

计算机应用专业系列教材

软件工程

陈明 编

中央广播电视大学出版社

计算机应用专业系列教材

软 件 工 程

陈 明 编

中央广播电视大学出版社

图书在版编目 (CIP) 数据

软件工程/陈明编. - 北京: 中央广播电视大学出版社, 2001.8
计算机应用专业系列教材
ISBN 7-304-02049-0

I. 软… II. 陈… III. 软件工程-电视大学-教材 IV. TP311.5

中国版本图书馆 CIP 数据核字 (2001) 第 054929 号

版权所有, 翻印必究。

计算机应用专业系列教材

软 件 工 程

陈 明 编

出版·发行/中央广播电视大学出版社

经销/新华书店北京发行所

印刷/北京市银祥福利印刷厂

开本/787×1092 1/16 印张/17.5 字数/402 千字

版本/2001 年 7 月第 1 版 2003 年 1 月第 4 次印刷

印数/36001 ~ 47000

社址/北京市复兴门内大街 160 号 邮编/100031

电话/66419791 68519502

(本书如有缺页或倒装, 本社负责退换)

书号: ISBN 7-304-02049-0/TP·165

定价: 24.00 元

计算机应用专业系列教材

软 件 工 程

策 划 钱辉镜

设 计 沈雅芬 徐孝凯 何晓新

顾 问 许卓群 任为民

课程建设指导小组（按姓氏笔画排序）

陈 明（石油大学 教授）

郑纪蛟（浙江大学 教授）

侯炳辉（清华大学 教授）

高金源（北京航空航天大学 教授）

内 容 提 要

本书介绍了软件工程学及应用的有关内容，主要包括可行性研究、需求分析、概要设计、详细设计、编码、测试和维护等内容。为了保持教材内容的先进性，也包含了面向对象软件工程学方面的内容。

本书注重概念和实际应用方法的介绍，在内容的组织和选材方面，注重实用性和先进性，并表现出较强的系统性和逻辑性。本书可作为高等院校的教材，也可作为从事软件开发与应用的工程人员的参考书。

前 言

为了克服与摆脱软件危机,诞生了软件工程学。从 1968 年软件工程概念的正式提出到现在,历经 30 多年的飞速发展,涌现出大量的科研成果,软件工程学逐渐成熟,现已成为计算机科学与技术领域中的一项重要学科。软件工程学的目标是以解决软件生产的质量与效率为宗旨,研究一套科学的工程方法,以及与此相应的方便的工具系统,用来指导软件的开发研究工作,在软件产业的发展中起到重要的技术保障与促进作用。

全书分为九章,较系统地介绍了软件工程学的内容,主要包括:

- 软件工程概述
- 可行性分析
- 需求分析
- 概要设计
- 详细设计
- 面向对象的分析与设计方法
- 编码
- 软件的质量与质量保证
- 项目计划与管理

本书在内容上较为系统,注重基本概念解释和方法的说明,各章都附有内容小结、习题,可以帮助同学们很好地巩固所学的知识。

本书由殷人昆教授主审,参加审阅的还有陈永义教授、高全泉研究员,他们提出了许多有益的意见,在此表示真诚的谢意。

在本教材的编写过程中,中央广播电视大学理工部副主任沈雅芬副教授、徐孝凯副教授、何晓新老师和袁薇老师提出了许多指导性意见,在此一并表示真诚感谢。

由于作者水平有限,书中不足之处在所难免,敬请读者批评指正。

陈 明

2001 年 5 月

目 录

第一章 软件工程概述	[1]
1.1 软 件	[1]
1.1.1 软件的发展	[1]
1.1.2 软件定义	[3]
1.1.3 软件的特点	[3]
1.1.4 软件的分类	[5]
1.2 软件工程概念	[7]
1.2.1 软件危机与软件工程定义	[7]
1.2.2 软件工程的基本内容与目标	[11]
1.2.3 软件工程的原则	[12]
1.3 软件生存周期与软件开发模型	[12]
1.3.1 软件生存周期	[13]
1.3.2 软件开发模型	[14]
 第二章 可行性研究	 [21]
2.1 可行性研究的任务	[21]
2.2 可行性研究的步骤	[23]
2.3 系统流程图	[24]
2.4 成本效益分析	[26]
2.4.1 成本估计	[26]
2.4.2 费用估计	[28]
2.4.3 几种度量效益的方法	[29]
 第三章 需求分析	 [31]
3.1 需求分析的任务与步骤	[31]
3.1.1 需求分析的任务	[31]

3.1.2	需求分析的步骤	[34]
3.1.3	需求分析的原则	[36]
3.2	需求分析方法	[37]
3.2.1	面向数据流的需求分析方法	[39]
3.2.2	数据流图	[39]
3.2.3	数据字典	[47]
3.3	图形工具	[51]
3.3.1	层次方框图	[52]
3.3.2	Warnier 图	[52]
3.3.3	IPO 图	[53]
3.4	需求规格说明与评审	[53]
3.4.1	需求规格说明的主要内容	[54]
3.4.2	需求分析的评审	[55]
3.4.3	需求分析评审的方法	[56]
第四章	概要设计	[58]
4.1	概要设计的任务与步骤	[59]
4.1.1	概要设计的任务	[59]
4.1.2	概要设计的过程	[59]
4.2	软件设计的概念与原则	[61]
4.2.1	模块化	[61]
4.2.2	抽象与逐步求精	[63]
4.2.3	信息局部化	[64]
4.2.4	模块独立性	[65]
4.2.5	结构设计原则	[70]
4.3	面向数据流的设计方法	[73]
4.3.1	基本概念	[74]
4.3.2	系统结构图 (SC 图) 的组成	[76]
4.3.3	变换分析	[77]
4.3.4	事务分析	[82]
4.3.5	设计优化	[83]
4.4	面向数据结构的分析设计方法	[84]
4.4.1	Jackson 系统开发方法 (JSD)	[84]
4.4.2	Warnier 方法	[95]

4.5	概要设计文档评审	[98]
第五章	详细设计	[100]
5.1	详细设计的任务与原则	[100]
5.1.1	详细设计的任务	[100]
5.1.2	详细设计的原则	[101]
5.2	详细设计的工具与方法	[101]
5.2.1	程序流程图	[102]
5.2.2	N-S图	[104]
5.2.3	PAD图	[105]
5.2.4	PDL (过程设计语言)	[108]
5.2.5	HIPO图	[109]
5.2.6	详细设计工具的选择	[110]
5.3	详细设计规格说明与复审	[111]
5.3.1	详细设计说明书	[111]
5.3.2	设计复审	[112]
第六章	面向对象的分析与设计方法	[114]
6.1	面向对象方法的基本概念和特征	[114]
6.1.1	面向对象方法概述	[115]
6.1.2	面向对象的软件工程 (OOSE)	[116]
6.1.3	面向对象的基本概念和特征	[120]
6.2	面向对象的分析	[129]
6.2.1	面向对象分析基本过程及原则	[129]
6.2.2	确定对象、类	[134]
6.2.3	确定属性	[137]
6.2.4	定义服务	[138]
6.2.5	对象间通信	[142]
6.3	面向对象设计	[149]
6.3.1	面向对象设计的基本概念	[149]
6.3.2	面向对象设计的方法	[150]
6.4	UML方法	[153]
6.4.1	UML的发展	[154]
6.4.2	UML的表示法	[158]

6.4.3	UML 软件开发过程概述	[175]
第七章	编 码	[178]
7.1	程序设计语言	[178]
7.1.1	程序设计语言分类	[178]
7.1.2	程序设计语言的特点	[181]
7.1.3	程序设计语言的选择	[183]
7.2	编码风格	[185]
7.2.1	源程序文档化	[185]
7.2.2	数据说明	[188]
7.2.3	语句结构	[188]
7.2.4	输入/输出 (I/O)	[191]
7.3	程序效率	[191]
7.3.1	有关程序效率的几条准则	[192]
7.3.2	算法对效率的影响	[192]
7.3.3	影响存储器效率的因素	[192]
7.3.4	影响输入/输出的因素	[193]
7.4	编程安全	[193]
7.4.1	冗余程序设计	[193]
7.4.2	防错程序设计	[194]
7.5	面向对象程序设计步骤	[195]
第八章	软件质量与质量保证	[197]
8.1	软件质量的定义	[197]
8.2	影响软件质量的因素	[198]
8.3	软件质量保证策略	[199]
8.4	软件质量保证活动	[200]
8.5	软件评审	[201]
8.5.1	设计质量的评审内容	[202]
8.5.2	程序质量的评审内容	[206]
8.6	软件质量保证的标准	[209]
8.7	结构化的软件测试	[210]
8.7.1	软件测试的概念和原则	[210]
8.7.2	软件测试技术	[213]

8.7.3	测试的步骤	[220]
8.7.4	软件纠错技术	[225]
8.8	面向对象的软件测试	[227]
8.8.1	面向对象分析和面向对象设计的模型测试	[227]
8.8.2	面向对象的测试策略	[229]
8.8.3	面向对象软件测试用例设计	[230]
8.9	软件测试计划与测试分析报告	[232]
8.10	软件维护	[234]
8.10.1	软件维护的定义、分类、特点	[234]
8.10.2	软件维护步骤及组织	[237]
8.10.3	软件的可维护性	[240]
8.10.4	软件维护的副作用	[242]
8.10.5	逆向工程和再生工程	[243]
第九章	项目计划与管理	[246]
9.1	软件项目特点、管理的特殊性及软件管理功能	[246]
9.1.1	软件项目的特点	[246]
9.1.2	软件管理的功能	[247]
9.1.3	确定软件项目的工作范围	[247]
9.2	确定软硬件资源	[248]
9.2.1	人力资源	[248]
9.2.2	硬 件	[248]
9.2.3	软 件	[249]
9.3	人员的计划和组织	[249]
9.4	成本估计及控制	[251]
9.4.1	软件开发成本估计方法	[251]
9.4.2	专家判定技术	[251]
9.4.3	成本估算模型	[252]
9.5	进度计划	[255]
9.5.1	软件工作的特殊性	[255]
9.5.2	各阶段工作量的分配	[255]
9.5.3	制定开发进度	[256]
9.6	软件配置管理	[257]
9.6.1	基 线	[257]

9.6.2 软件配置项 (SCI)	[258]
9.6.3 软件配置管理 (SCM) 过程	[260]
9.7 软件管理方案	[263]

参 考 文 献	[265]
---------	-------

第一章 软件工程概述

【学习目标】

1. 掌握：软件和软件工程的基本概念
2. 了解：软件生存周期及软件开发的各模型

【学习要点】

软件、软件工程、软件生存周期和软件开发模型

1.1 软 件

软件是一种产品，同时又是开发和运行产品的载体。作为一种产品，它表达了由计算机硬件体现的计算潜能。不管它是驻留在设备中，还是在主机中，软件是一个信息转换器，能够产生、管理、获取、修改、显示或转换信息。这些信息可以很简单，如一个 bit，也可以很复杂，如多媒体信息。作为开发运行产品的载体，软件是计算机工作的基础、信息通信的基础，也是创建和控制其他程序的基础。

信息是 21 世纪最重要的产品，软件充分地体现了这一点，软件处理数据，使得这些数据更为有用；软件管理商业信息增强了商业竞争力；它不仅提供了通往全球信息网络的途径，而且也提供了以各种形式获取信息的手段。

1.1.1 软件的发展

1. 程序设计阶段

在计算机发展早期阶段（20 世纪 50 年代初期至 60 年代中期）为程序设计阶段。在这个阶段硬件已经通用化，而软件的生产却是个体化的。这时程序规模小，几乎没有什么系统化的方法可遵循。对软件的开发没有任何管理方法，一旦计划延迟了或者成本提高了，程序员才开始手忙脚乱地弥补。在通用的硬件已经非常普遍的时候，软件却相反，对每一类应用均需自行再设计，应用范围很有限。软件产品处在初级阶段，大多数软件是由使用它们的人员

或组织自己开发的，可变性很低。因为这是个人化的软件环境，所以设计往往仅是人们头脑中的一种模糊想法，而文档就根本不存在。

2. 程序系统阶段

计算机系统发展的第二阶段（20 世纪 60 年代中期到 70 年代末期）为程序系统阶段。多道程序设计、多用户系统引入了人机交互的新概念。交互技术打开了计算机应用的新世界，以及硬件和软件配合的新层次，实时系统和第一代数据库管理系统出现了。这个阶段还有一个特点就是软件产品的使用 and 软件作坊的出现。被开发的软件可以在很宽广的范围中应用，能够有数百甚至上千的用户。

当发现错误时需要纠正程序源代码；当用户需求发生变化时需要修改；当硬件环境变化时需要适应，这些活动统称为软件维护。在软件维护上所花费的精力以惊人的速度消耗资源。更为严重的是，许多程序的个性化特性使得无法维护，软件危机出现了。

3. 软件工程阶段

计算机系统发展的第三阶段始于 20 世纪 70 年代中期并跨越了近十年，被称为软件工程阶段。在这一阶段，以软件的产品化、系列化、工程化、标准化为特征的软件产业发展起来，打破了软件生产的个体化特征，有了软件工程化的设计原则、方法、标准可以遵循。在分布式系统中，各台计算机同时地执行某些功能，并与其他计算机通讯，极大地提高了计算机系统的复杂性。广域网、局域网、高带宽数字通讯以及对即时数据访问需求的增加都对软件开发提出了更高的要求。然而，软件仍然主要应用于工业界和学术界，个人应用很少。

第三阶段的主要特点是微处理器的出现和广泛应用，微处理器孕育了一系列的智能产品的出现，从汽车到微波炉，从工业机器人到血液诊断设备，但哪一个也没有个人计算机那么重要，个人计算机成为了大众化的产品。

4. 第四阶段

计算机发展的第四阶段已经不再是着重于单台计算机和计算机程序，而是面向计算机和软件的综合影响。由复杂的操作系统控制的强大的桌面机、广域网络和局域网络，配以先进的软件应用已成为标准。计算机体系结构迅速地从集中的主机环境转变为分布的客户机/服务器环境。世界范围的信息网提供了一个基本结构，“信息高速公路”和“网际空间连通”已成为令人关注的热点问题。事实上，Internet 可以看作是能够被单个用户访问的软件，计算机发展正朝着社会信息化和软件产业化方向发展，从软件工程阶段过渡到社会信息化的计算机系统。随着第四阶段的进展，一些新技术开始涌现。面向对象技术将在许多领域中迅速取代传统软件开发方法。

表 1-1 给出了四个阶段典型技术的比较。

表 1-1 四个阶段典型技术

阶段	第一阶段	第二阶段	第三阶段	第四阶段
典型技术	·面向批处理 ·有限的分布 ·自定义软件	·多用户 ·实时 ·数据库 ·软件产品	·分布式系统 ·嵌入“智能” ·低成本硬件 ·消费者的影响	·强大的桌面系统 ·面向对象技术 ·专家系统 ·人工神经网络 ·并行计算 ·网络计算机

1.1.2 软件定义

计算机系统是通过运行程序来实现各种不同的应用。把各种不同功能的程序，包括用户为自己的特定目的编写的应用程序、检查和诊断机器系统的程序、支持用户应用程序运行的系统程序、管理和控制机器系统资源的程序等通常称为软件。它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。软件定义如下：

- (1) 在运行中能提供所希望的功能和性能的指令集（即程序）；
- (2) 使程序能够正确运行的数据结构；
- (3) 描述程序研制过程、方法所用的文档。

随着计算机应用的日益普及，软件变得越来越复杂，规模也越来越大，这就使得人与人、人与机器间相互沟通，保证软件开发与维护工作的顺利进行显得特别重要，因此，文档（即各种报告、说明、手册的总称）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。

1.1.3 软件的特点

软件在整个计算机系统中是一个逻辑部件，而硬件是一个物理部件。因此，软件相对硬件而言有许多特点。为了能全面、正确地理解计算机软件及软件工程的重要性，必须了解软件的特点。软件的特点可归纳如下：

(1) 软件是一种逻辑实体，而不是具体的物理实体，因而它具有抽象性。这个特点使它与计算机硬件或其他工程对象有着明显的差别。人们可以把它记录在介质上，但却无法看到软件的形态，而必须通过观察、分析、思考、判断去了解它的功能、性能及其他特性。

(2) 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。软件故障往往是在开发时产生，而在测试时没有发现的问题。所以要保证软件的质量，必须重视软件开发过程，并加强管理。

(3) 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。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，参阅图 1-1、图 1-2 和图 1-3。图 1-1 所示的是硬件的故障率随时间变化的曲线，图 1-2 所示的是在理想情况下软件故障率随时间变化的曲线，图 1-3 所示的是软件的实际故障率曲线。

(4) 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性。在软件的开发和运行中必须以硬件提供的条件为基础。为了解除这种依赖性，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。

(5) 软件的开发至今尚未完全摆脱手工的开发方式。由于传统的手工开发方式仍然占据统治地位，软件开发的效率受到很大的限制。因此，应促进软件技术进步，提出和采用新的开发方法。例如近年来出现的充分利用现有软件的复用技术、自动生成技术和其它一些有效的软件开发工具或软件开发环境，既方便了软件开发的质量控制，还提高了软件的开发效率。

(6) 软件的开发费用越来越高，成本相当昂贵。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。

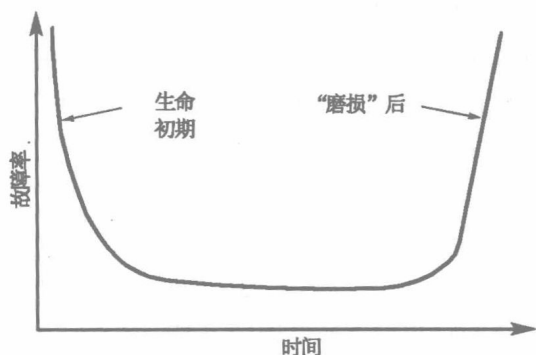


图 1-1 硬件的故障率随时间变化的曲线

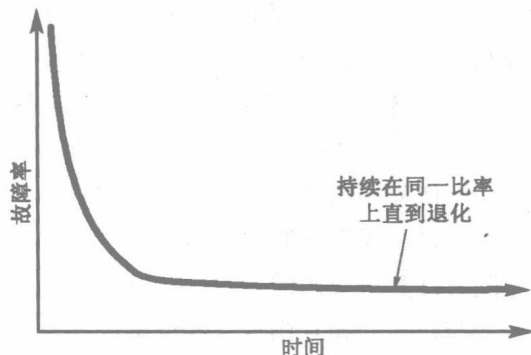


图 1-2 理想情况下的软件故障率随时间变化的曲线

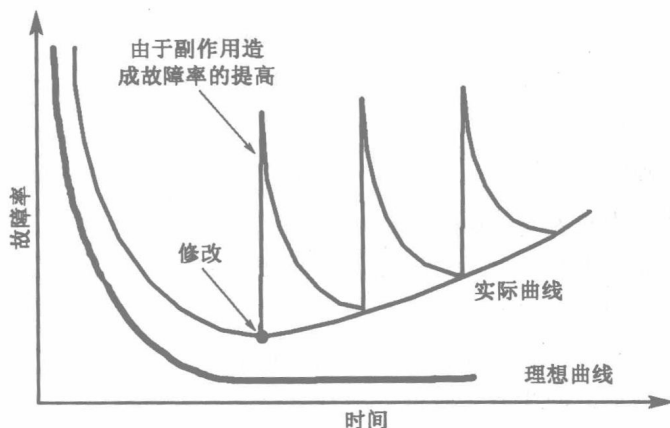


图 1-3 软件的实际故障率曲线

1.1.4 软件 的分类

1. 基于软件的功能划分

(1) 系统软件：是与计算机硬件紧密配合以使计算机各个部件与相关软件及数据协调、高效工作的软件，例如操作系统、数据库管理系统等。系统软件在工作时频繁地与硬件交往，以便为用户服务，共享系统资源，在这中间伴随着复杂的进程管理和复杂的数据结构的处理。系统软件是计算机系统必不可少的一个重要组成部分。

(2) 支撑软件：它是协助用户开发软件的工具性软件，包括帮助程序人员开发软件产品的工具，可分为：

①一般类型：包括文本编辑程序、文件格式化程序、程序库系统等。

②支持需求分析：包括 PSL/PSA 问题描述语言、问题描述分析器、关系数据库系统、一致性检验程序等。

③支持设计：包括图形软件包、结构化流程图绘图程序、设计分析程序、程序结构图编辑程序等。

④支持实现：包括编译程序、交叉编译程序、预编译程序、连接编译程序等。

⑤支持测试：包括静态分析程序、符号执行程序、模拟程序、测试覆盖检验程序等。

⑥支持管理：包括 PERT 进度计划评审方法、绘图程序、标准检验程序和库管理程序等。

(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。现在几乎所有的领域都使用了计算机，为这些计算机应用领域服务的应用软件种类繁多。其中商业数据处理软件是所占比例最大的一类，工程与科学计算软件大多属于数值计算问题。应用软件还包括计算机辅助设计/制造 (CAD/CAM)、系统仿真、智能产品嵌入软件 (如汽车油耗控制、仪表盘数字显示、刹车系统)，以及人工智能软件 (如专家系统、模式识别) 等，此外，在事务管理、办公自动化、中文信息处理、计算机辅助教学 (CAI) 等方面的软件也迅速得到发展，产生了惊人的生产效率和巨大的经济效益。

2. 基于软件工作方式划分

(1) 实时处理软件：指在事件或数据产生时立即处理，并及时反馈信号，控制需要监测和控制的过程的软件。主要包括数据采集、分析、输出三部分，其处理时间是应严格限定的，如果在任何时间超出了这一限制都将造成事故。

(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。这类软件接收用户给出的信息，但在时间上没有严格的限定，这种工作方式给予用户很大的灵活性。

(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，按顺序逐个处理的软件。