



软件职业技术学院“十一五”规划教材

软件开发生命周期

与统一建模语言UML

曹静主编 罗炜 刘洁 副主编



中国水利水电出版社
www.waterpub.com.cn

软件职业技术学院“十一五”规划教材

软件开发生命周期与统一建模语言 UML

曹 静 主 编

罗 炜 刘 洁 副主编

中国水利水电出版社

内 容 提 要

统一建模语言(UML)在面向对象的软件分析与设计中起着举足轻重的作用。本书以“实用为本”为原则,在介绍软件工程基本理论之后,通过大量实例讲解软件开发生命周期的主要方法;系统讲授了统一建模语言的应用,展示了在软件开发的各个阶段中如何使用UML;示范了如何利用UML选择合适的技术以满足应用的需求;并试图通过自己的尝试,在面向对象程序设计中吸收传统结构化设计方法,走出自己的道路。

本书结合实例陈述理论,深入浅出,可以作为计算机及相关专业软件工程课程或统一建模语言课程的教材,同时也适合作为广大编程爱好者的自学参考书。

本书电子教案及实例源代码可以从中国水利水电出版社网站免费下载,网址为:<http://www.waterpub.com.cn/softdown/>。

图书在版编目(CIP)数据

软件开发生命周期与统一建模语言UML/曹静主编. —北京:
中国水利水电出版社, 2008

软件职业技术学院“十一五”规划教材

ISBN 978-7-5084-5243-2

I. 软… II. 曹… III. 软件开发—高等学校:技术学校—
教材 IV. TP311.52

中国版本图书馆CIP数据核字(2008)第003233号

书 名	软件开发生命周期与统一建模语言 UML
作 者	曹 静 主 编 罗 炜 刘 洁 副 主 编
出 版 发 行	中国水利水电出版社(北京市三里河路6号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn
经 售	电话: (010) 63202266 (总机)、68331835 (营销中心)、82562819 (万水) 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京蓝空印刷厂
规 格	787mm×1092mm 16开本 10.5印张 235千字
版 次	2008年2月第1版 2008年2月第1次印刷
印 数	0001—4000册
定 价	20.00元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换
版权所有·侵权必究

序

随着信息技术的广泛应用和互联网的迅猛发展，以信息产业发展水平为主要特征的综合国力竞争日趋激烈，软件产业作为信息产业的核心和国民经济信息化的基础，越来越受到世界各国的高度重视。中国加入世贸组织后，必须以积极的姿态，在更大范围和更深程度上参与国际合作和竞争。在这种形势下，摆在我们面前的突出问题是人才短缺，计算机应用与软件技术专业领域技能型人才的缺乏尤为突出，无论是数量还是质量，都远不能适应国内软件产业的发展和信息化建设的需要。因此，深化教育教学改革，推动高等职业教育与培训的全面发展，大力提高教学质量，是迫在眉睫的重要任务。

2000年6月，国务院发布《鼓励软件产业和集成电路产业发展的若干政策》，明确提出鼓励资金、人才等资源投向软件产业，并要求教育部门根据市场需求进一步扩大软件人才培养规模，依托高等学校、科研院所，建立一批软件人才培养基地。2002年9月，国务院办公厅转发了国务院信息化工作办公室制定的《振兴软件产业行动纲要》，该《纲要》明确提出要改善软件人才结构，大规模培养软件初级编程人员，满足软件工业化生产的需要。教育部也于2001年12月在35所大学启动了示范性软件学院的建设工作，并于2003年11月启动了试办示范性软件职业技术学院的建设工作。

示范性软件职业技术学院的建设目标是：经过几年努力，建设一批能够培养大量具有竞争能力的实用型软件职业技术人才的基地，面向就业、产学结合，为我国专科层次软件职业技术人才培养起到示范作用，并以此推动高等职业技术教育人才培养体系与管理体制和运行机制的改革。要达到这个目标，建立一套适合软件职业技术学院人才培养模式的教材体系显得尤为重要。

高职高专的教材建设已经走过了几个发展阶段，由最开始本科教材的压缩到加大实践性教学环节的比重，再到强调实践性教学环节，但是学生在学习时还是反映存在理论与实践的结合问题。为此，中国水利水电出版社在经过深入调查研究后，组织了一批长期工作在高职高专教学一线的老师，编写了这套“软件职业技术学院‘十一五’规划教材”，本套教材采用项目驱动的方法来编写，即全书所有章节都以实例作引导来说明各知识点，各章实例之间并不是孤立的，每个实例都可以作为最终项目的一个组成部分；每一章章末还配有实习实训（或叫实验），这些实训组合起来是一个完整的项目。

采用这种方式编写的图书与市场上同类教材相比更具优越性，学生不仅仅学到了知识点，还通过项目将这些知识点连成一条线，开拓了思路，掌握了知识，达到了面向岗位的职业教育培训目标。

本套教材的主要特点有：

- (1) 课程主辅分明——重点突出，教学内容实用。
- (2) 内容衔接合理——完全按项目运作所需的知识体系结构设置。

(3) 突出实习实训——重在培养学生的专业能力和实践能力,力求缩短人才与企业间的磨合期。

(4) 教材配套齐全——本套教材不仅包括教学用书,还包括实习实训材料、教学课件等,使用方便。

本套教材适用于广大计算机专业和非计算机专业的大中专院校的学生学习,也可作为有志于学习计算机软件技术与开发的工程技术人员的参考教材。

编委会

·2006年7月·

前 言

软件是脑力劳动的产品，但它不同于追求个性化的艺术作品；软件是产品，但它又不同于工业化生产得到的有形产品。如何更经济、高效地开发出高质量、可维护、可重用的软件，已成为软件业广泛关注的问题。

为克服软件缺乏“可见性”的特点，建模软件系统一般从软件过程管理、开发方式、产品构成等方面着手，借鉴工业化生产的成功经验，对软件产品的生产过程加以严格的管理和控制。

本书以此为立足点，本着“实用为本”的原则，在介绍软件工程基本理论之后，通过大量实例讲解软件开发生命周期的主要方法，系统讲授了统一建模语言的应用，并试图通过自己的尝试，在面向对象程序设计中使用统一建模语言并吸收传统结构化设计方法，走出自己的道路。

本书介绍了软件工程的基本概念、软件需求分析方法、软件生命周期模型，以及面向对象的设计思想和统一建模语言；深入浅出地向读者展示了软件系统开发的整个过程；系统地讲解了如何利用统一建模语言构建信息系统，步步深入地探究开发过程，展示了在每一步中如何使用 UML；示范了如何利用 UML 选择合适的技术以满足应用的需求。各个章节都配有大量的实例，有助于读者更加直观地理解软件开发生命周期与 UML 的理论知识。

“越早开始写代码的人，就是越迟完成代码的人”，希望读者通过对本书的阅读与学习能更加高效地获得高质量的程序。

本书由曹静任主编，罗炜、刘洁任副主编，肖菲、刘嵩、胡蓉珍、肖英、董宁、赵丙秀、陈丹、张宇、陈娜、马力参加编写，罗炜统编全稿。

由于时间仓促，加之编者水平有限，书中不妥或错误之处在所难免，殷切希望广大读者批评指正。同时，恳请读者一旦发现错误，及时与作者联系，以便尽快更正，我们将不胜感激，作者 E-mail: jingcao_2002@163.com。

编者

2007 年 12 月

目 录

序	
前言	
第 1 章 软件工程概述	1
1.1 软件、软件危机和软件工程的定义	1
1.1.1 软件及其特点	1
1.1.2 软件危机	3
1.1.3 软件工程的定义	3
1.1.4 软件工程的基本原理	3
1.2 软件开发生命周期	4
1.3 软件过程模型	7
1.3.1 常见的生命周期模型	7
1.3.2 生命周期模型的应用	10
1.4 软件工程方法学	11
本章小结	11
第 2 章 结构化程序设计与面向对象程序设计	12
2.1 概述	12
2.2 面向对象方法与结构化方法比较	12
2.2.1 实例	12
2.2.2 面向对象方法与结构化方法比较	14
2.2.3 两种方法的综合运用	16
2.3 结构化方法的基本原理	18
2.3.1 结构化分析过程	18
2.3.2 结构化分析工具	18
2.4 面向对象方法学的基本概念和原则	25
2.4.1 基本概念	25
2.4.2 主要原则	26
本章小结	27
第 3 章 统一建模语言与现代软件工程	28
3.1 软件工程的的发展趋势及成功经验	28
3.1.1 现代软件工程研究中的热点内容	28
3.1.2 现代软件工程的成功经验	30
3.1.3 统一建模语言在现代软件工程中的作用	32

3.2 统一建模语言概述.....	33
3.2.1 什么是软件建模.....	33
3.2.2 什么是UML.....	33
3.2.3 UML的历史.....	34
3.2.4 UML中的五类图.....	34
3.2.5 UML建模的基本过程.....	34
本章小结.....	36
第4章 用例图.....	37
4.1 用例图概述.....	37
4.1.1 用例建模的目的.....	37
4.1.2 定义用例图.....	37
4.1.3 用例图的主要组件.....	38
4.2 识别参与者.....	40
4.2.1 捕获需求.....	40
4.2.2 识别参与者.....	40
4.3 识别用例.....	41
4.3.1 识别用例的方法.....	41
4.3.2 用例的命名规则.....	43
4.4 用例间的关系.....	43
4.4.1 泛化关系.....	43
4.4.2 包含关系.....	44
4.4.3 扩展关系.....	45
4.5 用例文档.....	46
4.6 重构系统的用例模型.....	48
4.7 用例建模实例——书店借书系统.....	49
4.7.1 建立用例模型的步骤.....	49
4.7.2 确定系统边界.....	50
4.7.3 识别参与者.....	50
4.7.4 识别用例.....	51
4.7.5 建立用例图.....	51
4.7.6 书写用例文档.....	51
4.7.7 通过关系整理用例.....	53
本章小结.....	53
实战模拟 A 家教网上发布系统的用例模型.....	54
实战模拟 B 超市收银系统的用例模型.....	54
第5章 静态模型.....	55
5.1 静态模型的基本概述.....	55

5.2	类图	55
5.2.1	类图概述	55
5.2.2	类图的基本组件	56
5.2.3	关系	60
5.2.4	关联关系的不同重数与代码的映射	66
5.2.5	如何建模类图	67
5.3	对象图	70
5.3.1	对象图的概念	70
5.3.2	对象图和类图的区别	70
5.4	包图	70
5.4.1	包图的概念	70
5.4.2	包图建模	71
	本章小结	71
	习题	71
	案例完善 书店借书系统的初始类图模型	73
	实战模拟 A 电子办公桌网络系统的类图模型	75
	实战模拟 B 超市收银系统的类图模型	75
第 6 章	动态模型	76
6.1	动态模型概述	76
6.2	活动图	76
6.2.1	定义活动图	77
6.2.2	如何建模活动图	78
6.2.3	实例——活动图在用例模型中的作用	78
6.2.4	活动图与其他模型	80
6.3	顺序图	82
6.3.1	定义顺序图	82
6.3.2	关于消息	83
6.3.3	对象的创建和销毁	84
6.3.4	顺序图的主要用途	85
6.3.5	顺序图实例	85
6.4	协作图	89
6.4.1	定义协作图	89
6.4.2	综合实例	89
6.5	状态图	93
6.5.1	定义状态图	93
6.5.2	为什么要建模状态图	94
6.5.3	状态图映射成代码	94

6.5.4 状态图实例	95
本章小结	98
实战模拟 A 完善“电子办公桌”网络系统的类图模型	99
实战模拟 B 超市收银系统的用例模型	99
第 7 章 UML 实现与部署	101
7.1 建模实现方式图的目的	101
7.2 组件图	101
7.2.1 组件图的概念	101
7.2.2 组件图的关键技术	102
7.2.3 组件图与类图、包图的关系	104
7.2.4 购物车的组件图实现	105
7.3 部署图	107
7.3.1 部署图的概念	107
7.3.2 节点	107
7.3.3 通信关联	108
7.3.4 部署图的建模	108
7.3.5 组件图与部署图	109
7.4 建模实现方式图	110
本章小结	110
实战模拟 A 书店借书系统的 UML 实现与部署	111
实战模拟 B 家教网上发布系统的 UML 实现与部署	111
第 8 章 新闻发布系统的实例	112
8.1 系统概述	112
8.2 需求分析	112
8.2.1 系统的功能与要求	112
8.2.2 技术方案选择	113
8.2.3 系统的体系结构	113
8.3 UML 用例建模	115
8.3.1 初始用例模型	115
8.3.2 用例文档	116
8.3.3 完成的用例图	117
8.4 静态模型	117
8.4.1 初始领域类图	117
8.4.2 使用的模式	117
8.4.3 数据库设计	121
8.5 主要的动态模型	122
8.6 实现与部署	123

8.6.1	系统设计	123
8.6.2	新闻发布系统的组件图和部署图.....	124
8.7	利用 JSP 技术实现考试系统的部分关键代码.....	124
第 9 章	Rational Rose 简介	128
9.1	Rational Rose 概述.....	128
9.2	Rational Rose 的安装	128
9.2.1	安装前的准备	128
9.2.2	安装步骤	128
9.3	Rational Rose 的使用	133
9.3.1	Rational Rose 的启动.....	133
9.3.2	Rational Rose 的主界面.....	134
9.3.3	Rose 模型的 4 种视图	136
9.3.4	Rose 的基本操作	136
9.3.5	关于用例图	140
9.3.6	活动图的画法	141
9.3.7	关于类图	141
9.3.8	关于顺序图	144
9.3.9	顺序图与协作图的相互转化.....	144
9.3.10	借助协作图的工具绘制对象图.....	145
9.3.11	关于状态图.....	146
9.3.12	图形的布局	147
9.3.13	发布模型	148
9.4	Rational Rose 的逆向工程.....	149
附录	书店借书系统的需求	153
参考文献	158

第 1 章 软件工程概述

教学要求

- 理解：软件和软件工程的**概念**；软件开发生命周期。
- 掌握：软件过程模型运用的**原则**。

1.1 软件、软件危机和软件工程的定义

- 软件开发是一个专业领域的人在为另一个专业领域的人服务，但开发出来的软件往往与用户的需求有偏差，用户往往在看到最终交付的产品时才真正明确自己的需求。
- 在软件开发过程中，需求可能经常在变，原因可能是用户本身需求的变化，也可能是我们对需求的理解在发生变化，但结果是每次需求变化会带来软件系统的开发延迟，甚至出现变更反复，被推倒了的内容又要重新确立。
- 在软件没有最终交付时，用户如何了解开发进展情况。
- 系统难以维护和扩展，经常重复开发类似的功能。
- 项目经常延期，实际成本往往远远高于估计成本。
- 开发团队内部使用了不同的技术，在交流时常常有障碍。
-

如何应对软件开发过程中的种种不确定因素，更有效地开发与维护软件呢？让我们首先立足于整体，从软件工程的**角度**认识一下软件系统。

1.1.1 软件及其特点

1. 软件定义

软件是计算机系统中与硬件相互依存的另一部分，与硬件合为一体完成系统功能，如图 1-1 所示。

软件不仅仅是程序， $\text{软件} = \text{程序} + \text{数据} + \text{文档}$ 。这里的数据包括初始化数据、测试数据、研发数据、运行数据、维护数据，以及软件企业积累的项目工程数据和项目管理数据。文档是开发、使用和维护程序所需要的图文资料。

随着计算机的日益普及，软件变得越来越复杂，规模越来越庞大，为保证软件开发与维护工作的顺利进行，人与人、人与计算机之间的相互沟通就显得特别重要。因此，文档是不可缺少的，特别是在软件日益成为产品的今天，文档的作用就更加重要了。

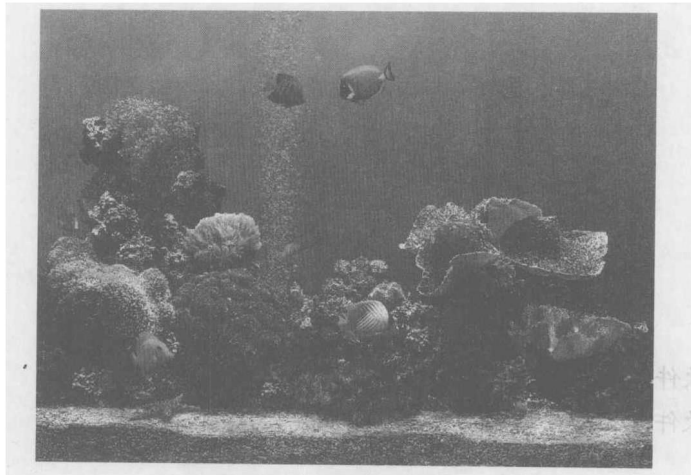


图 1-1 软件与硬件

2. 软件的特点

软件是脑力劳动的产品，但它不同于追求个性化的艺术作品；软件是产品，但它又不同于工业化生产得到的有形产品，如图 1-2 所示。软件是计算机系统中的一个逻辑部件，它的特点如表 1-1 所示。

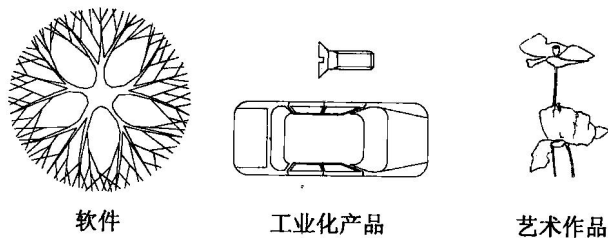


图 1-2 软件是一种特殊的产品

表 1-1 软件的特点

特点	描述
抽象性	软件是一种逻辑实体而不是具体的物理实体，必须通过测试、分析、思考、判断来了解它的功能、性能及其他特性
可复制性	软件是通过人们的智力活动，把知识与技术转化成信息的一种产品，是在研制、开发中被创造出来的。一旦某一软件项目研制成功，以后就可以大量地复制同一内容的副本，即其研制成本远远大于其生产成本
不会磨损	在软件的运行和使用期间，没有硬件那样的机械磨损、老化问题，但软件也会出现故障，软件维护比硬件维护要复杂得多，与硬件的维修有着本质的差别
依赖性	软件的开发和运行经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。为了消除这种依赖，在软件开发中提出了软件移植的问题，并且把软件的可移植性作为衡量软件质量的因素之一
开发效率低	软件的开发尚未完全摆脱手工的方式，依然有大量重复性的劳动
开发费用高	软件的研制工作需要投入大量的、复杂的、高强度的脑力劳动，需要较高的成本。软件的开发是一个复杂的过程，因而管理是软件开发过程中必不可少的内容

1.1.2 软件危机

1. 什么是软件危机

软件危机是指在计算机软件的开发和维护过程中所遇到的一系列严重问题。几乎所有软件都不同程度地存在着这些问题，它表现为多种形式，比如篇首所列举的那些。

概括地说，软件危机包含下述两方面的问题：如何开发软件，以满足对软件日益增长的需求；如何维护数量不断增加的已有软件。

2. 消除软件危机的途径

- 正确认识软件。软件=程序+数据+文档，在软件开发的各个阶段都要有完备的文档。
- 加强管理。软件开发应该是一种组织良好、管理严密、各类人员协同配合、共同完成的工程项目，应该推广使用在实践中总结出来的开发软件的成功的技术和方法。
- 使用软件工具。在软件开发的每个阶段都有许多烦琐重复的工作需要做，在适当的软件工具辅助下，开发人员可以把这类工作做得既快又好。

总之，为了解决软件危机，既要有技术措施（方法和工具），又要有必要的组织管理措施。软件工程正是从管理和技术两方面研究如何更好地开发和维护计算机软件的一门学科。

1.1.3 软件工程的定义

软件开发个性化的时代已永远成为过去，在网络、硬件等软件支持环境的迅猛发展下，软件规模不断扩大，复杂程度显著提高。如何更经济、高效地开发出高质量、可维护、可重用的软件，已成为软件业广泛关注的问题。

为了尽可能消除软件危机的影响，高效地开发出高质量的软件系统，软件工程作为一门学科应运而生，它的最终目的是实现软件的工业化生产。为克服软件缺乏“可见性”的特点，从软件过程管理、开发方式、产品构成等方面着手，借鉴工业化生产的成功经验，对软件产品的生产过程加以严格的管理和控制。

软件工程 = 管理 + 技术

软件工程是采用工程的概念、原理、技术和方法来指导计算机软件开发和维护的工程学科，该学科将正确的管理策略和最好的技术与开发方法结合起来。软件工程目标如图 1-3 所示。

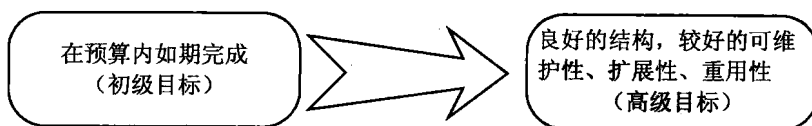


图 1-3 软件工程的目标

1.1.4 软件工程的基本原理

著名的软件工程专家 B.W.Boehm 综合众多学者的意见，于 1983 年提出了为确保软件产品质量和开发效率的 7 条基本原理，这 7 条原理至今依然具有很强的现实指导意义。

(1) 用分阶段的生命周期计划严格管理。统计发现，不成功的软件项目中有一半左右是由于计划不周造成的。因此，有必要制定完善的计划，分阶段地进行管理和控制。

(2) 坚持进行阶段评审。软件中的大部分错误是在编码之前造成的；错误发现与改正得越晚，所需付出的代价也就越高，如图 1-4 所示。因此，在每个阶段都进行严格的评审以尽早发现在软件开发过程中所犯的错误，是一条必须遵循的重要原则。

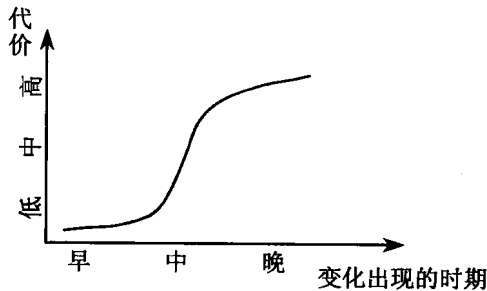


图 1-4 改正同一错误付出的代价随时间变化的趋势

(3) 实行严格的产品控制。软件开发过程中，需求的变更往往需要付出较高的代价，但这种改变又是难以避免的，因此不能硬性禁止客户提出改变需求的要求，而要依靠科学的产品控制技术来顺应这种要求，按照严格的规程进行变更控制。

(4) 采用现代程序设计技术。如“清晰第一、效率第二”的程序风格；面向对象的分析方法；各种框架技术的使用、模式的应用；软件建模方法的运用等。实践表明，采用先进的技术不仅可以提高软件开发和维护的效率，而且可以提高软件产品的质量。

(5) 结果应能清楚地审查。软件是脑力劳动的逻辑产品，应该根据软件开发项目的总目标及完成期限规定开发组织的责任和产品质量标准，制定出完备的文档，从而提高其“可见性”。

(6) 开发小组的人员应该少而精。

(7) 不断改进软件工程实践的经验和技巧。

1.2 软件开发生命周期

如何盖一栋大楼？通常要经历如图 1-5 所示的 5 个阶段。



图 1-5 大楼是怎样建成的

对于软件开发这样一个大型工程来说，同样需要划分成若干阶段以便于管理和控制。软

件开发生命周期可以划分成计划、设计、开发和运行维护 4 个时期，每个时期又进一步划分成若干阶段，如图 1-6 所示。

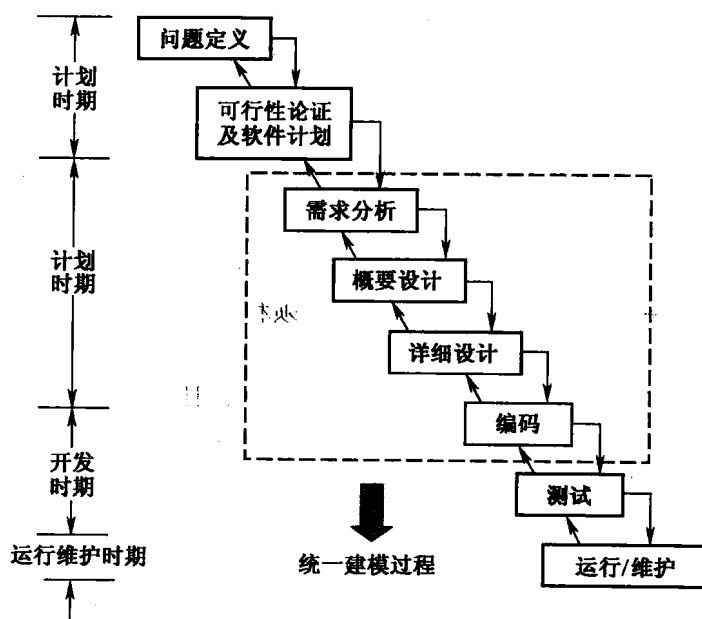


图 1-6 软件开发生命周期

下面介绍软件生命周期每个阶段的基本任务。

1. 问题定义——“要解决的问题是什么？”

通过对客户的访问调查，系统分析员扼要地写出关于问题性质、工程目标和工程规模的书面报告，经过讨论和必要的修改之后这份报告应该得到客户的确认。

2. 可行性论证及软件计划——“有行得通的解决办法吗？”

在进行任何一项较大的工程时，首先都要进行可行性分析和研究。目的就是用最小的代价在尽可能短的时间内确定该软件项目是否能够开发，是否值得去开发。

可行性研究的主要任务是：了解客户的要求及现实环境，从技术、经济和社会因素三方面研究并论证本软件项目的可行性，编写可行性研究报告，制定初步的项目开发计划。

具体步骤如下：

- (1) 确定项目规模和目标。
- (2) 研究正在运行的系统。
- (3) 建立新系统的高层逻辑模型。
- (4) 导出和评价各种方案。
- (5) 推荐可行的方案。
- (6) 编写可行性研究报告。

系统分析员需要进行一次大大压缩和简化了的系统分析和设计过程，也就是在较抽象的高层次上进行的分析和设计过程。可行性研究应该比较简短，这个阶段的任务不是具体解决问

题，而是研究问题的范围，探索这个问题是否值得去解决，是否有可行的解决办法。如果可行，制定出初步的开发计划。

可行性研究的结果是部门负责人作出是否继续进行这项工程的决定的重要依据，一般说来，只有投资可能取得较大效益的那些工程项目才值得继续进行下去。

3. 需求分析——“系统必须做什么？”

这个阶段的任务仍然不是具体地解决问题，而是确定目标系统必须具备哪些功能。

软件开发是一个专业领域的人在为另一个专业领域的人做事。用户了解他们所面对的问题，知道必须做什么，但是通常不能完整准确地表达出他们的要求，更不知道怎样利用计算机解决他们的问题；软件开发人员知道怎样用软件实现人们的要求，但是对特定用户的具体要求并不完全清楚。因此，系统分析员在需求分析阶段必须和用户密切配合，充分交流信息，以得出用户认可的各种模型。常用的有用例模型、活动图、顺序图、类图、数据流图、层次图等。

需求分析阶段的两个任务是捕获需求和分析整理需求，具体过程参见第4章。

需求分析阶段确定的系统逻辑模型是以后设计和实现目标系统的基础，因此必须完整地体现用户的要求。这个阶段的一项重要任务是，用正式文档准确地记录对目标系统的需求，即规格说明书。

4. 总体设计——“概括地说，应该怎样做？”

总体设计的基本任务是：

(1) 设计出实现目标系统的几种可能的方案。软件工程师用适当的表达工具描述每种方案，分析每种方案的优缺点，并在充分权衡各种方案的利弊的基础上推荐一个最佳方案。此外，还应该制定出实现最佳方案的详细计划。

(2) 设计软件体系结构。通常指划分模块，确定模块的功能及其相互之间的调用关系，确定模块间的接口等。

(3) 数据库设计。

(4) 编写概要设计文档。

5. 详细设计——“具体怎样做？”

总体设计阶段以比较抽象概括的方式提出了解决问题的办法，详细设计阶段的任务就是把解法具体化。

这个阶段的任务还不是编写程序，而是设计出程序的详细规格说明，这种规格说明应该包含必要的细节，程序员可以根据它们写出实际的程序代码。

6. 编码和单元测试

这个阶段的关键任务是写出正确的、容易理解、容易维护的程序模块。程序员应该根据目标系统的性质和实际环境选取一种适当的程序设计语言，把详细设计的结果翻译成用选定的语言书写的程序，并且仔细测试编写出的每一个模块。

7. 综合测试

软件测试的目的是希望以最低代价尽可能多地找出软件中潜在的各种错误和缺陷。软件测试并不是在软件交付之后才开始，而应尽早地、不断地进行，贯穿于软件定义与开发的整个期间。例如，在需求分析和设计阶段就要尽可能地考虑到如何提高软件的可测试性。