

软件工程专业核心课程系列教材

软件测试实践教程

蔡建平 倪建成 高仲合 编著

清华大学出版社

软件工程专业核心课程系列教材

软件测试实践教程

蔡建平 倪建成 高仲合 编著

清华大学出版社
北京

内 容 简 介

软件测试是软件工程的一个重要分支,它对测试人员的专业知识、专业技术、专业能力要求极高,而目前企业对测试人员的要求是要有较丰富的测试经验及较强的测试工具应用能力。本书作为《软件测试方法与技术》配套的实验教材,通过覆盖软件评测的各个环节和知识点,以主流的开源软件测试工具应用为基础,以实战能力培养为目的,为高等院校不同学历教育的软件工程专业和计算机相关专业开设软件测试课程提供了全方位的,并且是可行或可用的实践教学方案和实践教学平台以及配套的实践教学案例。

全书共 12 章,分为管理、静态分析、单元测试、GUI 测试、性能测试及软件综合评测共 6 大部分。主要内容包括软件缺陷管理、软件测试管理、程序理解、代码静态分析、xUnit 单元测试框架、单元覆盖测试、Java GUI 基础类库应用测试、Web 页面测试、Gtk+用户界面测试、单元性能测试、Web 应用性能测试以及软件综合评测工具等。

掌握软件测试技术、构建软件测试环境、编写软件测试用例、开展软件测试工作并有效进行软件测试管理,无论是对于软件管理人员、开发人员、质量保证人员还是测试人员,都具有较强的现实意义。本书针对软件测试的实验内容全面,实验方案完整,实践环境建设可行,实验步骤及过程讲解清晰,实验案例丰富实用,可作为高等院校不同学历教育的软件工程及计算机相关专业的“软件测试实验课程”教材(如本科生、研究生,甚至高职生或高专生等),也可作为软件测试实战培训教材,同时本书也是软件开发或管理人员、测试或质量保证人员非常好的自学参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

软件测试实践教程/蔡建平,倪建成,高仲合编著.--北京:清华大学出版社,2014
软件工程专业核心课程系列教材

ISBN 978-7-302-36040-7

I. ①软… II. ①蔡… ②倪… ③高… III. ①软件-测试-教材 IV. ①TP311.5

中国版本图书馆 CIP 数据核字(2014)第 065978 号

责任编辑:魏江江 薛 阳

封面设计:常雪影

责任校对:时翠兰

责任印制:宋 林

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 刷 者:北京市人民文学印刷厂

装 订 者:三河市漂源装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:25.25 插页:2 字 数:599 千字

版 次:2014 年 11 月第 1 版

印 次:2014 年 11 月第 1 次印刷

印 数:1~2000

定 价:45.00 元

产品编号:054100-01

前言

本书是本人编著的全国工程硕士专业学位教育指导委员会推荐教材及软件工程专业核心课程系列教材——《软件测试方法与技术》配套的实践教材。

《软件测试实验指导教程》自2009年11月发行到2013年2月销售殆尽，有二十多所高等院校将它指定为实验教材或实验参考书籍。在这期间，作者一直密切地与高校教材的使用者——授课老师交流，听取他们对本教材的意见。同时，也密切关注企业应用开源软件测试工具进行软件测试的情况，了解企业常用的开源测试工具都有哪些，并基于此做了部分调整和补充。最后，为保持其延续性，本书在教材结构和内容组织上没有做太大的调整，主要是在软件测试工具的升级、软件测试类型的覆盖、软件测试案例的补充以及软件测试工具的选型上进行了修改、补充和完善，使本教材在内容上更加新颖和完整。

软件测试是一门对工程实践要求极高，对学生动手能力要求极强的软件工程核心课程。目前许多高校不同学历教育的计算机类专业均开设了这门课程，并配有大量学时的实验课程或额外配套的课程设计实践课程。

本书充分考虑到现代软件测试贯穿软件工程整个软件生命周期，需要用到多种测试技术、方法的实际要求，以及国内大多数院校办学条件不足，实验教学经费有限，无法全方位引进商用软件测试工具，无法开展软件测试实验室建设的实际情况，对国内外主流的开源软件测试工具进行全面分析、研究和优选，并经过十多轮实践教学的检验，来设计本书的实验教学重点和实践能力要求。

本书与国内常见的软件测试实践教材重点讲授一般商用软件测试工具的方法不同，它涉及的实验内容非常广，软件测试知识非常多，开源软件测试工具实验非常全面，并且易于剪裁或扩充，无论是对于学生工具学习，还是教师实验指导，以及培训机构实战训练都是不可多得的实验教材。

本书的出版首先要感谢清华大学出版社的大力支持和帮助。

另外，本书的完成得益于许多学生的积极参与。在这里我要特别感谢我指导的研究生乔丽平、路翠、邱聃、刘泽伟等同学，他们在资料收集，章节起草，内容组织、实验完成以及图表制作等方面做了很多工作。我还要感谢北京工业大学软件工程专业03至06级的一些学生，如03级的安文怡、李征和刘欣宇等，04级的孙建和刘茜等，05级的杨天放、时永欣、赵京超和周丰等，06级的黄飞和霍晓珍等，本书的很多内容是来自于他们完成的软件测试课程设计实践总结报告。

其次，本书的大量内容是取材于互联网，并进行组织和修改的结果。遗憾的是很多网上资料由于转载或引用找不到原创处，在参考文献中无法准确标注，在这里一并表示感谢。

最后，对家人的感谢是必需的，教材、专著的编写离不开她们多年的支持和照顾。

目前国内软件测试的实验书籍或教材也逐渐多了起来，都有自己的特点或特色。但愿本实验教材能够一如既往地受到学生、教师以及软件工程师等广大读者的欢迎。当然，由于自身能力和水平有限，书中难免有许多不周到、不准确、疏漏或不足之处，恳请读者提出批评和建议，以便及时修正。

蔡建平

2014年10月 于北京

作者简介

蔡建平，原北京工业大学教授。曾在总装备部某研究所工作十余年，一直从事该所负责的全军军用共性软件系统项目的论证和研究，并在其中发挥重要作用，得到有关领导和专家的认可和好评。获军队科技进步一等奖、二等奖、三等奖多项；发表各类学术论文三十多篇，与他人合作著书一部。

在北京奥吉通科技有限公司任技术总监期间，除负责国防有关单位的软件工程、软件质量保证、软件测试以及嵌入式开发的技术咨询、提供解决方案和技术支持外，还主持开发了科锐时系列软件测试工具，并成功用于国防软件的测试。

2005年9月在北京工业大学软件学院任职教授以来，除了多年讲授软件测试课程和多次对外开展嵌入式软件测试技术培训外，在学院的学科建设、专业建设，如实验室建设、实践教学改革与创新、数字艺术方向和数字媒体技术专业建设、“211工程”建设、教育部和北京市特色专业建设，以及科研基地——科技创新平台建设等方面做了大量的工作，取得了突破性成果。

2013年10月在航天中认软件测评科技（北京）有限责任公司担任专家咨询委员会副主任，负责公司的业务规划。

2014年1月，在惠普国际软件人才及产业基地担任学术总监。同年4月，被聘为曲阜师范大学的兼职教授。

目 录

第 I 部分 管 理 篇

第 1 章 软件缺陷管理	3
1.1 缺陷管理工具介绍	3
1.1.1 Bugzilla	4
1.1.2 BugOnline	4
1.1.3 Bugzero	4
1.1.4 其他开源缺陷管理工具	5
1.2 缺陷管理工具 Mantis 及其应用	5
1.2.1 Mantis 功能介绍	5
1.2.2 Mantis 应用环境建立	11
1.2.3 Mantis 应用流程	18
1.3 Mantis 应用举例	28
1.3.1 Mantis 的应用过程举例	28
1.3.2 stock 软件中的缺陷处理 流程举例	33
实验习题	36
第 2 章 软件测试管理	37
2.1 软件测试管理工具	37
2.1.1 软件测试管理工具应具备 的功能	37
2.1.2 软件测试管理工具的 选择	38
2.1.3 常用软件测试管理 工具介绍	39
2.2 软件测试管理工具 TestLink 应用	41
2.2.1 TestLink 功能介绍	41
2.2.2 TestLink 应用环境建立	42
2.2.3 TestLink 使用流程	45

2.2.4 TestLink 应用举例	46
实验习题	65

第 II 部分 静态分析篇

第 3 章 程序理解工具	69
3.1 程序理解概述	69
3.1.1 程序理解的概念	69
3.1.2 程序理解的任务与内容	70
3.1.3 程序理解的相关技术	71
3.1.4 程序理解工具	72
3.2 Oink 程序理解工具	72
3.2.1 Oink 环境建立	73
3.2.2 Oink 工具及使用流程	75
3.2.3 Oink 应用举例	80
3.3 Eclipse PTP/CDT 程序理解 工具	83
3.3.1 PTP/CDT 介绍	83
3.3.2 PTP 环境建立	84
3.3.3 PTP 功能及使用流程	93
3.3.4 PTP 应用举例	94
实验习题	100
第 4 章 代码静态分析工具	101
4.1 代码静态分析工具及编程 规范检查	101
4.1.1 静态代码分析工具介绍	102
4.1.2 编程规范检查工具 CheckStyle	105
4.2 代码静态分析工具 FindBugs	110
4.2.1 FindBugs 环境建立	110
4.2.2 FindBugs 应用举例	119

- 4.2.3 FindBugs 的 Bug 级别介绍..... 123
- 4.3 代码静态分析工具 PMD 125
 - 4.3.1 PMD 功能介绍 125
 - 4.3.2 PMD 环境建立 126
 - 4.3.3 PMD 应用流程 130
- 4.4 开源代码静态分析工具 Splint 135
 - 4.4.1 Splint 的安装 136
 - 4.4.2 Splint 的应用 137
 - 4.4.3 Splint 与 IDE 的集成..... 142
- 实验习题 144

第III部分 单元测试篇

- 第 5 章 xUnit 单元测试框架..... 147
 - 5.1 xUnit 介绍 148
 - 5.2 JUnit 单元测试工具 150
 - 5.2.1 JUnit 单元测试环境建立..... 152
 - 5.2.2 JUnit 单元测试方法 158
 - 5.2.3 JUnit 单元测试应用举例..... 160
 - 5.2.4 JUnit4 与 JUnit3 的区别 167
 - 5.3 CppUnit 单元测试工具 169
 - 5.3.1 CppUnit 单元测试环境建立..... 169
 - 5.3.2 CppUnit 功能和使用流程..... 175
 - 5.3.3 CppUnit 单元测试应用举例..... 180
 - 5.4 基于标注的单元测试框架 TestNG 181
 - 5.4.1 TestNG 功能介绍 182
 - 5.4.2 TestNG 环境建立 183
 - 5.4.3 TestNG 应用流程 185
 - 5.4.4 TestNG 应用举例 189

- 5.4.5 TestNG 与 JUnit4 对比.....193
- 实验习题.....194
- 第 6 章 单元覆盖测试 195
 - 6.1 覆盖测试工具介绍196
 - 6.2 JUnit 下的覆盖测试工具 Eclemma196
 - 6.2.1 Eclemma 介绍197
 - 6.2.2 Eclemma 测试环境建立197
 - 6.2.3 Eclemma 测试功能及使用流程198
 - 6.2.4 Eclemma 测试应用举例202
 - 6.3 GCC 的覆盖测试工具 Gcov.....210
 - 6.3.1 Gcov 测试环境建立.....211
 - 6.3.2 Gcov 测试功能及使用流程211
 - 6.3.3 Gcov 覆盖测试应用举例213
- 实验习题.....222

第IV部分 图形用户界面测试篇

- 第 7 章 Java GUI 基础类库应用测试 227
 - 7.1 JFCUnit 单元测试工具介绍228
 - 7.2 JFCUnit 基本测试方法.....229
 - 7.3 JFCUnit 测试环境建立.....230
 - 7.4 JFCUnit 测试资源应用.....232
 - 7.4.1 JFCUnit 核心函数的应用方式232
 - 7.4.2 JFCUnit 的界面操作要点234
 - 7.4.3 JFCUnit 中主要的 GUI 类.....237
 - 7.5 JFCUnit 测试应用举例.....239
 - 7.6 JFCUnit XML 测试框架.....248

实验习题	259
第 8 章 Web 页面测试	261
8.1 Web 页面测试工具介绍	263
8.2 Web 页面测试工具之一 ——HttpUnit	264
8.2.1 HttpUnit 环境建立	266
8.2.2 HttpUnit 的工作方式	266
8.3 Web 页面测试工具之二 ——JWebUnit	272
8.3.1 JWebUnit 测试环境建立	273
8.3.2 JWebUnit 应用方法	274
8.3.3 JWebUnit 测试应用举例	277
8.3.4 JWebUnit 应用小结	280
8.4 Web 页面测试工具之三 ——Selenium	280
8.4.1 Selenium 环境建立	281
8.4.2 应用流程	283
8.4.3 应用举例	287
实验习题	289
第 9 章 Gtk+用户界面测试	291
9.1 Gtk+用户界面概述	292
9.2 Gtk+用户界面测试工具 Gerd	294
9.2.1 Gerd 测试环境建立	295
9.2.2 Gerd 功能及使用原理	296
9.2.3 界面测试应用举例	297
实验习题	301

第 V 部分 性能测试篇

第 10 章 单元性能测试	307
10.1 单元性能测试概念介绍	307
10.2 单元性能测试工具 p-unit	309
10.2.1 p-unit 测试环境建立	310
10.2.2 p-unit 测试功能及 使用流程	311

10.2.3 p-unit 测试应用举例	311
实验习题	326
第 11 章 Web 应用性能测试 工具 JMeter	327
11.1 Web 性能测试工具介绍	327
11.1.1 HP LoadRunner	328
11.1.2 Apache JMeter	328
11.2 应用 JMeter 进行 Web 性能测试	329
11.2.1 JMeter 测试环境建立	329
11.2.2 JMeter 测试功能及 使用流程	330
11.3 JMeter 测试应用举例	337
11.3.1 测试 HTTP 请求	337
11.3.2 FTP 测试	340
11.3.3 数据库测试	341
11.3.4 Web 应用测试	343
11.3.5 JMeter 工具小结	345
实验习题	346

第 VI 部分 软件综合评测篇

第 12 章 软件综合评测工具 EASTT	349
12.1 EASTT 工具介绍	350
12.2 EASTT 测试环境建立	352
12.3 EASTT 测试功能及使用 流程	354
12.3.1 EASTT 的主要功能	355
12.3.2 EASTT 的使用流程	356
12.4 EASTT 评测工具具体 使用举例	378
12.5 EASTT 应用小结	390
实验习题	390
参考文献	391

第 I 部分 管 理 篇

软件工程除了技术外，最重要的思想之一就是管理。软件测试作为软件工程的一个重要分支，其目标是保证软件生命周期中每个阶段的活动结果是正确的，这就是现代软件测试思想——全生命周期软件测试思想。软件测试管理是软件测试质量的重要保证手段，它要解决的问题是如何确保软件测试技术能使软件项目在软件生命周期内得到顺利实施，并产生预期的效果。

事实上，随着技术的发展，软件系统的规模急剧增大，采用国际协作的模式，由位于世界上不同国家不同城市的多个团队联合开发软件系统，已经成为目前软件开发的主要趋势。与之相适应，测试也需要物理上分布的多个团队共同参与。由于各个团队承担不同的任务，并有着不同的项目管理模式，为保证整个系统能得到一致、有效的质量控制，测试管理至关重要。测试管理有助于系统、规范地管理各种测试资源和测试活动，以提高测试的效率和质量。按照管理对象的不同，软件测试管理大致分为软件测试团队组织管理、软件测试计划管理、软件缺陷(错误)跟踪管理以及软件测试资源管理这四大部分。

软件测试团队组织管理，通俗地讲就是测试团队应该如何组建、人员应该如何分工与管理以及绩效应该如何考核等。

软件测试计划管理，通俗地讲就是安排好测试流程。这部分内容具体涵盖软件测试策划、软件测试技术剪裁、测试进度管理、测试成本管理等几个部分。其中：软件测试策划工作主要是指在具体测试活动实施之前做好策划工作，如起草测试大纲和测试计划；软件测试技术剪裁工作主要是指测试团队应根据软件项目的具体情况剪裁出所要实施的测试技术；测试进度管理工作主要是指排出各项测试的时间进度及人员安排，如有变动则应做相应调整；测试成本管理工作主要是开列出测试活动中会涉及的资源需求。

软件测试团队组织管理和软件测试计划管理属于软件项目管理的范畴，可根据软件测试的特点和 GB/T 15532—2008 计算机软件测试规范以及 GB/T 9386—2008 计算机软件测试文档编制规范来开展相关的管理工作。

软件缺陷(错误)跟踪管理，通俗地讲就是确保发现的缺陷(错误)已经被开发团队纠正或处理过，并且没有引入新的缺陷(错误)。具体来讲，当测试团队通过各种途径发现了文档或代码中的缺陷或错误以后，并不是交一份测试报告就草草了事，而是在递交报告以后继续督促开发团队及时关闭已知缺陷或错误(当然，如有必要，应对这些缺陷、错误做严重程度排序，以便开发团队能视轻重缓急安排处理顺序)。当开发团队关闭了测试报告中的缺陷(错误)以后，测试团队还需验证开发团队在关闭过程中有没有引入新的错误。通常，这个

过程称为回归测试。回归测试如发现问题，则继续报告给开发团队，按上述流程循环，直至回归测试最终通过。

软件测试资源管理，通俗地讲就是努力建设好测试团队的测试资源库，并对测试团队成员进行技能培训以帮助他们使用好这个测试资源库。软件测试资源库所包括的内容是测试团队在长期实践过程中逐步积累起来的经验教训、测试技巧、测试工具、规格文档以及一些经过少量修改便能推广至通用的测试脚本程序。测试资源管理工作做得越好，测试团队在实际测试过程中就越能少走弯路，测试团队内部的知识交流和传递就越充分，测试脚本或规格文档的重复开发工作也就越能被有效地避免。软件测试资源管理工作包括两部分：一个是建设，另一个是培训。建设工作大抵是收集各类测试文档、测试工具、测试脚本，也包括收集整理测试人员的会议发言、总结报告、技术心得等。培训工作大抵是通过技术讲座、正式或非正式团队会议、印发学习资料等形式进行。

软件测试管理工具是软件测试管理最重要的保证手段，对于大型系统测试来说，测试管理工具可以帮助组织测试资产、监督项目状态、集成自动化测试工具以及度量测试效果，能够为所有这些参与者提供一个交流和协作的平台，是项目管理中必不可少的。近年来，测试工具的应用越来越广泛，很多工具都能提供一定程度的测试管理功能。当然，商用软件测试管理工具不论从能力还是从成熟度来说都是有较强竞争力的，在条件许可的情况下应该作为首选。但是，开源软件测试管理工具发展越来越迅速，功能越来越强大，应用越来越广泛，是进行软件测试管理实践教学的最佳解决方案之一。本部分以开源为基础，重点讲解有关的软件测试管理工具。

第1章 软件缺陷管理

软件开发是引入软件错误或软件缺陷的过程，软件测试则是发现软件错误或软件缺陷的过程。对于大型软件来说，错误数目是非常可观的，必须借助工具才能对所发现的这些错误进行有效的管理，为软件缺陷或错误的消除或者软件质量的评价及软件开发的决策提供依据。

1.1 缺陷管理工具介绍

有些项目很简单，缺陷也就十几个、几十个，采用手工进行缺陷管理就可以了。而有些项目，特别是大型软件，会有成千上万个缺陷，甚至还有无法预计的缺陷。这时，手工进行缺陷管理就很不现实，而选用合理的缺陷管理工具便成了不可避免的问题。

本节主要介绍一些开源的缺陷管理工具，以便读者使用。例如 Mantis(免费)、Bugzilla(免费)、JIRA(免费)、TrackRecord(Compuware 公司)、ClearQuest(IBM Rational 公司)，这些都是专门的缺陷管理工具。此外，有些测试管理工具也具有缺陷管理的功能，如 HP 公司的 ALM、IBM Rational 公司的 TestManager，但这些都是商业软件。

商业软件有商业软件的好处，例如 MI 公司的 TestDirector，采用的是 B/S 构架模式，Windows 平台，可以定制流程、查询、功能域、用户角色及角色权限，可 E-mail 通知，可以生成各种报表并支持多种数据库，还可以与其他 MI 公司测试工具集成，安装配置也较为简单、有可优化的工作流、可使用 C 语言来改进优化系统。当然，开源软件也有其自身的优点：基本上都是基于 Web 的、提供源程序、免费使用、用户有更多的自由操作空间，等等。

基于 Web 的缺陷跟踪系统有很多好处，它简化了缺陷管理的相关工作。

(1) 实现地域上分散的项目人员高效协同工作，有效地降低软件测试成本，提高工作效率。

(2) 通过设置不同的用户权限，安全、准确地实现缺陷的管理和跟踪，且便于项目结束后的存档，以备将来参考。

(3) 系统维护简单，如果采用 B/S 结构，则只需要修改服务器端。现有的基于 Web 的缺陷跟踪系统所提供的功能都基本相同，但是在管理结构的实现上有一些不同，最简单的实现是一个系统有一个管理员，这个管理员负责管理所有的用户和项目。管理员的工作量

会随用户和项目的增加而增加,通信和管理成本也会增加,因此这样的系统不适合大型项目或者地域分散的软件生产商采用。

下面简单介绍几个常用的开源缺陷管理工具。

1.1.1 Bugzilla

Bugzilla(免费,跨平台)是一款开源的 Bug 追踪系统,可以用来帮助管理软件开发。Bugzilla 虽然是专门为 UNIX 定制开发的,但是在 Windows 平台下依然可以成功安装使用。

而且, Bugzilla 还能够被集成到 Testopia(测试用例管理系统)系统中。Bugzilla 的强大功能表现在以下几个方面。

- (1) 强大的检索功能。
- (2) 用户可配置通过 E-mail 来公布 Bug 变更。
- (3) 历史变更记录。
- (4) 通过跟踪和描述处理 Bug。
- (5) 附件管理。
- (6) 完备的产品分类方案和细致的安全策略。
- (7) 安全的审核机制。
- (8) 强大的后端数据库支持。
- (9) Web、XML、E-mail 和控制界面。
- (10) 友好的网络用户界面。
- (11) 丰富多样的配置设定。
- (12) 版本间向下兼容。

1.1.2 BugOnline

BugOnline(开源)是一款开源的 Bug 管理系统,功能强大,易于使用。BugOnline 基于 ASP.NET、SQL Server(包括 Express 版)及 AJAX 等技术。

BugOnline 具有如下一些特性。

- (1) 在线消息及 E-mail 自动通知功能。当有新 Bug 及 Bug 分配给用户时,会自动通知用户。
- (2) 优秀的人员分配、工作量统计功能。
- (3) 基于项目角色的权限管理、工作规划及流程化。
- (4) Bug 状态统计,便于掌控项目进度。
- (5) 基于 SSL 的数据传输,确保数据不被截取,保证安全性(也可设定为非 SSL)。
- (6) 强大的报表功能。

1.1.3 Bugzero

Bugzero(免费开源,跨平台)是一款多功能、基于网络并在浏览器下运行的 Bug 缺陷管理和跟踪系统,可用来记录、跟踪、并归类处理软件开发过程中出现的 Bug 和硬件系统中

存在的缺陷。Bugzero 还是一个完整的服务管理软件，集成了服务台热线流程管理，可用来记录各种日常事务、变更请求和问题报告，并追踪和处理各种客户询问、反馈和意见。

Bugzero 提供了一个可靠的中央数据库，使得公司内部团队成员以及外部客户能在任何地点、任何时间进行协调和信息交流，并且使任何记录都有据可查。它使用户省时省力。Bugzero 不但使用方便，而且功能齐全，变通性好，能够灵活设置各种实际工作流程，满足特定业务和产品环境下的需求。这种灵活、易用的缺陷跟踪流程不仅增强了项目开发的质量，同时也提高了整个机构的生产效率。

1.1.4 其他开源缺陷管理工具

Bugtracker 是一个完整的 Bug / Issue 管理系统，以 Java Servlet 作为 Web 前台，以 MySQL 数据库作为后台。

BugFree 是借鉴微软的研发流程和 Bug 管理理念，使用 PHP+MySQL 独立编写的一个 Bug 管理系统。它简单实用、免费并且开放源代码(遵循 GNU GPL)。

JTrac 是一个开源且可高度配置的用于缺陷追踪的 Web 应用程序。它可以跟踪网络应用程序，可方便地实现定制，增加自定义字段和下拉式。其特点包括可定制的工作流程、实地许可、电子邮件集成、文件附件和详细历史记录查询。

BugNet 是一个不错的开源 Bug 跟踪和项目管理系统。

eTraxis 是基于网页的免费 Bug 跟踪系统。其主要特点是完全自定义模板、先进的过滤器、LDAP 支持、电子邮件通知、订阅报刊、提醒、灵活的权限管理、图形化的项目指标等。

1.2 缺陷管理工具 Mantis 及其应用

Mantis 同样是一款开源的软件缺陷管理工具，是一个基于 PHP 技术的轻量级缺陷跟踪系统，其功能与商用的 JIRA 系统类似，都是以 Web 操作的形式来提供项目管理及缺陷跟踪服务。Mantis 在功能上可能没有 JIRA 那么专业，界面也没有 JIRA 漂亮，但在实用性上足以满足中小型项目的缺陷管理及跟踪。Mantis 包括客户端浏览器、Web 服务器和数据库服务器。当然，Web 服务器和数据库服务器也可以是同一台主机。重要的是它是开源的，不需要付任何费用。不过 Mantis 目前的版本还存在一些问题，期待在今后的版本中能够得以完善。

1.2.1 Mantis 功能介绍

Mantis 基于 PHP+MySQL，可以运行于 Windows/UNIX 平台上。作为一个 Bug 管理系统，其适用性是否符合实际工作的需要是至关重要的。Mantis 基本可以满足 Bug 管理日

常流程。而且, Mantis 是 B/S 架构的 Web 系统, 如果今后有需要, 还可以配置到 Internet 上, 实现异地 Bug 管理。在 Mantis 系统中, 有如下几种角色: 管理员、经理、开发人员、修改人员、报告人员、查看人员。每个角色所拥有的权限是不一样的, 从大到小依次排列是: 管理员→经理→开发人员→修改人员→报告人员→查看人员。

Mantis 的特点是免费、简洁灵活, B/S 结构的 Web 系统比较适合分布式协作开发和测试。关于 Mantis 的详细信息和技术支持, 可访问 <http://www.mantisbt.net/>。

1. Mantis 的基本特征

(1) 个人可定制的 E-mail 通知功能, 每个用户可根据自身的工作特点而只订阅相关的缺陷状态邮件。

(2) 支持多项目、多语言。

(3) 权限设置灵活, 不同角色有不同权限, 每个项目可设为公开或私有状态, 每个缺陷也可设为公开或私有状态, 每个缺陷可以在不同项目间移动。

(4) 主页可发布项目相关新闻, 方便信息传播。

(5) 方便的缺陷关联功能。除重复缺陷外, 每个缺陷都可以链接到其他相关缺陷。

(6) 缺陷报告可打印或输出为 CSV 格式。支持可定制的报表输出, 可定制用户输入域。

(7) 有各种缺陷趋势图和柱状图, 为项目状态分析提供依据, 如果不满足要求, 则可以把数据输出到 Excel 中进一步分析。

(8) 流程定制不够方便, 但该流程可满足一般的缺陷跟踪。

(9) 可以实现与 CVS 的集成, 即实现缺陷和 CVS 仓库中的文件相关联。

(10) 可以对历史缺陷进行检索。

2. Mantis 系统中缺陷状态的转换

缺陷状态是描述软件缺陷处理过程所处阶段的一个重要属性。对应于不同的状态, 软件测试人员能确定对该问题的处理已经进展到什么阶段, 还需要进行哪些工作, 需要哪些人员的参与等信息。缺陷跟踪系统的状态比较复杂, 这也是缺陷管理中的难点。在缺陷跟踪管理过程中, 将缺陷记录划分为不同的阶段、不同的状态来进行标记。Mantis 系统将缺陷的处理状态分为 New、Active、Invalid、Later、Resolve、Reopen、Closed 7 种, 如图 1-1 所示。

(1) 一个新的缺陷被提交, 即为 New。

(2) Active, 刚提交的缺陷, 在被项目经理确认并分发给研发人员修改前所处的状态。

(3) Invalid, 已提交的缺陷在当前版本中已不是问题或不需要修改。

(4) Later, 提交的缺陷在当前研发阶段无法对其进行修改。

(5) Resolve, 经软件工程师修改或给出相关意见后, 等待测试人员验证时所处的状态。

(6) Reopen, 已经关闭的缺陷重新出现, 测试人员将其状态设置为 Reopen, 分发缺陷时的操作与状态为 New 时的类似。

(7) Closed, 最终修改正确或不正确的缺陷报告, 经过验证或项目经理同意后, 可以关闭。处于关闭状态的缺陷报告可表现为已改正、符合设计、不能重现、不能改正、由报告人撤回。

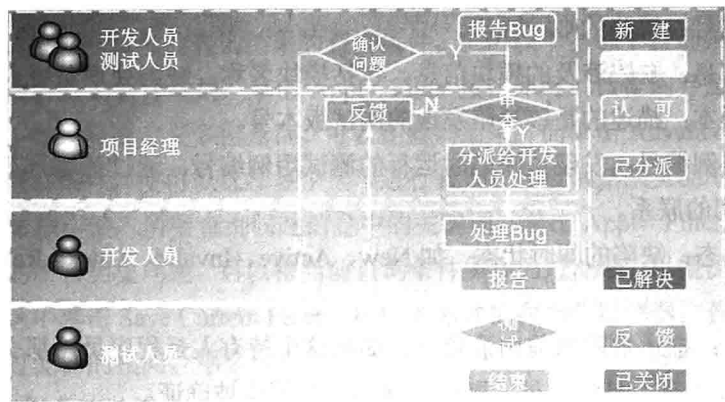


图 1-1 Mantis 缺陷状态转换图

3. Mantis 用户角色及权限的管理

在一个测试项目中, 存在各种不同的身份, 比如项目经理、测试经理、开发经理、程序员、测试员等。不同身份的用户使用系统时可以执行的操作理应是不同的, 例如不能让一个测试员来进行用户分工的工作。另一方面权限的要求是以对象为中心的, 比如对于缺陷这个对象, 它的填报信息只能由填报该缺陷的测试员来修改和维护, 其他的任何人都不能具有同等的操作权限。

Mantis 中用户角色和登录权限及方式如表 1-1 所示。

表 1-1 Mantis 中用户角色和登录权限及方式

用户	权限	工作范围
管理员 (administrator)	高 ↓ 低	管理和维护整个系统
项目经理 (manager)		对整个项目进行管理
开发人员 (developer)		负责整个软件的开发
修改人员 (updater)		负责修改Issue
报告人员 (reporter)		负责提交Bug报告
查看人员 (viewer)		查看Bug流程及情况

4. Mantis 的软件缺陷属性的定义

软件缺陷是按照能准确发现缺陷目标进行分类的, 分类之间应无重叠, 分类体系应覆盖所有的缺陷类型; 要与软件生命周期相结合。传统的分类方法可分为按照缺陷的来源和

缺陷错误性质这两种。如 Putnam 等人提出的分类方法和正交缺陷分类法以及 IEEE 制定的软件异常分类标准等。正交缺陷分类法定义的软件缺陷的 13 个属性在 Mantis 中得到了实现。

(1) 缺陷编号：缺陷的唯一标识。

(2) 模块信息：缺陷涉及的模块信息，包括模块名称、缺陷处理负责人、模块版本。

(3) 测试版本：描述的是该缺陷发现的测试版本号。

(4) 对应用例编号：发现该缺陷时运行的测试用例编号，通过该编号可以建立起测试用例和缺陷之间的联系。

(5) 缺陷状态：缺陷的即时状态，如 New、Active、Invalid、Later、Resolve、Reopen、Closed 等。

(6) 持有人：描述缺陷当前由谁负责，如果这个持有人是程序员，那么这个缺陷正在被修改；而如果持有人是测试员，则这个缺陷正在等待被确证。

(7) 报告者：报告缺陷的测试人员的编号或用户名。

(8) 报告日期：缺陷填报的日期。

(9) 重现性：可重现或不可重现。

(10) 重现步骤：和测试用例相关，描述的是发现这个缺陷的步骤。

(11) 严重等级：可定制，默认为 4 级——P1(致命)、P2(严重)、P3(一般)、P4(轻微)。

(12) 缺陷类型：可定制，默认分为功能缺陷、用户界面缺陷、边界值相关缺陷、初始化缺陷、计算缺陷、内存相关缺陷、硬件相关缺陷、文档缺陷。

(13) 缺陷优先级(报告者)：可定制，默认分为必须修复、立即修复、应该修复、考虑修复。

5. Mantis 的功能介绍

1) 多项目管理

在系统页面上单击 Manage|Manage Projects，可以进入项目管理界面。上面显示了已创建的项目列表，单击 Create New Project，可进入新建项目页面。可以设定新项目当前的状态，项目状态有 development、release、stable 和 obsolete 这几种。在已建项目列表中可以修改项目数据，包括修改项目状态(将项目修改为公开或私有)，添加和修改子项目，为该项目添加和修改 Categories，添加和修改项目发布版本，定义项目可使用的用户自定义域，添加和修改该项目用户及其权限属性。

2) 问题录入

在系统界面单击 Report Issue，可进入问题录入界面。如果在单击前，右上角项目选择为 All Project，那么在填报问题前需要先选择要填报的项目。可以选中 Make Default，这样在每次填报进入该界面时，所选择的的就是默认项目了。在问题填报界面选择并输入 Category、Reproducibility、Impact、Severity、Summary、Description、Additional Information 等信息，单击 Submit Report 按钮即可。在录入页面中还可以添加和上传附件。