

 **NAPT** 全国应用型人才培养工程指定教材

IT技术类

软件测试技术

IT技术类教材编写组 组编

杨堃 熊应高 主编



北京航空航天大学出版社

全国应用型人才培养工程指定教材
IT 技术类

软件测试技术

IT 技术类教材编写组 组编
杨 堃 熊应高 主编

北京航空航天大学出版社

内 容 简 介

本书是最新编写的《软件测试技术》课程的指定教材。主要内容包括:软件测试概述、黑盒测试方法、白盒测试方法、软件测试过程、系统测试技术与应用、软件测试报告与测试评价、软件测试项目管理、软件测试自动化与软件测试工具、WinRunner 测试工具等。

本书既可作为高职高专院校各专业相关课程的教材,也可以供广大计算机爱好者自学或计算机培训班使用。

图书在版编目(CIP)数据

软件测试技术/杨堃等主编. —北京:北京航空航天大学出版社,2009.9

ISBN 978-7-81124-876-0

I. 软… II. 杨… III. 软件—测试—高等学校—教材
IV. TP311.5

中国版本图书馆 CIP 数据核字(2009)第 138981 号

软件测试技术

IT 技术类教材编写组 组编

杨 堃 熊应高 主编

责任编辑 杨 昕

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100083) 发行部电话:(010)82317024 传真:(010)82328026

<http://www.buaapress.com.cn> E-mail: bhpress@263.net

北京市松源印刷有限公司印装 各地书店经销

*

开本:787×1 092 1/16 印张:13 字数:333 千字

2009 年 9 月第 1 版 2009 年 9 月第 1 次印刷 印数:5 000 册

ISBN 978-7-81124-876-0 定价:24.00 元

全国应用型人才培养工程 指定教材编委会

主 任 李希来 杨建中

副 主 任 赵匡名 吴志松 李若曦

编 委 (排名不分先后)

柳淑娟 唐 琴 谭继勇 倪永康 曹晓浩 吕 俊

朱志明 连成伟 郭训成 周 扬 付开明 曹福来

吴全勇 林 岚 徐飞川 王 睿 刘国成 臧乐全

李 勇 赵丰年 王建国 杨文林 王松海 邹大民

王树理 胡志明 闫作溪 刘关宾 彭 杨 秦 柯

龚 海 潘明桓 秦绪祥 曲东涛 杨光强 王 义

陈 鹏 黄天雄 罗勇君 陈 涛 何一川 廖智科

邹雨恒 曾天意 卿平武 邹 鹏 朱 鹏 罗伟臣

王 翔 郭胜荣 吴 平 张 明 李 伟

执行编委 康 悦 孙臣英 彭卫平 黎 阳 林 军

李国胜 万 鹏 邓 波 谢 飞 张云忠

丛书前言

社会要发展,人才是关键。随着知识经济时代的到来,人才资源在经济发展中的地位 and 作用日益突出,已经成为现代经济社会发展的第一资源。目前,国内各行业对于应用型人才的需求日益迫切,无论是 IT 技术、工程制造领域,还是经济管理,甚至社会科学领域,都是如此。

全国应用型人才培养工程是由中外科教联合现代应用技术研究院组织开展的面向现代企业用人需要的人才工程。工程坚持“职业能力为导向,职业素质为核心”的课程设计原则,重点突出“职业精神、职业素质、职业能力”的培养,以提高学员的职业能力为目的,弥补技术人才与岗位要求差距,提高学员的从业竞争力,培养适应现代信息社会需要的高技能应用型人才。

全国应用型人才培养工程包括培训、测评和就业三大部分。以企业对特定岗位的实际技术要求以及对从业人员的职业精神和素质要求为依据,通过课程嵌入或者集中培训的方式解决企业在岗前培训设置方面的诸多问题。人才工程还集合各专业、各方向社会普遍认可的考核、评测体系,通过整合及学分互认等方式,实现国家认证、国际学历的有益结合;实现职业资格、职业能力、专项技能和人才资格等多种认证的有益互补;实现紧缺人才库入库、技能大赛选拔以及人才择优推荐的有益支持,从而实现始于培训、专于认证、达于就业的完整的人才培养和服务体系。

全国应用型人才培养工程培训课程包括 IT 技术类、工程制造类、经济管理类和社会科学类 4 大类,13 个专业方向,共 100 多门课程。

为了更好地配合全国应用型人才培养工程在全国的推广工作,我们专门成立了教材编写组,负责指定教材的编写工作。在编写过程中,依照人才工程所开设课程的考核标准,设定教材的编写纲目,分解知识点,选择常用经典实例,组织知识模块。

本套指定教材的特点体现在以下几个方面。

1. 行业特点

人才工程标准教材根据全国各级院校的专业教师、大中型培训机构培训师和企业相关技术人员提出的对新世纪本、专科学生培养的明确目标而设定内容,因此具备了明显的符合当前行业细分原则的侧重点与方向,更加符合企业用人的技术要求。

2. 内容侧重

人才工程主要解决当前本、专科学生所学知识内容与企业实际需要之间的差距问题;人才工程的指定教材则以企业对用人的实际技能需求为设定依据,按照“理论够用为度”的原则,对各个专业的核心课程进行了梳理整合,并以实训内容为侧重点编写。因此,本套教材不仅适用于人才工程培训,亦适用于普通的本、专科院校。

3. 编写团队

全国应用型人才培养工程教研中心负责标准教材的组织和编写工作。本套教材由教研工作经验较为丰富的专业团队负责编写,既可以解决教学实践与工程案例的接口问题,也可以有效地提高实训教材的实用性。

4. 编写流程

注重整体策划。本套教材在策划以及编写过程中,严格按照“岗位群→核心技能→知识点→课程设置→各课程应掌握的技能→各教材的内容”的编写流程,保证了教学环节内容的设定和教材的编写与当前企业的实际工作需要紧密衔接。

为了方便教学,我们免费为选择本套教材的教师提供部分专业的整体教学方案以及教学相关资料:

- ◇ 所有教材的电子教案。
- ◇ 部分教材的习题答案。
- ◇ 部分教材的实例制作过程中用到的素材。
- ◇ 部分教材中实例的制作效果以及一些源程序代码。

本套教材的编写是在教育部、中国科学院、工业和信息化部、人力资源和社会保障部众多领导和专家的支持和帮助下才顺利完成的,在此我们表示衷心的感谢。同时,我们也欢迎读者朋友们能够对于本套教材给予指正和建议。来信请发至 napt.untis@gmail.com。

全国应用型人才培养工程指定教材编委会

2009年6月

前 言

软件测试在软件开发过程中具有重要的地位,它是保证软件质量的一种重要手段。软件测试是一门需要独特技术的专门学科。软件测试并非只凭直觉,测试人员必须掌握软件测试方面的专业知识,并掌握专业的测试技术,还要具有相当多的实践经验。

随着人们对软件测试技术认识的不断加深,软件测试已经成为软件产业中的新兴门类并迅速发展起来,软件测试专业机构和组织也正在迅速发展,软件测试人才需求旺盛,测试职业的价值也日益凸显。

本书作为“全国应用型人才培养工程”指定教材之一,全面、系统地讲述了软件测试的基础知识和专门技术。全书共分9章。

第1章是软件测试概述,介绍了软件生命周期模型、软件质量保证、理解软件测试、软件测试的分类、软件测试的工作流程、软件测试人员的基本素质等内容。

第2、3章分别介绍了软件测试中经常用到的两种测试方法:黑盒测试和白盒测试。

第4章为软件测试过程,对软件测试过程中包含的单元测试、集成测试、系统测试、验收测试、回归测试、系统排错等不同阶段进行了详尽的阐述。

第5章讲述了系统测试中常用的基本技术以及测试技巧。

第6、7章分别介绍了软件测试报告与测试评价、软件测试项目管理这两方面的内容。

第8章为软件测试自动化与软件测试工具概述。

第9章结合实例,详细介绍了软件测试工具 WinRunner,便于学生在前面所学理论知识的基础上,进行一些软件测试方面的具体实践。

本书由杨堃、熊应高主编。此外,在本书编写过程中,吴洪伟、徐振成、彭小琦、史磊、陈赞等同志,在资料收集与整理方面给予了大量的帮助,在此表示衷心的感谢。

由于编写时间较为仓促,书中难免会有疏漏和不足之处,恳请广大读者提出宝贵意见。如有问题可以通过电子邮件(wooystudio@263.net)与编者联系。

编 者
2009年7月

目 录

第 1 章 软件测试概述	1
1.1 软件生命周期模型	1
1.1.1 软件生命周期	1
1.1.2 软件生命周期模型	2
1.2 软件质量保证	4
1.2.1 质量与质量模型	4
1.2.2 软件质量保证	6
1.3 理解软件测试	6
1.3.1 软件测试的定义及目的	6
1.3.2 软件测试与开发过程的关系	7
1.3.3 软件测试与质量保证的区别	8
1.4 软件测试的分类	9
1.4.1 黑盒测试和白盒测试.....	11
1.4.2 静态测试和动态测试.....	11
1.4.3 测试的不同阶段.....	11
1.4.4 测试目的和内容.....	14
1.4.5 测试的各种分类之间的关系.....	15
1.5 软件测试的工作流程.....	15
1.5.1 测试工作的主要步骤.....	15
1.5.2 测试信息流.....	16
1.6 软件测试人员的基本素质.....	17
习 题	18
第 2 章 黑盒测试方法	19
2.1 黑盒测试方法概述.....	19
2.2 边界值测试.....	20
2.2.1 边界值分析法.....	20
2.2.2 边界值分析法测试用例.....	20
2.2.3 边界值分析法测试实例.....	22
2.2.4 边界值分析法的局限性.....	23
2.3 等价类测试.....	24
2.3.1 等价类.....	24

2.3.2 等价类测试实例	26
2.3.3 指导原则	27
2.4 基于决策表的测试	27
2.5 错误推测法	33
习 题	33
第3章 白盒测试方法	34
3.1 白盒测试的基本概念	34
3.1.1 流 图	35
3.1.2 环形复杂度	36
3.1.3 图矩阵	37
3.2 程序插桩	37
3.2.1 方法简介	38
3.2.2 断言语句	40
3.3 逻辑覆盖	42
3.3.1 逻辑覆盖标准	42
3.3.2 最少测试用例数字计算	45
3.4 独立路径测试	47
3.5 循环测试	47
3.6 面向对象的白盒测试	48
3.7 其他白盒测试方法	49
习 题	51
第4章 软件测试过程	52
4.1 软件测试过程概述	52
4.2 单元测试	53
4.2.1 单元测试的主要任务	53
4.2.2 单元测试的执行过程	55
4.2.3 单元测试技术和测试数据	56
4.2.4 单元测试人员	57
4.3 集成测试	57
4.3.1 集成测试的主要任务	57
4.3.2 集成测试方法	58
4.3.3 集成测试技术和测试数据	60
4.3.4 集成测试遵循的原则	61
4.3.5 集成测试人员	61
4.4 系统测试	61
4.4.1 系统测试的任务	62
4.4.2 系统测试技术和测试数据	63
4.4.3 系统测试人员	63
4.5 验收测试	63

4.5.1	验收测试的主要任务	64
4.5.2	验收测试技术和测试数据	64
4.5.3	验收测试人员	65
4.5.4	α 、 β 测试	65
4.6	回归测试	65
4.6.1	回归测试技术和测试数据	66
4.6.2	回归测试的范围	67
4.6.3	回归测试人员	67
4.7	系统排错	68
	习 题	69
第 5 章	系统测试技术与应用	70
5.1	系统测试技术	70
5.1.1	功能测试	71
5.1.2	错误处理测试	72
5.1.3	内存泄漏测试	73
5.1.4	用户界面测试	74
5.1.5	性能测试	80
5.1.6	安全测试	81
5.1.7	压力测试	82
5.1.8	安装与卸载测试	82
5.1.9	升级测试	83
5.1.10	兼容性测试	84
5.1.11	冒烟测试	85
5.1.12	文档测试	85
5.2	测试技巧	86
5.2.1	植入缺陷	86
5.2.2	回归测试	86
5.3	Web 应用系统测试要点	87
	习 题	89
第 6 章	软件测试报告与测试评价	90
6.1	软件缺陷的概念和种类	90
6.2	正确面对软件缺陷	93
6.3	软件缺陷的生命周期	94
6.4	软件缺陷的严重性和优先级	95
6.5	报告软件缺陷	96
6.5.1	报告软件缺陷的基本原则	96
6.5.2	IEEE 软件缺陷报告模板	98
6.5.3	软件缺陷数据库跟踪系统	99
6.5.4	手工报告和跟踪软件缺陷	100

6.6	分离和再现软件缺陷	101
6.7	测试总结报告	102
6.8	测试的评测	103
6.8.1	覆盖评测	104
6.8.2	质量评测	105
6.8.3	性能评测	109
	习 题	111
第7章	软件测试项目管理	112
7.1	测试项目管理概述	112
7.1.1	测试项目与测试项目管理	112
7.1.2	测试项目的范围管理	114
7.2	测试文档	114
7.2.1	测试文档的作用	114
7.2.2	测试文档的类型	115
7.3	软件测试计划	120
7.3.1	制订测试计划的目的	120
7.3.2	制订测试计划的原则	121
7.3.3	制订测试计划时面对的问题	121
7.3.4	制订测试计划	122
7.4	测试的组织与人员管理	127
7.4.1	测试的组织与人员管理概述	127
7.4.2	测试人员的组织结构	128
7.4.3	测试人员	129
7.4.4	人员的通信方式	130
7.4.5	测试人员管理的激励机制	130
7.4.6	测试人员的培训	130
7.4.7	测试的组织与人员管理中的风险管理	131
7.5	软件测试过程管理	131
7.5.1	测试项目的跟踪与监控	131
7.5.2	测试的配置管理	132
7.5.3	软件测试风险管理	134
7.5.4	软件测试的成本管理	136
	习 题	140
第8章	软件测试自动化与软件测试工具	141
8.1	软件自动化测试概述	141
8.2	自动化测试的作用和优势	142
8.3	软件测试工具分类	148
8.4	常用软件测试工具简介	151
8.4.1	QACenter	151

8.4.2 WinRunner	152
8.4.3 LoadRunner	152
习 题.....	153
第9章 WinRunner 测试工具	154
9.1 WinRunner 简介	154
9.1.1 主界面	155
9.1.2 测试模式	157
9.1.3 测试过程	157
9.1.4 样本软件	158
9.1.5 测试套件	159
9.2 GUI Map	159
9.2.1 GUI 对象属性的查看	159
9.2.2 GUI map file 模式	161
9.3 录制测试脚本	166
9.3.1 选择录制模式	166
9.3.2 Context Sensitive 模式下录制	167
9.3.3 Analog 模式下录制	168
9.3.4 测试脚本执行	169
9.3.5 测试结果分析	170
9.3.6 录制时建议	170
9.4 同步点	171
9.5 GUI 对象检查点	174
9.6 图像检查点	176
9.7 编辑测试脚本	178
9.8 数据驱动测试脚本	181
9.9 文字检查点	185
9.10 批次测试.....	187
9.11 维护测试脚本.....	189
习 题.....	191
参考文献	192

第 1 章 软件测试概述

本章要点

- 软件生命周期模型
- 软件质量保证
- 软件测试的定义及目的
- 软件测试的分类
- 软件测试的工作流程
- 软件测试人员的基本素质

学习要求

- 了解 3 种典型的软件生命周期模型
- 理解软件测试与软件开发过程的关系
- 掌握软件测试与质量保证的区别
- 领会不同的软件测试分类方法及其区别
- 掌握软件测试的基本工作流程

1.1 软件生命周期模型

1.1.1 软件生命周期

软件测试是软件开发过程中的一个重要组成部分。当一个项目组开发一个软件时,需要按照一系列步骤来进行,这些步骤构成了软件开发过程。在软件开发过程中,每个步骤都应该有明确的输入、输出和实施方法,有时,一个步骤会被分解成若干个子步骤。

软件开发过程和一些常见的工程活动(如建筑房屋、修建公路等)是很相似的,一般来说,软件开发过程需要经历以下主要阶段:

- ① 定义。明确软件开发的目标、软件的需求。
- ② 计划。制订软件开发所涉及的各种计划。
- ③ 实现。进行设计、编码、文档编写工作,完成所要求开发的软件特性。
- ④ 稳定化。以测试和缺陷修复工作为主,确保提交的软件具有优良的品质。
- ⑤ 部署。安装、提交开发完成的软件,建立可以供用户使用的环境。

实际应用过程中,在软件开发过程经过这 5 个主要阶段之后,软件还需要经历一个使用、

维护,直到被停止使用的阶段。加上这个阶段之后,就形成了一个软件从诞生到最后被停止使用的完整周期,这个周期被称为软件生命周期。也就是说,软件生命周期是包含了定义、计划、实现、稳定化、部署、运行与维护这6个阶段的过程。

1.1.2 软件生命周期模型

从20世纪70年代开始,人们就在不断研究软件开发过程,到目前为止,已经形成了一系列过程模型,也称为软件生命周期模型。以下将介绍几种最为常见的软件生命周期模型。

1. 瀑布过程模型

瀑布过程模型反映了人们早期对软件工程的认识水平,是人们所熟悉的一种线性思维的体现。

瀑布过程模型强调阶段的划分及其顺序性、各阶段工作及其文档的完备性,是一种严格线性的、按阶段顺序的、逐步细化的开发模式,如图1-1所示。

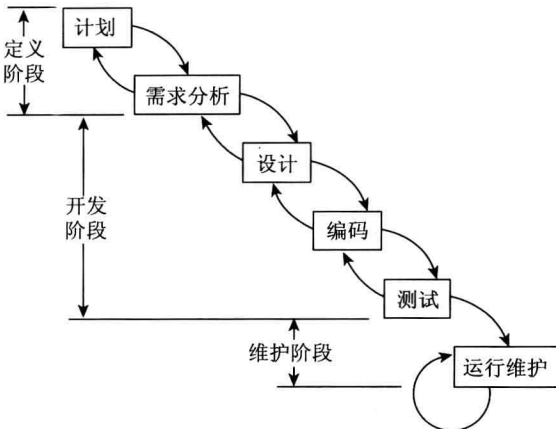


图 1-1 瀑布过程模型

该模型主要由顺序的6个阶段组成,包括计划、需求分析、设计、编码、测试和运行维护等。

(1) 计划阶段

该阶段的主要任务包括:确定软件开发的总目标;给出软件的功能、性能、可靠性以及接口等方面的设想;研究完成该项软件任务的可行性,探讨问题解决的方案;对可供开发使用的资源(如计算机硬件、软件、人力等)、成本、可取得的效益和开发的进度做出估计;制订完成开发任务的实施计划。

(2) 需求分析

所谓需求分析,是指收集产品的需求,对开发的软件进行详细的定义,由软件开发人员和用户共同讨论决定,确定哪些需求是可以满足的,并且给予确切的描述,写出软件需求说明书等文档,提交管理机构审查。

(3) 设计

设计是软件过程的技术核心。在设计阶段,应把已确定的各项需求转换成相应的体系结构,在结构中,每一组成部分是功能明确的模块,每个模块都能体现相应的需求。这一步称为总体设计。然后进行详细设计,即对某个模块要完成的工作进行具体的描述,以便为程序编写打下基础。

(4) 编码阶段

编码阶段的主要工作是编写各种层次的代码,包括由高级的可视化开发系统生成的代码和用第4代语言编写的代码等。

(5) 软件测试

软件测试,是为了发现程序中的错误而执行的过程。

(6) 运行维护

运行维护,主要是对应用进行修复和改动,使它能够持续发挥作用。

实际上,图1-1所示各阶段并不是严格按照顺序执行的,因为过程中的各部分之间都有某种程度的重叠。造成这种重叠的原因是,绝大多数应用系统都是复杂的,非线性的,上述任何一个阶段都不可能在下一阶段开始之前完全结束。实际上,人们很少使用单纯的瀑布过程模型,除非是很小的项目或者是开发的产品与自己以前做过的项目类似。

2. 螺旋过程模型

螺旋过程模型需要经历多次需求分析、设计、实现、测试。这样做有多种原因,其中主要的原因是基于规避风险的需要;另一个原因是需要在早期构造产品的局部版本时即交给客户以获得反馈;最后是为了避免像瀑布过程模型一样一次集成大量的代码。

螺旋过程模型的基本思路是,依据前一个版本的结果构造新的版本,这个不断重复迭代的过程形成了一个螺旋上升的路径,如图1-2所示。

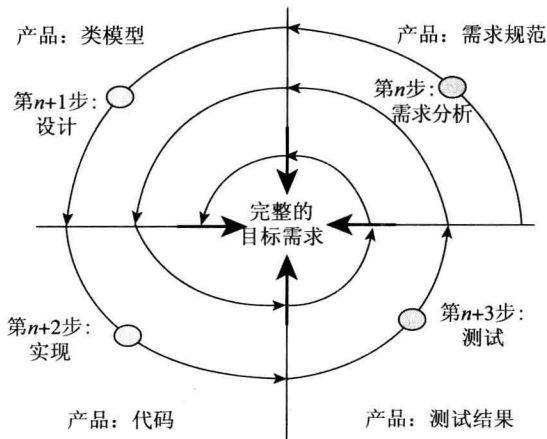


图 1-2 螺旋过程模型

螺旋过程模型的一个主要优点是,能够在每次迭代中都收集到过程中产生的各种度量数据。例如,在第一次迭代时记录下小组进行设计和实现所耗费的时间,以此即可改进后续设计,实现耗费时间的估计。这对于没有任何历史数据的开发组织尤其有用。

螺旋过程模型符合典型项目的发展特点,但是跟简单的瀑布过程模型相比,它需要投入更多的精力来更细致地管理其过程。一个最直接的原因是,每次迭代完成之后都必须保证文档的一致性。特别是代码应该实现文档中描述的设计并且满足文档中记录的需求。此外为了提高小组的生产效率,往往会在前一个迭代结束之前就开始新的一次迭代,这为协调文档的一致性增加了难度。

螺旋开发过程需要多少次迭代取决于具体情况。一个典型的工作量为3人/月、耗时4个月的项目大概需要两次或者3次迭代。项目若采用5次迭代,则所需要的管理费用通常会耗掉新增迭代所创造的价值。

当迭代的速度加快,每次迭代只是在前一次的基础上增加少量功能的时候,螺旋过程模型就演变为增量过程模型。

3. 增量过程模型

有些时候可能会用一种几乎连续的过程小幅度地推进项目,这就是增量过程模型,如图1-3所示。

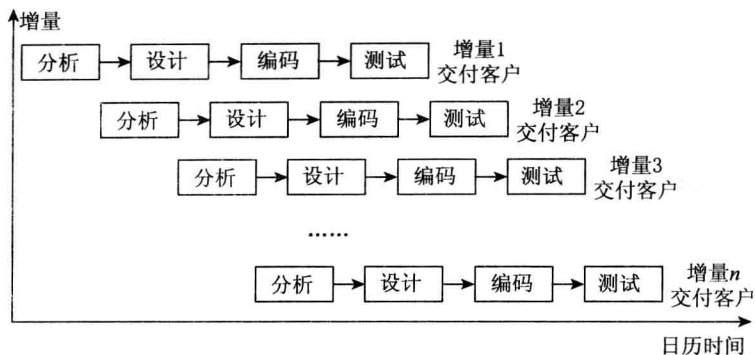


图 1-3 增量过程模型

这种过程模型在项目的后期尤其适用,例如,当项目处于维护阶段,或者立项的产品与原先开发出来的产品结构非常相似的时候。

以上 3 种模型只是众多软件生命周期模型中较为典型的,此外,还有快速原型法、喷泉模型、统一软件开发过程模型等。

软件开发机构形成一套完整而成熟的软件过程,需要一个从无序到有序、从特殊到一般、从定性到定量、从静态到动态的过程。也就是说,软件机构在形成成熟的软件过程之前,必须经历一系列的成熟阶段。因此,有必要建立一个软件能力成熟度模型来对过程作出一个客观、公正的评价,以促进软件开发组织改进软件过程。

1.2 软件质量保证

软件质量是软件的生命,它直接影响软件的使用与维护。软件开发人员和用户都十分重视软件的质量问题。软件质量通常由以下几方面进行评价:

- ① 软件需求是衡量软件质量的基础,不符合需求的软件就不具备软件质量。设计的软件应在功能、性能等方面都符合要求,并能可靠地运行。
- ② 软件结构良好,易读,易于理解,易于修改、维护。
- ③ 软件系统具有友好的用户界面,便于用户使用。
- ④ 软件生存周期中各阶段文档齐全、规范,便于配置、管理。

1.2.1 质量与质量模型

软件的质量因素很多,如正确性、精确性、可靠性、容错性、性能、效率、易用性、可理解性、简洁性、可复用性、可扩充性及兼容性等。

软件质量因素也称为软件质量特性,反映了质量的本质。讨论一个软件的质量,问题最终要归结到定义软件的质量特性。

面对众多的质量因素如何取折衷,这实际上就是区分质量因素对软件质量影响程度轻重的问题,这个问题已经有了解决方案,即软件质量模型。

图 1-4 所示为 ISO/IEC 9126—1991 标准规定的软件质量度量模型。它由 3 层组成,其中第 1 层称为质量特性,第 2 层称为质量子特性,第 3 层称为度量。

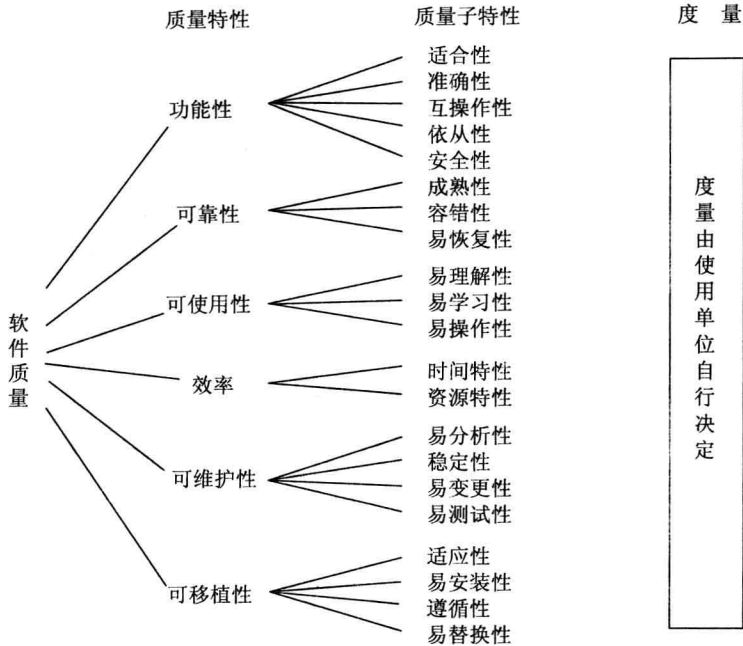


图 1-4 ISO 软件质量度量模型

软件质量评价的目的是为了直接支持开发并获得能满足用户要求的软件。最终目标是保证产品能提供所要求的质量,即满足用户明确的和隐含的要求。软件产品的一般评价过程是,确定评价需求,然后规定、设计和执行评价,如图 1-5 所示。

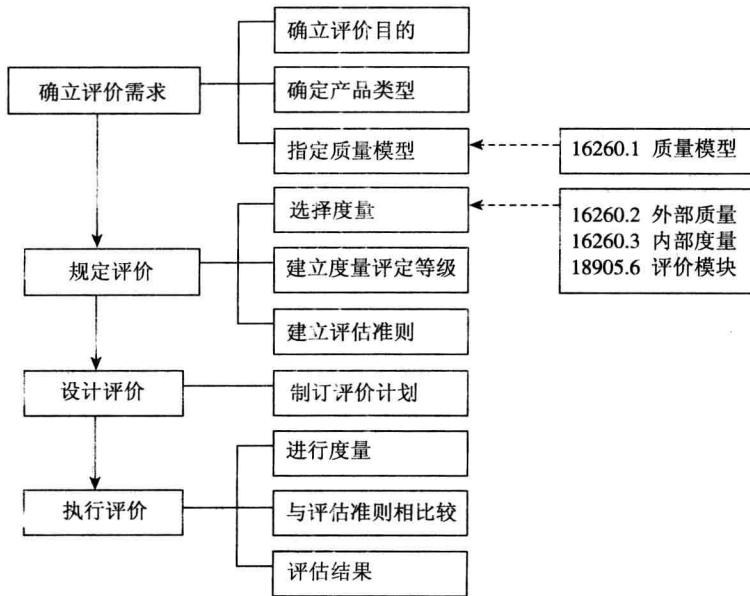


图 1-5 软件评价过程