

Redmine 和 Jira 的优缺点对比

Redmine 介绍

开源，可免费使用

- 自带 wiki 功能
- 每个项目具有用户论坛
- 操作更加便捷
- 界面更加简洁友好，不花哨
- 易于 SCM 集成
- 更易于追踪问题
- 集成邮件系统（包括创建问题、以及问题变更的通知）
- 对子任务的支持

JIRA pro's

- IDE 集成

Redmine CONS:

- Redmine 更关注于问题与项目多过于用户，很难取得较好的报告呈现方式，包括工作完成情况、工作进度以及其他。
- 没有通知安排，只有变化通知，没有升级通知，只通知指定用户插件添加。
- 没有问题的保障计划
- 没有清晰定义，因而每个角色可以看到问题追踪。
 - 无法从项目一开始就选择与其相关的子项目
 - 问题的时间评估功能不够强大的，JIRA 甚至允许用 3w 4d 12h 的时间格式。
- Redmine 的绝大部分领域都没有基于 Ajax 的自动填充。
 - 只有 7 种类型的自定义字段。JIRA 允许有 23 种，包括创建 “User Picker

”以及其他多层次领域。

- 活动日志没有过滤器/标签，无法在某一页面中看到当前的所有问题。
 - 无法对时间进行评论
 - 无法设定条件控制问题状态的变化

Redmine CONS that are secondary:

- 外观没有经过优化，大部分东西在页面中使用更少的空间。
 - 无法手动设置问题
 - 无多国语言的问题类型
 - 没有增加问题可读性的图标
 - 没有键盘快捷键
 - Redmine

的文件管理功能比较薄弱，无法描述所有的东西。不过，大部分都是比较容易懂，除了一些特殊的任务，

althou, most of everything is easy understandable, but for some

specific tasks

Jira 与 Redmine 试用对比报告

Redmine pro's

开源，可免费使用 自带 wiki 功能 每个项目具有用户论坛 操作更加便捷
界面更加简洁友好，不花哨 易于 SCM 集成 更易于追踪问题
集成邮件系统（包括创建问题、以及问题变更的通知）

对子任务的支持

JIRA pro's

IDE 集成

Redmine CONS:

Redmine 更关注于问题与项目多过于用户，很难取得较好的报告呈现方式，包括工作完成情况、工作进度以及其他。

没有通知安排，只有变化通知，没有升级通知，只通知指定用户插件添加。 没有问题的保障计划.没有清晰定义，因而每个角色可以看到问题追踪。 无法从项目一开始就选择与其相关的子项目

问题的时间评估功能不够强大的，JIRA 甚至允许用 3w 4d 12h 的时间格式。 Redmine 的绝大部分领域都没有基于 Ajax 的自动填充。

只有 7 种类型的自定义字段.JIRA 允许有 23 种，包括创建 “User Picker”以及其他多层次领域。 活动日志没有过滤器/标签，无法在某一页面中看到当前的所有问题。 无法对时间进行评论

无法设定条件控制问题状态的变化

Redmine CONS that are secondary:

外观没有经过优化，大部分东西在页面中使用更少的空间。 无法手动设置问题 无多国语言的问题类型 没有增加问题可读性的图标 没有键盘快捷键

Redmine 的文件管理功能比较薄弱，无法描述所有的东西。不过，大部分都是比较容易懂，除 了 一 些 特 殊 的 任 务 ，
althou, most of everything is easy understandable, but for some specific tasks

It doesn't support internal (private) tickets and the time-tracking commit message patch, doesn't support the trunk version.

JIRA CONS:

很难理解用户界面， workflow 创建流程，许多功能性的选择容易令人困惑。 所做的修改并没有直接显示出来.

不易创建子任务。

JIRA CONS that are secondary:

并非开源，只能依赖现有的功能和系统。 慢并且复杂，绝大多数功能并不需要用户系统。

虽然 JIRA 的文件管理功能很强大，但是系统太复杂以至于你需要文档定制每一个任务。

JIRA 适合多人的团队(100+)，而 Redmine 适合中小型团队

Redmine 是用 Ruby 开发的基于 web 的项目管理软件，是用 ROR 框架开发的一套跨平台项

目管理系统，据说是源于 Basecamp 的 `ror` 版而来，支持多种数据库，有不少自己独特的功能，例如提供 `wiki`、新闻台等，还可以集成其他版本管理系统和 `BUG` 跟踪系统，例如 `SVN`、`CVS`、`TD` 等等。这种 `Web` 形式的项目管理系统通过“项目（`Project`）”的形式把成员、任务（问题）、文档、讨论以及各种形式的资源组织在一起，大家参与更新任务、文档等内容来推动项目的进度，同时系统利用时间线索和各种动态的报表形式来自动给成员汇报项目进度。

功能

- 多项目和子项目支持
- 里程碑版本跟踪
- 可配置的用户角色控制
- 可配置的问题追踪系统
- 自动日历和甘特图绘制

支持 `Blog` 形式的新闻发布、`Wiki` 形式的文档撰写和文件管理

- `RSS` 输出和邮件通知

- 每个项目可以配置独立的 `Wiki` 和论坛模块
- 简单的任务时间跟踪机制

- 用户、项目、问题支持自定义属性

- 支持多种版本控制系统的绑定（`SVN`、`CVS`、`Git`、`Mercurial` 和 `Darcs`）
- 支持多 `LDAP` 用户认证
- 支持用户自注册和用户激活

- 多语言支持（已经内置了 `zh` 简体中文）
- 多数据库支持（`MySQL`、`SQLite`、`PostgreSQL`）
- 外观模版化定制（可以使用 `Basecamp`、`Ruby` 安装）

`JIRA` 是集项目计划、任务分配、需求管理、错误跟踪于一体的商业软件。`JIRA` 创建的问题类型包括 `New Feature`、`Bug`、`Task` 和 `Improvement` 四种，还可以自己定义，所以它也是一过程管理系统。`JIRA` 融合了项目管理、任务管理和缺陷管理，许多著名的开源项目都采用了 `JIRA`。

`JIRA` 功能全面，界面友好，安装简单，配置灵活，权限管理以及可扩展性方面都十分出色。

- 问题追踪和管理：用它管理项目，跟踪任务、`bug`、需求，通过 `jira` 的邮件通知功能进行协作通知，在实际工作中使工作效率提高很多

- 问题跟进情况的分析报告：可以随时了解问题和项目的进展情况
- 项目类别管理功能：可以将相关的项目分组管理

- 组件/模块负责人功能：可以将项目的不同组件/模块指派相应的负责人，来处理所负责的组件的 `Issues`

- 项目 `email` 地址功能：每个项目可以有不同的 `email`（该项目的通知邮件从该地址发出）

- 无限制的工作流：可以创建多个工作流为不同的项目使用

`JIRA` 的优点

使用 `JIRA` 管理项目，跟踪任务、`bug`，通过 `JIRA` 的邮件通知功能进行协作通知，在实际工作中可以使工作效率显著提高，安全性及可扩展性也有所体现。`JIRA` 不仅仅是一个缺陷跟踪系统，通过 `JIRA`，可以整合客户、开发人员、测试人员，通过各自和共同的模块，信息能够及时得到交流和反馈。

`Eclipse` 和 `IDEA` 下的 `JIRA` 插件，主要为开发人员服务，实时将信息反馈给开发人员，开发人员同时迅速地将修复的结果信息反馈到跟踪系统中，最后通过持续集成，软件可以迅速完成了更新。

`JIRA` 还拥有众多插件的支持，包括生成报表，`XP` 编程，与 `SVN` 集成等，能够极大地丰富其功能。

跟同类软件产品 `TestTracker`、`ClearQuest`、`TestDirector` 相比，`JIRA` 的性价比最好！`JIRA` 的缺点

对于测试需求、测试用例等都没有提供直接的方式进行管理。不过可以利用 `JIRA` 的 `Issue Type` 的可定制性,来进行需求和测试用例方面的管理,也可以与 `Testlink` 集成。

`JIRA` 的主要功能如下：

1. 项目类别管理功能：可以将相关的项目分组管理。
2. 组件/模块负责人功能：可以将项目的不同组件/模块指派相应的负责人，来处理所负责的组件的 Issues。
3. 项目 email 地址功能：每个项目可以有不同的 email（该项目的通知邮件从该地址发出）。
4. LDAP 配置器：用于将 JIRA 连接到公司的 LDAP 目录上。
5. Issue 安全级别：在一个项目里可以精确地控制对 Issue 的访问。
6. 可订制的工作流：可以创建多个工作流为不同的项目使用；
7. 快速搜索和 JQL 高级查询功能。
9. 子任务功能：可以在某个 Issue 下创建子任务；
10. 共享过滤器功能：一个团队成员可以访问共享的过滤器；
11. 多个 dashboards 功能：可以定制自己的 JIRA 显示面板；
12. 与 CVS、SVN，ClearCase, Mercurial 的集成；通过 FishEye 还可以连接 Git 代码库。
13. 与 GreenHopper 集成后支持敏捷开发；
14. 通过 IDE 连接器，可以将 JIRA 集成到 Eclipse、IntelliJ IDEA、MS Visual Studio。