



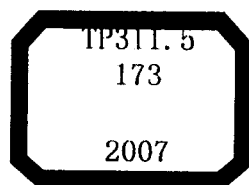
高等院校规划教材
软件工程系列

软件工程

瞿中 吴渝 刘群 刘冰 编著



机械工业出版社
CHINA MACHINE PRESS



高等院校规划教材·软件工程系列

软 件 工 程

瞿 中 吴 渝
刘 群 刘 冰 编著

机 械 工 业 出 版 社

本书从实用的角度出发,根据教育部高教司主持评审的《中国计算机科学与技术学科教程 2002》中对软件工程的要求组织编写,并参照美国 ACM 和 IEEE Computing Curricula 2001 教程关于软件工程的描述。本书吸取了国内外软件工程的精华,详细介绍了软件开发的过程,包括:软件计划、需求分析、总体设计、详细设计、编码、软件测试、软件维护、软件工程标准化和软件文档、软件工程质量、软件工程项目管理、开发实例等知识。每章配有习题,以指导读者深入地进行学习。

本书内容丰富,结构合理,既可作为高等学校计算机专业课程的教材或教学参考书,也可作为通信、电子信息、自动化等相关专业的软件工程课程教材,还可供软件工程师、软件项目管理者 and 应用软件开发人员阅读参考。

图书在版编目 (CIP) 数据

软件工程/瞿中等编著. —北京:机械工业出版社, 2007.4

(高等院校规划教材·软件工程系列)

ISBN 978-7-111-21184-6

I. 软... II. 瞿... 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 037005 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策 划: 胡毓坚

责任编辑: 张宝珠

责任印制: 李妍

保定市中画美凯印刷有限公司印刷

2007 年 4 月第 1 版·第 1 次印刷

184mm×260mm·20.25 印张·496 千字

0001—4000 册

标准书号: ISBN 978-7-111-21184-6

定价: 29.00 元

凡购本书,如有缺页,倒页,脱页,由本社发行部调换

销售服务热线电话: (010) 68326294

购书热线电话: (010) 88379639 88379641 88379643

编辑热线电话: (010) 88799739

封面无防伪标均为盗版

出版说明

计算机技术的发展极大地促进了现代科学技术的发展，明显地加快了社会发展的进程。因此，各国都非常重视计算机教育。

近年来，随着我国信息化建设的全面推进和高等教育的蓬勃发展，高等院校的计算机教育模式也在不断改革，计算机学科的课程体系和教学内容趋于更加科学和合理，计算机教材建设逐渐成熟。在“十五”期间，机械工业出版社组织出版了大量计算机教材，包括“21世纪高等院校计算机教材系列”、“21世纪重点大学规划教材”、“高等院校计算机科学与技术‘十五’规划教材”、“21世纪高等院校应用型规划教材”等，均取得了可喜成果，其中多个品种的教材被评为国家级、省部级的精品教材。

为了进一步满足计算机教育的需求，机械工业出版社策划开发了“高等院校规划教材”。这套教材是在总结我社以往计算机教材出版经验的基础上策划的，同时借鉴了其他出版社同类教材的优点，对我社已有的计算机教材资源进行整合，旨在大幅提高教材质量。我们邀请多所高校的计算机专家、教师及教务部门针对此次计算机教材建设进行了充分地研讨，达成了许多共识，并由此形成了“高等院校规划教材”的体系架构与编写原则，以保证本套教材与各高等院校的办学层次、学科设置和人才培养模式等相匹配，满足其计算机教学的需要。

本套教材包括计算机科学与技术、软件工程、网络工程、信息管理与信息系统、计算机应用技术以及计算机基础教育等教材系列。其中，计算机科学与技术系列、软件工程系列、网络工程系列和信息管理与信息系统系列是针对高校相应专业方向的课程设置而组织编写的，体系完整，讲解透彻；计算机应用技术系列是针对计算机应用类课程而组织编写的，着重培养学生利用计算机技术解决实际问题的能力；计算机基础教育系列是为大学公共基础课层面的计算机基础教学而设计的，采用通俗易懂的方法讲解计算机的基础理论、常用技术及应用。

本套教材的内容源自致力于教学与科研一线的骨干教师与资深专家的实践经验和研究成果，融合了先进的教学理念，涵盖了计算机领域的核心理论和最新的应用技术，真正在教材体系、内容和方法上做到了创新。而且本套教材根据实际需要配有电子教案、实验指导或多媒体光盘等教学资源，实现了教材的“立体化”建设。本套教材将随着计算机技术的进步和计算机应用领域的扩展而及时改版，并及时吸纳新兴课程和特色课程的教材。我们将努力把这套教材打造成为国家级或省部级精品教材，为高等院校的计算机教育提供更好的服务。

对于本套教材的组织出版工作，希望计算机教育界的专家和老师能提出宝贵的意见和建议。衷心感谢计算机教育工作者和广大读者的支持与帮助！

机械工业出版社

前 言

当今,软件工程已成为计算机科学中的一个重要分支,并且一直是一个非常活跃的研究领域。软件工程是指导计算机软件开发与维护的工程学科,它采用工程的概念、原理、技术和方法来开发与维护软件,把经过时间考验而证明正确的管理技术和当前能够得到的最好技术方法结合起来,以便经济地开发高质量的软件并有效地维护它。软件工程是计算机专业的一门非常重要的专业课程,它的研究范围非常广泛,包括技术、方法、工具和管理等许多方面。严格遵循软件工程的方法,可以大大提高软件的开发效率和成功率,减少软件开发和维护中的问题。

本书从实用角度出发,详细介绍了软件开发的全过程,包括:软件计划、需求分析、总体设计、详细设计、编码、软件测试、软件维护、软件工程标准化和软件文档、软件工程质量、软件工程项目管理、开发实例等知识。本书在编写的过程中注重理论、方法与应用相结合,不仅对软件的分析、设计、开发到维护过程进行了全面讲述,而且配有丰富的实例,每章还提供典型习题。

本书文字通俗易懂、概念清晰、深入浅出、实例丰富、实用性强,既可作为高等学校计算机专业课程的教材或教学参考书,也可作为通信、电子信息、自动化等相关专业的软件工程课程教材,还可供软件工程师、软件项目管理者 and 应用软件开发人员阅读参考。

本书的编者都是长期在高校从事软件工程教学的教师,有丰富的教学经验和科研开发能力。本书的第4、5、12章由瞿中编写;第6、10、11章由吴渝编写;第1~3章由刘群编写;第7~9章由刘冰编写。肖春雷、熊炜、范军、胡德华、杨帆、康健辉、王功孝、韩加亮以及唐衡、杨婷婷、代永亮、梁凯威、张辉、马庆伟、伍明、郭凯、刘纬地、邓军、施佳、徐丽丽、黄青蓝等参与了文字录入,并对书中的实例及图表做了大量的工作。本书的顺利出版,要感谢领导和有关老师给予的大力支持和帮助,也得到了计算机教育界许多同行的关心,在此一并致谢。

与本教材配套的电子教案和习题答案将于本书正式出版后,向使用本教材的单位与个人提供,如有需要可与机械工业出版社或编者联系。

目前,国内外有关软件工程技术与设计方面的资料很多,新理论、新技术层出不穷。由于软件工程发展迅速和编者水平有限,书中不妥和错误之处,恳请读者批评指正,并提出宝贵意见,以便进一步完善。

感谢您阅读本书。请将您的宝贵建议和意见发送至:jsjfw@mail.machineinfo.gov.cn。

作 者

目 录

出版说明

前言

第1章 概论	1
1.1 软件的概念	1
1.1.1 软件技术的发展阶段	1
1.1.2 软件的定义及特点	2
1.1.3 软件的分类	4
1.2 软件危机	6
1.2.1 软件危机的定义	6
1.2.2 产生软件危机的原因	7
1.2.3 解决软件危机的途径	8
1.3 软件工程	9
1.3.1 软件工程的定义和研究对象	9
1.3.2 软件工程的基本原理	10
1.3.3 软件工程项目的目标	11
1.3.4 软件工程的开发原则	11
1.4 软件过程和软件生存期	12
1.4.1 软件过程	12
1.4.2 软件生存周期	12
1.5 软件开发过程模型	13
1.5.1 瀑布模型	13
1.5.2 快速原型模型	15
1.5.3 螺旋模型	15
1.5.4 增量模型	17
1.5.5 喷泉模型	18
1.5.6 构件组装模型	18
1.5.7 统一过程(RUP)模型	19
1.5.8 形式化方法模型	23
1.5.9 第四代技术模型	23
1.6 软件开发方法简述	23
1.6.1 结构化方法	23
1.6.2 面向数据结构的开发方法	24
1.6.3 面向对象的方法	24
1.6.4 视觉化开发方法	26
1.7 软件工程的最新发展动向	27

1.8 小结	27
1.9 习题	28
第2章 软件计划	30
2.1 可行性研究	30
2.1.1 问题定义	30
2.1.2 可行性研究的任务	30
2.1.3 可行性研究的步骤	33
2.2 系统流程图	34
2.2.1 系统流程图的符号	34
2.2.2 系统流程图举例	35
2.2.3 分层	36
2.3 制定软件计划	36
2.3.1 确定软件计划	36
2.3.2 复审软件计划	38
2.4 成本/效益分析	39
2.4.1 成本估算技术	39
2.4.2 成本/效益分析的方法	41
2.5 小结	43
2.6 习题	43
第3章 需求分析	46
3.1 需求分析的概念和任务	46
3.1.1 需求分析的概念	46
3.1.2 需求分析的层次	46
3.1.3 需求分析的目标和任务	47
3.1.4 需求分析的原则	48
3.1.5 需求规格说明书	48
3.1.6 评审	51
3.2 获取需求的方法	53
3.2.1 存在问题	53
3.2.2 常用方法	53
3.2.3 需求分析的过程	55
3.2.4 需求分析方法	55
3.3 传统的软件建模	57
3.3.1 数据模型的建立	57
3.3.2 功能模型、行为模型的建立及数据字典	58
3.3.3 结构化分析实例	60
3.3.4 原型法分析实例	61
3.4 用例建模过程	62
3.5 面向对象建模	63

3.5.1 面向对象概述	64
3.5.2 面向对象分析建模	66
3.5.3 对象模型的建立	66
3.5.4 行为模型的建立	68
3.5.5 功能模型的建立	68
3.6 统一建模语言(UML)	69
3.6.1 UML 的基本实体	70
3.6.2 UML 目标及范畴	70
3.6.3 UML 图	71
3.7 面向对象分析实例	73
3.8 小结	76
3.9 习题	76
第4章 总体设计	79
4.1 总体设计的任务及过程	79
4.1.1 总体设计的任务	79
4.1.2 总体设计的过程	79
4.2 总体设计的原理	80
4.2.1 软件结构和过程	80
4.2.2 模块设计	81
4.2.3 结构设计	87
4.3 设计准则	89
4.4 总体设计的常用方法及工具	90
4.4.1 面向数据流的设计方法	90
4.4.2 面向对象结构的分析设计方法	94
4.5 模型—视图—控制器框架	97
4.5.1 MVC 模式	97
4.5.2 MVC 中的模型类、视图类和控制类	98
4.5.3 MVC 的实现	98
4.6 总体设计说明书编写规范	98
4.7 小结	101
4.8 习题	101
第5章 详细设计	103
5.1 详细设计阶段的任务	103
5.2 详细设计的原则	104
5.3 详细设计的方法和工具	106
5.3.1 详细设计的方法	106
5.3.2 详细设计的工具	107
5.3.3 详细设计工具的选择	114
5.4 详细设计规格说明与复审	115

5.4.1 详细设计说明	115
5.4.2 设计复审	116
5.5 面向数据结构的结构化设计方法	116
5.6 Jackson 程序设计方法	117
5.6.1 Jackson 方法的基本思想	117
5.6.2 Jackson 结构图	117
5.6.3 Jackson 方法的设计技术	117
5.7 Warnier 程序设计方法	118
5.7.1 Warnier 方法的基本思想	118
5.7.2 Warnier 方法的设计技术	118
5.8 基于组件的设计方法	118
5.9 界面设计	119
5.9.1 用户界面设计的一般原则和步骤	119
5.9.2 字符界面设计	121
5.9.3 菜单设计	122
5.9.4 对话框设计	123
5.9.5 多窗口界面设计	124
5.10 软件体系结构	125
5.10.1 软件体系结构的兴起	125
5.10.2 软件体系结构的概念	125
5.10.3 软件体系结构的现状及发展方向	126
5.10.4 软件体系结构的描述方法	129
5.11 软件体系结构与操作系统	130
5.11.1 分层结构	130
5.11.2 微内核结构	132
5.12 小结	134
5.13 习题	135
第6章 编码	137
6.1 程序设计语言	137
6.1.1 程序设计语言的发展及分类	137
6.1.2 选择程序设计语言的标准	140
6.2 程序设计风格	141
6.2.1 源程序文档化	141
6.2.2 数据说明	142
6.2.3 表达式和语句结构	143
6.2.4 输入和输出	143
6.3 程序效率	144
6.3.1 代码效率	145
6.3.2 存储器效率	145

6.3.3 输入/输出效率	145
6.4 编程安全	146
6.4.1 冗余程序设计	146
6.4.2 防错程序设计	146
6.5 程序设计方法	147
6.5.1 结构化程序设计方法	147
6.5.2 面向对象的程序设计方法	149
6.6 程序的复杂性及度量	152
6.6.1 代码行度量法	152
6.6.2 McCabe 度量法	153
6.6.3 Halstead 度量法	153
6.7 小结	154
6.8 习题	155
第7章 软件测试	157
7.1 软件测试的任务和目标	157
7.1.1 软件测试的意义	157
7.1.2 基本概念	157
7.1.3 软件测试的目的、任务、原则和研究对象	158
7.1.4 软件测试的发展历史及趋势	160
7.2 软件测试的方法	161
7.2.1 静态测试和动态测试	161
7.2.2 黑盒测试法和白盒测试法	162
7.3 软件测试的步骤	167
7.3.1 单元测试	168
7.3.2 集成测试	171
7.3.3 确认测试	176
7.3.4 系统测试	176
7.3.5 验收测试	180
7.3.6 其他测试种类	183
7.4 软件测试	185
7.4.1 软件测试角色	185
7.4.2 软件测试的需求规格说明	186
7.4.3 软件测试设计说明	191
7.5 面向对象软件测试	192
7.5.1 面向对象测试模型	192
7.5.2 面向对象分析的测试	192
7.5.3 面向对象设计的测试	192
7.5.4 面向对象编程的测试	193
7.5.5 面向对象的单元测试	193

7.5.6 面向对象的集成测试	193
7.5.7 面向对象的系统测试	194
7.6 软件测试工具	194
7.6.1 自动软件测试的优点	194
7.6.2 测试工具分类	195
7.6.3 自动测试的相关问题	197
7.6.4 常用软件测试工具及特点	198
7.7 排错	204
7.8 实例	204
7.9 小结	207
7.10 习题	208
第8章 软件维护	210
8.1 软件维护的概念	210
8.1.1 软件维护的定义	210
8.1.2 软件维护的分类	211
8.1.3 软件维护成本	213
8.2 软件维护的特点和过程	213
8.2.1 软件维护的特点	214
8.2.2 软件维护的过程	215
8.3 软件维护的步骤	217
8.4 软件的可维护性	218
8.4.1 影响软件可维护性的因素	218
8.4.2 软件的可维护性的定义	220
8.4.3 软件可维护性的度量	220
8.4.4 提高软件的可维护性方法	220
8.5 逆向工程和再工程	223
8.5.1 预防性维护	223
8.5.2 软件的逆向工程和再工程	223
8.5.3 软件再工程过程	224
8.5.4 软件再工程的方法	224
8.6 小结	225
8.7 习题	225
第9章 软件工程标准化和软件文档	228
9.1 软件工程标准化的概念	228
9.1.1 什么是软件工程标准化	228
9.1.2 软件工程标准化的意义	228
9.1.3 软件工程标准化的类型	229
9.2 软件工程标准的制定与推行	230
9.3 软件工程标准的层次和体系框架	231

9.3.1	软件工程标准的层次	231
9.3.2	软件工程过程中版本控制与变更控制处理过程	232
9.3.3	中国的软件工程标准化工作	232
9.4	ISO 9000 国际标准简介	234
9.5	软件文档	236
9.5.1	软件文档的作用和分类	236
9.5.2	对软件文档编制的质量要求	239
9.5.3	软件文档的管理和维护	240
9.6	小结	241
9.7	习题	241
第 10 章	软件工程质量	243
10.1	软件质量特性	243
10.1.1	软件质量的定义	243
10.1.2	软件质量的特性	243
10.2	软件质量的度量模型	245
10.2.1	软件度量和软件质量的度量	245
10.2.2	软件质量的度量管理模型	249
10.3	软件质量保证	252
10.3.1	什么是软件质量保证	252
10.3.2	软件质量保证的主要任务	252
10.3.3	软件质量保证策略	253
10.4	技术评审	253
10.4.1	准备评审	254
10.4.2	举行评审会议	255
10.4.3	跟踪与审核	255
10.5	软件质量管理体系	256
10.5.1	软件产品质量管理的特点	256
10.5.2	软件质量管理的指导思想	257
10.5.3	软件质量管理体系	258
10.6	小结	261
10.7	习题	261
第 11 章	软件工程项目管理	263
11.1	软件项目管理	263
11.1.1	软件项目管理的特点	263
11.1.2	软件项目管理的主要职能	263
11.2	常见管理技术及工具简介	264
11.2.1	软件项目管理的主要内容	264
11.2.2	常用工具简介	264
11.3	软件项目管理活动	266

11.3.1	计划项目	266
11.3.2	组织项目	267
11.3.3	控制项目	268
11.3.4	终结项目	269
11.4	成本估算	269
11.5	计划和组织	270
11.5.1	项目计划的制定	270
11.5.2	项目组人员管理原则	271
11.5.3	人员组织与管理	271
11.6	进度计划	273
11.6.1	制定开发进度计划	273
11.6.2	甘特图与时间管理	274
11.6.3	工程网络与关键路径	274
11.6.4	项目进度跟踪与控制	276
11.7	风险管理	278
11.7.1	风险识别与分类	278
11.7.2	风险评估与分析	281
11.7.3	风险策划与管理	282
11.7.4	风险规避与监控	283
11.8	软件成熟度模型	284
11.8.1	CMM 概述	284
11.8.2	CMM 成熟度级别	285
11.9	利用 CMM 对软件机构进行成熟度评估	287
11.10	项目管理认证体系 IPMP 与 PMP	287
11.10.1	IPMP 概述	287
11.10.2	PMP 概述	288
11.10.3	我国目前的项目管理认证体系的发展状况	289
11.11	小结	289
11.12	习题	289
第 12 章	开发实例	291
12.1	项目论证和计划	291
12.1.1	系统调查	291
12.1.2	系统的总体功能需求和性能要求	291
12.1.3	系统开发的框架	292
12.2	可行性分析	292
12.2.1	技术可行性	292
12.2.2	经济可行性	293
12.2.3	管理可行性	293
12.2.4	开发环境可行性	293

12.3 需求分析	293
12.3.1 数据流分析	293
12.3.2 系统流程图	295
12.3.3 数据字典	295
12.4 总体设计	296
12.4.1 功能模块图	296
12.4.2 层次方框图	297
12.4.3 IPO 图	297
12.4.4 系统的功能结构图	298
12.4.5 人事管理工作中其工作流程模型图	298
12.4.6 系统数据库关系说明图	298
12.5 详细设计	299
12.5.1 查询功能流程图	299
12.5.2 登录界面程序流程图	299
12.5.3 添加功能流程图	299
12.5.4 系统程序流程图	300
12.5.5 系统功能流程图	300
12.6 系统实现	300
12.6.1 实现工具	300
12.6.2 开发平台	301
12.6.3 数据库系统工作结构图	301
12.7 测试与维护	301
12.7.1 测试结果	301
12.7.2 系统维护	302
12.8 小结	304
12.9 习题	304
参考文献	307

第 1 章 概 论

本章要点

- 软件、软件工程的概念以及开发的主要原则
- 软件开发过程的模型以及开发方法
- 软件工程标准及开发文档的编制管理方法
- 软件工程发展的最新动向

1.1 软件的概念

1.1.1 软件技术的发展阶段

历史上第一个写软件的人是 Ada(Augusta Ada Lovelace),17 世纪 60 年代,她开天辟地的第一次为 Babbage(Charles Babbage)的分析机(Analytic Machine)编制程序,其中包括计算三角函数的程序、级数相乘程序、伯努利函数程序等等。Ada 编制的这些程序,即使到了今天,计算机软件界的后辈仍然不敢改动任何一条指令,人们公认她是世界上第一位软件工程师。在 20 世纪 40 年代末,软件伴随着第一台电子计算机 ENIAC(Electronic Numerical Integrator And Calculator)在美国的问世而诞生,从此以写软件为职业的人陆续开始出现,他们大多是经过训练的数学家和电子工程师。到了 20 世纪 60 年代,美国大学里开始出现专门教授人们编写软件的专业,并且对该专业毕业的大学生、研究生授予计算机专业的学位。伴随着信息产业的迅速发展,软件对人类社会的作用越来越重要,人们对软件的认识也更为深刻。

在几十年的发展过程中,软件技术主要经历了以下 4 个发展阶段。

1. 第一阶段:程序设计阶段

在计算机系统发展的早期阶段(20 世纪 50 年代初期至 20 世纪 60 年代中期),软件技术经历了它的第一个阶段,即程序设计阶段。这个阶段的软件生产以个体化为主。由于软件规模不大,几乎没有什么系统化的标准可遵循,对软件的开发也无良好的管理方法。大多数的软件由使用者自己开发、编写,然后个人应用,也很少涉及到软件文件的制作。程序设计阶段早期并没有软件的概念,开发工作主要是围绕硬件进行的。工程规模很小,所使用的工具也较为单一,开发者之间也没有明确的分工。

2. 第二阶段:程序系统阶段

当计算机发展到第二阶段(20 世纪 60 年代中期至 70 年代末期)时,软件技术发展到了程序系统阶段。多道程序设计、多用户系统引入了人机交互的新概念。这时候出现了实时系统和第一代数据库管理系统。这个阶段的另一个特点就是软件产品的使用 and 软件作坊的出现。软件的应用范围更广阔,一个程序能够有多达上百的用户。

在软件的使用中,纠正源代码错误、变更程序功能、适应硬件环境被称为软件的维护。随着计算机软件规模越来越大,应用范围越来越广,软件的维护也花费人们更多的精力和资源,

却依然没有解决许多程序个人化特性的问题。这时,人们开始有了“软件危机”感。

3. 第三阶段:软件工程阶段

从 20 世纪 70 年代中期开始,软件技术发展到了软件工程阶段,该阶段跨越了近 10 年。随着这个时候的分布式系统、高带宽数字通信系统、实时数据访问控制系统等应用技术的迅速发展,人们对计算机软件的需求变得更高,同时也使得软件开发的效率和质量成为人们关注的焦点。因此,以软件产品化、系列化、工程化、标准化为特征的软件产业迅猛发展,推动了软件工程学的进步。

4. 第四阶段

计算机系统发展的第四阶段已经不再着重于单台计算机系统和程序,而是面向计算机和软件的综合影响。Internet 和世界范围的信息网提供了一个基本架构,使得计算机体系结构迅速从集中的主机环境转变为分布式的客户机/服务器环境,由复杂的操作系统控制强大的桌面机、广域网络和局域网络,配以先进的软件应用已成为标准。计算机科学与软件技术正朝着社会信息化和软件产业化的方向发展。一些新技术的蓬勃兴起,面向对象的开发方法和其他技术方法在许多领域中表现出强大的生命力。

表 1-1 给出了 4 个阶段典型技术的比较。

表 1-1 4 个阶段典型技术比较

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、网格计算

1.1.2 软件的定义及特点

“软件”这一名词在 20 世纪 60 年代初从国外引进,当时人们无法说清它的具体含义,也无法解释它的英文单词“software”,于是有人把它翻译成“软件”或“软制品”,现在应该统一称其为软件。软件对于人类而言是一个全新的概念,其发展历史不过四、五十年,人们对软件的认识也经历了一个由浅到深的过程。在计算机系统发展的初期,硬件通常用来执行一个单一的程序,而这个程序又是为一个特定的目的而编制的。早期,当硬件的通用性成为很平常的事情时,软件的通用性却是很有限制的。大多数软件是由使用该软件的个人或机构研制的,软件往往带有强烈的个人色彩。早期的软件开发也没有什么系统的方法可以遵循,软件设计是在某个开发者的头脑中完成的。而且,软件成品除了源代码外几乎没有软件说明书等文件。现在看来,能够被普遍接受的软件的定义是:软件是计算机系统中与硬件(hardware)相互依存的另一部分,它包括程序(program)、相关数据(data)及其说明文档(document)。其中程序是按照事先设计的功能和性能要求执行的指令序列;数据是程序能正常操纵信息的数据结构;说明文档是与程序开发维护和使用有关的各种图文数据。

随着计算机应用的逐渐普及,软件变得越来越复杂,规模变得越来越大,与此同时,人与人、人与机器间的相互沟通变得越来越困难。说明文档(即各种程序文本、报告、说明、手册等)也在软件开发与维护中体现出越来越重要的价值,甚至超出了软件产品本身。因此,“软件就是程序”的观念已经落后,需要对其有新的正确认识。

软件有如下特点:

1) 软件是一种抽象的逻辑实体。人们无法看到其具体形态,只能通过观察、分析、思考、判断等方式去了解它的特性功能。

2) 软件是一种通过人们智力活动,把知识与技术转化为信息的一种产品,是在研制、开发中被创造出来的。在软件的开发过程中没有明显的制造过程,也不像硬件那样,一旦研制成功,就可以重复制造,在制造过程中进行质量控制,以保证产品质量。而一旦某一软件研制成功,以后就只是大量复制同一内容的副本,所以软件的质量主要取决于软件的开发,在软件的制造过程中几乎不会引入新的质量问题。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,所以为了保证软件质量,必须在软件开发的过程中加强管理。由于软件的复制是非常容易的,所以必须在技术上和法律上采取有力的措施,严格控制任意复制软件的行为。

3) 在软件的运行和使用期间,没有硬件那样的机器磨损、老化问题。但是软件也存在退化问题,也需要维护。软件的退化问题主要是因为在软件的生存期中,为了它能够克服之前没有发现的故障、使它能够适应硬件、软件环境的变化以及用户新的要求,必须多次修改(维护),每次修改都会引入新的不可知的错误,连续多次后会造成软件失效率升高。

图 1-1 描述了硬件与软件使用过程中产生的故障曲线。

软件退化的根本原因在于在软件维护的过程中,新的错误多在设计或编码阶段产生。因此,软件维护比硬件维护要复杂得多,与硬件维护有着本质的区别。

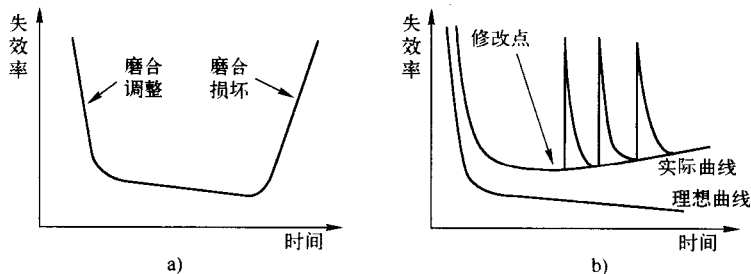


图 1-1 失效率曲线

a) 硬件失效率曲线 b) 软件失效率曲线

4) 软件的开发和运行受到计算机硬件系统的限制。在软件的开发和运行中必须以硬件提供的条件为基础,有的软件依赖于某种硬件系统,有的依赖于某种操作系统,给软件的使用造成诸多不便。为了消除这种依赖关系,在软件开发中提出了软件移植问题,并且将软件的可移植性作为衡量软件质量的一个重要因素。

5) 软件开发至今尚未摆脱手工开发方式。很多软件仍然是“定制”的,这使得软件的开发效率受到很大限制。近年来出现的软件复用技术、自动生成技术和其他一些有效的软件开发工具或软件开发环境,在一定程度上提高了软件的开发效率,但在软件项目中采用的比率较低。对软件开发工作而言,仍是一种高强度的脑力工作。

6) 软件的开发是一个复杂的过程。软件的复杂性可能来自于它所反映的实际问题的复杂度,也可能来自于程序逻辑结构的复杂度,因而管理是软件开发过程中必不可少的内容。

7) 软件的成本相当昂贵。软件开发需要投入大量的、高强度的脑力劳动,成本很高,风险也非常大。现在软件的开销已大大超过了硬件的开销。

8) 相当多软件的开发涉及到社会因素。如许多软件的开发和运行涉及到机构、体制、管