

溢米  
辅导

技术中心

BUG 处理流程规范

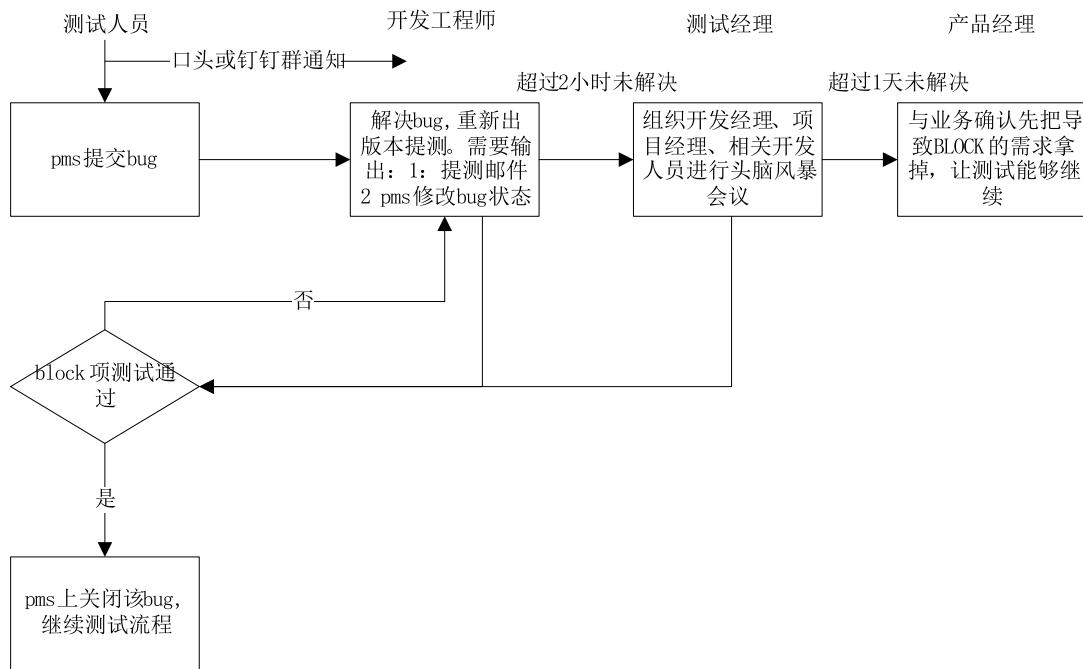
V1.0

文件编码	技术中心-LC181001		
发布时间	20181010		
文件密级	<input type="checkbox"/> 绝密 <input checked="" type="checkbox"/> 秘密 <input type="checkbox"/> 内部公开 <input type="checkbox"/> 外部公开		
文件流转			
类型	岗位	姓名	
文件编制人	客户端总监 项目管理高级经理	李德星 薛亚兰	
文件评审人	技术副总裁	沈菁	
文件会签人	各二级部门负责人	李德星、黄佳庆、许学文、王碧舟、黄桢浩、薛亚兰	
文件签发人	技术副总裁	沈菁	
文件管理人	项目管理高级经理	薛亚兰	
文件执行人	技术中心全员		
版本更新			
序号	版本号	更新内容	更新人
1	V1.0	BUG 处理和管理规范首次制定	薛亚兰
			2018 年 10 月

一 测试阶段 bug 处理流程	4
1. Block BUG 处理流程	4
2. 非 BLOCK BUG 的处理流程	5
二 线上 bug 处理流程	6
1. P0 故障的处理流程	6
2. 非 P0 故障的处理流程	6
三 BUG 处理规范	7
1. bug 等级划分	7
2. bug 状态定义	7
3. bug 解决结果	8
4. bug 的状态迁移	9
5. bug 的提交管理	9
6. bug review 会议	11

# 一 测试阶段 bug 处理流程

## 1. Block BUG 处理流程



说明：

(1) 测试需求出现 block 后，第一时间将该 JIRA 需求置为 测试阻塞状态，同时口头或通过钉钉群的方式通知到开发人员，JIRA 需求设置为阻塞状态，参考下图：

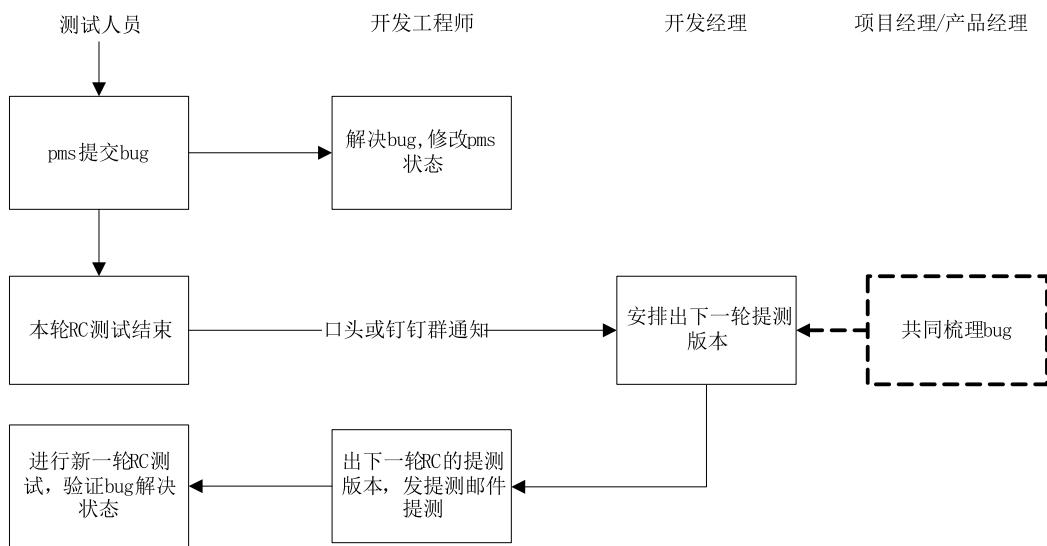


(2) block 的 bug 仍然需要输入到 JIRA 系统中，并与 JIRA 对应需求关联，以便后续统计和跟踪，与 JIRA 需求关联步骤参考下图：



(3) Block 发生后，原则上需要在 2h 之内解决，若超出解决时限，需测试经理或者项目经理 PM 发起头脑风暴会议，制定相应解决方案和步骤

## 2. 非 BLOCK BUG 的处理流程

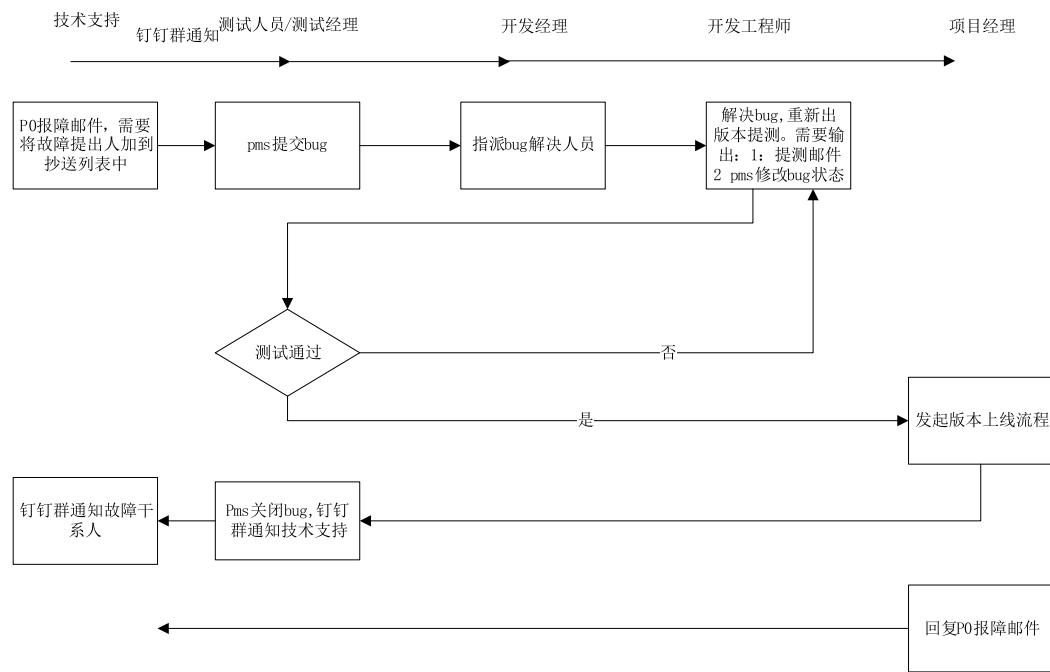


说明：

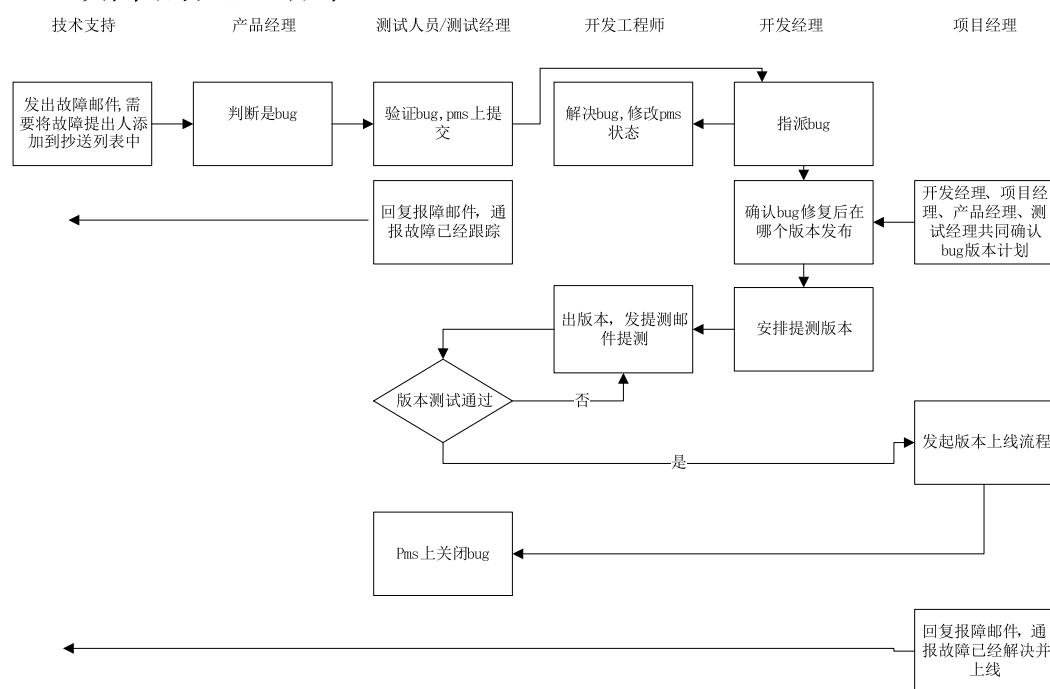
- (1) 项目开发计划中，应该明确各个模块的开发人员，测试人员提交 bug 时，直接指定该开发人员为经办人
- (2) 测试创建 bug，需要将 bug 与测试需求做关联（也就是说一条需求测试出来的相关 bug 均需关联至该需求），以方便 bug 跟踪和统计
- (3) 开发人员在接收到 JIRA 指派的 bug 后，应该在 24 小时内做出响应，要么将 bug 状态修改为解决中，要么去沟通协调，将 bug 的经办人指派给相应的处理人
- (4) 通常情况下，bug 经过一番调研后，需要在 JIRA 内添加相应的调研备注，比如需要其他组或者成员解决的，需要添加备注，写明自己已经调研到哪一步，发现哪些问题等
- (5) 原则上，上一轮 RC 的所有 bug 都解决后，才能提交下一轮 RC 的提测版本；如果遇到特殊的情况，比如有些 bug 出现的概率很低，很难定位，为了不影响下一轮 RC 的测试时间，开发经理和项目经理需要进行 bug 梳理，决定部分 bug 是否可以下个 RC 解决，或者说本次版本内不解决
- (6) 版本计划里，测试需要写明第一轮 RC 测试的周期，方便进行提测版本的管理

## 二 线上 bug 处理流程

### 1. P0 故障的处理流程



### 2. 非 P0 故障的处理流程



说明：

(1) bug 版本计划的原则：开发经理、项目经理、产品经理、测试经理需要共同确定该 BUG 在哪个版本发布。

备注：bug 解决后，如果一周内有版本发布计划，则该修改加入到这个版本中一同发布；如果一周内没有版本发布计划，则新增加一个发布版本，将 bug 的修改通过这个新版本计划发布出去

(2) 明确 bug 在哪个版本中发布后，同时更新该 JIRA BUG 的 修复的版本 字段

### 三 BUG 处理规范

#### 1. bug 等级划分

BLOCKER (P0-Block) :造成测试进程无法继续的严重故障，如：主要需求未实现；功能设计与需求严重不符；系统主要功能异常；系统崩溃、死机；数据库死锁；无测试环境可测试等。

A 类故障 (P1-Major)：严重影响系统要求或基本功能实现的，如：功能实现与需求不符；功能错误影响用户正常使用；操作结果与预期严重不符；卡顿严重；主流设备未兼容；用户数据错误；

B 类故障 (P2-Normal)：一般性错误，如：操作反应速度较慢但不影响使用；部分小众设备未兼容；极端操作顺序导致的错误；复现概率很低的错误；操作时间长、查询时间长、格式错误、删除没有确认框等；

C 类故障 (P3-Minor)：比较轻微的错误，一般是使用方面的问题，不影响操作功能的执行，可以优化性能的方案等。如：错别字、界面格式不规范，不该显示的要隐藏，描述不清楚，提示语丢失，文字排列不整齐，光标位置不正确等；

D 类故障 (P4-Improve)：建议类的故障，如操作优化的建议；文字、提示语优化等。

#### 2. bug 状态定义

- (1) OPEN：测试人员初次提交 bug 后的状态；
- (2) 解决中：开发人员确认了 bug，正在着手解决；
- (3) 已解决：开发人员解决了 bug，但尚未被测试人员验证；
- (4) 验证中：表示测试人员开始验证；

(5) 第三方处理中：如果 bug 是需要第三方解决的，则在解决中状态时，选择转第三方，将 bug 置为第三方处理中状态，如下图：（选择转第三方处理时，需要写明应该写明第三方给出的解决方案和解决时间）



The screenshot shows a software interface for managing bugs. At the top, there is a navigation bar with icons for 'tests' and other functions. Below the navigation bar, there is a toolbar with several buttons: '编辑' (Edit), '备注' (Notes), '分配' (Assign), '更多' (More), '已解决' (Solved), '转第三方' (Transfer to Third Party), and '管理' (Management). The '转第三方' button is highlighted with a red box. Below the toolbar, there is a section titled '详情' (Details) with the following information:

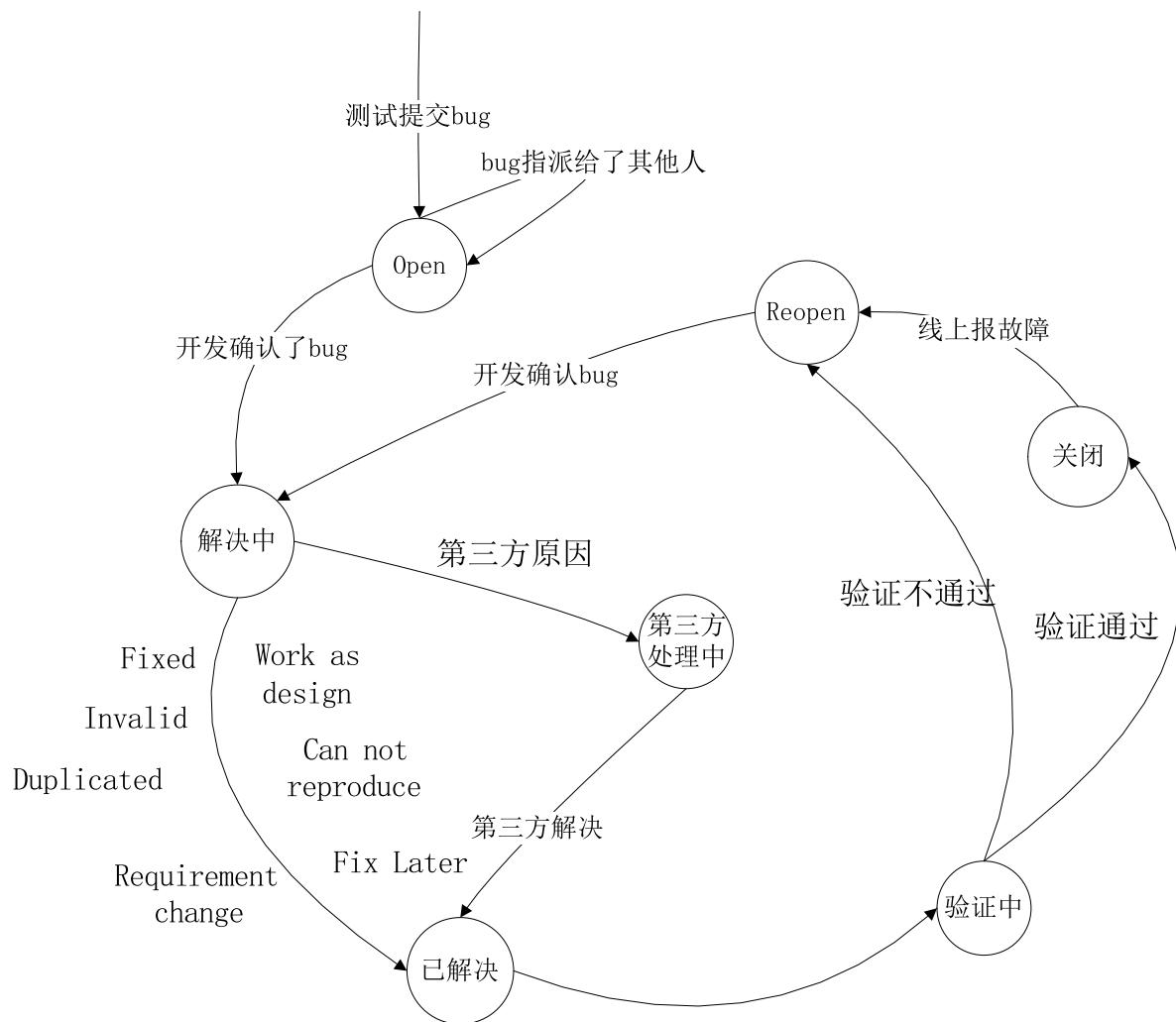
类型:	<input checked="" type="checkbox"/> 故障	状态:	<span>解决中 (查看工作流)</span>
优先级:	<span>P4-Improve</span>	解决结果:	Can not reproduce

(6) 关闭：测试人员验证 bug 已经被解决。

### 3. bug 解决结果

- (1) Fixed: 开发人员通过修改代码解决了故障，测试人员确认故障已解决后，才能关闭该 bug；
- (2) Invalid: bug 无效，测试操作错误等；
- (3) Duplicated: bug 的产生原因与另外的 bug 一致，原 bug 解决，此 bug 也会解决；
- (4) Work as design: 产品设计如此。测试人员需要跟产品经理确认，才能关闭该 bug；
- (5) Requirement change: 设计方案有缺陷，需要修改方案。开发人员、测试人员需要跟产品经理确认，如果确实需要修改方案，则将 bug 指派到产品经理处理；
- (6) Can not reproduce: bug 无法复现。测试人员如果确认确实无法复现，则该 bug 需要待上线一周之后，观察无报障再关闭；
- (7) Fix Later: 开发人员提出该 bug 在当前版本下无法短时间解决，且 bug 影响度很小，和产品、测试等商量后建议延后（放置未来版本）解决的问题；

#### 4. bug 的状态迁移



#### 5. bug 的提交管理

##### (1) bug 的代码提交

线上 p0 bug 的修改代码在 hotfix 分支上提交，上线后合并到所有分支。master 分支上需要打标签，标签格式如下：

版本号\_code 号\_年月日时分\_fixbug\_bug 编号

线上非 p0 bug 在新建 feature 分支上提交，确定好版本计划后，合并到对应的版本的相应分支上：(客户端: release) / (后端: master)

测试阶段 bug 的修改，在对应的需求 feature 分支上提交

bug 修改代码要单独提交，不要和其他需求修改混在一起，comment 格式如下：

【解决 bug】 bug 编号\_解决 xxxx 故障

## (2) bug 的 JIRA 提交

JIRA 标题:

测试环境: 【BLOCK】(无 BLOCK 不写) - 【影响版本号】【复现概率】模块名-问题简要描述 (20字以内)

Eg: 【BLOCK】【V1.9.5】【必现】合同管理-无法创建合同

Eg: 【V1.10.0】【9/16】合同管理-创建合同按钮颜色异常

生产环境: 【线上 P1】【版本号】【复现概率】模块名-问题简要描述 (20 字以内)

Eg: 【线上 P0】【V1.10.0】【9/16】合同管理-创建合同按钮颜色异常

JIRA BUG 条目内容:

### 【描述】

包含 bug 的一些现象描述、严重度描述等，以及涉及的课程，需要填写课程编号；涉及老师和学生，需要填写老师、学生的账号

### 【重现步骤】

填写测试重现步骤；

填写输入参数；

### 【实际结果】

导入错误截图

填写错误输出参数

### 【期望结果】

填写程序应该输出的正确参数/结果；

如需修复数据需声明，并填写预期正确结果

### 【是否必现】是/否

### 【复现概率】

【模块】：影响的模块，以及版本号迭代项目经理设定的固定模块，诸如 ERP 后台，安卓客户端等

【影响版本】：影响版本要明确填写版本号，不要只填写“主干”

【优先级】：bug 的等级，按照上文提及的等级选择填充

## (3) bug 解决后上线的通知邮件

项目经理需要回复 P0/P1/P2 的报障邮件 (Reply to all)，格式如下：

邮件标题: XXXX 故障问题已解决

邮件内容: XXX 故障已于 XX 日 XX 时 XX 分解决，请各位知悉，本次修复内容包括：

bug 编号      内容描述

解决版本（客户端的 bug 需附上）

邮件接收人： p0/p1/p2 故障邮件接收群及原邮件 list 内的相关人员

## 6. bug review 会议

版本比较稳定后（发版前 2-3 天），由测试发起 bug review 会议，确定本次版本必须要解决的剩余 bug 有哪些，review 的 bug 包括状态是 Open、解决中和第三方处理中，以及状态是已解决，但是解决结果是 Fix later、Work as design、Invalid 的 bug

参会人员：测试经理、产品经理、项目经理，相关开发人员和测试人员（有些 bug 甚至需要涉及业务方的确认）

输出：确认哪些 bug 本次版本必须解决，哪些可以延迟到下一个版本解决