21世纪高等学校计算机科学与技术教材

面向对象技术导论
——系统分析与设计

刁成嘉　主编

机械工业出版社
China Machine Press

21世纪高等学校计算机科学与技术教材

面向对象技术导论

——系统分析与设计

刁成嘉　主编

机 械 工 业 出 版 社

本书系统、全面地阐述了面向对象技术的基本概念，详细介绍了统一建模语言 UML 及其开发过程，以具体案例为模型全面介绍面向对象系统开发方法。以一个集成案例贯穿各章，讲解循序渐进、前后贯通。使学习者能够较快地掌握面向对象系统的分析、设计方法。

本书还介绍了面向对象技术的高级内容，如通用设计样式、持久对象、分布式对象技术、COM+、EJB、CORBA 等对象接口技术。另外，本书也介绍了软件复用技术和面向对象软件开发 CASE 集成环境。

本书可作为高等院校计算机与科学技术专业的相关课程教材，也适合作为广大软件开发人员学习面向对象技术的自学指导书和技术参考书。

图书在版编目（CIP）数据

面向对象技术导论——系统分析与设计/刁成嘉主编.
-北京：机械工业出版社，2004.7
（21 世纪高等学校计算机科学与技术教材）

ISBN 7-111-14934-3

Ⅰ.面… Ⅱ.刁… Ⅲ.面向对象语言-程序设计 Ⅳ TP312

中国版本图书馆 CIP 数据核字（2004）第 070698 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）
责任编辑：夏孟瑾　　版式设计：侯哲芬
三河市宏达印刷有限公司印刷·新华书店北京发行所发行
2004 年 9 月第 1 版第 1 次印刷
787mm×1092mm 1/16·18 印张·421 千字
0001-5000 册
定价：27.00 元

前　言

当代软件工程的发展正面临着从传统的结构化范型到面向对象范型的转移，这需要有新的语言、新的系统和新的方法学的支持，面向对象技术就是这种新范型的核心技术。面向对象的分析和设计方法已逐渐取代了传统的方法，成为我国当前计算机软件工程学的主流方法。

面向对象方法包括面向对象分析（OOA）、面向对象设计（OOD）、面向对象实现（OOI）、面向对象测试（OOT）和面向对象系统维护（OOSM）。其核心思想就是利用面向对象的概念和方法为软件需求建立模型，进行系统设计，采用面向对象程序设计语言完成系统实现，并对建成的系统进行面向对象的系统测试和系统维护。在今天，特别是随着 Internet/Intranet 的发展，网络分布计算的应用需求日益增长，面向对象技术为网络分布计算提供了基础性核心技术支持。

本书系统地介绍了面向对象技术的基本概念，面向对象的开发方法、类的封装、对象类和实例、对象的关联、继承、多态性、构件和接口等，还介绍了通用设计样式、持久对象、往返工程、逆向工程、COM+、EJB、CORBA、分布式对象、软件的复用及 CASE 集成环境。

书中详细介绍了 OMG（对象管理组织）推荐的统一建模语言 UML 及其开发过程。通过案例模型全面介绍如何采用面向对象的方法开发一个软件项目，从客户需求出发，利用 CASE 集成环境采用循环、反复、渐增的方法设计系统对象的逻辑模型、物理模型、静态模型和动态模型。每个阶段都配以案例，通过多个案例全面展示系统模型及其产生过程。

本书共 9 章，各章内容如下：

第 1 章概要地介绍了软件方法学的演变历程及面向对象方法学的发展过程，面向对象的基本概念和几种经典的面向对象方法论。

第 2 章重点介绍统一建模语言 UML 开发过程。从客户需求分析（OOA）到系统设计（OOD）及系统实现（OOI）、系统测试（OOT）和系统维护（OOSM），并介绍了一些实用的方法。

第 3~7 章详细讲述了利用统一建模语言 UML 开发建立一个软件项目模型的全过程，从建立系统的用例图、对象类图、时序图、活动图、状态图、协作图到构件图、部署图及相应的包图，通过实际案例引导完成全部系统建模。并介绍了在面向对象系统设计中经常出现、具有实用价值的一些通用设计样式。

第 8 章介绍了面向对象高级技术，如 COM+、EJB、CORBA 等构件接口技术、持久对象与关系数据库、面向对象数据库，以及客户机/服务器模型及分布式对象模型。

第 9 章重点讲述了面向对象的软件复用的方法和组织实施。还对 CASE 工具及集成环境的类型、发展及 OOCASE 集成环境的功能与结构进行了讨论。并对 Rose 2002 集成环境及其使用作了简单介绍。

本教材适用于一学期36课时的课程使用，建议安排如下：

第1章：面向对象技术概论［2课时］

第2章：统一建模语言UML［4课时］

第3章：用例建模［2课时］

第4章：类和对象建模［6课时］

第5章：动态建模（一）［4课时］

第6章：动态建模（二）［4课时］

第7章：物理体系结构建模［4课时］

第8章：高级对象技术［6课时］

第9章：CASE工具与集成化环境［4课时］

教学建议

建议在本课程开始时，为每个同学选择一个拟开发的课题作为一个案例。在教学过程中，随着课程的深入，逐步开发、完善案例的系统模型设计。系统的实现不拘泥何种程序设计语言，同学可以参看C++、Java等，力争在学期末，能有一个完整的系统实现。

在本教材的编写过程中，杨志真、刁奕、李炜、刘胜斐、罗仕波、郑莹莹、漆芳敏等对书中的部分图示进行了绘制。由于编者水平所限，加之时间仓促，疏漏、欠妥、谬误之处在所难免，敬请读者批评指正。

编者

目　录

第 1 章　面向对象技术概论

面向对象技术的概念和方法，本质上是一种合理的思维方法，是不依赖程序设计语言的应用软件开发的基本核心技术。学习和掌握面向对象技术的基本要点，是软件开发者必备的基础知识；越是深入理解面向对象技术的理论和方法，就越能在自己的应用领域中最大限度地发挥思维能力和创造本领，开发出高质量的软件系统。

本章目的：

- 了解软件开发的基本方法
- 理解面向对象技术的基本概念
- 了解面向对象方法的特点和优点
- 初步了解面向对象分析（OOA）的方法和步骤
- 初步了解面向对象设计（OOD）的工作要点

自 20 世纪 40 年代发明计算机以来，计算机在各个领域得到了广泛的应用，使得计算机技术蓬勃发展。然而，长期以来计算机软件开发的低效率制约着计算机软件行业的发展。计算机业界努力探索和研究解决软件危机的途径，提出了软件工程的思想和方法，极大地提高了软件开发的效率和软件的质量。当代软件工程的发展正面临着从传统的结构化范型到面向对象范型的转移，这需要有新的语言、新的系统和新的方法学的支持，面向对象技术就是这种新范型的核心技术。面向对象的分析和设计方法已逐渐取代了传统的方法，成为我国当前计算机软件工程学的主流方法。

计算机科学发展的每一步几乎都在软件设计和程序设计语言中得到体现。软件是一个发展的概念，在早期的概念里，程序就是软件。从 20 世纪 40 年代开始，人们从在 ENIAC 计算机上编制程序开始到软件工程术语提出为止的二十多年的时间里，对软件开发的理解就是编程序，且编程是在一种无序的、崇尚个人技巧的状态中完成。同今天的软件开发相比，那时的编程有以下特点：

（1）软件规模相对较小。由于硬件发展水平的限制，只有少数专业人员才去关心软件，而且这些专业人员必须懂得硬件，只能根据硬件的性能编写一些简单、单一的应用软件。从根本上阻碍了软件的广泛应用，限制了软件的规模。

（2）编程无规范与标准。大部分软件人员不太关心别人的工作，他们往往醉心于自己的编程技巧，决定软件质量和开发时间的惟一因素就是编程人员的素质，即个人的经验、智慧和技巧。编程是作为一门艺术被人们所津津乐道。

（3）缺少有效的软件开发方法和软件工具支持。这也是为什么人们把编程作为一门艺术来对待的原因。

（4）不重视软件开发的管理。由于人们重视个人的技能，而且软件开发的过程能见度

低，许多管理人员甚至根本不知道他们的软件技术人员究竟做得如何，造成管理活动很少甚至不存在。直到今天，这种观念在一些软件开发机构中仍有残留，因此，我们对软件设计开发的管理必须要给予足够的重视。

（5）软件维护困难。进入20世纪60年代以后，由于客观实际的需求，要制作一些大型的软件系统。在这种背景的驱动下，软件的开发方法和管理受到重视，逐渐产生出各种不同类型的软件开发方法和模型。随着时间的推移和实际工作的需要，软件开发方法也在不断更新、变化。

1.1　软件开发方法概述

随着软件开发规模的扩大和开发方法的变化，程序设计开始被人们作为一门科学来对待，它研究程序设计和实现的各种性质和规律。经过多年研究，在计算科学中发展了许多程序设计方法和技术。例如，自顶向下和自底向上的程序设计方法、程序推导设计方法、程序变换设计方法、函数式程序设计技术、面向对象的程序设计技术、程序验证技术、约束程序设计技术、分布式程序设计技术和并发程序设计技术等。程序设计方法和技术在各个时期的发展不仅直接导致了一大批风格各异的高级语言的诞生，而且对计算机理论、硬件、软件和计算机应用技术等多方面具有深刻的影响。在软件开发方法出现以前，就出现了程序设计方法学（PROGRAMMING METHODOLOGY），程序设计方法相对比较成熟，可以分为以下几种：

1. 结构化程序设计方法

结构化程序有3个主要特征：首先，其控制结构仅由顺序、选择与重复等结构复合而成；其次，程序自底向上逐步抽象成一个函数块；每个函数块都有一个入口和一个出口。在结构化程序设计中，应尽量避免或少量使用goto语句。对编写完成的程序采用3个步骤，即测试（testing）、确认（validation）与验证（verification）来证明程序设计的正确性。结构化程序设计方法使用的具有代表性的程序设计语言有结构Basic、结构FORTRAN和结构COBOL等。

2. 模块化程序设计方法

模块化程序设计方法就是将程序分成若干可单独命名和编址的部分，它们被称为模块。模块化使程序能够被有效地管理和维护，有效地分解和处理复杂问题。模块如何划分有以下几种观点：有人认为最佳途径是用函数；有人认为一个模块应容纳一个数据结构；有人认为一个模块应只做且仅做一件事情；还有人认为应以事件来驱动。模块之间的接口应尽可能简明清晰；单独模块的修改不影响其他模块的功能；模块化应具有可修改性、易读性和可验证性。模块化程序设计方法使用的具有代表性的程序设计语言有Modula和Simula等。

3. 面向对象程序设计方法

面向对象程序设计方法追求的是现实问题空间与软件系统解空间的近似和直接模拟。

不是以函数过程和数据结构为中心，而是以对象代表问题的中心环节，采用“对象+消息”的程序设计模式，使人们对复杂系统的认识过程与系统的程序设计实现过程尽可能地一致。面向对象技术采用以类为中心的封装、继承、多态性等技术，不仅支持软件复用，而且使软件维护工作可靠有效，可实现软件系统的柔性制造。面向对象程序设计方法使用的具有代表性的程序设计语言有 Smalltalk、C++、Java 和 C#等。

软件开发方法是来自于程序设计方法的发展，同时程序设计方法的研究又影响程序设计语言，反之亦然，彼此互相补充、共同发展。可以这样说，程序设计方法深刻地影响着软件开发方法和程序设计语言。

软件开发方法就是软件开发所遵循的办法和步骤，以保证所得到的运行系统和支持的文档满足质量要求。软件开发方法有很多种，其中针对系统分析和设计活动的软件开发方法和针对系统全局的软件开发方法尤为重要。下面简略地介绍其中的 4 种软件开发方法。

1.1.1　结构化方法

结构是指系统内各组成要素之间的相互联系、相互作用的框架。结构化方法强调系统结构的合理性以及所开发的软件的结构合理性，因此提出了一组提高软件结构合理性的准则，如分解和抽象、模块的独立性、信息隐蔽等。面向数据流方法是结构化方法家族中的一员，它具有明显的结构化特征。

1. 结构化分析的步骤

（1）分析用户当前需求，创建实体-关系图。
（2）根据实体-关系图作出相应的物理模型数据流图（DFD）。
（3）推导出等价的逻辑模型的数据流图。
（4）生成数据字典。
（5）创建控制流图。
（6）建立人机接口界面，提出可供选择的目标系统的物理模型数据流图。
（7）确定各种方案的成本和风险等级，据此对各种方案进行分析。
（8）选择一种方案。
（9）建立完整的需求规约。

结构化设计给出一组帮助设计人员在模块层次上区分设计质量的原理与技术，通常和结构化分析衔接起来使用，以数据流图为基础得到软件模块结构。

2. 结构化设计的步骤

（1）评审和细化数据流图。
（2）确定数据流图的类型。
（3）把数据流图映射到软件模块结构，设计出模块结构的上层。
（4）基于数据流图逐步分解高层模块，设计中下层模块。
（5）对软件模块结构进行优化，得到更为合理的软件结构。
（6）描述模块接口。

1.1.2 模块化方法

把一个待开发的软件系统分解成若干较为简单的部分，称为模块（modules）。每个模块分别独立地开发、测试，最后再组装出整个软件系统。这种开发方法是对复杂的系统“分而治之，各个击破”，将软件系统开发的复杂性在分解过程中降低。

在考虑模块化时，模块定义多大合适，模块设计规则的制定成为关键。下面 5 条标准可供参考：

（1）模块可分解性。如果一种设计方法提供了将问题分解成子问题的系统化机制，它就能降低整个系统的复杂性，从而实现一种有效的模块化解决方案。

（2）模块可组装性。如果一种设计方法使现存的（可复用的）设计模块能够被组装成新系统，它就能提供一种不用一切从头开始的模块化解决方案。

（3）模块可理解性。如果一个模块可以作为一个独立的单位（不用参考其他模块）被理解，那么它就易于构造和修改。

（4）模块连续性。如果对系统需求的微小修改只导致对单个模块，而不是整个系统的修改，则修改引起的副作用就会被最小化。

（5）模块保护。如果模块内出现异常情况，并且它的影响限制在模块内部，则错误引起的副作用就会被最小化。

1.1.3 面向数据结构方法

面向数据结构方法有很多种，它实际上是结构化方法的变形，着重数据结构而不是数据流。我们只对 Jackoon 系统开发方法（简称 JSD）作简单介绍，其开发步骤如下：

1. JSD 需求分析步骤

（1）标识系统中的实体与相应动作。

（2）生成实体结构图。

（3）初建系统模型。

2. JSD 系统设计步骤

（1）扩充功能过程：对系统分析产生的模型进行功能补充、完善。

（2）系统定时：确定实现必须满足的时间约束。

（3）实现：确定系统中所有软硬件成分，形成一个完整设计。

虽然每个面向数据结构的开发方法都有自己的一套开发步骤和过程，但概括起来有如下共同特点：

- 将分析结果作为设计基础，无明显分界；
- 都必须标识关键实体和动作；
- 信息具有层次性；
- 提供一组将层次化的数据结构映射到程序结构的步骤；

- 数据结构由顺序、选择和重复构造成分表示。

1.1.4　面向对象方法

面向对象方法起源于面向对象编程语言。面向对象方法包括面向对象分析（OOA）、面向对象设计（OOD）、面向对象实现（OOI）、面向对象测试（OOT）和面向对象系统维护（OOSM）。其核心思想就是利用面向对象的概念和方法为软件需求建立模型，进行系统设计，采用面向对象程序设计语言完成系统实现，并对建成的系统进行面向对象的系统测试和系统维护。在今天，特别是随着 Internet/Intranet 的发展，网络分布计算的应用需求日益增长，面向对象技术为网络分布计算提供了基础性核心技术支持。

面向对象方法的意义：

（1）面向对象方法是一种建立在已有的软件开发经验基础上的新的思考方式。

（2）将数据和行为结合成对象。

（3）核心是封装。

（4）面向对象方法建立的基础是：

- 软件工程概念；
- 计算机科学概念；
- 工程管理；
- 数据库信息模型；
- 传统软件开发方法。

面向对象方法有几十种，综合起来，其基本观点如下：

- 现实客观世界是由对象组成。任何客观的事物和实体都是对象，复杂对象可以由简单对象组成；
- 具有相同的数据和操作的对象可归并为一个类，具有封装性，形成一个包装；对象是类的一个实例，一个类可以产生很多对象；类能够被开发、再使用或购买；
- 类可以派生出子类，继承能避免共同行为的重复；
- 对象之间通过传递消息进行联系。

软件工程学家 Codd 和 Yourdon 认为：

面向对象 = 对象 + 类 + 继承 + 通信

面向对象的方法已经成为新趋势，自 20 世纪 80 年代开始，5 年之内面向对象方法论从 5 种开发到 50 种以上。其中布什（Booch）的面向对象软件工程在这些方法之中居于领导地位。比较著名的面向对象方法有：

- Booch 方法由布什发明；
- OMT 方法由詹幕斯·云豹（James Rumbaugh）发明；
- OOSE 方法由雅寇森（Jacobson）发明；
- 其他还有尤顿（Coad-Yourdon）、雪梨-米勒（Shlaer-Mellor）等开发方法。

在 1995 年，布什、云豹和雅寇森三位大师开始合作，整合 Boosh、OMT 和 OOSE 等方法的概念，从其他大师的方法中吸取最好的想法，并纳入企业最好的实务经验。在 1997