第3章 软件缺陷管理

3.1 软件缺陷

软件缺陷(Software Defect),常常被叫作 Bug。软件缺陷是对软件产品预期属性的偏离现象。缺陷的存在会导致软件产品在某种程度上不能满足用户的需求。IEEE729-1983对缺陷的定义是:从产品内部看,缺陷是软件产品在开发和维护过程中存在的错误、缺点等问题;从产品外部来看,缺陷是系统所需要实现的某种功能的失效或违背。

缺陷的表现形式不仅体现在功能的失效方面,还体现在其他方面。主要类型有:软件没有实现产品规格说明所要求的功能模块;软件中出现了产品规格说明指明不应该出现的错误;软件实现了产品规格说明没有提到的功能模块;软件没有实现虽然产品规格说明没有明确提及但应该实现的目标;软件难以理解,不容易使用,运行缓慢,或从测试员的角度看,最终用户会认为不好。

软件缺陷是影响软件质量的重要因素之一,发现并排除缺陷是软件生命周期中的一项重要工作。

3.2 软件缺陷的属性

软件缺陷的属性包括缺陷标识、缺陷类型、缺陷严重级别、缺陷优先级、缺陷起源和缺陷状态等。

1. 缺陷标识

缺陷标识是标记某个缺陷的唯一标识,可通过该标识来跟踪缺陷。

2. 缺陷类型

缺陷的类型包括功能、用户界面、性能、接口、兼容性、文档、软件包等。

3. 缺陷严重级别

软件缺陷一旦被发现,就应该设法找出引起这个缺陷的原因,并分析对软件产品质量的影响程度,然后确定处理这个缺陷的优先顺序。一般来说,问题越严重,其处理的优先级越高,越需要得到及时的修复。

缺陷严重级别是指因缺陷引起的故障对被测试软件的影响程度。在软件测试中，缺陷的严重级别应该从软件最终用户的观点出发来判断，考虑缺陷对用户使用所造成的后果的严重性。由于软件产品应用的领域不同，因此软件企业对缺陷严重级别的定义也不尽相同。软件缺陷一般包括 5 个级别，如表 3-1 所示。

表 3-1 缺陷严重级别示例

缺陷级别	描述
严重缺陷(Critical)	不能执行正常工作功能或重要功能，使系统崩溃或资源严重不足。如： 1. 由程序所引起的死机，非法退出 2. 死循环 3. 数据库发生死锁 4. 错误操作导致的程序中断 5. 严重的计算错误 6. 与数据库连接错误 7. 数据通讯错误
较严重缺陷(Major)	严重地影响系统要求或基本功能的实现，且没有办法更正(重新安装或重新启动该软件不属于更正办法)。如： 1. 功能不符 2. 程序接口错误 3. 数据流错误 4. 轻微数据计算错误
一般缺陷(Average Severity)	影响系统要求或基本功能的实现，但存在合理的更正办法。如： 1. 界面错误(附详细说明) 2. 打印内容、格式错误 3. 简单的输入限制未放在前台进行控制 4. 删除操作未给出提示 5. 数据输入没有边界值限定或不合理
次要缺陷(Minor)	使操作者不方便或遇到麻烦，但它不影响执行工作或功能实现。如： 1. 辅助说明描述不清楚 2. 显示格式不规范 3. 系统处理未优化 4. 长时间操作未给用户进度提示 5. 提示窗口文字未采用行业术语
改进型缺陷(Enhancement)	1. 对系统使用的友好性有影响，例如名词拼写错误、界面布局或色彩问题、文档的可读性、一致性等 2. 改进建议

缺陷的严重级别可根据项目的实际情况制定，一般在系统需求评审通过后，由开发人员、测试人员等组成相关人员共同讨论，达成一致，为后续的系统测试的 Bug 级别判断提供依据。

4. 缺陷优先级

缺陷的优先级是指缺陷必须被修复的紧急程度。一般地，严重级别高的缺陷具有较高的优先级。严重性高说明缺陷对软件造成的质量危害大，需要优先处理，而严重性低的缺陷可能只是软件的一些局部的、轻微的问题，可以稍后处理。但是，严重级别和优先级并不总是一一对应的。有时候严重级别高的缺陷，优先级不一定高，而一些严重级别低的缺陷却需要及时处理，因此具有较高的优先级。

缺陷优先级如表 3-2 所示。

表 3-2 缺陷优先级示例

缺陷优先级	描　　述
I 级(Resolve Immediately)	缺陷必须被立即解决
II 级(Normal Queue)	缺陷需要正常排队等待修复或列入软件发布清单
III 级(Not Urgent)	缺陷可以在方便时被纠正

5. 缺陷起源

缺陷起源是指缺陷引起的故障或事件第一次被检测到的阶段。缺陷起源如表 3-3 所示。

表 3-3 缺陷起源示例

缺陷起源	描　　述	缺陷起源	描　　述
需求(Requirement)	在需求阶段发现的缺陷	代码(Code)	在编码阶段发现的缺陷
架构(Architecture)	在架构阶段发现的缺陷	测试(Test)	在测试阶段发现的缺陷
设计(Design)	在设计阶段发现的缺陷		

6. 缺陷状态

缺陷状态是指缺陷通过一个跟踪修复过程的进展情况。缺陷管理过程中的主要状态如表 3-4 所示。

表 3-4 缺陷状态示例

缺陷状态	描　　述
新缺陷(New)	已提交到系统中的缺陷
接受(Accepted)	经缺陷评审委员会的确认，认为缺陷确实存在
已分配(Assigned)	缺陷已分配给相关的开发人员进行修改
已打开(Open)	开发人员开始修改缺陷，缺陷处于打开状态
已拒绝(Rejected)	拒绝已经提交的缺陷，不需修复或不是缺陷或需重新提交
推迟(Postpone)	推迟修改
已修复(Fixed)	开发人员已修改缺陷
已解决(Resolved)	缺陷被修改，测试人员确认缺陷已修复
重新打开(Reopen)	回归测试不通过，再次打开状态
已关闭(Closed)	已经被修改并测试通过，将其关闭

除了以上主要状态外，在缺陷管理过程中，还存在其他一些状态。

Investigate(研究)：当缺陷分配给开发人员时，开发人员并不是都直接可以找到相关的解决方案的。开发人员需要对缺陷和引起缺陷的原因进行调查研究，这时候我们可以将缺陷状态改为研究状态。

Query & Reply(询问/回答)：负责缺陷修改的开发工程师认为相关的缺陷描述信息不够明确，或希望得到更多和缺陷相关的配置和环境条件，或引起缺陷时系统产生的调试命令和信息等。

Duplicate(重复)：缺陷评审委员会认为这个缺陷和某个已经提交的缺陷是同一个问题，因此设置为重复状态。

Reassigned(再分配)：缺陷需要重新分配。

Unplanned(无计划)：在用户需求中没有要求或计划。

Wontfix(不修复)：问题无法修复或者不用修复。

3.3 软件缺陷的类型

根据缺陷的自然属性来划分，缺陷可分为：功能问题、接口问题、逻辑问题、计算问题、数据问题、用户界面问题、兼容性问题等。缺陷类型的详细描述如表 3-5 所示。

表 3-5 软件缺陷的类型

编号	缺陷类型	描述	子类型	
			编号	名称
01	功能问题 F-Function	影响了重要的特性、用户界面、产品接口、硬件结构接口和全局数据结构，并且设计文档需要正式的变更。如指针、循环、递归、功能等缺陷	0101	功能错误
			0102	功能缺失
			0103	功能超越
			0104	设计二义性
			0105	算法错误
02	接口问题 I-Interface	与其他组件、模块或设备驱动程序、调用参数、控制模块或参数列表相互影响的缺陷	0201	模块间接口
			0202	模块内接口
			0203	公共数据使用
03	逻辑问题 L-Logic	需要进行逻辑分析，进行代码修改，如循环条件等	0301	分支不正确
			0302	重复的逻辑
			0303	忽略极端条件
			0304	不必要的功能
			0305	误解
			0306	条件测试错误
			0307	循环不正确
			0308	错误的变量检查
			0309	计算顺序错误
			0310	逻辑顺序错误

续表

编号	缺陷类型	描　　述	子类型	
			编号	名　称
04	计算问题 C-Computation	等式、符号、操作符或操作数错误，精度不够、不适当的数据验证等缺陷	0401	等式错误
			0402	缺少运算符
			0403	错误的操作数
			0404	括号用法不正确
			0405	精度不够
			0406	舍入错误
			0407	符号错误
05	数据问题 A-Assignment	需要修改少量代码，如初始化、控制块、声明、重命名、范围、限定等缺陷	0501	初始化错误
			0502	存取错误
			0503	引用的错误变量
			0504	数组引用越界
			0505	不一致的子程序参数
			0506	数据单位不正确
			0507	数据维数不正确
			0508	变量类型不正确
			0509	数据范围不正确
			0510	操作符数据错误
			0511	变量定位错误
			0512	数据覆盖
			0513	外部数据错误
			0514	输出数据错误
			0515	输入数据错误
			0516	数据检验错误
06	用户界面问题 U-User Interface	人机交互特性：屏幕格式、确认用户输入、功能有效性、页面排版等方面的缺陷	0601	界面风格不统一
			0602	屏幕上的信息不可用
			0603	屏幕上的错误信息
			0604	界面功能布局和操作不合常规
07	文档问题 D-Document	影响发布和维护，包括注释等缺陷	0701	描述含糊
			0702	项描述不完整
			0703	项描述不正确
			0704	项缺少或多余
			0705	项不能验证
			0706	项不能完成
			0707	不符合标准
			0708	与需求不一致
			0709	文字排版错误
			0710	文档信息错误
			0711	注释缺陷
08	性能问题 P-Performance	不满足系统可测量的属性值，如执行时间、事务处理速率等缺陷		

续表

编号	缺陷类型	描　　述	子类型	
			编号	名　称
09	配置问题 B-Build/ package/merge	由配置库、变更管理或版本控制引起的错误	0901	配置管理问题
			0902	编译打包缺陷
			0903	变更缺陷
			0904	纠错缺陷
10	标准问题 N-Norms	不符合各种标准的要求，如编码标准、设计符号等缺陷	1001	不符合编码标准
			1002	不符合软件标准
			1003	不符合行业标准
11	环境问题 E-Environments	由于设计、编译和运行环境引起的问题	1101	设计、编译环境
			1102	运行环境
12	兼容性问题	软件之间不能正确地交互和共享信息	1201	操作平台不兼容
			1202	浏览器不兼容
			1203	分辨率不兼容
13	其他问题 O-Others	以上问题不包含的其他问题		

3.4 软件缺陷管理

软件缺陷管理(Defect Management)是在软件生命周期中识别、管理、沟通任何缺陷的过程(从缺陷的识别到缺陷的解决和关闭)，确保缺陷被跟踪管理而不丢失。缺陷管理主要实现以下目标：

(1) 及时了解并跟踪每个被发现的缺陷；

(2) 确保每个被发现的缺陷都能被处理；

(3) 收集缺陷数据并根据缺陷趋势曲线识别测试过程阶段；

(4) 收集缺陷数据并在其上进行数据分析，作为组织过程的财富。

一般地，软件缺陷管理需要跟踪管理工具来帮助进行缺陷全流程管理。

1. 缺陷管理中的角色

在缺陷管理中涉及测试人员和开发人员，他们在缺陷管理中扮演着不同的角色，完成相应的工作。

1) 项目测试负责人

项目测试负责人负责制定缺陷管理计划和流程，将测试工程师发现的问题指派给指定开发工程师，协调缺陷管理流程中的问题。

2) 测试工程师

测试工程师将发现的问题提交到缺陷管理系统中，写明问题的描述、严重程度，以及问题重现方法，并负责重新测试开发工程师修改过的缺陷。

3）开发工程师

开发工程师需要确认并修改指定给自己的软件缺陷。

4）质量工程师

质量工程师监控项目组缺陷管理规程执行情况。

2. 缺陷管理流程

为正确跟踪软件中缺陷的处理过程，通常将软件测试中发现的缺陷作为记录输入到缺陷跟踪管理系统。在缺陷管理系统中，缺陷的状态主要有提交、确认、拒绝、修正和已关闭等。缺陷的生命周期一般要经历从被发现和报告，到被打开和修复，再到被验证和关闭等过程。缺陷的跟踪和管理一般借助于工具来实施。缺陷管理的一般流程如图 3-1 所示。

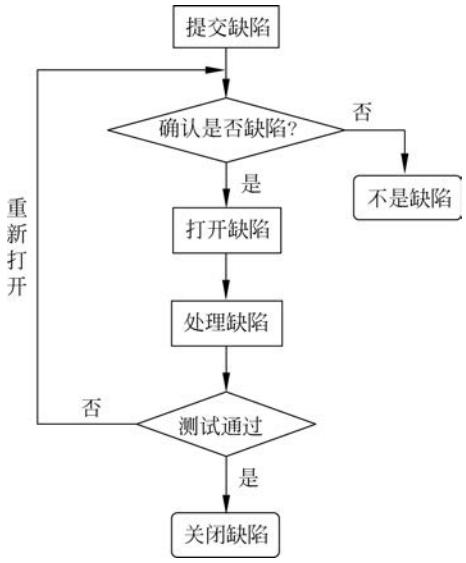

图 3-1　缺陷管理的一般流程

缺陷管理的流程说明如下。

（1）测试人员发现软件缺陷，提交新缺陷入库，缺陷状态设置为 New。

（2）软件测试经理或高级测试经理对新提交的缺陷进行确认。若确认是缺陷，则分配给相应的开发人员，将缺陷状态设置为 Open 状态。若不是缺陷（或缺陷描述不清楚），则拒绝，设置为 Declined 状态。

（3）开发人员对标记为 Open 状态的缺陷进行确认，若不是缺陷，状态修改为 Declined，若是缺陷，则进行修复，修复后将缺陷状态改为 Fixed。对于不能解决的缺陷，提交到项目组会议评审，以做出延期或进行修改等决策。

（4）测试人员查询状态为 Fixed 的缺陷，然后通过测试（即回归测试）验证缺陷是否已解决。如果缺陷已经解决，则将此缺陷的状态置为 Closed。如果缺陷依然存在或者还引入了新的缺陷，则置缺陷状态为 Reopen。

异常过程：对于被验证后已经关闭的缺陷，由于种种原因被重新打开，测试人员将此类缺陷标记为 Reopen，重新经历修正和测试等阶段。

在缺陷管理过程中，应加强测试人员与开发人员之间的交流，对于那些不能重现的缺陷

或很难重现的缺陷，可以请测试人员补充必要的测试用例，给出详细的测试步骤和方法。同时，还需要注意下列细节。

(1) 软件缺陷跟踪过程中的不同阶段是测试人员、开发人员、配置管理人员和项目经理等协调工作的过程，要保持良好的沟通，尽量与相关的各方人员达成一致。

(2) 测试人员在评估软件缺陷的严重性和优先级上，要根据事先制定的相关标准或规范来判断，应具独立性、权威性，若不能与开发人员达成一致，由产品经理来裁决。

(3) 当发现一个缺陷时，测试人员应分给相应的开发人员。若无法判断合适的开发人员，应先分配给开发经理，由开发经理进行二次分配。

(4) 一旦缺陷处于修正状态，需要测试人员的验证，而且应围绕该缺陷进行相关的回归测试，并且包含该缺陷修正的测试版本是从配置管理系统中下载的，而不是由开发人员私下给的测试版本。

(5) 只有测试人员有关闭缺陷的权限，开发人员没有这个权限。

3. 缺陷描述

测试人员发现缺陷后，需要对缺陷进行翔实的描述。对缺陷的描述一般包含以下内容。

(1) 缺陷 ID。唯一的缺陷标示符，可以根据该 ID 追踪缺陷。

(2) 缺陷标题。描述缺陷的名称。

(3) 缺陷状态。标明缺陷所处的状态，如“新建”“打开”“已修复”“关闭”等。

(4) 缺陷的详细说明。对缺陷进行详细描述，说明缺陷复现的步骤等。对缺陷描述的详细程度直接影响开发人员对缺陷的修改，描述应该尽可能详细。

(5) 缺陷的严重程度。指因缺陷引起的故障对软件产品的影响程度。

(6) 缺陷的紧急程度。指缺陷必须被修复的紧急程度(优先级)。

(7) 缺陷提交人。缺陷提交人的名字。

(8) 缺陷提交时间。缺陷提交的时间。

(9) 缺陷所属项目/模块。缺陷所属的项目和模块，最好能较精确地定位至模块。

(10) 缺陷解决人。最终解决缺陷的人。

(11) 缺陷处理结果描述。对处理结果的描述，如果对代码进行了修改，要求在此处体现出修改的内容。

(12) 缺陷处理时间。缺陷被修正的时间。

(13) 缺陷复核人。对被处理缺陷复核的验证人。

(14) 缺陷复核结果描述。对复核结果的描述(通过、不通过)。

(15) 缺陷复核时间。对缺陷复核的时间。

(16) 测试环境说明。对测试环境的描述。

(17) 必要的附件。对于某些文字很难表达清楚的缺陷，使用图片等附件是必要的。

除上述描述项外，配合不同的统计的角度，还可以添加“缺陷引入阶段”“缺陷修正工作量”等属性。

4. 缺陷报告原则

缺陷报告是测试过程中提交的最重要的东西，它的重要性丝毫不亚于测试计划，并且比

测试过程中产生出的其他文档对产品质量的影响更大。对缺陷的描述要求准确、简洁、步骤清楚、有实例、易再现、复杂问题有据可查(截图或其他形式的附件)。

有效的缺陷报告需要做到以下几点。

(1) 单一。每个报告只针对一个软件缺陷。

(2) 完整。提供完整的缺陷描述信息。

(3) 简洁。使用专业语言,清晰而简短地描述缺陷,不要添加无关的信息。确保所包含信息是最重要的,而且是有用的,不要写无关信息。

(4) 客观。用中性的语言客观描述事实,不带偏见,不用幽默或者情绪化的语言。

(5) 可再现。不要忽视或省略任何一项操作步骤,特别是关键性的操作一定要描述清楚,确保开发人员按照所描述的步骤可以再现缺陷。

(6) 特定条件。必须注明缺陷发生的特定条件。

5. 缺陷报告模板

软件缺陷报告是软件测试过程中最重要的文档。它记录了缺陷发生的环境,如各种资源的配置情况、Bug 的再现步骤以及 Bug 性质的说明。更重要的是它还记录着 Bug 的处理进程和状态。Bug 的处理进程从一定角度反映了测试的进程和被测软件的质量状况以及改善过程。

缺陷报告模板如附表 A-4 所示。

3.5 软件缺陷度量

3.5.1 缺陷数据分析

通过对缺陷的分析,可以了解更多的产品质量信息。同时可以根据缺陷的状态来判断测试的进展情况和开发人员的编程质量,修正缺陷的进度等。另外,通过对缺陷的分析,还可以完成对产品质量的评估,确定测试是否到达结束的标准,也即软件的质量是否达到可发布水平。最常用的缺陷分析方法可以分为两类:缺陷趋势分析和缺陷分布分析。

1. 缺陷趋势分析

缺陷趋势分析是在时间轴上对缺陷进行分析,有助于进度控制和测试过程的管理。缺陷趋势分析主要考察缺陷随时间变化的趋势,如将缺陷的各种状态的数量作为时间的函数进行二维显示。

在一个成熟的软件开发过程中,缺陷趋势一般会遵循一种和预测比较接近的模式向前发展。在测试初期,缺陷数量的增长率较高,但达到峰值后,缺陷将随时间以较低速率降低,如图 3-2 所示。

从图中可以看到,当每天发现的新缺陷的数量呈下降趋势(假定工作量是恒定的),则每发现一个缺陷所消耗的成本会呈现出上升的趋势。所以,到某个点以后,继续进行测试的成本将会超过进行额外测试所需要的成本。发布日期的确定就是对这种情况的时间进行估计。在进行发布日期估计过程中,要考虑以下因素:

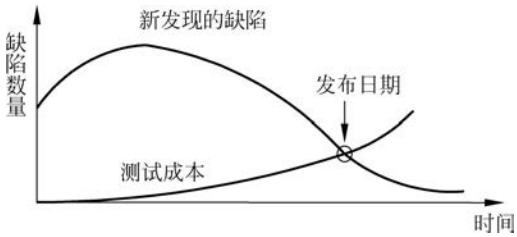

图 3-2 合理的缺陷趋势分布图

(1) 未发现 Bug 的级别是未知的,这可采用基于风险的技术,可在一定程度上弥补其中的不足。

(2) 测试中所发现 Bug 级别的趋势,采用基于风险的技术,能期待缺陷发现率下降,还能期待发现的缺陷的严重级别也在下降。若没有这种趋势,则说明系统还不能交付使用,即没有达到可发布标准。

2. 缺陷分布分析

缺陷分布是缺陷的横向分布,即空间上的分布。缺陷分布分析可以针对一个或多个缺陷分类进行缺陷分析。定义缺陷分类可能有多个维度,分析时可以将发现的缺陷按照所属功能模块、缺陷严重程度、缺陷来源、缺陷类型、注入阶段、发现阶段、修复阶段、缺陷性质等方面进行分类和统计,计算各种缺陷的分布情况。

按缺陷所属功能模块分析,可以了解哪些功能模块处理比较难,了解哪些功能模块程序的质量比较差等。分析缺陷分布,不仅可以帮助测试经理决定哪些功能领域和性能领域需要增强测试,而且还可以使开发人员的注意力集中到频繁产生缺陷的模块或单元。

按缺陷来源分析,可以帮助找出缺陷产生的根本原因,从而可以更新相关检查列表。

总之,通过缺陷分布分析,可以进一步优化测试时间的分配,以及改进软件开发流程等,有重点地避免缺陷发生,提高软件产品的质量。

3. 缺陷密度分析

根据缺陷集群的原则,在测试中发现缺陷多的地方,其潜在的缺陷也会更多。这个原则背后的原因在于:在发现缺陷多的地方,漏掉的缺陷可能性也会大,或者说测试效率没有被显著改善之前,在纠正缺陷时将引入较多的错误。其数学表达就是缺陷密度的度量——缺陷密度。缺陷密度越低意味着产品质量越高。

缺陷密度(Dd)是以每千行代码的缺陷数(Defects/KLOC)来测量的,其测量单位是 defects/KLOC。缺陷密度的定义如下:

$$缺陷密度 = 缺陷数量 / 代码行 \tag{3-1}$$

可按照以下步骤来计算一个程序的缺陷密度:

(1) 统计开发过程中每个阶段发现的缺陷总数(D)。

(2) 统计程序中新开发的和修改的代码行数(N)。

(3) 计算每千行的缺陷数 Dd=1000×D/N。例如,一个 10 万行的源程序总共有 50 个缺陷,则缺陷密度是:Dd=1000×50/100000=0.2defects/KLOC。

缺陷密度是软件缺陷的基本度量，可用于设定产品质量目标，支持软件可靠性模型预测潜伏的软件缺陷，进而对软件质量进行跟踪和管理，支持基于缺陷计数的软件可靠性增长模型，对软件质量目标进行跟踪并评判能否结束软件测试。

如果缺陷密度跟上一个版本相同或更低，就应该分析当前版本的测试效率是不是降低了？如果不是，意味着质量的前景是乐观的；如果是，那么就需要额外的测试，还需要对开发和测试的过程进行改善。

如果缺陷密度比上一个版本高，那么就应该考虑在此之前是否为显著提高测试效率进行了有效的策划，并在本次测试中得到实施？如果是，虽然需要开发人员更多的努力去修正缺陷，但质量还是得到更好的保证；如果没有，意味着质量恶化，质量很难得到保证。这时，要保证质量，就必须延长开发周期或投入更多的资源。

3.5.2 测试有效性度量

如何对软件测试工程师的工作进行考核，业界一直在进行探索和思考。有的企业采用测试工程师发现的缺陷数量作为绩效考核的指标之一。采用此方法很容易导致开发工程师和测试工程师的对立，同时对缺陷的严重级别如何划定更容易造成测试工程师和开发工程师的对立，从而导致测试工程师不去真正重视测试的质量，而仅仅重视缺陷发现的数量。

对测试有效性的度量通常采用缺陷消除率(DRE)和缺陷损耗等指标来进行。

1. 缺陷消除率

对于一个软件工作产品，软件缺陷可以分为通过评审或测试等方法发现的已知缺陷(Known Defect)和尚未发现的潜在缺陷(Latency Defect)两种。

将测试期间发现的缺陷和尚未发现的缺陷两个缺陷指标，可以构造一个测试有效性的度量指标，即缺陷消除率：

$$\mathrm{DRE}=\frac{\text{测试期间发现的缺陷数量}}{\text{测试期间发现的缺陷数量}+\text{未发现的缺陷数量}} \tag{3-2}$$

未发现的缺陷数量通常等于客户发现的缺陷数量(尽管客户也不可能发现所有的缺陷)。DRE 是测试有效性的一个非常好的度量，但必须考虑以下问题。

(1) 缺陷的严重级别和分布状况。

(2) 观察客户在以前项目或版本中报告的缺陷趋势，以确定客户发现“绝大多数的”缺陷所需要的时间。

(3) 虽然这种度量属于马后炮性质的度量，对当前项目的测试有效性度量毫无裨益，但对自己所在组织的测试有效性的长期趋势进行度量就非常有意义。

(4) 计算缺陷数量时，在计算过程中采用同一起点，以便于比较。

DRE 有时还用来度量某一特定测试等级的有效性，如系统测试的 DRE，这时就应把在系统测试中发现的缺陷数作为分子，并把这个数与系统测试中未发现的缺陷数之和作为分母。

2. 缺陷损耗

测试有效性的另外一个有用的度量是缺陷潜伏期，通常也称为阶段潜伏期。缺陷潜伏期是一种特殊类型的缺陷分布度量。从软件测试的原则中可知，发现缺陷的时间越晚，则该

缺陷带来的损失就越大，修复该缺陷的成本也就越高。表 3-6 显示了某个项目度量缺陷潜伏期的尺度。需要注意的是，在不同的组织和项目，可对这个尺度进行适当调整，以反映企业所在项目开发过程各个阶段、各个测试等级的数量和名称。

表 3-6 某项目的缺陷潜伏期尺度

缺陷造成阶段	发现阶段								
	需求	概要设计	详细设计	编码	单元测试	集成测试	系统测试	β测试	产品推广
需求	0	1	2	3	4	5	6	7	8
概要设计		0	1	2	3	4	5	6	7
详细设计			0	1	2	3	4	5	6
编码				0	1	2	3	4	5

缺陷损耗是修复缺陷所耗费的成本的一种度量。缺陷损耗可定义为

$$\text{损耗} = \frac{\text{缺陷数量} \times \text{发现的阶段潜伏期尺度}}{\text{缺陷总量}} \tag{3-3}$$

一般而言，损耗的数值越低，说明缺陷的发现过程就越有效，最理想的数值应是 1。作为一个绝对值，损耗几乎没有任何意义，但用损耗来度量测试有效性的长期趋势时，就显示出它的价值。

3.6 软件缺陷管理工具

1. ClearQuest

IBM 公司的 ClearQuest 提供基于活动的变更和缺陷跟踪。ClearQuest 以灵活的工作流管理所有类型的变更要求，包括缺陷、改进、问题和文档变更，能够方便地定制缺陷和变更请求的字段、流程、用户界面、查询、图表和报告，并提供了预定义的配置和自动电子邮件通知和提交。ClearQuest 与 Rational ClearCase 一起提供完整的 SCM 解决方案，拥有“设计一次，到处部署”的能力，从而可以自动改变任何客户端界面（Windows、Linux、UNIX 和 Web）。ClearQuest 可与 IBM WebSphereStudio、Eclipse 和 Microsoft .NET IDE 进行紧密集成，从而可以即时访问变更信息。支持统一变更管理，以提供经过验证的变更管理过程支持。易于扩展，因此无论开发项目的团队规模、地点和平台如何，均可提供良好支持。包含并集成于 IBM Rational Suite 和 IBM Rational Team Unifying Platform，提供生命周期变更管理。

网站地址为 http://www.ibm.com/software/rational。

2. Mantis

Mantis 是一个基于 PHP 技术的轻量级的开源缺陷跟踪系统，以 Web 操作的形式提供项目管理及缺陷跟踪服务。在功能上和实用性上足以满足中小型项目的管理及跟踪。

Mantis 易于安装，易于操作，基于 Web，支持任何可运行 PHP 的平台（Windows、Linux、Mac、Solaris、AS400/i5 等），支持多个项目，为每一个项目设置不同的用户访问级别，跟踪缺陷变更历史，定制我的视图页面，提供全文搜索功能，内置报表生成功能（包括图形报表），通

过 Email 报告缺陷，用户可以监视特殊的 Bug，附件可以保存在 Web 服务器上或数据库中（还可以备份到 FTP 服务器上），自定义缺陷处理工作流，支持输出格式包括 csv、Microsoft Excel、Microsoft Word，集成源代码控制（SVN 与 CVS），集成 WiKi 知识库与聊天工具（可选/可不选），支持多种数据库（MySQL、MS SQL、PostgreSQL、Oracle、DB2），提供 Web Service（SOAP）接口，提供 Wap 访问。

网站地址为 http://www.mantisbt.org/。

3. Bugzilla

Bugzilla 是一个开源免费的 Bug 管理工具。作为一个产品缺陷的记录及跟踪工具，它能够建立一个完善的 Bug 跟踪体系，包括报告 Bug、查询 Bug 记录并产生报表、处理解决、管理员系统初始化和设置 4 部分。Bugzilla 具有如下特点。

（1）基于 Web 方式，安装简单，运行方便快捷，管理安全。

（2）有利于缺陷的清楚传达。该系统使用数据库进行管理，提供全面详尽的报告输入项，产生标准化的 Bug 报告。提供大量的分析选项和强大的查询匹配能力，能根据各种条件组合进行 Bug 统计。

（3）系统灵活，强大的可配置能力。Bugzilla 工具可以对软件产品设定不同的模块，并针对不同的模块设定开发人员和测试人员，这样可以实现提交报告时自动发给指定的责任人，并可设定不同的小组，权限也可划分。设定不同的用户对 Bug 记录的操作权限不同，可有效进行管理。允许设定不同的严重程度和优先级在 Bug 的生命期中管理 Bug，从最初的报告到最后的解决，确保了 Bug 不会被忽略，同时可以使注意力集中在优先级和严重程度高的 Bug 上。

（4）自动发送 Email，通知相关人员。根据设定的不同责任人，自动发送最新的动态信息，有效地帮助测试人员和开发人员进行沟通。

网站地址为 https://www.bugzilla.org/download/。

4. JIRA

JIRA 是 Atlassian 公司出品的项目与事务跟踪工具，被广泛应用于缺陷跟踪、客户服务、需求收集、流程审批、任务跟踪、项目跟踪和敏捷管理等工作领域。JIRA 配置灵活、功能全面、部署简单、扩展丰富，多语言支持，界面友好和其他系统（如 CVS、Subversion（SVN）Perforce、邮件服务等）整合得相当好，文档齐全，安装配置简单，可用性以及可扩展性方面都十分出色，拥有完整的用户权限管理。

JIRA 推出云服务和下载版，均提供 30 天的免费试用期。

网站地址为 https://www.atlassian.com/software/jira。

3.7 Mantis 的安装及使用

3.7.1 Mantis 简介

Mantis 是一个开源的 Bug 管理系统，基于 PHP＋MySQL，可以运行在 Windows/

UNIX 平台上。Mantis 是 B/S 结构的 Web 系统，可以配置到 Internet 上，实现异地 Bug 管理。

1. Mantis 基本特性

(1) 个人可定制的 Email 通知功能，每个用户可根据自身的工作特点只订阅相关缺陷状态邮件。

(2) 支持多项目。

(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) 具有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，如果不能满足要求，可以把数据输出到 Excel 中进一步分析。

(8) 流程定制方便且符合标准，满足一般的缺陷跟踪。

(9) 可以实现与 CVS 集成，将缺陷和 CVS 仓库中文件实现关联。

(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。在缺陷跟踪管理过程中，将缺陷记录划分为不同的阶段和状态来进行标记。Mantis 系统将缺陷的处理状态分为 New(新建)、Feedback(反馈)、Acknowledged(认可)、Confirmed(已确认)、Assigned(已分派)、Resolved(已解决)、Closed(已关闭) 7 种，如图 3-3 所示。

New(新建)：一个新的缺陷被提交。

Feedback(反馈)：对此 Bug 存有异议，就将其反馈。由测试人员和开发人员讨论评估后，决定是否将其关闭。

Acknowledged(认可)：经理认为报告员提交的问题是个 Bug，对这个 Bug 表示认可。

Confirmed(已确认)：开发人员确认存在此 Bug，并准备修改，将其设为已确认。

Assigned(已分派)：经理将认可的问题单分派给某个开发人员。

Resolved(已解决)：被分派的开发人员已经进行修改，测试人员可以进行验证测试，确认 Bug 已经解决。

Closed(已关闭)：Bug 修改后，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。

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，例如项目经理、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权限不同。

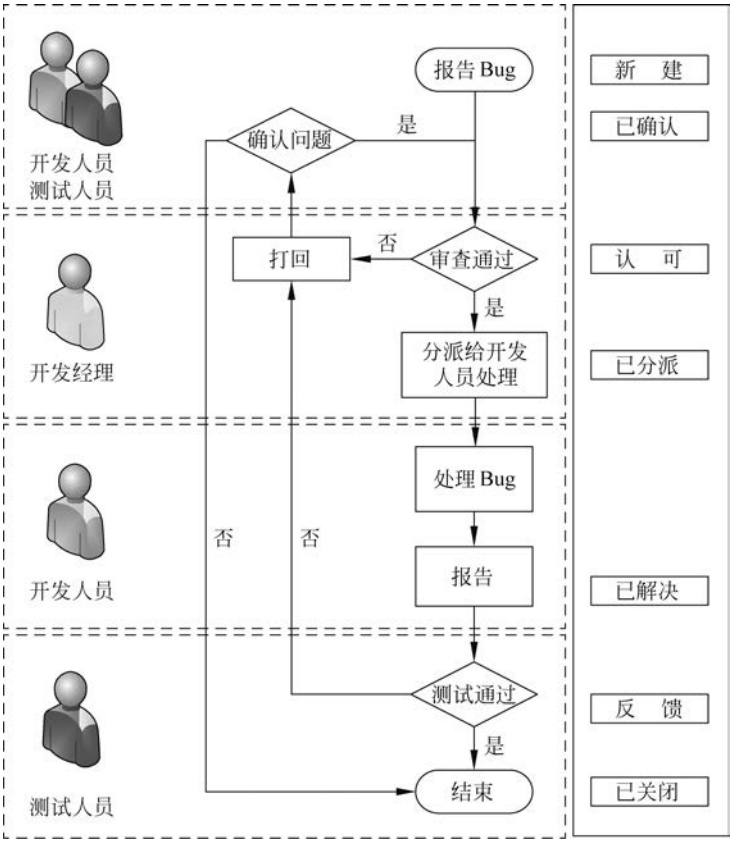

图 3-3 Mantis 缺陷状态转换图

在 Mantis 系统中，分别有下列几种角色：管理员、开发经理、开发人员、修改人员、报告人员、查看人员。

Mantis 中用户角色和权限如表 3-7 所示。权限的从大到小依次排列是：管理员→项目经理→开发人员→修改人员→报告人员→查看人员。

表 3-7 Mantis 中用户角色和权限

用 户	权 限
管理员(Administrator)	管理和维护整个系统
开发经理(Manager)	对整个软件项目进行管理
开发人员(Developler)	负责软件项目的开发
修改人员(Updater)	负责修改 Issue(问题)
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

在一个项目组或团队中，不同的人有不同的职责和分工，在 Mantis 中对应不同的角色，其角色和权限可以由管理员进行设置。

4. Mantis 软件缺陷属性的定义

Mantis 的软件缺陷属性的定义如下。

(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。描述的是发现该缺陷的测试版本号。

(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,如新建、反馈、已分派、已确认、已关闭等。

(6) 报告者。报告缺陷的测试人员的编号或用户名。

(7) 报告日期。缺陷填报的日期。

(8) 重现性。可重现或不可重现。

(9) 重现步骤。和测试用例相关,描述的是发现这个缺陷的步骤。

(10) 严重等级。可定制,默认为 4 级,Pl(致命)、P2(严重)、P3(一般)、P4(轻微)。

(11) 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。

(12) 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。

3.7.2 Mantis 的安装

1. 安装 Mantis

安装 Mantis 之前需要安装 Apache 服务器、MySQL 和 PHP 运行环境,本书采用 XAMP 集成环境,其安装步骤见 2.7.1 节。Mantis 的安装与 TestLink 的安装类似。

在 Mantis 官方网站(http://www.mantisbt.org/)上下载最新版本软件系统,这里下载的是 mantisbt-1.2.19。

安装好 XAMP 之后,将 Mantis 的安装文件解压到"C:\xampp\htdocs"目录下,并将文件名改为 mantis。打开 IE 浏览器,在地址栏中输入"http://localhost/mantis",即可进入安装页面,如图 3-4 所示。

在 Password(for Database)输入框中输入密码,然后单击 Install/Upgrade Database 按钮,进入安装检查页面。如果后面的状态栏全部为绿色,则安装成功。注:Password(for Database)的密码是安装 XAMP 时设置的数据库密码。

在 IE 地址栏中输入"http://localhost/mantis/",即可进入登录页面,如图 3-5 所示。

首次进入登录页面,会出现下列提示。

Warning: You should disable the default 'administrator' account or chang its password.(警告:建议禁止缺省管理员账号或修改账号密码。)

Warning: Admin directory should be removed.(警告:建议删除 admin 的目录。)

使用 Mantis 默认的用户名(administrator)和密码(root)登录系统,进入我的账户(My Account),修改密码,然后退出 Mantis 系统。将 Mantis 安装路径 C:\xampp\htdocs\mantis 中的 admin 文件夹删除。重新打开 Mantis 登录页面,此时页面中将不再有警告信息。

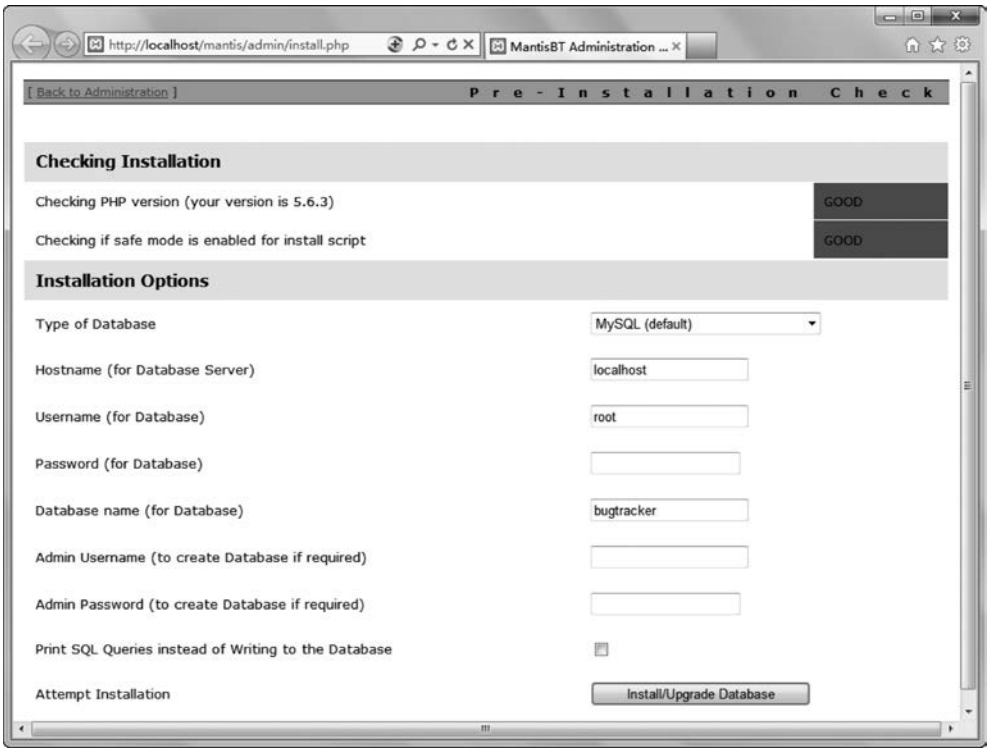

图 3-4　Mantis 安装页面

图 3-5　Mantis 登录页面

2. 配置

在 mantis 目录下新建配置文件 config_inc. php，在里面进行数据库配置、邮件服务配置和语言配置。配置文件加载顺序：先加载 config_defaults_inc. php，后加载 config_inc. php。config_inc. php 中的值会覆盖 config_defaults_inc. php。在 config_inc. php 中撰写下列代码。

```
1   ##########################################
2   # Database Configuration   数据库配置
3   ##########################################
4   $g_hostname = 'localhost';
5   $ g_db_type = 'mysql';
6   $ g_database_name = 'mantis';
7   $ g_db_username = 'root';
8   $ g_db_password = '';                    #以上内容由系统自动生成,不用修改
9   ##########################################
10  # Mantis Email Configuration   邮件服务配置
11  ##########################################
12  $g_phpMailer_method = PHPMAILER_METHOD_SMTP;
13  # select the method to mail by:   0 - mail()   1 - sendmail   2 - SMTP
14  $g_phpMailer_method   = 2;               #以 smtp 发送邮件
15  $g_smtp_host     = 'smtp.163.com:25';    #邮件服务器的地址,后面加上端口号 25
16  $g_smtp_username = 'xxxxx';              #邮箱的用户名
17  $g_smtp_password = '*****';              #邮箱的密码
18  $g_administrator_email = 'xxxx@163.com';  #xxxx@xxx.com 是要修改为相应的邮箱名称
19  $g_webmaster_email   = 'xxxx@163.com';  #xxxx@xxx.com 是要修改为相应的邮箱名称
20  $g_from_name     = 'Mantis Bug Tracker';
21  # the 'From: ' field in emails
22  $g_from_email     = 'xxxx@163.com';      #xxxx@xxx.com 是要修改为相应的邮箱名称
23  # the return address for bounced mail
24  $g_return_path_email = 'xxxx@163.com';  #xxxx@xxx.com 是要修改为相应的邮箱名称
25  $g_email_receive_own = OFF;
26  $g_email_send_using_cronjob = OFF;
27  # allow email notification
28  $g_enable_email_notification = ON;
29  ##########################################
30  # Language Configuration   语言设置
31  ##########################################
32  $g_default_language = 'chinese_simplified';  #设置语言为中文
```

邮件系统的配置建议用 SMTP 方式。一般公司都有自己的邮件服务器,让管理员提供一个 Mantis 的专用信箱。本例采用的是 163 邮件服务。

【注】 如果 Mantis 不使用邮件系统(Email),修改配置文件 config_inc.php 中的语句:

```
$g_enable_email_notification = ON;
```

将其改为:

```
$g_enable_email_notification = OFF;
```

然后保存此文件。

如果不使用邮件系统,用户创建和管理的方法如下。

(1) 以管理员身份登录 Mantis 系统,输入账号和真实姓名,创建用户。此时新创建用户的密码为空,可以由新创建的用户登录 Mantis 系统后自行修改。

(2) 如果用户忘记了密码,可以让管理员登录 Mantis 系统,进入管理→用户管理→选择用户→重设密码,则该用户的密码将被置为空,由该用户登录后修改。

3.7.3 管理员的操作

管理员是管理整个系统运作的工作人员，他不仅是整个系统操作流程中权限最高的工作人员，而且还可以对项目和用户账户进行创建和管理等，下面将详细说明。管理员登录到系统之后，可以先进入自己的主界面，然后再根据工作要求，选择页面上方的菜单栏来进入相应的界面。

1. 我的视图

在系统界面，单击菜单栏中的“我的视图”项，管理员将会看到以下界面，如图 3-6 所示。

图 3-6 我的视图

从页面上看，Bug 根据其工作状态被分类成几个表格来显示，符合这些工作状态的 Bug 都被一一罗列。

(1) 分派给我的(未解决的)(assigned)。

(2) 未分派的(unassigned)。

(3) 我报告的(reported by me)。

(4) 已解决的(resolved)。

(5) 最近修改(recently modified)。

(6) 我监视的(monitored by me)。

在页面上还可以进行下列操作。

(1) 切换项目。单击页面右上角“项目”的下拉式菜单，选择其中的项目，然后单击“切换”按钮，来切换所选项目。

(2) 跳转到某问题。在页面右上角，“问题 #”文本框中输入问题编号，单击“前往”按钮，可根据问题编号进行查询，直接进入该问题的详细信息界面，可进行相应操作。

(3) 转向其他操作界面。单击主页面上方的菜单栏，即可进入到相应的操作界面。

2. 查看问题

1）查看问题

在系统界面，单击菜单栏中的“查看问题”项，可以进入问题查询结果页面，如图 3-7 所示。

图 3-7　查看问题

页面上部相当于一个过滤器，页面下部是根据过滤器显示 Bug 的数据列表。如果管理员没有对给予的参数选择设置，那么默认就是没有对数据进行过滤，则页面下部则显示所有 Bug 数据。此外，还可以在搜索框中输入 Bug 编号直接查询，那么在页面下部就会出现查询结果。

用户可以自己创建过滤器。在对参数进行设置完成后，可对当前的过滤设置进行保存，如图 3-8 所示。填入相应内容后即可保存下来。如果标记为公有，则其他工作人员（除了查看人员）都能共享这个过滤器。

图 3-8　创建过滤器

在显示 Bug 的数据列表中可以看到，页面下部的表格头部显示查看问题的当前数量，并且在旁边提供了“打印报告”“导出为 CSV”和“导出为 Excel”功能链接。在 Bug 表中显示了下列信息：Bug 优先级、Bug 编号、Bug 分类、Bug 严重性、Bug 状态、最后更新、Bug 摘要。

在此表格还可以进行下列操作。

（1）打印报告。单击“打印报告”进入打印报告页面，如图 3-9 所示。在页面中列出了需要导出打印的 Bug 列表，可以根据需要通过复选框选中需要打印的 Bug。在选择完毕后，根据需要单击 Word 图标或网页图标，Bug 数据便相应地导出到该类型的文件中，实现打印输出的需求。

（2）导出为 CSV 或者导出为 Excel。单击功能链接，可将报告保存为相应格式文件，并下载存储到本地。

图 3-9　打印报告

(3) 按指定方式排序。单击标题栏的列属性,可以进行排列,并出现上三角形图标或下三角形图标代表是按升序或降序排列。

(4) 更新问题属性。可以通过选中 Bug 列表的复选框,也可以选中“全选”的复选框,然后选择下拉列表中的操作命令,再单击“确定”按钮,则可以对这些 Bug 进行相应操作,如图 3-10 所示。

图 3-10　更新问题属性

① 移动。可以把选中的 Bug 从当前项目转移到别的项目中。

② 复制。可以把选中的 Bug 从当前项目复制到别的项目中。

③ 分派。可以把选中的 Bug 直接分派给指定的工作人员。

④ 关闭。当被选中的 Bug 确认已解决,或确认不是 Bug,管理员可以直接采用这个命令将 Bug 状态设为关闭。

⑤ 删除。当被选中的 Bug 是垃圾数据,在整理数据的时候可以直接进行删除。

⑥ 解决。如果 Bug 确认已经解决,则选中将其状态置为“解决”。但是如果 Bug 当前状态为“已解决”或以上状态,则不能进行此项操作。

⑦ 更新优先级。使用这项操作,可以更新选中 Bug 的优先级。

⑧ 更新状态。使用这项操作,可以更新选中 Bug 的流程状态。

⑨ 更新视图状态。使用这项操作,可以将选中的 Bug 重新设置为公共或私有状态。

【注】 对上述命令的操作和当前用户的权限有关系,如果不能进行该命令的操作,系统会出现“你无权执行这项操作”的提示性语句。而对于管理员来说,他具有完全的权限。

(5) 在 Bug 列表中,在每个 Bug 编号的前面,有一个编辑图标 ,可以单击此图标进入这个 Bug 信息的修改界面。注意:是否能出现这个图标和权限有关系。

（6）单击 Bug 编号可以直接进入其详细信息界面。

（7）单击注释数目可以直接进入对应 Bug 的详细信息界面，并将界面焦点定位在 Bug 注释信息。

2）问题更新

单击页面中的问题编号，可进入该问题的详细页面，并对该问题进行修改，图 3-11 所示。

图 3-11　问题更新

在这里可进行下列操作。

(1) 编辑(Edit)。修改问题的各项基本属性,并添加注释。

(2) 分派给(Assign To)。将问题分派给某个开发人员处理,分派之后系统将自动向被分派人发送邮件通知,被分派人打开 Mantis 之后将在“我的视图”页面看到被分派的问题。

(3) 状态改为(Change Status to)。这里是指问题状态的转变。状态包括新建、反馈、认可、已确认、已解决和已关闭。这是 Mantis 比较重要的一个功 能,问题的每次变动都会发生状态的改变,以此来标记问题的处理情况。

(4) 监视(Monitor)。单击此按钮后,用户就可以对该问题进行监视,也就是说只要该问题有改动,系统就会自动发邮件通知本人。这在“我的视图”页也可以体现出来。

(5) 创建子问题(Clone)。可以创建该问题的子项问题。

(6) 移动(Move)。可以将该问题移动到别的项目中(需要相应的权限)。

(7) 删除问题(Delete)。删除无用的问题,已处理完毕的问题建议不必删除,关闭即可,以保留问题记录。

(8) 关联(Relationships)。可以指定问题之间的关联关系,具体关联方式见下拉菜单。

(9) 上传文件(Upload File)。可以上传与问题相关的文件,大小暂时限制为 5MB。

(10) 问题历史(Issue History)。此项为问题处理的历史记录。

3. 提交问题

在系统界面,单击菜单栏中的“提交问题”项,进入问题录入界面。如果单击前,右上角项目选择为“所有项目”,那么填报问题前,需要先选择要填报的项目,如图 3-12 所示。可以选择“设为默认值”复选框,这样每次填报进入该界面时,就为默认项目了。

图 3-12 选择项目

单击“选择项目”按钮后,进入问题填报界面,如图 3-13 所示。页面中可以看到一个提交 Bug 的表单,根据具体情况填写后提交即可。

在提交报告时请注意,带 * 号的填写项是必填项,页面还提供文件上传功能,只要是低于 2MB 的文件都允许上传,支持 doc、xls、zip 等格式的文件。这样在报告 Bug 的时候,可以上传相关的文件,为 Bug 的解决提供更多的信息。全部填写完毕之后,单击“提交报告”按钮,即可提交报告。之后系统会提示用户操作成功。返回“我的视图”中查看,就可以看到新提交的报告。

4. 修改日志

在系统界面,单击菜单栏中的“变更日志”项,则直接进入预设项目的日志信息,如图 3-14 所示。页面列出了该项目下已解决的 Bug 编号、所属组别、Bug 摘要以及该项目产品的版本号变化,单击 Bug 编号还可进入其详细信息页面。

填写问题详情

*分类	(请选择)
出现频率	总是
严重性	很严重
优先级	中
选择平台配置	
⊖ 或填写	
平台	
操作系统	
操作系统版本	
产品版本	
分派给	
目标版本	
*摘要	
*描述	
问题重现步骤	
附注	
上传文件 (最大大小: 5,000k)	浏览...
查看权限	公开　私有
继续报告	报告更多的问题

* 必填　　提交报告

图 3-13　填写问题信息

LxBlog博客系统 - 变更日志

LxBlog博客系统 - 1.1 (未发布) [查看问题]
==========
- 0000001: [功能测试] 登录用户名未区分大小写 (liuhai) - 已关闭.

[1 问题]

Copyright © 2000 - 2015 MantisBT Team
foxlan888@163.com

mantis BUG TRACKER

图 3-14　日志信息

5. 路线图

在系统界面，单击菜单栏中的“路线图”项，如果开始没有指定项目的情况下，则首先进入项目选择界面，如果指定了默认值，则直接进入默认项目的路线图。路线图相当于一个Bug的日志信息，如图3-15所示。页面列出了该项目下已解决的Bug编号、所属组别、Bug摘要以及该项目产品的版本号变化，单击Bug编号还可进入其详细信息页面。

LxBlog博客系统 - 路线图

LxBlog博客系统 - 1.1 (待发布 2015-08-17) [查看问题]

33%

- 0000003: [功能测试] 查看日志页面表格内容未对齐 (liuhai) - 已分派.
- 0000002: [功能测试] 日志的附件文件大小未限制，可上传超大文件，严重影响系统性能。 (liuhai) - 已分派.
- ~~0000001: [功能测试] 登录用户名未区分大小写 (liuhai) - 已关闭.~~

已解决 1 个问题(共 3 个问题)，进度(33%)。

图3-15 路线图

6. 统计报表

在系统界面，单击菜单栏中的“统计报表”项，将会出现一个包含所有Issue报告的综合报表，页面上还提供了“打印报告”和“统计报表”的功能链接，如图3-16所示。在这个综合报表中，按照Bug报告详细资料中的项目，将所有的报告按照不同的分类进行了统计。这个统计报表有助于管理员及经理掌握Bug报告处理的进度，而且很容易就能把没有解决的问题与该问题的负责人、监视人联系起来，提高了工作效率。这个页面中还提供了按不同要求分类的统计图表，如按状态统计、按优先级统计、按严重性统计、按项目分类统计和按处理状况统计。

统计报表

按项目	未解决	已解决	已关闭	合计
LxBlog博客系统	2	0	1	3
» LxBlog博客系统功能测试	0	0	0	0
» LxBlog博客系统安全性测试	0	0	0	0
» LxBlog博客系统性能测试	0	0	0	0
» LxBlog博客系统用户界面测试	0	0	0	0

按问题状态	未解决	已解决	已关闭	合计
已分派	2	-	-	2
已关闭	-	-	1	1

按严重性	未解决	已解决	已关闭	合计
细节	1	0	0	1
很严重	1	0	1	2

按分类	未解决	已解决	已关闭	合计
功能测试	2	0	1	3

已解决问题的耗时(天)	
耗时最长的问题	~~0000001~~
最长耗时	0.02
平均时间	0.02
合计时间	0.02

处理员(未解决/已解决/已关闭/合计)	未解决	已解决	已关闭	合计
liuhai	2	0	1	3

按日期 (天)	已报告	已解决	差值
1	3	1	+2
2	3	1	+2
3	3	1	+2
7	3	1	+2
30	3	1	+2
60	3	1	+2
90	3	1	+2
180	3	1	+2
365	3	1	+2

最活跃	得分
0000003 - 查看日志页面表格内容未对齐	3
0000002 - 日志的附件文件大小未限制，可上传超大文件，严重影响系统性能。	3

最长耗时	天
0000002 - 日志的附件文件大小未限制，可上传超大文件，严重影响系统性能。	0
0000003 - 查看日志页面表格内容未对齐	0

按处理状况	未解决	已解决	已关闭	合计
未处理	2	0	0	2
已修正	0	0	1	1

按优先级	未解决	已解决	已关闭	合计
低	1	0	0	1
中	1	0	1	2

图3-16 统计报表

如果需要，还可以单击界面上方的“打印报告”，将所有的 Bug 显示出来。

7. 个人资料

在系统界面，单击菜单栏中的“个人资料”项，即进入了账户管理界面，如图 3-17 所示。此页面中包含个人资料、更改个人设置、管理列和管理平台配置的功能操作，当前默认界面为个人账户编辑页面。

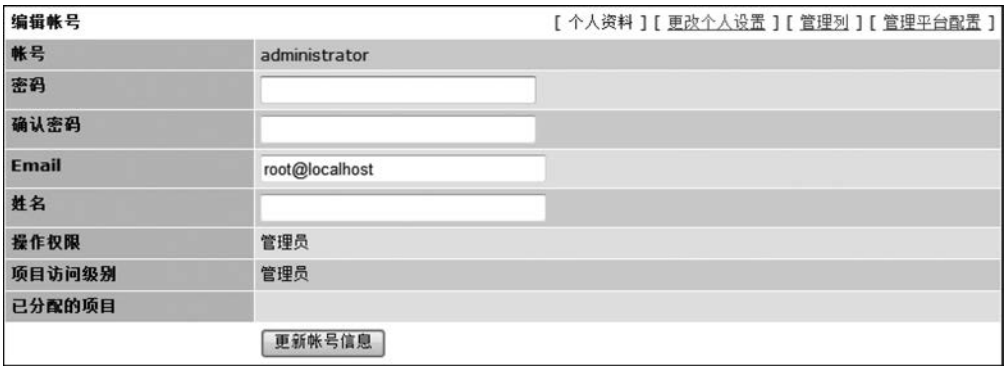

图 3-17　个人资料

1）个人资料

在个人资料页面，可设置个人信息，其中包括修改密码、Email、姓名等信息。

2）更改个人设置

单击“更改个人设置”，进入个人设置页面，如图 3-18 所示。在这里可对页面相关项进行重新设置。

图 3-18　更改个人设置

3）管理列

管理列的信息如图 3-19 所示。

4）管理平台配置

通过管理平台可以增加平台设置，也可以对现有的平台数据进行编辑或删除。这样，在自己报告 Bug 的时候，采用“高级报告”的报告报表就可以直接选用对应的平台数据而不需要自行输入，节省工作时间。管理平台配置的内容如图 3-20 所示。

【注】 管理平台配置的内容只限于本人采用。

图 3-19　管理列

图 3-20　管理平台

8. 管理

在系统界面，单击菜单栏中的“管理”项，即可进入管理界面，管理界面包含用户管理、项目管理和自定义字段管理等内容。

1）用户管理

用户管理页面是“管理”功能的默认页面，如图 3-21 所示。管理者可以按用户账号的字母顺序筛选用户，修改用户权限和信息，也可以单击“创建新账号”按钮来建立新账户。

全部 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 0 1 2 3 4 5 6 7 8 9 未用[3](从未登录) 新的[5](一周内)

管理帐号 [5] 创建新帐号　隐藏非活动的　显示已禁用　筛选

帐号	姓名	Email	操作权限	启用	已保护	创建日期	最近一次登录
administrator		root@localhost	管理员	X		2015-08-17 07:09	2015-08-17 12:51
lanrui	蓝蕊	364100060@qq.com	复查员	X		2015-08-17 07:29	2015-08-17 07:29
liuhai	刘海	liu678@163.com	开发员	X		2015-08-17 12:47	2015-08-17 12:47
liuwei	刘伟	45623124@qq.com	经理	X		2015-08-17 12:49	2015-08-17 12:49
wangyu	王雨	714622475@qq.com	报告员	X		2015-08-17 07:36	2015-08-17 07:37

搜索　管理用户

图 3-21　用户管理

(1) 新的(新的账号)。显示一周之内添加到该项目的新用户。

(2) 未用(从未登录)。显示目前存在却至今从未登录过的用户,对此用户可以单击“清理账号”功能链接将其清除。

(3) 管理账号。在这里可以添加新用户和更新已有用户。单击“创建新账号”按钮,就可以添加新的用户,并指定其工作身份。单击现有账号名称,就可以对当前账号的资料进行更新,更新之后单击“重置”按钮,系统就接受更新信息了。

建立新账户时,可以设置是否启用账户,以及账户操作权限。用户权限包括报告员、复查员、修改员、开发员、经理和管理员。各用户的操作权限可以定制。

2) 项目管理

在系统界面,单击菜单栏中的“项目管理”项,即可查看当前的所有项目。

(1) 创建新项目。

单击“创建新项目”按钮,进入新项目创建页面,如图 3-22 所示。添加项目时,可以设定

添加项目

* 项目名称	
状态	开发中
继承全局类型	☑
查看权限	公开
描述	

添加项目

图 3-22　添加项目

新项目的状态，状态包括开发中（development）、发布（release）、稳定（stable）和停止维护（obsolete）。填写好项目资料后，单击“添加项目”按钮，新的项目就添加到系统中了。

（2）管理项目。

在系统界面，单击菜单栏中的“项目管理”项，将进入项目管理页面，如图 3-23 所示。在

图 3-23　管理项目

弹出的页面中，可以看到已经创建的项目列表，列表中包括各个项目的名称、状态、查看状态以及说明列属性。单击列表中的项目名称，就可以看到项目的具体情况。在这个页面上可以进行下列操作。

① 编辑项目。在这里经理可以对项目的名称、状态、查看状态、上传文件的存放路径以及说明等内容进行更新。

② 删除项目。单击“删除”按钮，即可将当前项目从库里删除。

③ 子项目。可以创建属于该项目的子项目，或者指定某个项目为该项目的子项目。

④ 添加分类。填入类别名称，单击“添加分类”便可在当前项目里增加类别。

⑤ 编辑分类。单击“编辑”按钮，进入类别编辑页面，可以将当前的类别分配给指定的工作人员，这样会在该项目下提交一个新 Bug 的时候，直接分派给该指定的工作人员处理。

⑥ 删除分类。单击“删除”按钮，即可删除当前的分类。

⑦ 版本。可以对已有的项目版本进行更新或者删除，也可以添加新的版本。

⑧ 自定义字段。可以从已存在的自定义字段中选择出所需要的添加到该项目中的自定义字段里，也可以删除已添加的自定义字段。自定义字段添加至项目后，在“提交问题”中会显示为必填字段。

⑨ 添加用户至项目。将与项目相关的用户添加进来。

⑩ 管理账号。对该项目中所有的相关人员的账号进行管理，可以删去那些在项目中不需要的账号。

3）自定义字段管理

自定义字段管理是用于在提交问题的时候，系统给予的填写项不满足实际需求，这样可以自行在这个功能页面里定义自己需要的字段，以便能更好地描述 Bug 的情况，而且在过滤器中也会增加这个字段的属性，可以根据其进行数据过滤。

在管理页面中，单击“自定义字段管理”菜单项，即可进入自定义字段管理页面，在这里可以新建自定义字段、修改已有的自定义字段。

自定义域的类型有字符串（String）、数值（Numeric）、浮点型（Float）、枚举类型（Enumeration）、电子邮件（Email）、选择框（Checkbox）、列表（List）、多选列表（Multiselection List）、日期（Date）等。

新建自定义字段时，可以设置类型、可能取值、默认取值，是否在报告、更新、解决、关闭页面显示和必填，是否仅在高级查询条件页面显示，并且可以设置关联自定义字段到项目。

4）插件管理

在系统界面，单击菜单栏中的“插件管理”项，进入插件管理页面，如图 3-24 所示。在这里可以看到已装插件和可用插件。在可用插件的右侧单击“安装”按钮，可以安装相应的插件。

5）注销

在系统界面，单击菜单栏中的“注销”项，即可退出登录，返回至初始登录界面。

3.7.4 权限用户的操作

1. 经理

经理是整个软件开发过程中较为重要的管理人员。经理在该系统下的使用权限比管理

已装插件

插件	说明	依赖于	优先级	已保护	操作
MantisBT Core 1.2.19	Core plugin API for the Mantis Bug Tracker. 作者： MantisBT Team 网址： http://www.mantisbt.org	没有依赖项			
MantisBT格式化 1.0b	官方文本处理和格式化工具. 作者： MantisBT Team 网址： http://www.mantisbt.org	MantisBT Core 1.2.0	3		[卸载]
Mantis图表 1.0	官方图表插件. 作者： MantisBT Team 网址： http://www.mantisbt.org	MantisBT Core 1.2.0	3		[卸载]

升级

可用插件

插件	说明	依赖于	操作
导入/导出问题 1.0	为MantisBT添加基于XML的导入导出功能. 作者： MantisBT Team 网址： http://www.mantisbt.org	MantisBT Core 1.2.0	[安装]

图 3-24　插件管理

员稍微低一些，由于前面已经详细说明了各个菜单功能的使用，因而这里主要说明经理在本系统所能使用的功能与管理员有什么不同。

对于前面提及的菜单功能，经理在使用的时候和管理员基本相同，但存在以下几个差异。

(1) 只能对自己的过滤器进行操作，对于管理员设为共有的过滤器只能使用而不能进行操作。

(2) 在“查看问题”时，可以通过复选框来对某条 Bug 进行命令操作，但经理级别的工作人员不能执行“删除”操作。如果执行“删除”操作，系统会提示：你无权执行该项操作。

(3) 在“管理”功能中，不能对用户进行管理，包括新建、删除用户等，也不能新建项目，只能管理现有项目的信息。

2. 开发人员

开发人员是负责整个软件开发的工作人员。使用 Mantis 缺陷跟踪管理系统，开发人员可以及时地发现和解决软件缺陷。在该系统中，开发人员的权限比经理的权限低一些，从开始登录进入系统就能看出来，其主菜单栏的功能相对少一些。比起经理和管理员的菜单栏，少了“统计报表”和“管理”功能。

前面已经详细说明了各个菜单功能的使用，这里主要说明开发人员在本系统所能使用的功能与管理员的不同之处。

(1) 开发人员只能设置私有的过滤器，可以共享管理员和经理创建的且属性设置为公有的过滤器。

(2) 在“查看问题”的时候，可以通过复选框来对某条 Bug 进行命令操作，但开发人员不能执行“删除”操作。如果执行“删除”操作，系统会提示：你无权执行该项操作。

3. 修改人员

修改人员是负责修改问题的工作人员。修改人员的主菜单和开发人员的一样，但其使用权限还是比开发人员稍低一些，下面来具体说明操作区别。

(1) 不能创建过滤器，但是可以使用由其他工作人员创建且属性被设为公有的过滤器。

(2) 在“查看问题”的时候,可以通过复选框来对某条 Bug 进行命令操作,修改人员只能进行“复制”“更新优先级”“更新状态”“更新视图状态”操作,对于其他的命令操作则无权执行。

(3) 在“我的视图”页面,没有“分派给我的(尚未解决)”状态的 Bug 数据列表,因为对于修改人员来说,不能将 Bug 直接指派给他。因此,相应的界面上没有这类数据的显示。

4. 报告人员

报告人员是专门负责提交 Bug 报告的工作人员。报告人员的主菜单与开发人员和修改人员的一样,但是其使用权限比修改人员低一些。下面来具体说明操作区别。

(1) 不能创建过滤器,但是可以使用由其他工作人员创建且属性被设为公有的过滤器。

(2) 在“查看问题”的时候,对 Bug 数据列表不能进行任何命令操作。

(3) 在“我的视图”页面,没有“分派给我的(尚未解决)”状态的 Bug 数据列表,因为对于报告人员来说,也不能将 Bug 直接指派给他。因此,相应的界面上没有这类数据的显示。

5. 查看人员

查看人员,顾名思义就是只具有查看权限的工作人员,在 Mantis 系统中,其权限最低,功能主菜单也相应更少。查看人员对于 Mantis 系统的操作功能基本上没有使用权限(除了个人资料的功能外),而只能查看各个项目的 Bug 流程及具体情况。

3.7.5 指派给我的工作

当工作人员登录系统后,在“我的视图”界面,单击“分派给我的(未解决)”中的问题编号链接,将进入分派任务的问题界面。在这个页面,将问题的信息分为 7 个数据块来显示。

1. 查看问题详细信息

在页面上的第一个数据块显示问题(Issue)的详细资料,在这个数据表格可以进行下列操作。

(1) 查看注释。单击此链接,则直接跳转到该页面的注释数据。

(2) 发送提醒。单击此链接,则可以对某个工作人员发送提醒,直接在该工作人员的监视 Issue 视图里生成,并在该 Issue 的注释里也增加一条记录。注意。这个功能除了查看人员之外,其他工作人员都可以使用。

(3) 问题历史。单击此链接,则直接跳转到该页面的问题历史数据。

(4) 打印。单击此链接,则直接在网页上生成这个问题的详细数据,可以通过浏览器提供的打印功能进行打印。

(5) 编辑。如果问题的信息需要修改,则单击“编辑”按钮,直接进入问题编辑页面。然后单击“更改信息”按钮,根据需要修改信息,如果不修改,可以单击“返回到问题”按钮,回到前面的页面。注意:修改问题信息只有管理员、经理、开发人员和修改人员能使用。

(6) 分派给。这个功能可以把当前的问题直接分派给选定的工作人员。注意:分派功能只有管理员、经理、开发人员以及修改人员具有,而且只有管理员才能指派给自身。

(7) 状态改为。这个功能可以当前的问题流程状态进行修改。注意:修改状态功能只

有管理员、经理、开发人员可以使用。

(8) 监视。这个功能可以把当前这个问题置为所监视的问题范围之内。注意：这个功能除了查看人员之外，其他工作人员都可以使用。

(9) 创建子问题。这个功能主要是创建与当前问题相关的问题。单击“创建”按钮进入如图 3-25 所示的页面。在“与上级问题关联”中设定创建子项后和当前问题的关系。同时在“关联”数据块中会增加一条记录。注意：创建子项功能只有管理员、经理、开发人员以及修改人员才能使用。

图 3-25 创建子问题

(10) 移动。单击“移动”按钮，将打开如图 3-26 所示的页面。选择下拉列表中的项目，然后单击“移动问题”按钮，即可以将该问题转移到所选的项目中。注意：移动功能只有管理员、经理、开发人员才能使用。

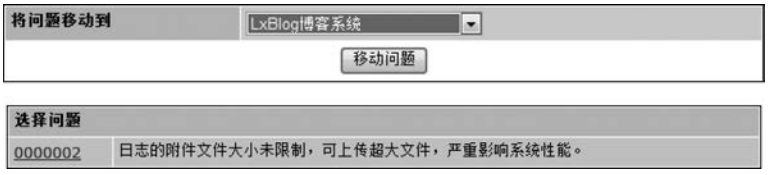

图 3-26 移动问题

(11) 关闭。将该问题关闭，不再修改、讨论。注意：关闭功能只有管理员、经理才能使用。

(12) 删除问题。单击“删除问题”按钮，将直接删除该问题。注意：只有管理员才有这个权限。

2. 关联

关联主要是描述当前问题是否和别的问题有什么关系。在此界面可以进行如下操作。

(1) 增加关联。增加与当前问题有关联的问题，选择关系，输入问题编号，单击“添加”按钮即可完成添加。注意：只有管理员、经理、开发人员和修改人员才能使用这项功能。

(2) 查看相关问题。单击关联列表的问题编号链接，可以直接查看该关联问题的详细信息。

(3) 删除关联。对已经存在的关联数据进行删除操作。注意：只有管理员、经理、开发人员和修改人员才能使用这项功能。

3. 上传文件

在报告问题时，可以上传文件。如果当时没有上传，可以在此处重新上传。上传文件则