作过程和维护过程,它们覆盖了需求、设计、编码、测试以及支持等软件工程活动。在软件工程活动中,分析活动包括问题分析和需求分析。问题分析获取需求定义,又称软件需求规约。需求分析生成功能规约。设计活动一般包括概要设计和详细设计。概要设计建立整个软件系统结构,包括子系统、模块以及相关层次的说明、每一模块的接口定义。详细设计产生程序员可用的模块或者类说明。编码活动把设计结果转换为可执行的程序代码。测试活动贯穿于整个软件开发过程,实现完成后的确认,保证最终产品满足用户的要求。维护活动包括使用过程中的扩展、修改与完善。伴随以上这些过程,还包括管理过程、支持过程和培训过程等。

管理过程、支持过程、培训过程贯穿软件开发过程的始终,对软件开发过程起着至关重要的作用。

1.1.4 软件工程的原则

问题引入

为了达到软件工程的目标,在软件开发过程中针对项目设计、支持以及管理必须遵循哪些基本原则?

解答问题

在软件开发过程中针对项目设计、支持以及管理必须遵循的基本原则是:

- ① 选取适宜的软件开发模型。
- ② 采用合适的软件开发方法。
- ③ 提供高效的开发支撑环境。
- ④ 重视软件开发过程的管理。
- ⑤ 建设高素质的软件开发团队。

分析问题

正如任何事物一样,软件也有其孕育、诞生、成长、成熟以及衰亡的生命过程,一般称其为“软件生命周期”。软件生命周期一般分为6个阶段,即制定计划、需求分析、设计、编码、测试、运行和维护。软件开发的各个阶段包含了一系列的活动,活动之间的关系可以是顺序的、重复的、并行的、嵌套的或者是由条件引发的。在软件工程中,这个复杂的过程用软件开发模型来描述和表示。在软件生命周期中,软件开发的各个阶段相互关联,用户需求的频繁变更会对软件的开发过程产生重大影响。因此,必须认识需求定义的易变性,采用适宜的软件开发模型予以控制,以保证软件产品满足用户的要求。

在软件开发中,通常要考虑软件的模块化、抽象与信息隐藏、可移植性、局部化,以及可适应性等方面的问题。合适的软件开发方法,如面向对象软件开发方法等,有助于这些方面的实现,以达到软件工程的目标。

在软件工程中,软件开发工具与开发环境对软件过程的支持尤为重要。软件工程项目的质量与开发成本直接取决于对软件工程所提供的支撑环境。

软件工程的管理,直接影响到可用资源的有效利用、满足目标的软件产品的生产、软件组织的生产能力的提高等问题。因此,只有在软件过程得以有效管理时,才能实现有效的软件工程。

正如 Tom DeMacro 和 Timothy Lister 在《人件》中所说的“人与人之间的交互是复杂的,并且其效果从来都难以预料,但却是工作中最为重要的方面”。过程和方法对于项目的影

响只是次要的,而首要的影响是人。因此,只有当开发人员发挥其效力时,才能达到软件工程的目标。软件工程的目的是生产满足用户需求的高质量、高生产率、低开销的软件产品;实施一个软件工程要选取适宜的开发模型,采用合适的开发方法,提供高效的支撑环境,实行开发过程的有效管理,建设能相互沟通、极具社会责任感的软件开发团队。软件工程主要包括需求、设计、编码、测试和支持等活动,每一活动可根据特定的软件工程项目,采用合适的开发模型、方法、支持过程以及过程管理。

1.2 UML 简介

UML (Unified Modeling Language, 统一建模语言) 是一种定义良好、易于表达、功能强大的用于对软件密集型系统建模的图形语言。它支持从需求分析开始的面向对象软件开发的全过程。

UML 作为一种建模语言,使软件开发人员专注于建立系统的模型和结构,而不是选用具体的程序设计语言和算法来实现。当模型建立之后,模型可以被 UML 工具转化成指定的程序设计语言代码和数据库结构。

UML 1.4 有如下九种图：

- ① 用例图：用于业务建模、需求捕获，作为测试的依据。
- ② 类图：描述类以及类之间的相互关系。
- ③ 对象图：描述对象以及对象之间的相互关系。
- ④ 构件图：描述构件及其相互依赖关系。
- ⑤ 部署图：描述构件在各个结点上的部署情况。
- ⑥ 顺序图：强调时间顺序的交互图。
- ⑦ 协作图：强调对象协作的交互图。
- ⑧ 状态图：描述类所经历的各种状态以及状态之间的转换关系。
- ⑨ 活动图：用于对工作流程建模。

对于一般系统，常常使用类图来产生程序代码。而对于嵌入式系统，则用状态图生成程序代码。

本书阐述了使用 UML 全程建立系统模型的过程，并指导读者应用建模工具 Rational Rose 针对案例项目建立模型。

Rational Rose 是一种支持 UML 1.4 的便于进行面向对象分析和设计的可视化的建模工具。它提供了一个集成化的建模环境，可以用来创建、查看和修改 UML 模型、视图、图和模型元素。

Rational Rose 使用图形用户界面，包括如下元素：

- 菜单栏；
- 标准工具栏；
- 图形工具栏；
- 浏览器窗口；
- 图形窗口；
- 文档窗口。

图 1-1 即为 Rational Rose 的图形用户界面。浏览器窗口用于查看、处理和切换模型中的项，浏览器中的项按层次结构排列，其操作与 Windows 系统的资源管理器类似。在 Rational Rose 的浏览器窗口中包含四个视图：

- Use Case View (用例视图)；
- Logical View (逻辑视图)；
- Component View (构件视图)；
- Deployment View (部署视图)。

每个视图可以包含特定类型或多种类型的图和模型元素。用例视图可以包括用例图、顺序图、协作图、状态图和活动图。逻辑视图可以包括类图、状态图、顺序图和协作图。构件视图包括一个或多个构件图。部署视图包括一个部署图。

图形窗口用于显示、创建和修改 UML 图。文档窗口用于创建、查看和修改选定的模型元素、图或视图的文档。标准工具栏包括当前处于活动状态的各种类型的图可使用的工具，类似于一般的 Windows 窗口的标准工具栏。图形工具栏提供为一个图添加模型元素的图形工具。