

# git与bug管理系统 安装和学习报告

---

## Git:

### 安装Git:

在Linux上安装Git,通过一条`sudo apt-get install git`就可以直接完成Git的安装,非常简单。

Windows下要使用很多Linux/Unix的工具时,需要Cygwin这样的模拟环境, Git也一样。Cygwin的安装和配置都比较复杂,就不建议你折腾了。不过,有高人已经把模拟环境和Git都打包好了,名叫msysgit,只需要下载一个单独的exe安装程序,其他什么也不用装,绝对好用。msysgit是Windows版的Git,从<http://msysgit.github.io/>下载,然后按默认选项安装即可。

安装完成后,在开始菜单里找到“Git”->“Git Bash”,蹦出一个类似命令行窗口的东西,就说明Git安装成功!

安装完成后,还需要最后一步设置,在命令行输入:

```
$ git config --global user.name "Your Name"
```

```
$ git config --global user.email "email@example.com"
```

注意`git config`命令的`--global`参数,用了这个参数,表示你这台机器上所有的Git仓库都会使用这个配置,当然也可以对某个仓库指定不同的用户名和Email地址。

初始化一个Git仓库,使用`git init`命令。

添加文件到Git仓库,分两步:

- 第一步,使用命令`git add <file>`,注意,可反复多次使用,添加多个文件;
- 第二步,使用命令`git commit`,完成。

Git鼓励大量使用分支:

查看分支: `git branch`

创建分支：git branch <name>

切换分支：git checkout <name>

创建+切换分支：git checkout -b <name>

合并某分支到当前分支：git merge <name>

删除分支：git branch -d <name>

Git虽然极其强大，命令繁多，但常用的就那么十来个，掌握好这十几个常用命令，你已经可以得心应手地使用Git了。

## Bugzilla

### 1. Bugzilla简介

#### 1.1. 产生

Bugzilla是一个共享的免费的产品缺陷记录及跟踪工具。由Mozilla公司提供。创始人是Terry Weissman，开始时使用一种名为“TCL”的语言创建的，后用Perl语言实现，并作为Open source发布。

#### 1.2. 麻烦

1. 需要Perl和配置MYSQL数据库;
2. Bug和Case管理部署在两个不同的工具上面;

#### 1.3. 特点

Bugzilla能够建立一个完善的bug跟踪体系：报告bug、查询bug记录并产生报表、处理解决bug、管理员系统初始化和设置四部分。（Bug管理系统的通性：比如TFS）

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。本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。当缺陷在它的生命周期中变化时，开发人员、测试人员、及管理人员将及时获得动态的变化信息，允许你获取历史记录，并在检查缺陷的状态时参考这一记录。
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，可有效控制进行管理。允许设定不同的严重程度和优先级。可以在缺陷的生命期中管理缺陷。从最初的报告到最后的解决，确保了缺陷不会被忽略。同时可以使注意力集中在优先级和严重程度高的缺陷上。

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，有效的帮助测试人员和开发人员进行沟通。（每个人收到邮件后要自觉的进行相关处理）

## 2. Bugzilla操作说明

### 2.1. 用户登录及设置

#### 2.1.1. 用户登录

1. 用户输入服务器地址http://192.168.11.18/bugzilla。
2. 进入主页面后，点击【login in】进入。
3. 输入用户名和密码即可登录。用户名为Email地址，初始密码为用户名缩写。
4. 如忘记密码，点击【Forgot Password】，输入用户名根据收到的邮件进行重新设置。

#### 2.1.2. 用户属性设置

Login登录后，点击【Preferences】进行属性设置

##### a) 账号设置 (Name and Password)

在这里你可以改变你账号的基本信息，如口令，Email地址，真实姓名

为了安全起见，在此页进行任何更改之前你都必须输入你当前的口令

当你变更了你的Email地址，系统会给你的新老Email地址分别发一封确认邮件，你必须到邮件中指定的地址对你的更改进行确认

##### b) Email设置 (Email Preferences)

你可以在此通过选择告诉系统，你希望在什么条件下收到和你相关的邮件

##### c) 保存的查询 (Saved Searches)

##### d) 常规属性 (General Preferences)

##### e) 用户权限 (Permissions)

### 2.2. 报告Bug

#### 2.2.1. 测试人员报告Bug

1. 请先进行查询，确认要提交的bug报告不会在原有纪录中存在，若已经存在，不要提交，若有什么建议，可在原有纪录中增加注释，告知其属主，让bug的属主看到这个而自己去修改。
2. 若Bug不存在，创建一份有效的bug报告后进行提交。
3. 操作：点击New，选择产品后，填写下表。
4. 填表注意：

Assigned to: 为空则默认为设定的 owner, 也可手工制定。

CC: 可为多人，需用","隔开。

Description中要详细说明下列情况：

- a) 发现问题的步骤
  - b) 执行上述步骤后出现的情况。
  - c) 期望应出现的正确结果。
5. 操作结果：Bug状态（status）可以选择Initial state 为New或Assigned.
- a) 系统将自动通过Email通知项目组长或直接通知开发者。
6. 帮助： Bug writing guidelines

### 2.2.2. 开发人员报告Bug.

1. 具体方法同测试人员报告。
2. 区别： Bug初始状态将自动设为Assigned，待测试人员确定后变为“New”.

## 2.3. 处理Bug

### 2.3.1. 处理情况1

Bug的属主 (owner) 处理问题后，提出解决意见及方法。

1. 给出解决方法并填写Additional Comments，还可创建附件（如：更改提交单）
2. 具体操作（填表项如下）
3. 填表注意：
  - FIXED 描述的问题已经修改
  - INVALID 描述的问题不是一个bug (输入错误后，通过此项来取消)
  - WONTFIX 描述的问题将永远不会被修复。
  - LATER 描述的问题将不会在产品的这个版本中解决.
  - DUPLICATE 描述的问题是一个存在的bug的复件。
  - WORKSFORME 所有要重新产生这个bug的企图是无效的。如果有更多的信息出现，请重新分配这个bug，而现在只把它归档。

### 2.3.2. 处理情况2

项目组长或开发者重新指定Bug的属主。（owner）

1. 为此bug不属于自己的范围，可置为 Assigned,等待测试人员重新指定。
2. 为此bug不属于自己的范围，但知道谁应该负责，直接输入被指定人的Email， 进行 Resigned。
3. 操作：（可选项如下）
  - Accept bug (change status to ASSIGNED)
  - Reassign bug to
  - Reassign bug to owner and QA contact of selected component
4. 操作结果：此时bug状态又变为New，此bug的owner变为被指定的人。

### 2.3.3. 处理情况3

测试人员验证已修改的 Bug.

1. 测试人员查询开发者已修改的bug，即Status为"Resolved",Resolution为"Fixed".进行重新测试。（可创建test case附件）
2. 经验证无误后，修改Resolution为VERIFIED。待整个产品发布后，修改为CLOSED。

若还有问题，REOPENED，状态重新变为“New”，并发邮件通知。

3. 具体操作（可选择项）

- Leave as RESOLVED FIXED
- Reopen bug
- Mark bug as VERIFIED
- Mark bug as CLOSED

#### 2.3.4. 处理情况4

Bug报告者（reporter）或其他有权限的用户修改及补充Bug

1. 可以修改Bug的各项内容。
2. 可以增加建立附件，增加了相关性，并加一些评论来解释你正在做些什么和你为什么做。
3. 操作结果：

每当一些人修改了bug报告或加了一个评论，他们将会被加到CC列表中，bug报告中的改变会显在要发给属主、写报告者和CC列表中的人的电子邮件中。

### 2.4. 查询Bug

1. 直接输入Bug Id，点击find 查询。可以查看Bug的活动纪录。
2. 点击Search，输入条件进行查询。

根据查找的需要，在界面中选择对象或输入关键字

查找功能能够进行字符或字串的匹配查找

查找功能具有布尔逻辑检索功能

你可以通过在查找页面中选择“Remember this as my default query”将当前检索页面中设定的项目保存。以后可以从页脚中的My bugs中直接调用这个项目进行检索

你还可以通过在“Remember this query, and name it:”后面输入字符，将你当前检索页面中设定的项目保存命名，同时选中“and put it in my page footer”。则以后这个被命名的检索将出现在页脚中。

3. 查询Bug活动的历史
4. 产生报表。
5. 帮助：点击Give me some help.

### 2.5. 关于权限的说明

1. 组内成员对bug具有查询的权利,但不能进行修改。
2. Bug的owner 和 reporter 具有修改的权利。
3. 具有特殊权限的用户具有修改的权利

### 2.6. BUG处理流程

1. 测试人员或开发人员发现bug后，判断属于哪个模块的问题，填写bug报告后，通过Email通知项目组长或直接通知开发者。
2. 项目组长根据具体情况，重新reassigned分配给bug所属的开发者。
3. 开发者收到Email信息后，判断是否为自己的修改范围。
  - a) 若不是,重新reassigned分配给项目组长或应该分配的开发者。

- b) 若是，进行处理，resolved并给出解决方法。（可创建补丁附件及补充明）
- 4. 测试人员查询开发者已修改的bug，进行重新测试。（可创建test case附件）
  - a) 经验证无误后，修改状态为VERIFIED。待整个产品发布后，修改为CLOSED
  - b) 还有问题，REOPENED，状态重新变为“New”，并发邮件通知。
- 5. 如果这个BUG一周内一直没被处理过。Bugzilla就会一直用email骚扰它的owner，直到采取行动。

## 2.7. 表单描述

- 模块 (Component)  
产生Bug的软件模块
- 版本 (Version)  
产生Bug的软件版本
- 严重性 (Severity)
- 硬件平台 (Hardware)
- 操作系统 (OS)
- 优先级 (Priority)  
分五个等级即P1-P5，P1的优先级别最高之后逐级递减
- 初始状态 (Initial State)
- 指定处理人 (Assigned To)  
可以指定一个处理人。如不指定处理人，则系统指定管理员为默认处理人
- 相关人员 (CC)  
如需要抄送多人，可将邮件地址用“，”分隔
- 估计时间 (Estimated Hours)
- 最后期限 (Deadline)
- 超链接 (URL)  
输入超链接地址，引导处理人找到与报告相关联的信息
- 概述 (Summary)  
概述部分“Summary”的描述，应保证处理人在阅读时能够清楚提交者在进行什么操作的时候发现了什么问题。  
如果是通用组件部分的测试，则必须将这一通用组件对应的功能名称写入概述中，以便今后查询。
- 从属关系 (Bug “ID” depends on, Bug “ID” blocks)  
“Bug “ID” depends on”如果该Bug必须在其他Bug修改以后才能够修改，则在此项目后填写那个Bug的编号  
“Bug “ID” blocks”如果该Bug的存在影响了其他Bug的修改，则在此项目后填写被影响的Bug编号
- 附加描述 (Description)  
在Bug跟踪过程中测试与开发人员通过这里进行沟通  
开发人员可以在这里填写处理意见和处理记录  
测试人员可以在这里填写返测意见和对在返测过程中发现的新问题进行描述

- 附件 (Attachment)  
Bug相关的文件或截图

### 3. 相关名词解释

#### 3.1. Bug报告状态 (Status)

- 待确认的 (Unconfirmed)
- 新提交的 (New)
- 已分配的 (Assigned)
- 问题未解决的 (Reopened)
- 待返测的 (Resolved)
- 待归档的 (Verified)
- 已归档的 (Closed)

#### 3.2. Bug处理意见 (Resolution)

- 已修改的 (Fixed)
- 不是问题 (Invalid)
- 无法修改 (Wontfix)
- 以后版本解决 (Later)
- 重复 (Duplicate)
- 无法重现 (Worksforme)
- 移除 (Moved)

#### 3.3. Bug严重性 (Severity)

- Blocker, 阻碍开发和/或测试工作
- Critical, 死机, 丢失数据, 内存溢出
- Major, 较大的功能缺陷
- Normal, 普通的功能缺陷
- Minor, 较轻的功能缺陷
- Trivial, 产品外观上的问题或一些不影响使用的小毛病, 如菜单或对话框中的文字拼写或字体问题等等
- Enhancement, 建议或意见

#### 3.4. Bug优先级 (Priority)

- 分五个等级即P1-P5, P1的优先级别最高之后逐级递减

#### 3.5. Bug硬件平台 (Hardware)

- All
- PC
- IBM
- Other

### 3.6. Bug操作系统 (OS)

- All
- Windows
- WindowsXP
- Windows2003
- AIX
- AIX5.3
- AIX6.1
- Linux
- Other