

高等学校计算机类课程应用型人才培养规划教材

UML 建模语言 及其开发工具 Rose

UML modeling language and its development tools Rose

余永红 陈晓玲 编著



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

高等学校计算机类课程应用型人才培养规划教材

UML 建模语言及其 开发工具 Rose

余永红 陈晓玲 编著

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书介绍了统一建模语言(UML)和 Rational Rose 软件的基础知识与使用方法,从实用角度介绍了 UML 与 Rose 的有机结合,及其在面向对象开发中的有效应用。主要内容包括 UML 的基础知识和 Rational Rose 的使用方法、UML 静态建模、UML 动态建模、UML 双向工程、RUP 等。本书结构合理,语言简练易懂,论述深入浅出,并配有大量的实例分析以帮助读者更好地理解 UML 与 Rose。

本书适合作为高等学校计算机应用类专业以及信息管理与信息系统专业的本科教材,也可作为 UML 建模人员的参考资料和相关培训教材。

图书在版编目(CIP)数据

UML 建模语言及其开发工具 Rose / 余永红, 陈晓玲编
著. —北京: 中国铁道出版社, 2011. 5
ISBN 978-7-113-12690-2
高等学校计算机类课程应用型人才培养规划教材
I. ①U… II. ①余… ②陈… III. ①面向对象语言,
UML—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 053250 号

书 名: UML 建模语言及其开发工具 Rose
作 者: 余永红 陈晓玲 编著

策划编辑: 严晓舟
责任编辑: 周海燕
编辑助理: 何 佳
封面设计: 付 巍
版式设计: 李 路

读者热线: 400-668-0820
封面制作: 白 雪
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: 北京新魏印刷厂

版 次: 2011 年 5 月第 1 版 2011 年 5 月第 1 次印刷

开 本: 787mm×1 092mm 1/16 印张: 15.75 字数: 367 千

印 数: 3 000 册

书 号: ISBN 978-7-113-12690-2

定 价: 25.00 元

版权所有 侵权必究

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。

编审委员会

主 任：何新贵 教授、中国工程院院士 (北京大学)

副主任：(按姓氏笔画排序)

刘瑞挺	教授	(南开大学)
罗晓沛	教授	(中国科学院研究生院)
岳丽华	教授	(中国科学技术大学)
周广声	教授	(上海海事大学)
徐洁磐	教授	(南京大学)

委 员：(按姓氏笔画排序)

王 浩	教授	(合肥工业大学)
王晓峰	教授	(上海海事大学)
史九林	教授	(南京大学)
白晓颖	教授	(清华大学)
刘 强	副教授	(清华大学)
许 勇	教授	(安徽师范大学)
孙志挥	教授	(东南大学)
李龙澍	教授	(安徽大学)
李银胜	副教授	(复旦大学)
李盛恩	教授	(山东建筑工程学院)
李敬兆	教授	(安徽理工大学)
杨吉江	教授	(清华大学)
何守才	教授	(上海第二工业大学)
余 粟	副教授	(上海工程技术大学)
张 莉	教授	(北京航空航天大学)
张 瑜	教授	(上海工程技术大学)
张燕平	教授	(安徽大学)
陈世福	教授	(南京大学)
陈涵生	研究员	(上海华东计算技术研究所)
迟洪钦	副教授	(上海师范大学)
林钧海	教授	(南京航空航天大学)
金志权	教授	(南京大学)
周鸣争	教授	(安徽工程科技学院)
周根林	教授级高工	(中电集团)
郑人杰	教授	(清华大学)
胡学钢	教授	(合肥工业大学)
徐永森	教授	(南京大学)
殷人昆	教授	(清华大学)
郭学俊	教授	(河海大学)
黄上腾	教授	(上海交通大学)
董继润	教授	(山东大学)
蔡瑞英	教授	(南京工业大学)

当前,世界格局深刻变化,科技进步日新月异,人才竞争日趋激烈。我国经济建设、政治建设、文化建设、社会建设以及生态文明建设全面推进,工业化、信息化、城镇化和国际化深入发展,人口、资源、环境压力日益加大,调整经济结构、转变发展方式的要求更加迫切。国际金融危机进一步凸显了提高国民素质、培养创新人才的重要性和紧迫性。我国未来发展关键靠人才,根本在教育。

高等教育承担着培养高级专门人才、发展科学技术与文化、促进现代化建设的重大任务。近年来,我国的高等教育获得了前所未有的发展,大学数量从1950年的220余所已上升到2008年的2200余所。但目前高等教育与社会经济发展不相适应的问题越来越凸显,诸如学生适应社会以及就业和创业能力不强,创新型、实用型、复合型人才紧缺等。2010年7月发布的《国家中长期教育改革和发展规划纲要(2010—2020)》提出了高等教育要“建立动态调整机制,不断优化高等教育结构,重点扩大应用型、复合型、技能型人才培养规模”的要求。因此,新一轮高等教育类型结构调整成为必然,许多高校特别是地方本科院校面临转型、准确定位的问题。这些高校立足于自身发展和社会需要,选择了应用型发展道路。应用型本科教育虽早已存在,但近几年才开始大力发展,并根据社会对人才的需求,补充了新的教育理念,现已成为我国高等教育的一支重要力量。发展应用型本科教育,也已成为中国高等教育改革与发展的重要方向。

应用型本科教育既不同于传统的研究型本科教育,又区别于高职高专教育。研究型本科培养的人才将承担国家基础型、原创型和前瞻型的科学研究,它应培养理论型、学术型和创新型的研究人才。高职高专教育培养的是面向具体行业岗位的高素质、技能型人才,通俗地说,就是高级技术“蓝领”。而应用型本科培养的是面向生产第一线的本科层次的应用型人才。由于长期受“精英”教育理念的支配,脱离实际、盲目攀比,高等教育普遍存在重视理论型和学术型人才培养的偏向,忽视或轻视应用型、实践型人才的培养。在教学内容和教学方法上过多地强调理论教育、学术教育而忽视实践能力的培养,造成我国“学术型”人才相对过剩,而应用型人才严重不足的被动局面。

应用型本科教育不是低层次的高等教育,而是高等教育大众化阶段的一种新型教育层次。计算机应用型本科的培养目标是:面对现代社会,培养掌握计算机学科领域的软硬件专业知识和专业技术,在生产、建设、管理、生活服务等第一线岗位,直接从事计算机应用系统的分析、设计、开发和维护等实际工作,维持生产、生活正常运转的应用型本科人才。计算机应用型本科人才有较强的技术思维能力和技术应用能力,是现代计算机软、硬件技术的应用者、实施者、实现者和组织者。应用型本科教育强调理论知识和实践知识并重,相应地其教材更强调“用、新、精、适”。所谓“用”,是指教材的“可用性”、“实用性”和“易用性”,即教材内容要反映本学科基本原理、思想、技术和方法在相关现实领域的典型应用,介绍应用的具体环境、条件、方法和效果,培养学生根据现实问题选择合适的科学思想、概念、理论、技术和方法去分析、解决实际问题的能力。所谓“新”,是指教材内容应及时反映本学科的最新发展和最新技术成就,以及这些新知识和新成就在行业、生产、管理、服务等方面的最新应用,从而有效地保证学生

“学以致用”。所谓“精”，不是一般意义的“少而精”。事实常常告诉我们“少”与“精”是有矛盾的，数量的减少并不能直接导致质量的提高。而且，“精”又是对“宽与厚”的直接“背叛”。因此，教材要做到“精”，教材的编写者对教材的内容，要在“用”和“新”的基础上再进行去粗取精的工作，精选学生终身受益的基础知识和基本技能，力求把含金量最高的知识传承给学生。“精”是最难掌握的原则，是对编写者能力和智慧的考验。所谓“适”，是指各部分内容的深度、难度和信息量要适合应用型本科的教育层次、适合培养目标的既定方向、适合应用型本科学生的理解程度和接受能力。教材编写应贯彻启发式、深入浅出、理论联系实际、适合教学实践，使学生能够形成对专业知识的整体认识。以上四个方面不是孤立的，而是相互依存，相互促进的。“用”是教材建设的唯一目的和出发点，“用”是“新”、“精”、“适”的最后归宿。“精”是“用”和“新”的进一步升华。“适”是教材与计算机应用型本科培养目标符合度的检验，是教材与计算机应用型本科人才培养规格适应度的检验。

中国铁道出版社同高等学校计算机类课程应用型人才规划教材编审委员会经过近两年的前期调研，专门为应用型本科计算机专业学生策划出版了理论深入、内容充实、材料新颖、范围较广、叙述简洁、条理清晰的系列教材。本系列教材在以往教材的基础上大胆创新，在内容编排上努力将理论与实践相结合，尽可能反映计算机专业的最新发展；在内容表达上力求由浅入深、通俗易懂；编写的内容主要包括计算机专业基础课和计算机专业课；在内容和形式体例上力求科学、合理、严密和完整，具有较强的系统性和实用性。

本系列教材是针对应用型本科层次的计算机专业编写的，是作者在教学层次上采纳了众多教学理论和实践的经验及总结，不但适合计算机等专业本科生使用，也可供从事IT行业或有关科学研究工作的人员参考，适合对该新领域感兴趣的读者阅读。

在本系列教材出版过程中，得到了计算机界很多院士和专家的支持和指导，中国铁道出版社多位编辑为本系列教材的出版做出了很大贡献，本系列教材的完成不但依靠了全体作者的共同努力，同时也参考了许多中外有关研究者的文献和著作，在此一并致谢。

应用型本科是一个日新月异的领域，许多问题尚在发展和探讨之中，观点的不同、体系的差异在所难免，本系列教材如有不当之处，恳请专家及读者批评指正。

“高等学校计算机类课程应用型人才规划教材”编审委员会
2011年1月

统一建模语言（UML）是一种通用的可视化建模语言，用于对软件进行描述、可视化处理、构造和建立软件系统工作的文档。它记录构建系统的有关决策，可用于对系统的理解、设计、浏览、配置、维护和控制。UML 适用于各种软件开发方法、软件生命周期的各个阶段、各种应用领域以及各种开发工具，旨在统一过去建模技术的经验，吸收当今软件开发的实践经验从而形成一种标准的方法。UML 包括语义概念、表示法和指导规范，它提供了静态、动态、系统环境及组织结构的模型，为交互式的可视化建模工具所支持，支持现今大部分面向对象的开发过程，其目的是简化和强化现有面向对象的开发方法。Rose 是目前广泛使用的面向对象可视化建模工具之一，可用于对系统的建模、设计与编码，还可对已有的系统实施逆向工程，实现代码模型转换，以便更好地开发与维护系统。UML 与 Rose 的有机结合，在开发大型面向对象应用中发挥着巨大的作用。

目前，市场上关于 UML 与 Rose 的教材很多，但对 UML 和 Rose 技术的介绍大多基于语言和软件使用本身的层面，而对一些较为实用的技术则浅尝辄止，难以对读者独立开发项目有实质性的帮助。同时由于教材需求层次多、类型广，因此需要有适应不同需求特色的教材，鉴于此，作者在实际教学经验基础上，编写了本教材。本书在内容的编排上注重实用性，在强调基本知识理解与基本技能训练的同时，更注重对读者可扩展素质和创新能力的培养。

全书共分 14 章，各章的具体内容安排如下：

第 1 章：概要介绍面向对象技术，包括不同软件开发方法的比较、面向对象的基本概念、面向对象分析、面向对象设计、面向对象实现以及 UML 与面向对象方法之间关系描述等基本概念。

第 2 章：概要介绍 UML 的基础知识，包括 UML 的特性及发展历史、UML 概念模型、UML 的体系结构和 UML 的基本图形分类等。

第 3 章：概要介绍 Rational Rose 的基础知识，包括 Rational Rose 的安装、组织、建模等使用方法。

第 4 章：具体介绍用例和用例图的基础知识，主要包括参与者、用例、用例间关系、用例图建模等知识。

第 5 章：具体介绍类图和对象图的知识，主要包括类、类之间的关系、类图、对象图、类图建模等知识。

第 6 章：具体介绍包图的基础知识，包括包图、包图应用及包图建模等知识。

第 7 章：具体介绍交互图的基础知识，包括顺序图、通信图、时间图及彼此之间关系比较方面的知识。

第 8 章：具体介绍活动图的基础知识，包括活动图的组成元素、活动图的应用、活动图建模等知识。

第 9 章：具体介绍状态机图的基础知识，包括状态图、复杂状态图的转换、复合状态图的转换、状态图的应用、状态图的建模等知识。

第 10 章：具体介绍构件图的基础知识，包括构件图的概念、构件图的应用、构件图建模等知识。

第 11 章：具体介绍部署图的基础知识，包括部署图的概念、部署图的应用、部署图建模等知识。

第 12 章：具体介绍双向工程的基础知识，包括双向工程的概念、正向工程、逆向工程及双向工程的实施等知识。

第 13 章：具体介绍 UML 和 Rational 统一过程的基础知识，包括 RUP 二维开发模型、UML 与 RUP 关系等知识。

第 14 章：系统综合，以网上书店的具体实例完整地介绍 UML 和 Rational Rose 在整个项目开发过程中的使用。

本教材具有如下特点：①内容全面细致，具有系统性。书中内容既包括面向对象理论介绍，又全面介绍了 UML 的基础知识，特别是对 Rose 支持的图和模型元素进行了详细的讲解，同时结合案例给出了相关 Rose 的具体操作。本书集理论、操作于一体。②案例讲解深入透彻。书中使用网上书店系统的建模案例贯穿于各个 UML 基础知识的章节，每一章都力图给出建模时详细的分析过程，而非泛泛的建模结果，让读者在学习的过程中知道如何做以及为什么这样做，有助于读者边学习、边思考实践。③Rose 介绍难度适中。本书在案例建模过程中介绍 Rose 的使用，既非给出详细的帮助文档和规范说明，也非浅尝即止的泛泛而谈，而是按照建模的过程步步深入，将细化的模型元素细节通过 Rose 绘制出来。④图文并茂，通俗易懂。本书在介绍每个章节、知识点、案例以及 Rose 的使用时配有大量的图表，有助于读者更加直观地理解 UML 的理论知识，掌握 Rose 的使用技巧。

本书适合作为高等学校计算机应用类专业以及信息管理与信息系统专业的本科教材，也可作为 UML 建模人员的参考资料和相关培训教材。

本书主要由余永红、陈晓玲编写，全书由南京大学徐洁磐教授审阅并提出许多宝贵意见。本书编写过程中还得到许多老师的支持和帮助，他们参与了资料的收集、分析和部分编写工作，并给出了许多好的建议，在此一并表示衷心的感谢。

在本书的编写过程中，作者参阅借鉴了大量的参考资料，在此谨向诸多学者表示衷心的感谢。由于作者水平有限，虽对本书作反复的审核，书中错误与缺点在所难免，希望读者给予批评指正，多提宝贵意见。

编 者

2011 年 2 月

第 1 章 面向对象技术.....	1
1.1 不同软件开发方法的比较分析.....	1
1.2 面向对象的基本概念.....	6
1.2.1 对象与类.....	6
1.2.2 消息与方法.....	9
1.2.3 继承.....	10
1.2.4 多态与动态绑定.....	11
1.3 面向对象分析、设计与实现.....	12
1.3.1 面向对象分析.....	13
1.3.2 面向对象设计.....	14
1.3.3 面向对象实现.....	17
1.4 UML 与面向对象系统开发.....	18
1.4.1 软件工程中的分析与设计.....	18
1.4.2 常用面向对象分析与设计方法的比较.....	22
1.4.3 统一建模语言 (UML).....	25
1.4.4 在软件开发过程中使用 UML.....	28
1.4.5 UML 与 Rose 的有效结合.....	29
1.5 UML 应用领域.....	30
小结.....	31
复习思考题.....	31
第 2 章 UML 概述.....	33
2.1 什么是 UML.....	33
2.1.1 UML 的特性.....	33
2.1.2 UML 发展历程与现状.....	33
2.2 为什么学习 UML——建模.....	35
2.2.1 什么是模型.....	35
2.2.2 为什么要建模.....	35
2.2.3 UML 可以建立什么模型.....	36
2.3 用 UML 建立概念模型.....	37
2.3.1 事物.....	37
2.3.2 关系.....	41
2.3.3 图.....	42
2.3.4 UML 规则.....	42
2.3.5 公共机制.....	43

2 UML 建模语言及其开发工具 Rose

2.4 UML 体系结构与图	45
2.4.1 UML 体系结构——“4+1”视图	45
2.4.2 UML 图形分类	47
2.5 UML 中常用概念与面向对象概念的对应关系	48
小结	49
复习思考题	49
第 3 章 Rational Rose 简介	50
3.1 准备工作	50
3.1.1 Rational Rose 的功能与特点	50
3.1.2 Rational Rose 的获取与安装	51
3.2 Rational Rose 的使用	51
3.2.1 应用程序界面	51
3.2.2 Rational Rose 视图的组织方法	55
3.2.3 Rational Rose 建模基本操作	57
3.2.4 Rational Rose 建模辅助操作	60
小结	61
复习思考题	62
第 4 章 用例和用例图	63
4.1 概述	63
4.2 参与者	64
4.2.1 参与者概念	64
4.2.2 寻找和确定参与者	64
4.2.3 参与者的分类	65
4.2.4 参与者之间的关系	65
4.3 用例	66
4.3.1 用例的概念	66
4.3.2 发现用例	66
4.4 用例间的关系	66
4.5 边界	68
4.6 事件流与用例描述	69
4.7 用例图建模要点	71
4.8 用例图建模实例	72
4.8.1 实例分析	72
4.8.2 在 Rose 中创建用例图	75
小结	78
复习思考题	78
第 5 章 类图和对象图	80
5.1 类定义	80

5.1.1	类命名	80
5.1.2	类属性	81
5.1.3	类操作	81
5.2	类之间的关系	82
5.2.1	关联	82
5.2.2	特殊关联——聚合和组合	86
5.2.3	泛化	87
5.2.4	依赖	88
5.3	涉及类的其他概念	89
5.3.1	抽象类	89
5.3.2	接口和实现关系	89
5.3.3	模板类	90
5.3.4	边界类、控制类和实体类	90
5.4	类图	91
5.4.1	类图的应用和抽象层次	91
5.4.2	类图建模	93
5.5	面向对象设计的原则	93
5.5.1	开闭原则	94
5.5.2	Liskov 替换原则	94
5.5.3	依赖倒置原则	95
5.5.4	接口分离原则	95
5.6	对象图介绍	96
5.6.1	对象	96
5.6.2	对象图	97
5.7	类图建模实例	97
5.7.1	案例分析	97
5.7.2	在 Rose 中创建类图	103
	小结	112
	复习思考题	112
第 6 章	包图	115
6.1	包及其之间的关系	115
6.1.1	包	115
6.1.2	包之间的关系	116
6.2	包图的应用	117
6.2.1	对成组元素建模	117
6.2.2	对体系结构建模	118
6.2.3	其他应用	118
6.3	包图建模实例	118
6.3.1	实例	118

4 UML 建模语言及其开发工具 Rose

6.3.2 使用 Rose 绘制包图	119
小结	121
复习思考题	122
第 7 章 交互图	123
7.1 概述	123
7.2 顺序图	123
7.2.1 顺序图的建模元素	124
7.2.2 UML2.0 中的新概念——交互片段	127
7.3 通信图	130
7.3.1 通信图的建模元素	130
7.3.2 顺序图和协作图的比较	131
7.4 时间图	131
7.5 交互图建模实例	132
7.5.1 实例分析	132
7.5.2 使用 Rose 绘制交互图	136
小结	143
复习思考题	143
第 8 章 活动图	144
8.1 概述	144
8.2 活动图的组成元素	144
8.2.1 初始结点和活动终点	145
8.2.2 转换	145
8.2.3 分支与监护条件	145
8.2.4 分叉与汇合	145
8.2.5 泳道	146
8.2.6 对象流	147
8.2.7 活动的分解	147
8.3 活动图的应用	148
8.4 活动图建模实例	149
8.4.1 实例分析	149
8.4.2 在 Rose 中创建活动图	151
小结	155
复习思考题	155
第 9 章 状态机图	156
9.1 概述	156
9.1.1 状态	156
9.1.2 状态机	156
9.2 简单状态图	157

9.3	包含复杂转换的状态图	159
9.4	包含复合状态的状态机图	161
9.5	状态机图的应用	165
9.6	状态机图建模实例	165
9.6.1	状态机图实例	165
9.6.2	在 Rose 中创建状态图	167
小结		171
复习思考题		172
第 10 章	构件图	173
10.1	概述	173
10.2	构件	174
10.2.1	构件表示法	174
10.2.2	构件和类	174
10.2.3	构件的分类	174
10.2.4	构件的接口	175
10.2.5	UML2.x 中构件的嵌套	176
10.3	构件图的应用	176
10.4	构件图建模实例	177
10.4.1	构件图实例	177
10.4.2	在 Rose 中创建构件图	178
小结		182
复习思考题		182
第 11 章	部署图	183
11.1	概述	183
11.2	部署图中的元素	183
11.2.1	结点	183
11.2.2	连接	184
11.3	部署图的应用	184
11.4	部署图建模实例	185
11.4.1	部署图实例	185
11.4.2	在 Rose 中创建部署图	185
小结		189
复习思考题		190
第 12 章	双向工程	191
12.1	概述	191
12.2	正向工程	191
12.3	逆向工程	197
12.4	双向工程实例	198

小结	210
复习思考题	210
第 13 章 UML 与统一开发过程	211
13.1 Rational 统一过程	211
13.2 RUP 二维开发模型	212
13.2.1 RUP 二维开发模型简介	212
13.2.2 开发过程的动态结构——4 个阶段	212
13.2.3 迭代过程	214
13.2.4 开发过程中的静态结构	214
13.2.5 核心工作流	215
13.3 UML 与 RUP	218
13.3.1 业务模型	218
13.3.2 需求捕获阶段	218
13.3.3 分析阶段	219
13.3.4 设计阶段	219
13.3.5 实现阶段	220
小结	220
复习思考题	221
第 14 章 系统综合——网上书店	222
14.1 网上书店需求分析	222
14.2 系统动态模型	224
14.2.1 活动图	224
14.2.2 顺序图与通信图	227
14.2.3 状态图	229
14.3 系统静态模型	230
14.3.1 类模型	230
14.3.2 包图	233
14.4 系统的配置与实现	234
14.4.1 构件图	234
14.4.2 部署图	234
小结	234
参考文献	235

第1章 面向对象技术

20 世纪 60 年代中期以来, 基于计算机的系统规模越来越大, 系统的复杂性也不断增加, 人们很难把握软件开发过程, 开发出来的软件正确性越来越难以保证, 引起生产率急剧下降。由于大型软件开发的工程性和实际开发工作中手工性的不相适应, 出现了软件危机。为解决上述问题, 在 1968 年的北大西洋公约学术会议上提出了软件工程的思想, 希望采用工程方法来开发软件。40 多年来这一思想逐步为人们所重视, 其发展迅速, 并取得了令人瞩目的成就。但这一时期该领域采用的方法主要是结构化的分析与设计方法, 这种方法对现代日益复杂的软件开发而言存在严重的不足, 由此产生的软件危机并未真正得到解决。因此, 人们正设法寻找一种新的有效途径来解决这一问题。

面向对象概念自 20 世纪 60 年代出现至今, 面向对象技术已成为一种完整的思想与方法体系, 并且在计算机领域中得到广泛应用, 如在程序设计中的面向对象程序设计、在人工智能中的面向对象知识表示、在数据库中的面向对象数据库、在人机界面中的面向对象图形用户界面、在计算机体系结构中的面向对象结构体系等都有突出表现。由于面向对象技术在软硬件开发方面呈现出巨大的优越性, 人们将其视为解决软件危机的一个很有希望的突破口。面向对象方法的出现是与软件开发方法的发展密切相关的, 为使读者对面向对象方法有一个全面深刻的理解, 有必要对在过去几十年中所采用的软件开发方法作一个比较分析。

1.1 不同软件开发方法的比较分析

计算机软件是问题求解的一种表述形式, 一个计算机软件通常由程序、数据与文档构成。程序是用程序设计语言描述的、适合计算机处理的语句序列, 数据是程序处理的客观实体, 文档是一种数据媒体和其所记录的数据, 文档记录软件的开发活动和阶段成果供人阅读。一般计算机的发展过程大致可分为两个阶段: 第一阶段, 这个阶段计算机的应用领域比较窄, 主要是科学计算, 其研究、开发的主要内容是顺序程序, 对数据的考虑非常少。在这个阶段主要是考虑程序设计, 而数据是依附于程序中的, 注重对系统功能的实现, 是典型的面向过程的软件开发方式。第二阶段, 这个阶段计算机应用领域逐步拓宽, 出现了大量的数据处理问题。这个阶段的软件特点是数据量非常大, 并要求把这些数据存储在文件系统及数据库中以实现数据的独立和共享, 而对数据的处理则相对简单。这个阶段主要考虑数据设计, 而程序则是依附于数据的, 注重对系统数据的实现, 是典型的面向数据的软件开发方式。

从计算机软件发展过程中, 我们可以看到程序和数据在软件开发中所处的地位和作用以及与软件开发方法的密切关系。随着程序和数据在软件开发过程中地位和作用的改变, 出现了不

同的与程序和数据在软件开发过程中的地位和作用相适应的软件开发方法。过去几十年中，在实际系统开发中成功使用了面向过程的方法、面向数据的方法以及两者并重的信息工程方法，下面对这几种方法作一个简单介绍。

(1) 面向过程的方法：面向过程的方法主要考虑事务流和要完成的功能，数据被当做系统整体的一个部分来处理，常常是分析过程的一个副产品。其代表方法是结构化分析与设计方法，结构化方法主要用于在目标范围确定以后对事务功能活动的分析，使用自顶向下逐层分解的方式抽象出实现基本功能的数据。结构化方法采用明显的需求输入—处理—输出方式，数据同转换数据的处理是分开考虑的。系统行为虽然重要，但在结构化分析中只能扮演一个次要角色，结构化方法着重功能分解，其理解、表达及验证的内容只是用户的功能需求，功能是它的出发点与归宿。结构化方法只能部分地反映客观世界，由于问题空间包含所有可能的实体及相互关系，其隐含所有可能的功能需求。而实际上问题空间是相对稳定的，变化的只是用户功能需求。功能需求应在问题空间确立之后进行，而结构化方法则忽视了这一点，因此它不可能完整、准确地反映客观世界，且其分析基础是易变的用户功能需求，因而造成软件结构与问题结构不一致，而功能的经常改变使设计、编程和维护极为困难。

(2) 面向数据的方法：面向数据的方法将数据作为一种独立的资源对待，数据是支持所有应用项目的基础，处理过程也是很重要的，但是它们必须存取数据库中的数据。其代表方法是数据库设计方法。在现有数据库基础上开发应用项目，系统开发人员要描述稳定的数据库中数据的内在性质，而这个数据库是独立于任何所要编程的功能或数据流的。数据库设计方法侧重于用数据来描述现实世界，它从分析目标功能领域中的管理活动开始，标识各种不同类型的管理活动中涉及的实体、实体间的联系以及它们的特性，从而得到一个描述这些实体、实体间联系及实体特性的数据集，然后用目标范围内的数据处理要求对该数据集进行验证并加以必要的修改与补充。数据库设计方法直接将问题域映射为模型中的实体，但由于面向数据的方法较少考虑客观实体的行为，因此还需要考虑更为详细的映射方法。

(3) 信息工程的方法：信息工程是在软件工程和数据库基础上形成并发展起来的，其方法采用的前提是数据位于现代数据处理系统的中心，数据类型不会发生很大的变化，因此这种方法是面向数据的方法。但同时在采用这种方法进行系统开发时又充分考虑了对数据的创建与使用的过程处理(U/C 矩阵)，并充分利用结构化的自顶向下的设计方法，因此该方法又具有面向过程的特点，是一种综合运用自顶向下和自底向上设计方法的分析与设计方法。

上述三种系统开发方法虽然在其所处的计算机软件发展阶段中获得了很大的成功，但采用这些开发方法实现的软件系统在可靠性、可维护性和可重用性方面都存在很多的不足。人们在对问题空间理解的无数次实践中发现，一个系统各个部分的稳定程度依次为：系统各部分之间的接口极其不稳定，功能模块非常不稳定，各模块的先后顺序非常不稳定，一直保持的数据则较稳定，问题空间的对象最稳定。从提高软件的可靠性、可维护性与可重用性等质量的角度来看，基于系统非常不稳定的功能模块部分的开发方式难以满足要求，因为这种方式针对具体问题功能，每次功能的改变都需作大量重复而烦琐的工作。由于基于系统较稳定的数据部分的开发方式具有较稳定的软件体系结构，因此该方式在软件质量方面比基于功能模块的开发方式有较大的提高，但这种方式没有考虑进一步转向过程的研究，因此这种方式在软件维护和重用方面还存在问题，具体的比较分析为：

(1) 面向过程方法主要考虑事务流和需要完成的功能，它侧重于功能与功能抽象，而恰恰

忽视了功能的主体——对象。任何成功的系统在其生存期内都不可避免地要经历多次变动，其中变动最多的是系统功能，而对象才是问题空间中最稳定的部分。由于面向过程的方法建立在非常不稳定的系统功能基础上，因此面向过程的方法不能解决数据抽象和信息隐藏的问题，不能很好地适应极度复杂系统的扩展。

(2) 面向数据方法建立在较为稳定的系统数据基础上，将数据作为一种独立的资源对待，对数据进行抽象。但面向数据的方法在对数据进行抽象时只注重对客观实体及实体属性的抽象，而客观实体不仅有属性还有操作属性的行为，因为最终每个数据处理系统都必须有数据与处理。由于面向数据方法在系统开发过程中很少考虑实体的行为处理，因而只是部分地实现了问题空间到解空间的映射。

(3) 信息工程方法既是面向数据的方法，又具有面向过程的特征，是一种比较灵活的系统开发方法。但该方法在对面向过程和面向数据的综合过程中，进行数据抽象时，对属性和操作属性的方法(行为)是分开考虑的，而没有把属性和方法当做一个不可分割的整体，而这种属性和方法的封装对一个系统模型的稳定性有很大的影响，因此信息工程的方法也没能很好地解决数据抽象和信息隐藏问题，也是部分地实现了问题空间到解空间的映射。

虽然上述几种分析与设计方法在实际应用中取得了很大的成就，但由于这几种方法都存在一定的缺陷，其根本原因是人们认识一个系统的过程和方法同人们用于分析、设计和实现一个系统的过程和方法不一致。长久以来，人们一直都致力于解决认识一个系统的过程和方法与用于分析、设计和实现一个系统的过程和方法不一致的矛盾，人们在不断的实践中意识到解决这个问题的根本方法应是使描述问题的问题空间和解决问题的方法空间在结构上尽可能保持一致。人们发现若基于系统最稳定部分的对象进行软件开发，可以比较容易地实现描述问题的问题空间和解决问题的方法空间在结构上的一致，而这正是面向对象方法的出发点和追求的基本原则。

面向对象方法认为客观世界是由各种对象组成的，每个对象有其自身的运动规律和内部状态，每个对象都属于某个对象类，是该对象类的一个元素；复杂对象由相对简单的对象按某种方式构成，不同对象的组合及相互作用构成人们要分析和构造的客观系统；通过类比发现对象间的相似性，是构成对象类的依据，对象类按类、子类和超类构成对象类的层次关系，下层继承上层的所有特性；对于对象的功能，可通过定义一组方法来说明，对象间的相互联系通过消息传递来完成，属性与方法是封装在对象类内的，它们对外界是隐藏的，因此面向对象方法自然地模拟了人类认识客观世界的方式。

面向对象方法综合了传统软件开发的优点，同时克服了上述传统软件开发方法的不足，形成了其独特的优越性，具体表现为以下几个方面：

(1) 与人类习惯的思维方式一致。传统的程序设计技术是面向过程的设计方法，这种方法以算法为核心，把数据和过程作为相互独立的部分，数据代表问题空间中的客体，程序代码则用于处理这些数据。

把数据和代码作为分离的实体，反映了计算机的观点，因为在计算机内部数据和程序是分开存放的。但是，这样做的时候总存在使用错误的数据调用正确的程序模块，或使用正确的数据调用错误的程序模块的风险。使数据和操作保持一致，是程序员的一个重要任务，在多人分工合作开发一个大型软件系统的过程中，如果负责设计数据结构的人中途改变了某个数据的结构而又没有及时通知所有人员，则会发生许多不该发生的错误。

传统的程序设计技术忽略了数据和操作之间的内在联系，用这种方法所设计出来的软件系