中国计算机学会教育专业委员会 推荐
全国高等学校计算机教育研究会 出版
高等学校规划教材

软件工程
——方法与实践

许家珆 主编

软件工程课程群

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

高等学校规划教材

软件工程——方法与实践

许家珆　主编
白忠建　彭德中　吴磊　编

電子工業出版社
Publishing House of Electronics Industry
北京·BEIJING

内 容 简 介

软件工程作为一门指导计算机软件系统开发和维护的工程学科，近年来随着我国信息化建设的深入发展，对软件产业的支撑作用凸现。

本书是在吸取了国内外有关教材的精华，并结合编者多年进行软件工程教学及软件开发的实践经验、体会的基础上编写的。内容注重科学性、先进性，强调实践性。重点介绍面向对象的方法及 UML 统一建模语言，以及 CMM 软件成熟度模型、ERP 企业资源规划等先进管理技术。

本书可作为高等院校计算机及信息类专业“软件工程”课程的教材，也可作为广大工程技术人员和科研人员的参考书。

图书在版编目（CIP）数据

软件工程——方法与实践/许家珆主编.—北京：电子工业出版社，2007.9
（高等学校规划教材）
ISBN 978-7-121-04956-9

Ⅰ. 软… Ⅱ. ①许… ②白… ③彭… Ⅲ. 软件工程-高等学校-教材 Ⅳ. TP311.5

中国版本图书馆 CIP 数据核字（2007）第 135913 号

责任编辑：韩同平　　特约编辑：杨逢仪
印　　刷：北京牛山世兴印刷厂
装　　订：
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编：100036
开　　本：787×1092　1/16　印张：20.25　字数：518.4 千字
印　　次：2007 年 9 月第 1 次印刷
印　　数：4000 册　　定价：28.50 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

前　言

软件是信息化的核心，软件产业是增长最快的朝阳产业，是高投入/高产出、无污染、低能耗的绿色产业。软件产业关系到我国经济发展和文化安全，体现了国家的综合实力，是决定21世纪国际竞争地位的战略性产业。

随着计算机技术和Internet的日益普及和广泛应用，软件系统的规模和复杂度与日俱增，软件技术面临着新的挑战。大型复杂软件系统的开发是一项特殊的工程，不仅需要按照一般工程化的方法去组织管理，而且软件开发和管理更具特殊性和复杂性。软件工程作为一门指导计算机软件系统开发和维护的工程学科，近年对软件产业的支撑作用凸现。

为使我国的软件产业能尽快在国际竞争中占领一席之地，必须提高软件产业的管理水平和软件工程的应用水平，培养一大批复合型的软件人才。

本教材正是基于这样的目的而编写的，是对作者20多年来从事“软件工程”课程教学和软件开发的实践经验的总结，也是对我校的“软件工程”课程评为教育部优秀的“软件工程网络课程”、教育部-微软精品课程、四川省精品课程的总结。

全书共分13章，在系统介绍软件工程的基本内容、开发及管理技术的基础上，重点介绍面向对象的方法及UML统一建模语言，以及CMM软件成熟度模型、ERP企业资源规划等先进管理技术。内容涵盖了软件工程教育知识体系，保证了内容的科学性和先进性。

“软件工程”是一门实践性很强的课程，只有通过软件开发的实践才能真正掌握和应用软件工程的理论知识，为此，在第13章“软件工程课程设计”中，提供了一个综合性的设计型实验，以及三个软件开发的典型案例。读者在学习“软件工程”课程的同时，可选择完成一个适当的项目或者自己所承担的实际课题，以“项目驱动”促进“软件工程”课程的学习。本教材还介绍了面向对象的软件开发工具Rational Rose。

软件工程是一门处于前沿地位的重要学科，并在迅速不断的发展之中。我们希望读者通过对本书的学习，能将理论知识应用于软件开发的实践，并在实践中不断创新。

本书由许家珆教授主编，黄迪明教授主审。第1、2、4、5章由许家珆教授编写，第3、6、11章由白忠建博士编写，第7、8、10章由彭德中博士编写，第9、12、13章由吴磊博士编写。曾翎副教授对本书的编写提出的有益的意见及建议，侯晶及吴知硕士为本书提供了大量宝贵的素材，黄金艳硕士为本书绘制和修改了部分插图，在此一并感谢。

为便于广大读者自学，同时还提供电子科技大学软件工程精品课程网站，以及包括高质量的全程授课“软件工程”电子教案、中英文两个版本的“软件工程网络课件”、在线自测、电子教材、案例分析、资料查阅、在线讨论等丰富的网络资源。

软件工程精品课程网址：

教育网：　http://202.115.21.138/wlxt/ncourse/SE/web/new/default.asp

公网：　http://218.6.168.52/wlxt/ncourse/SE/web/new/default.asp

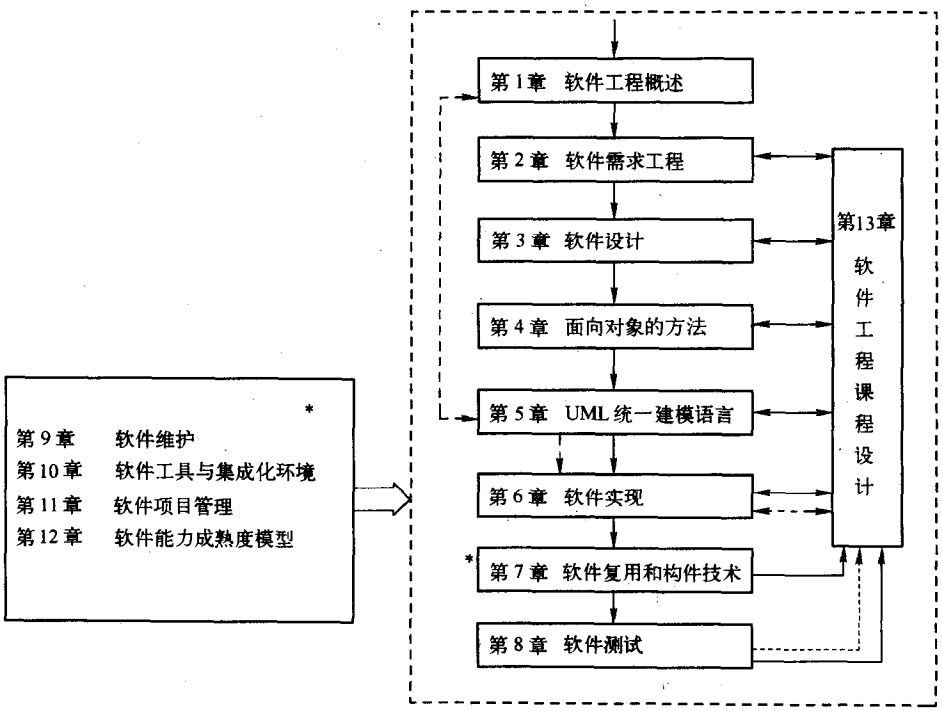
由于作者水平有限，疏漏、欠妥、谬误之处在所难免，恳请读者指正。

作　者

于电子科技大学

软件工程课程学习指导

本书的结构如下：



建议学习路径：

1. 学习第2章时，即可开始“软件工程课程设计”，课程设计与基本教学内容同步进行。上图中实线箭头表示学习顺序，双向箭头表示执行过程的交叠，即在进行课程设计时，可随时返回进行基本教学内容的学习。

2. 对软件工程方法有一定基础的读者，也可以跳过部分章节，建议按照虚线箭头的方向执行，即在学习第1章后，学习第5章，并开始课程设计。

3. 第9章至第12章的部分内容可安排自学，并在“软件工程课程设计”过程中实施。

4. 第7章为选学内容。

作者的E-mail：jiayixu@uestc.edu.cn

目　录

第 1 章　软件工程概述

1.1　软件工程的产生和发展

软件工程（Software Engineering）是在克服 20 世纪 60 年代末所出现的“软件危机”的过程中逐渐形成与发展的。自 1968 年在北大西洋公约组织（NATO）举行软件可靠性的学术会议上正式提出“软件工程”的概念以来，在不到 40 年的时间里，软件工程在理论和实践两方面都取得了长足的进步。

软件工程是一门指导计算机软件系统开发和维护的工程学科，是一门新兴的边缘学科，它涉及计算机科学、工程科学、管理科学、数学等多学科，其研究范围广，不仅包括软件系统的开发方法和技术、管理技术，还包括软件工具、环境及软件开发的规范。

软件是信息化的核心，国民经济、国防建设、社会发展及人民生活都离不开软件。软件产业关系到国家经济发展和文化安全，体现了国家综合实力，是决定 21 世纪国际竞争地位的战略性产业。因此，大力推广应用软件工程的开发技术及管理技术，提高软件工程的应用水平，对促进我国软件产业与国际接轨，推动软件产业的迅速发展将起着十分重要的作用。

1.1.1　软件工程的发展过程

软件工程的产生和发展是与软件的发展过程紧密相关的。自从第一台电子计算机诞生以来，就开始了软件的生产。从“软件工程”的概念提出至今，它的发展已经历了四个重要阶段：

（1）第一代软件工程

20 世纪 60 年代末，软件生产主要采用“生产作坊方式”。随着软件需求量、规模及复杂度的迅速增大，生产作坊的方式已不能够适应软件生产的需要，出现了所谓“软件危机（Software Crisis）”，其表现为软件生产效率低，大量质量低劣的软件涌入市场或软件在开发过程中夭折。由于“软件危机”的不断扩大，对软件生产已经产生了严重危害。

为了克服“软件危机”，在著名的 NATO（北大西洋公约组织）软件可靠性会议上第一次提出了“软件工程”这个术语，将软件开发纳入了工程化的轨道，基本形成了软件工程的概念、框架、技术和方法。这一阶段又称为传统的软件工程。

（2）第二代软件工程

20 世纪 80 年代中期，以 Smalltalk 为代表的面向对象的程序设计语言相继推出，面向对象的方法与技术得到发展；从 20 世纪 90 年代起，研究的重点从程序设计语言逐渐转移到面向对象的分析与设计，演化为一种完整的软件开发方法和系统的技术体系。20 世纪 90 年代以来，出现了许多面向对象的开发方法的流派，面向对象的方法逐渐成为软件开发的主流。所以这一阶段又称为对象工程。

（3）第三代软件工程

随着软件规模和复杂度的不断增大，开发人员也随之增多，开发周期也相应增长，加之软件是知识密集型的逻辑思维产品，这些都增加了软件工程管理的难度。人们在软件开发的

实践过程中认识到：提高软件生产效率，保证软件质量的关键是对“软件过程”的控制和管理，是软件开发和维护中的管理和支持能力。提出了对软件项目管理的计划、组织、成本估算、质量保证、软件配置管理等技术与策略，逐步形成了软件过程工程。

（4）第四代软件工程

20 世纪 90 年代起，基于构件（Component）的开发方法取得重要进展，软件系统的开发可通过使用现存的可复用构件组装完成，而无须从头开始构造，以此达到提高效率和质量，降低成本的目的。软件复用技术及构件技术的发展，对克服软件危机提供了一条有效途径，将这一阶段称为构件工程。

1.1.2 软件危机

1. 软件危机的产生

软件危机的出现是由于软件的规模越来越大，复杂度不断增加，而软件需求量也不断增大，生产作坊式的软件开发模式及技术已不能满足软件发展的需要。

软件开发过程是一种高密集度的脑力劳动，需要投入大量的人力、物力和财力；由于软件开发的模式及技术不能适应软件发展的需要，致使大量质量低劣的软件产品涌向市场，有的甚至在开发过程中就夭折了。国外在开发一些大型软件系统时，遇到了许多困难，有的系统最终彻底失败了；有的系统则比原计划推迟了好多年，而且费用大大超过了预算；有的系统不能符合用户当初的期望；有的系统则无法进行修改维护。典型的例子有：

IBM 公司的 OS/360，共约 100 万条指令，花费了 5000 个人年，经费达数亿美元，而结果却令人沮丧，错误多达 2000 个以上，系统根本无法正常运行。OS/360 系统的负责人 Brooks 这样描述开发过程的困难和混乱：“像巨兽在泥潭中作垂死挣扎，挣扎得越猛，泥浆就沾得越多，最后没有一个野兽能够逃脱淹没在泥潭中的命运……”。

1967 年前苏联“联盟一号”载人宇宙飞船，由于其软件忽略一个小数点的错误，导致返航时打不开降落伞，当进入大气层时因摩擦力太大而烧毁，造成机毁人亡的巨大损失。

还有，可以称为 20 世纪世界上最精心设计，并花费了巨额投资的美国阿波罗登月飞行计划的软件，也仍然没有避免出错。例如，阿波罗 8 号太空飞船由于计算机软件的一个错误，造成存储器的一部分信息丢失；阿波罗 14 号在飞行的 10 天中，出现了 18 个软件错误。

2. 软件危机的表现

20 世纪 60 年代末期所发生的软件危机，反映在软件可靠性没有保障、软件维护工作量大、费用不断上升、进度无法预测、成本增长无法控制、程序人员无限度地增加等各个方面，以至于形成人们难以控制软件开发的局面。

软件危机主要表现在两个方面：

① 软件产品质量低劣，甚至在开发过程中就夭折。

② 软件生产效率低，不能满足需要。软件危机所造成的严重后果已使世界各国的软件产业危机四伏，面临崩溃，克服软件危机刻不容缓。从 NATO 会议以来，世界各国的软件工作者为克服软件危机进行了许多开创性的工作，在软件工程的理论研究和工程实践两个方面都取得了长足的进步，缓解了软件危机。但距离彻底克服软件危机这个软件工程的最终目标，任重道远，还需要软件工作者付出长期艰苦的努力。

1.1.3 软件工程的定义及研究的内容

1. 软件工程的定义

自从 1968 年提出软件工程这个术语，对于软件工程就有了各种各样的定义，但是它们的基本思想都是强调在软件开发过程中应用工程化原则的重要性。

例如，1983 年，IEEE（国际电气与电子工程师协会）所下的定义是：软件工程是开发、运行、维护和修复软件的系统方法。1990 年，IEEE 又将定义更改为：对软件开发、运作、维护的系统化的、有规范的、可定量的方法之应用，即是对软件的工程化应用。

从软件工程的定义可见，软件工程是一门指导软件开发的工程学科，它以计算机理论及其他相关学科的理论为指导，采用工程化的概念、原理、技术和方法进行软件的开发和维护，把经实践证明的科学的管理措施与最先进的技术方法结合起来。软件工程研究的目标是“以较少的投资获取高质量的软件”。

2. 软件工程研究的内容

软件工程有方法、工具和过程三个要素。软件工程方法就是研究软件开发是“如何做”的；软件工具是研究支撑软件开发方法的工具，为方法的运用提供自动或者半自动的支撑环境，软件工具的集成环境，又称为计算机辅助软件工程（Computer Aided Software Engineering，CASE）；软件工程过程则是指将软件工程方法与软件工具相结合，实现合理、及时地进行软件开发的目的，为开发高质量软件规定各项任务的工作步骤。

软件工程是一门新兴的边缘学科，涉及的学科多，研究的范围广。归结起来软件工程研究的主要内容有以下 4 个方面：方法与技术、工具及环境、管理技术、标准与规范。

软件开发方法，主要讨论软件开发的各种方法及其工作模型，它包括多方面的任务，如软件系统需求分析、总体设计，以及如何构建良好的软件结构、数据结构及算法设计等，同时讨论具体实现的技术。

软件工具为软件工程方法提供支持，研究计算机辅助软件工程，建立软件工程环境。

软件工程管理，是指对软件工程全过程的控制和管理，包括计划安排、成本估算、项目管理、软件质量管理。

软件工程标准化与规范化，使得各项工作有章可循，以保证软件生产效率和软件质量的提高。软件工程标准可分为 4 个层次：国际标准、行业标准、企业规范和项目规范。

必须要强调的是，随着人们对软件系统研究的逐渐深入，软件工程所研究的内容也不是一成不变的。

3. 软件工程的基本原则

过去，软件工程的基本原则是抽象、模块化、清晰的结构、精确的设计规格说明。但今天的认识已经发生了很大的变化，现已提出的软件工程 4 条基本原则是：

① 必须认识软件需求的变动性，以便采取适当措施来保证产品能最好地满足用户要求。在软件设计中，通常要考虑模块化、抽象与信息隐蔽、局部化、一致性等原则。

② 稳妥的设计方法将大大方便软件开发，以达到软件工程的目标。软件工具与环境对软件设计的支持来说，颇为重要。

③ 软件工程项目的质量与经济开销直接取决于对它所提供的支撑质量与效用。

④ 只有在强调对软件过程进行有效管理的情况下，才能实现有效的软件工程。

近年来，印度的软件产业迅速发展，其成功的经验是，严格按照国际规范进行科学管理。在本教材中，虽然主要讨论软件开发技术，只在第 10 章讨论软件管理技术，但软件管理仍然是软件开发成功的关键，因此，将介绍当前质量管理的国际规范 CMM（软件成熟度的度量）。

1.2 软件与软件过程

软件工程是在软件生产中采用工程化的方法，并采用一系列科学的、现代化的方法和技术来开发软件的。这种工程化的思想贯穿软件开发和维护的全过程。

为了进一步学习有关软件工程的方法和技术，先介绍软件、软件生存期及软件工程过程这几个重要的概念。

1.2.1 软件的概念和特点

1. 软件及其特点

“软件就是程序，开发软件就是编写程序”是一个错误观点。这种错误观点的长期存在，影响了软件工程的正常发展。

事实上，正如 Boehm 指出的：软件是程序，以及开发、使用和维护程序所需的所有文档。它是由应用程序、系统程序、面向用户的文档及面向开发者的文档四部分构成的。

软件的特点如下。

① 软件是一种逻辑实体，不是具体的物理实体。

② 软件产品的生产主要是研制。

③ 软件具有“复杂性”，其开发和运行常受到计算机系统的限制。

④ 软件成本昂贵，其开发方式目前尚未完全摆脱手工生产方式。

⑤ 软件不存在磨损和老化问题，但存在退化问题。

图 1.1 给出了硬件的失效率曲线，它是一个“U 型”曲线（浴盆曲线），说明硬件随着使用时间的增加，失效率急剧上升。

图 1.2 所描述的软件失效率曲线，它没有“U 型”曲线的右半翼，表明软件随着使用时间的增加，失效率降低；因为软件不存在磨损和老化问题，但存在退化问题。

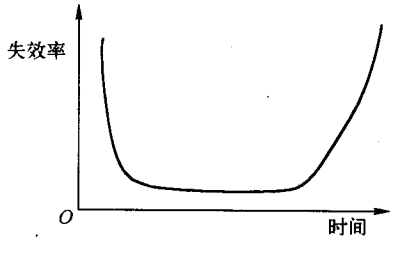


图 1.1 硬件失效率曲线

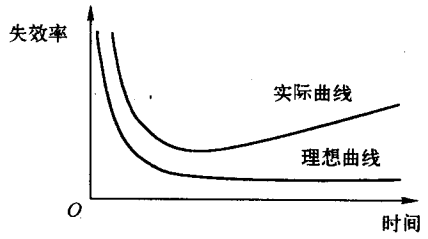


图 1.2 软件失效率曲线

2. 软件生存期

软件生命周期（SDLD）是指一个从用户需求开始，经过开发、交付使用，在使用中不断地增补修订，直至软件报废的全过程，亦称软件生存期（Life Cycle）。

GB 8567 中规定，软件生命周期分为 7 个阶段：

① 可行性研究和项目开发计划。该阶段必须要回答的问题是“要解决的问题是什么”。

② 需求分析。该阶段的任务不是具体地解决问题，而是准确地确定“软件系统必须做什么”，确定软件系统必须具备哪些功能。

③ 概要设计。概要设计就是设计软件的结构，该结构由哪些模块组成，这些模块的层次结构是怎样的，这些模块的调用关系是怎样的，每个模块的功能是什么。同时还要设计该项目的应用系统的总体数据结构和数据库结构，即应用系统要存储什么数据，这些数据是什么样的结构，它们之间有什么关系等。

④ 详细设计。即对每个模块完成的功能进行具体描述，要把功能描述变为精确的、结构化的过程描述。

⑤ 编码。该阶段把每个模块的控制结构转换成计算机可接受的程序代码，即写成以某特定程序设计语言表示的“源程序”。

⑥ 测试。它是保证软件质量的重要手段，其主要方式是在设计测试用例的基础上检验软件的各个组成部分。测试分为模块测试、组装测试、确认测试等。

⑦ 维护。软件维护是软件生存期中时间最长的阶段。已交付的软件投入正式使用后，便进入软件维护阶段，它可以持续几年甚至几十年。

在大部分文献中将生存期划分为 5 个阶段，即要求定义、设计、编码、测试及维护。其中要求定义阶段包括可行性研究和项目开发计划及需求分析，设计阶段包括概要设计和详细设计。

为了描述软件生存期的活动，提出了多种生存期模型，如瀑布模型、循环模型、螺旋模型、喷泉模型、智能模型等。

1.2.2 软件工程过程

按照 IEEE 计算机学会和 ACM 联合推出的软件工程知识体系（Software Engineering Body Knowledge，SWEBOK），将软件工程过程（Software Engineering Process）知识域（Knowledge Areas）划分为 6 个知识子域：基本概念、过程基础设施、过程度量、过程定义、定性分析和过程实施与变更。也就是说，软件工程过程是指在软件工具的支持下，所进行的一系列软件工程活动。通常包括以下 4 类基本过程：

① 软件规格说明：规定软件的功能及其运行环境。

② 软件开发：产生满足规格说明的软件。

③ 软件确认：确认软件能够完成客户提出的要求。

④ 软件演进：为满足客户的变更要求，软件必须在使用的过程中演进。

软件工程过程具有可理解性、可见性（过程的进展和结果可见）、可靠性、可支持性（易于使用 CASE 工具支持）、可维护性、可接受性（为软件工程师接受）、开发效率和健壮性（抵御外部意外错误的能力）等特性。

如图 1.3 所示，在软件工程的三要素中，软件过程将人员、方法与规范、工具和管理有机结合，形成一个能有效控制软件开发质量的运行机制。

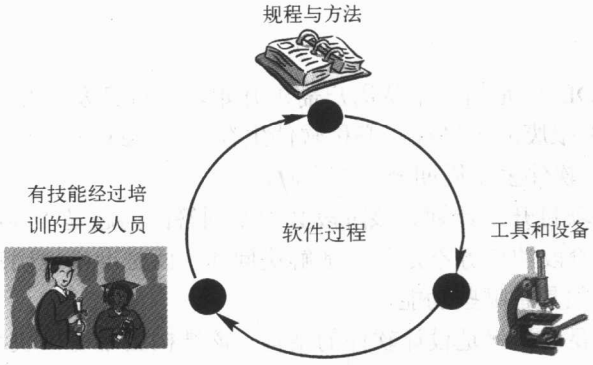


图 1.3　软件工程过程

1.3　软件过程模型

软件过程模型也称为软件生存期模型或软件开发模型，是描述软件过程中各种活动如何执行的模型。它确立了软件开发和演绎中各阶段的次序限制以及各阶段活动的准则，确立开发过程所遵守的规定和限制，便于各种活动的协调以及各种人员的有效通信，有利于活动重用和活动管理。

目前常见的软件过程模型如下。

1. 瀑布模型

瀑布模型是经典的软件开发模型，是 1970 年由 W.Royce 提出的最早的软件开发模型。如图 1.4 所示，瀑布模型将软件开发活动中的各项活动规定为依线性顺序连接的若干阶段，形如瀑布流水，最终得到软件系统或软件产品。换句话说，它将软件开发过程划分成若干个互相区别而又彼此联系的阶段，每个阶段中的工作都以上一个阶段工作的结果为依据，同时作为下一个阶段的工作基础。每个阶段的任务完成之后，产生相应的文档。该模型适合于需求很明确的软件项目开发。

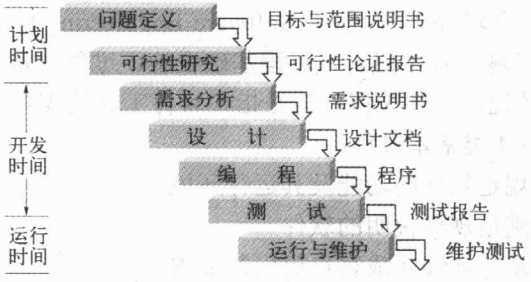


图 1.4　瀑布模型

在软件工程的第一阶段，瀑布模型得到了广泛的应用，它简单易用，在消除非结构化软件，降低软件的复杂性，促进软件开发工程化方面起了很大的作用。但在软件开发实践中也逐渐暴露出它的缺点。由于瀑布模型是一种理想的线性开发模式，它将一个充满回溯的软件开发过程硬性分割为几个阶段，无法解决软件需求不明确或者变动的问题。这些缺点对软件开发带来了严重影响，由于需求不明确，会导致开发的软件不符合用户的需求而夭折。

2. 增量模型

增量模型是一种非整体开发的模型。根据增量的方式和形式的不同，分为基于瀑布模型的渐增模型和基于原型的快速原型模型。一般的增量模型如图 1.5 所示。该模型具有较大的灵活性，适合于软件需求不明确、设计方案有一定风险的软件项目。

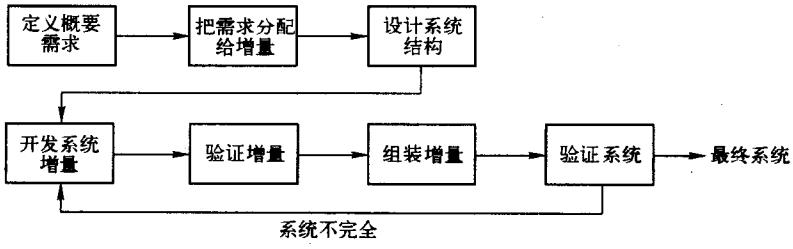


图 1.5　增量模型

增量模型和瀑布模型之间的本质区别是：瀑布模型属于整体开发模型，它规定在开始下一个阶段的工作之前，必须完成前一阶段的所有细节。而增量模型属于非整体开发模型，它推迟某些阶段或所有阶段中的细节，从而较早地产生工作软件。

3. 螺旋模型

对于大型软件，只开发一个原型往往达不到要求。螺旋模型将瀑布模型和增量模型结合起来，并加入了风险分析。它是由 TRW 公司的 B.Boehm 于 1988 年提出的。该模型将开发过程划分为制定计划、风险分析、实施工程和客户评估 4 类活动。如图 1.6 所示，沿着螺旋线

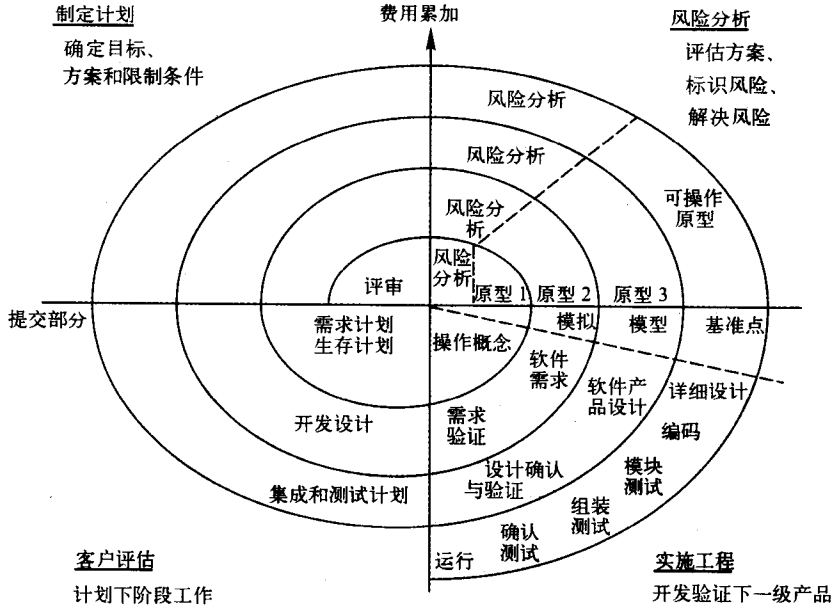


图 1.6　螺旋模型

每转一圈，表示开发出一个更完善的新的软件版本。如果开发风险过大，开发机构和客户无