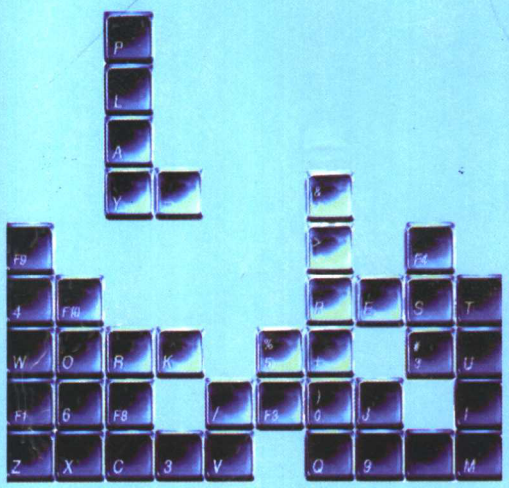
高等院校计算机专业教材

软件工程导论

刁成嘉　邵秀丽
马广慧　刁　奕　等编著

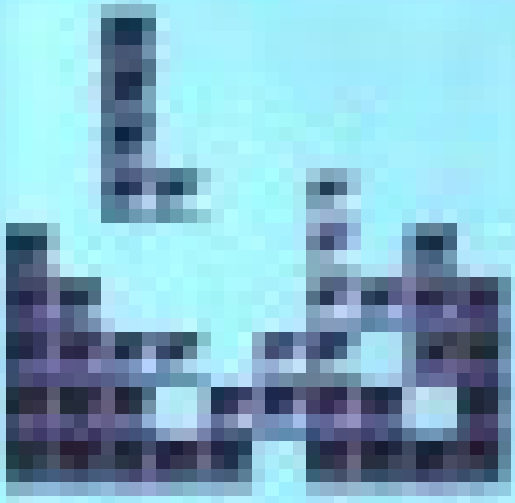
Software Engineering



南开大学出版社

软件工程导论

Software Engineering



软件工程导论

刁成嘉　邵秀丽
马广慧　刁　奕
等编著

南开大学出版社
天　津

内容提要

本教材对软件工程传统方法进行了简要介绍，重点对近年来国内外涌现的最新的软件工程方法和技术进行了系统的介绍，包括 UML、软件复用和构件接口技术（JRB/J2EE、CORBA 等）、软件过程工程建模、软件产品化技术、软件项目管理和软件开发 CASE 集成环境，等等。本书附有一个完整的标准文档格式供读者参考。

本书可以作为高等院校计算机和信息技术专业高年级软件工程课程教材，也适合作为相关专业研究生或广大软件开发人员学习软件工程技术的自学指导书和技术参考书。

图书在版编目(CIP)数据

软件工程导论 / 刁成嘉等编著. —天津：南开大学出版社，2006.8

ISBN 7-310-02583-0

Ⅰ.软... Ⅱ.刁... Ⅲ.软件工程 Ⅳ.TP311.5

中国版本图书馆 CIP 数据核字(2006)第 077769 号

南开大学出版社出版发行

出版人：肖占鹏

地址：天津市南开区卫津路 94 号 邮政编码：300071

营销部电话：(022)23508339 23500755

营销部传真：(022)23508542 邮购部电话：(022)23502200

*

河北省迁安万隆印刷有限责任公司印刷

全国各地新华书店经销

*

2006 年 8 月第 1 版 2006 年 8 月第 1 次印刷

787×1092 毫米 16 开本 20.25 印张 499 千字

定价：32.00 元

如遇图书印装质量问题，请与本社营销部联系调换，电话：(022)23507125

前 言

本教材薄古厚今，对软件工程的传统方法进行了简要的一般性介绍，对国内外涌现的当前最新的软件工程方法和技术进行了系统的重点介绍，涉及当今世界上软件工程的各种新技术、新概念和新设计开发方法。书中以作者多年相关教学及项目开发经验为基础，以一个完整的工程项目为案例，贯穿项目管理、需求分析、系统分析、系统设计、系统实现、系统调试、系统维护等各个阶段，给读者一个完整的软件工程开发全过程的实例。书后附录还附有一个实际的软件开发全过程的文档资料，作为读者进行自主软件开发的参考和依据。

本教材介绍了 OMG（对象管理组织）推介的统一建模语言 UML 及其开发过程；介绍了分布式对象技术、COM+、EJB、CORBA 等现代接口技术；另外还介绍了软件工程管理、软件产品化、软件过程及改进、软件复用技术和软件开发 CASE 集成环境。本教材可以作为高等院校计算机专业软件工程课程教材，也可作为相关专业研究生和软件开发人员学习软件工程的自学指导书和技术参考书。

本教材分为九章，内容简介如下。

第一章概要地介绍了软件方法学的演变历程及软件工程的发展过程，传统的软件工程方法模型，面向对象的基本概念和几种经典的面向对象方法论。

第二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及案例。

第三章详细讲述了软件过程的基本概念、过程工程框架、过程模型、构造方法和实施机制。还讨论了软件过程的度量及改进方法，并详细介绍了几个软件过程成熟度模型及其改进模式。

第四、五章系统地介绍了软件工程中需求分析、系统设计、系统实现、系统文档、系统测试、系统维护和逆向工程等技术、模型、方法和案例。

第六章介绍了软件工程与项目管理，包括软件项目需求、估算与进度、配置、风险、质量和资源管理。

第七章全面介绍了软件工程的新技术，如 COM+、EJB、CORBA 等构件接口技术，以及客户机/服务器模型及分布式对象模型。

第八章介绍了软件产品化技术。包括软件评审、生存周期软件开发 V 模型和软件自动化测试技术等内容。

第九章重点讲述了软件工程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 2004 集成环境及其使用作了简单介绍。

教学建议

建议在本课程开始时，请每位同学选择一个拟开发的课题（也可由教师提供题目）作为一个课程学习的设计案例。在教学过程中，通过案例模型逐步介绍如何采用软件工程的方法开发一个软件项目，而同学们则利用 CASE 集成环境采用循环、反复、渐增的方法设计系统的各种逻辑模型、物理模型、静态模型和动态模型。随着课程的深入，逐步开发完善学生自己案例的系统模型分析与设计。系统的实现不拘泥于某种程序设计语言，建议同学首先参看 C++和 Java 语言。在学期末，希望每位同学都能有一个完整的系统实现。

刁成嘉负责全书的策划和第 1、2、3、7 章的编写，邵秀丽编写了第 4、5、6、8 章和附录中文档标准格式，马广慧参加了第 2 章及附录的编写工作，刁奕编写了第 9 章。另外，在本教材的编写过程中，陈艳秋、张琳、吴宝琪、金士英、高建国、旷昊、李玉福、李军、邢恩军、杨茹、李昕、肖鹏、程玉鹏、侯乐彩、黄硕、费志泉、郜业军、蓝炳伟等参加了部分章节示例习题编写和排版等工作。

本书编写过程中得到南开大学信息学院和南开大学出版社各位领导的大力支持和帮助，责任编辑尹建国老师和李正明老师对书稿提出了许多宝贵意见，并对书中的错误和疏漏一一加以改正，他们严谨的工作作风和对读者的认真负责的精神令人敬佩，在此一并致以诚挚的感谢！

由于编者水平所限，加之时间仓促，疏漏、欠妥与谬误之处在所难免，敬请读者批评指正。

编　者

2006 年 6 月于南开园

目 录

第 1 章　软件工程技术发展与演变

软件工程（software engineering）是应用计算机科学的理论与技术以及项目工程管理的原则与方法，按照预算和进度，实现满足用户要求的软件产品的定义、开发、发布和维护的工程化方法或以之为研究对象的学科。软件工程技术就是研究和应用如何以系统化、规范化、可度量的方法开发、运行和维护软件的一种层次化技术，包括过程、方法和工具三个要素。学习和掌握软件工程技术的基本要点、基础知识及其理论和方法，是软件开发者学会如何欣赏他人的产品优势、增强团队合作意识与能力、设计开发高质量软件系统所必备的素质。

本章目的

- 理解软件的基本概念和特点
- 了解软件的发展过程及软件开发过程
- 了解软件开发的方法
- 理解面向对象技术的基本概念及开发过程
- 了解几种典型的面向对象方法

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。到了 20 世纪 80 年代，计算机及其软件向世界展示了一幅全新的认知画面，人们能够通过它看到未来世界的行为方式，充分显示出了软件工程的巨大推动作用。软件工程作为工程学科家族中的新成员，它指导人们科学地开发软件、制作软件新产品、集成计算机系统，以期降低成本、提高质量、符合进度要求，如果说软件是计算机及其应用的灵魂的话，那么可以毫不夸张地说，软件工程是计算机和信息产业的支柱，或不可分割的组成部分，其重要性于此可见一斑。

1.1　软件的概念、特点和分类

计算机科学发展的每一次进步几乎都在软件设计和程序设计语言中得到体现。软件一直是一个发展的概念。从 20 世纪 40 年代人们在 ENIAC 计算机上开始编制程序，到有人提出软件工程术语的二十多年的时间里，大家对软件开发的理解就是编程序。而且编程是在一种无序的、崇尚或表现个人技巧的状态中完成的。在高级语言出现以前，汇编语言（机器语言）是编程的工具，表达软件模型的基本概念（或语言构造）是指令，表达模型处理逻辑的主要概念（机制）是顺序和转移。显然，这一工具的抽象层次是比较低级的。随着诸如 FORTRAN

语言、PASCAL 语言、C 语言等高级语言的出现，软件开发人员开始使用变量、标识符、表达式等概念作为语言的基本构造元素，并使用“顺序、选择、循环”3 种基本控制结构来表达软件模型的计算逻辑，软件开发人员可以在一个更高的抽象层次上进行程序设计。随后出现了一系列开发范型和结构化程序设计技术，实现了模块化的数据抽象和过程抽象，提高了人们表达客观世界的抽象层次，并使开发的软件具有一定的构造性和演化性。

近 20 年来，面向对象程序设计语言的诞生并逐步流行，为人们提供了一种以对象为基本计算单元，以消息传递为基本交互手段的软件模型。面向对象方法的实质是以拟人化的观点来看待客观世界，即客观世界是由一系列对象构成，这些对象之间的交互形成了客观世界中各式各样的系统。面向对象方法中的概念和处理逻辑更接近人们解决计算问题的思维模式，使开发的软件具有更好的构造性和演化性。目前，人们更加关注软件复用问题，关注构建比对象粒度更大、更易于复用的基本单元—构件，并研究以构件复用为基础的软件构造方法，更好地凸现软件的构造性和演化性。易于复用的软件就是具有很好构造性和演化性的软件。

那么，软件的定义应该是什么呢？概括来讲，软件就是对客观世界中问题空间与解空间的具体描述，是客观事物的一种反映，是知识的提炼和“固化”。形象一点说，软件就是程序以及开发、使用、维护程序所需要的所有文档，即：软件=程序+文档。由于客观世界是不断变化的，因此，构造性和演化性是软件的本质特征。那么如何使软件模型具有更强的表达能力、更符合人类的思维模式，也即如何提升计算环境的抽象层次，在一定意义上来讲，就是围绕软件的本质特征——构造性和演化性实施的问题。详细地讲，今天的软件开发与过去相比具有以下特点：

1．软件规模相对较大。过去，由于硬件发展水平的限制，只有少数专业人员关心了解软件，而且这些专业人员必须懂得硬件，才能根据硬件的性能编写一些复杂、单一的应用软件。硬件从根本上阻碍了软件的应用，限制了软件的规模。现今，软件已逐渐走向成熟，也成为了一种系统的工程技术，可以完成一些大规模或者超大规模的项目工程。

2．编程逐步规范并趋于标准化。以往大部分软件开发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唯一因素就是编程人员的素质，即个人的经验、智慧和技巧。如今，一个规范化、标准化的编程更加注重团队的合作开发，而且相互可以比较容易地整合成为一个整体。

3．软件开发方法多，有大量的软件工具支持。计算机刚刚发展的时候，由于缺少有效的软件开发方法和软件工具支持，使人们错误地认为编程仅仅是一门艺术，其实并非如此。

4．更加重视软件开发的管理。在传统的软件开发中，由于人们重视个人的技能，而且软件开发的过程能见度低，许多管理人员根本搞不清楚软件技术人员的工作进展情况，造成管理活动很少甚至不存在。而今天，这种情况在软件开发机构中几乎看不到，取而代之的是大家极为重视软件设计、开发、运行等各个环节的管理工作。

5．软件维护相对过去较为容易。

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。

在某种程度上，软件产品在某些方面有些相似于其他工程中的有形产品，如桥梁、建筑物、机床、计算机硬件等，但其间也确有一些重要的差别，从而不能简单地把一般工程方面

的知识、方法和技术直接应用到软件工程上来。软件工程技术无时无刻都在发展，至今还不能说其已经完全成熟。软件是逻辑产品而不是实物产品，像磁盘、集成电路块等只是软件的载体。由于软件是逻辑产品，使得它的功能只能依赖于硬件的运行环境以及人们对它的操作，才能得以体现。没有计算机相关硬件的支持，软件难有实用价值。同样，没有软件支持的计算机硬件，也会是毫无价值的机器。软件与硬件密切相关的程度是其他工程所没有的。一般情况下，软件产品要完成的功能多种多样，以使用户获得清晰、准确的结果，而且软件要具备较强的可靠性、易移植性、易使用性。

以上讨论的是今天的软件区别于以往软件所具备的特征。那么，究竟软件有哪些类型呢？事实上，要给计算机软件做出科学的分类是很难的，但鉴于不同类型的工程对象，对其进行开发和维护有着不同的要求和处理方法，因此仍然需要对软件的类型进行必要的划分。在此我们从各个不同的角度对其分类，比较符合实际情况。

1. 按软件的功能划分

（1）系统软件：能与计算机硬件紧密配合在一起，使计算机系统各个部件、相关的软件和数据协调、高效地工作的软件。例如，操作系统、设备驱动程序以及通信处理程序等。

（2）支撑软件：是协助用户开发软件的工具性软件，其中包括帮助程序人员开发软件产品的工具，也包括帮助管理人员控制开发进程的工具。

（3）应用软件：是在特定领域内开发、为特定目的服务的一类软件。应用软件的种类繁多，涉及范围也很广。如商业数据处理软件、工程与科学计算软件、事务管理与办公自动化软件、中文信息处理软件、计算机辅助教学软件等，甚至有系统仿真软件、人工智能软件等。

2. 按软件的规模划分

（1）微型：一个人在一周之内编制完成的软件，程序语句不超过 5 百行，一般仅供个人专用。

（2）小型：大约 2 千行程序语句，如数值计算问题或数据处理问题。

（3）中型：一般由一个开发小组花费一年多时间完成在 5 千行至 5 万行之间的程序。这种软件需要开发人员之间、开发人员与用户之间频繁联系、密切协调，因而要配备必要的计划、资料以及技术审查文档等。

（4）大型甚至超大型：数个研发小组，甚至是成百上千人组成的开发团队，利用 3 到 5 年甚至更长时间才能完成的软件项目。这种软件往往需要进行二级开发，或者将项目划分为若干个子项目，每个子项目都是一个中型以上的软件。子项目之间具有复杂的接口，而且必须具备完备、严密的审查环节，采用统一的标准。例如编译程序、分时系统、实时控制系统以及实时处理系统、多任务系统、大型数据库管理系统等。

3. 按软件工作方式划分

（1）实时处理软件：指在事件或数据产生时，立即予以处理，并及时反馈信号，对过程需要进行监测和控制的软件。包括数据采集、分析、输出三部分，它对响应时间有严格的限定。

（2）分时软件：允许多个联机用户同时使用计算机。系统把处理机时间轮流分配给各联机用户，使各用户都感到只是自己在使用计算机的软件。

（3）交互式软件：能实现人机通信的软件。就是接收用户输入的信息，主要用于终端设备，实现人机交互。

（4）批处理软件：把一组输入作业或一批数据以成批处理的方式一次运行，按顺序逐个处理作业与数据的软件。

4. 按软件服务对象的范围划分

（1）项目软件：受某个特定客户的委托，由开发机构在合同的约束下开发出来的软件。这种软件带有试验性研究性质，一般需要软件开发机构拥有良好的质量管理手段、软件技术实力、项目开发经验以及信誉等。

（2）产品软件：是由软件开发机构开发出来直接提供给市场，或是为行业用户服务的软件。例如，文字处理软件、游戏软件、财务软件、人事管理软件等。

5. 按使用的频度划分

（1）低频次使用软件：如人口普查、工业普查软件以及世界级大型运动会、展览会软件若干年才使用一次，属于低频次使用软件。

（2）高频次使用软件：如天气预报软件可能天天使用，则属于使用频度较高的软件。

6. 按软件失效的影响程度划分

（1）一般性软件：如有的软件因发生故障而造成软件失效，但它对整个系统带来的影响不大，这就属于一般性软件。

（2）关键性软件：有的软件，如使用于财务金融、国防军工、航空航天等领域中，可靠性要求高，一旦失效将会造成灾难性的后果，这类软件就属于关键性软件。

1.2 软件的发展与软件工程

计算机软件的发展过程伴随着计算机科学的发展进程而不断发展完善，但由于软件危机的产生，促使人们对软件及其特性和软件设计方法进行更深入的研究，逐步形成了软件工程学科。同时，结构化程序设计法、快速原型法、面向对象设计法等软件设计方法相继产生和发展，加快了软件工程的建设，减少了软件危机所带来的影响。

事实上，早期的软件发展之路是异常艰难的，所设计的软件产品经常发生错误。1962 年 6 月，美国飞向金星的第一个空间探测器水手 1 号，因飞行舱中计算机导航程序中的一条语句的语义出错，致使其偏离航线，导致整个计划失败。阿波罗 8 号太空飞船的一个计算机软件错误，造成存储器的一部分信息丢失；而阿波罗 14 号在整个飞行过程中，出现了 18 个软件错误。可以说，软件系统的可靠性几乎得不到保证。另外，在当时，计算机硬件成本每隔 2～3 年降低一半，内存和外存的成本每年降低 40%左右，硬件性能价格比每十年提高一个数量级，但所需的软件很少能在开发成本、时间进度、功能规模、维护能力等方面达到要求，特别是可靠性难以符合人们的需要。那时在开发一些大型软件系统上，遇到的困难是相当多的，有些系统甚至彻底失败；有些系统虽然完成了，但比原定计划推迟了好几年，而且费用大大超过了预算；有些系统未能圆满地符合用户当初的期望；有些系统则无法进行修改维护。例如 IBM 公司的 OS/360 系统和美国空军后勤系统都花费了几千人年的努力，但最终失败了，结果令人失望。

究其原因，大型软件系统由于极大地增加了软件复杂性，使技术复杂性和管理复杂性呈指数上升。就拿软件维护来说，也要比计算机硬件来得复杂。美国 Dalg 研究了一个 16 万条汇编语句与 17 万个逻辑门组成的大型实时系统的记录，他发现：