

<<软件测试原理与实践>>

图书基本信息

书名：<<软件测试原理与实践>>

13位ISBN编号：9787111255062

10位ISBN编号：7111255062

出版时间：2009-2

出版时间：Srinivasan Desikan、Gopaldaswamy Ramesh、韩柯、李娜 机械工业出版社 (2009-02出版)

作者：(印)迪西肯 (Desikan, S) 等著

页数：285

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<软件测试原理与实践>>

### 前言

当今世界，软件变得无处不在。

消费者对此的期望大幅度增加，“软件出错很正常，我们必须接受这个事实”的旧观点已经不再适用。

如今，人们期望软件能每时每刻地正常运行，并且要满足客户不断变化的需求。

早先，软件系统用于后台管理服务和非关键业务的操作。

现在，越来越多的关键应用都在全球实现。

对无差错运行软件期望的提高，导致了对软件供应商高品质产量需求的增长。

反过来，在过去十几年中，市场关注的焦点已从单纯的编程和开发转向更全面的目標：生产的软件一直正常工作，因而更加关注对软件的测试。

在过去的十几年中，测试已经引起人们的极大兴趣。

以下列举一些事实：测试工作量的成倍增加提供了广阔的职业发展机会；测试相关职位的薪资正在上升；测试已经成为重要的外包机会；过去五年，关于测试的会议和其他类似活动有了明显的增加；越来越多的专业人士考虑把软件测试作为职业。

测试的工作方法为了跟上需求的增加也经历了彻底的变化。

首先，全球化是持久的。

现在的组织利用地域时差和全球的人才，向各大洲派遣开发和测试团队，这些团队共同无缝地工作。

为了成功地推动这种新的全球化运作方式，公司必须掌握异地分布团队这项工作的艺术。

其次，测试已经从即兴和偶然的尝试，转变为一项系统的、有计划的活动，以完成所有过程并且通过科学的度量。

第三，现在成功的测试需要谨慎地利用各种技术，满足产品上市时间的要求。

贯穿于测试生命周期的测试自动化也已经成为必需品而不是奢侈品。

最后，人们对测试职业的看法也经历了一种巨变——成功的公司应为测试专业人士提供职业发展途径，鼓励他们发挥聪明才智，以保证他们能够长期供职于该公司和从事软件测试这个职业。

本书及时满足了测试从业人员的需要，同时鼓舞了有志投身测试行业的专业人士和学生。

## <<软件测试原理与实践>>

### 内容概要

《软件测试原理与实践》全面论述了软件测试的基本原理和最佳实践，介绍了最近出现的极限测试和即兴测试等新的测试方法。

《软件测试原理与实践》介绍了全球团队的个人、组织结构和模型等问题。在介绍综合性理论知识的同时，强调实际经验。

《软件测试原理与实践》在介绍黑盒测试和白盒测试等传统方法的同时，还介绍了测试的很多实际问题，例如国际化测试和回归测试等。

《软件测试原理与实践》的突出特点是从工程实践的角度，比较全面地讨论棘手问题的具体应对方法和相应的风险，站在比较高的层次上讨论软件测试工程的整体把握方法。

全书在各章附有许多实际问题的思考题，帮助读者更深刻地理解这些现实问题。

《软件测试原理与实践》可作为高等院校软件工程和测试方面的基础教材，对软件开发和测试人员解决实际问题也有较高的参考价值。

## <<软件测试原理与实践>>

### 作者简介

作者：(印度)Srinivasan Desikan (印度)Gopaldaswamy Ramesh 译者：韩柯 李娜 Srinivasan Desikan，印度班加罗尔市的西贝尔系统公司质量工程部主任，具有16年的产品测试经验。所测试的产品正在被全世界数以百万计的客户使用。Srinivasan在测试自动化、测试管理、测试过程和测试团队建立等方面有丰富的经验。他经常在国际测试会议上发表演讲，定期在多所大学举办讲座。Srinivasan是印度技术学院的客座教授，并担任多家印度测试公司的荣誉董事主任。

## 书籍目录

出版者的话译者序序前言第一部分 写作线索第1章 测试原理1.1 生产软件中的测试背景1.2 本章介绍1.3 不完美的车1.4 Dijkstra定律1.5 及时测试1.6 圣人和猫1.7 首先测试测试用例1.8 杀虫剂悖论1.9 护航舰队与破布1.10 桥上的警察1.11 钟摆的终结1.12 黑衣人1.13 自动化综合症1.14 小结第2章 软件开发生存周期模型2.1 软件项目的阶段2.1.1 需求获取和分析2.1.2 策划2.1.3 设计2.1.4 开发或编码2.1.5 测试2.1.6 部署和维护2.2 质量、质量保证和质量控制2.3 测试、验证和确认2.4 表示不同阶段的过程模型2.5 生存周期模型2.5.1 瀑布模型2.5.2 原型和快速应用开发模型2.5.3 螺旋或迭代模型2.5.4 V字模型2.5.5 改进型V字模型2.5.6 各种生存周期模型比较第二部分 测试类型第3章 白盒测试3.1 白盒测试的定义3.2 静态测试3.2.1 人工静态测试3.2.2 静态分析工具3.3 结构测试3.3.1 单元 / 代码功能测试3.3.2 代码覆盖测试3.3.3 代码复杂度测试3.4 白盒测试中的挑战第4章 黑盒测试4.1 黑盒测试的定义4.2 黑盒测试的意义4.3 黑盒测试的时机4.4 黑盒测试的方法4.4.1 基于需求的测试4.4.2 正面和负面测试4.4.3 边界值分析4.4.4 决策表4.4.5 等价划分4.4.6 基于状态或基于图的测试4.4.7 兼容性测试4.4.8 用户文档测试4.4.9 领域测试4.5 小结第5章 集成测试5.1 集成测试的定义5.2 集成测试作为一种测试类型5.2.1 自顶向下集成5.2.2 自底向上集成5.2.3 双向集成5.2.4 系统集成5.2.5 选择集成方法5.3 集成测试作为一个测试阶段5.4 场景测试5.4.1 系统场景5.4.2 用例场景5.5 缺陷围歼5.5.1 选择缺陷围歼的频度和持续时间5.5.2 选择合适的产品版本5.5.3 对缺陷围歼的目标进行沟通5.5.4 建立和监视实验室5.5.5 采取行动解决问题5.5.6 优化缺陷围歼所涉及的工作5.6 小结第6章 系统测试和确认测试6.1 系统测试概述6.2 实施系统测试的原因6.3 功能测试与非功能测试6.4 功能系统测试6.4.1 设计 / 体系结构验证6.4.2 业务垂直测试6.4.3 部署测试6.4.4 贝塔测试6.4.5 符合性的认证、标准和测试6.5 非功能系统测试6.5.1 设置配置6.5.2 提出进入与退出准则6.5.3 平衡关键资源6.5.4 可伸缩性测试6.5.5 可靠性测试6.5.6 压力测试6.5.7 互操作性测试6.6 确认测试6.6.1 确认准则6.6.2 选择确认测试的测试用例6.6.3 执行确认测试6.7 测试阶段小结6.7.1 多阶段测试模型6.7.2 多个发布版本的处理6.7.3 谁负责实施与何时实施第7章 性能测试7.1 引论7.2 决定性能测试的要素7.3 性能测试的方法论7.3.1 收集需求7.3.2 编写测试用例7.3.3 自动化性能测试用例7.3.4 执行性能测试用例7.3.5 分析性能测试结果7.3.6 性能调谐7.3.7 性能基准测试7.3.8 能力策划7.4 性能测试工具7.5 性能测试的过程7.6 挑战第8章 回归测试8.1 回归测试的定义8.2 回归测试的类型8.3 回归测试的时机8.4 回归测试的方法8.4.1 实施第一次“冒烟”或“摸底”测试8.4.2 理解选择测试用例的准则8.4.3 测试用例分类8.4.4 选择测试用例的方法论8.4.5 重新设置测试用例以进行回归测试8.4.6 总结回归测试的结果8.5 回归测试的最佳实践第9章 国际化[I18n]测试9.1 引言9.2 国际化介绍9.2.1 语言的定义9.2.2 字符集9.2.3 属地9.2.4 本章使用的术语9.3 国际化测试的测试阶段9.4 有效化测试9.5 属地测试9.6 国际化确认9.7 假语言测试9.8 语言测试9.9 本地化测试9.10 国际化使用的工具9.11 挑战与问题第10章 即兴测试10.1 即兴测试概述10.2 伙伴测试10.3 结对测试10.4 探索式测试10.5 迭代式测试10.6 敏捷与极限测试10.6.1 xP workflow10.6.2 通过例子进行小结10.7 缺陷播种10.8 小结第三部分 特殊测试专题第11章 面向对象系统的测试11.1 引言11.2 面向对象软件入门11.3 面向对象测试的差别11.3.1 一组类的单元测试11.3.2 将类组合在一起——集成测试11.3.3 面向对象系统的系统测试与互操作11.3.4 面向对象系统的回归测试11.3.5 面向对象系统的测试工具11.3.6 小结第12章 可使用性与易获得性测试12.1 可使用性测试的定义12.2 可使用性测试的途径12.3 可使用性测试的时机12.4 实现可使用性的方法12.5 可使用性的质量因素12.6 美感测试12.7 易获得性测试12.7.1 基本易获得性12.7.2 产品易获得性12.8 可使用性工具12.9 可使用性实验室的建立12.10 可使用性的测试角色12.11 小结第四部分 测试中的人员和组织问题第13章 常见人员问题13.1 关于测试的感觉和错误概念13.1.1 “测试没有什么技术挑战”13.1.2 “测试没有为我提供职业成长道路”13.1.3 “我被派来测试——我到底怎么了？”13.1.4 “这些人是我的对手”13.1.5 “测试是如果我有时间最终会做的工作”13.1.6 “测试的拥有者毫无意义”13.1.7 “测试只是破坏”13.2 测试与开发工作的比较13.3 为测试人员提供职业发展道路13.4 生态系统的角色与行动要求13.4.1 教育系统的角色13.4.2 高级管理层的角色13.4.3 测试界的角色第14章 测试团队的组织结构14.1 组织结构的要素14.2 单产品公司的结构14.2.1 单产品公司的测试团队结构14.2.2 按组件组织的测试团队14.3 多产品公司的结构14.3.1 测试团队作为“首席技术官办公室”的一部分14.3.2 针

## &lt;&lt;软件测试原理与实践&gt;&gt;

对所有产品的单一测试团队14.3 3按产品组织的测试团队14.3.4 针对不同测试阶段的独立测试团队14.3.5 混合模型14.4 全球化与地域分散的团队对产品测试的影响14.4.1 全球化的业务影响14.4.2 全时区开发, 测试团队模型14.4.3 测试能力中心模型14.4.4 垒球团队面临的挑战14.5 测试服务公司14.5.1 测试服务的业务需求14.5.2 测试作为一种服务与产品测试公司之间的差别14.5.3 测试服务公司的典型角色和责任14.5.4 测试服务公司面临的挑战与问题14.6 测试公司的成功因素第五部分 测试管理与自动化第15章 测试策划、管理、执行与报告15.1 引言15.2 测试策划15.2.1 准备测试计划15.2.2 范围管理：决定要测试和不测试的特性15.2.3 确定测试方法和策略15.2.4 确定测试准则15.2.5 确定责任、人员和培训计划15.2.6 确定资源需求15.2.7 确定测试的可交付产品15.2.8 测试任务：规模与工作量估计15.2.9 活动分解与进度估计15.2.10 沟通管理15.2.11 风险管理15.3 测试管理15.3.1 标准的选择15.3.2 测试基础设施管理15.3.3 测试人员管理15.3.4 与产品发布集成15.4 测试过程15.4.1 把各种要素放在一起并确定测试计划基线15.4.2 测试用例规格说明15.4.3 可跟踪性矩阵的更新15.4.4 确定有可能实现自动化的测试用例15.4.5 测试用例的开发和基线确立15.4.6 测试用例的执行与可跟踪性矩阵的更新15.4.7 指标的采集与分析15.4.8 准备测试总结报告15.4.9 推荐产品发布准则15.5 测试报告15.6 最佳实践15.6.1 与过程相关的最佳实践15.6.2 与人员相关的最佳实践15.6.3 与技术相关的最佳实践附录A：测试策划检查单附录B：测试计划模板第16章 软件测试自动化16.1 测试自动化的定义16.2 自动化使用的术语16.3 自动化所需的技能16.4 自动化的对象与范围16.4.1 确定自动化负责的测试类型16.4.2 自动化不太可能变更的部分16.4.3 自动化测试符合标准16.4.4 自动化的管理问题16.5 自动化的设计和体系结构16.5.1 外部模块16.5.2 场景与配置文件模块16.5.3 测试用例与测试框架模块16.5.4 工具与结果模块16.5.5 报告生成器与报告 / 指标模块16.6 测试工具 / 框架的一般需求16.7 自动化的过程模型16.8 测试工具的选择16.8.1 选择测试工具的准则16.8.2 工具选择与部署步骤16.9 极限编程模型的自动化16.10 自动化中的挑战16.11 小结第17章 测试指标和度量17.1 指标和度量的定义17.2 测试中指标的意义17.3 指标类型17.4 项目指标17.4.1 投入偏差（计划投入与实际投入）17.4.2 计划偏差（计划与实际）17.4.3 不同阶段内的投入分布17.5 进度指标17.5.1 测试缺陷指标17.5.2 开发缺陷指标17.6 生产力指标17.6.1 每100小时测试发现的缺陷数17.6.2 每100小时的测试用例执行数17.6.3 每100小时的测试开发测试用例数17.6.4 每100个测试用例发现的缺陷数17.6.5 每100个失败的测试用例缺陷数17.6.6 测试阶段有效性17.6.7 已关闭缺陷的分布17.7 发布指标17.8 小结参考文献

## &lt;&lt;软件测试原理与实践&gt;&gt;

## 章节摘录

第一部分 写作线索第1章 测试原理1.1 生产软件中的测试背景我们今天使用的几乎一切东西都包含软件。

在软件发展的早期，软件用户的数量与大公司相比还是很少的。

现在，一个典型的工作场所（或家里），差不多每个人都在使用计算机及软件。

管理人员使用生产率很高的办公软件（代替以前的打字机）。

会计师及财务人员使用电子表格软件和其他财务软件包，比使用计算器（甚至手工）要快得多。

公司和家里的每个人都用电子邮件和互联网进行娱乐、教育、通信和交互，获取任何想要的信息。

另外，“技术”人员使用程序设计语言、建模工具、仿真工具和数据库管理系统完成以前主要靠手工完成的任务。

上面只是说明软件的使用对于用户来说“很明显”的几个例子。

但是，软件的无处不在和广泛普及远不止以上这些例子所揭示的那样。

现在的软件就像20世纪初的电一样普及。

我们在办公室和家庭所使用的几乎每一台设备都嵌入大量的软件，例如手机、电视、手表和冰箱以及厨房的每一件电器都有嵌入式软件。

另一个值得注意的现象是软件在任务关键场合的使用，在这些场合出现失效是根本不能接受的。

对于心脏起搏器软件，决不能提议“请关机并重启系统”！

我们离不开的几乎所有服务中都有软件。

银行、航空管制、汽车等，驱动它们的软件都是绝对不能失效的。

这些软件系统必须每时每刻、永久、可靠、可预见地运行。

这些无所不在、广泛使用和关键之处都对软件的开发和部署提出了一定的要求。

首先，开发软件产品或提供服务的公司必须尽全力减少、最好消除每件所交付的软件产品或服务中的缺陷。

用户越来越不能容忍劣质的软件产品。

从软件开发公司的角度看，发布有缺陷的软件产品在经济上也不是可行的。

比如，在电视机发运给成千上万的用戶后，其中的嵌入式软件被发现有一个缺陷。

怎么可能发送“补丁”给这些用户，要求他们“安装补丁”？

因此，唯一的解决方案是在产品交付用户之前就一次做好。

## <<软件测试原理与实践>>

### 编辑推荐

《软件测试原理与实践》从实用的角度对软件测试进行了全面的阐述，讨论了像极限测试和即兴测试这类新兴的领域。

《软件测试原理与实践》特色：关注分散在全球地域的团队。

讨论全球化团队的人员、组织结构和模型问题。

提供印度在测试方面的丰富经验。

越来越多的产品测试工作是在印度完成的，但是研究印度经验或印度业务模型的专著却很少。

《软件测试原理与实践》通过实例讨论了印度的最佳测试实践。

在保持完整的理论体系基础上，强调实践经验。

《软件测试原理与实践》在介绍诸如等价类划分和圈复杂度等传统方法的同时，还讨论了测试的一些实际问题，例如国际化测试和回归测试。



<<软件测试原理与实践>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>