

**RIL版本发布流程**

文档版本：V1.0

更新日期：2017.06.01

版权声明

版权所有©2017 深圳市广和通无线股份有限公司。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

注意

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

商标申明

为深圳市广和通无线股份有限公司的注册商标，由所有人拥有。



版本记录

|  |  |  |
| --- | --- | --- |
| 文档版本 | 更新日期 | 说明 |
| V1.0 | 2017-06-01 | 初始版本 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

适用型号

|  |  |  |
| --- | --- | --- |
| 序号 | 产品型号 | 说明 |
| 1 | RIL | Android 4.0 - 4.4 RIL Android5.0 - 7.0 RIL |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**目录**

[1 目的 5](#_Toc26593)

[2 涉及范围 5](#_Toc5841)

[2.1 涉及产品 5](#_Toc28819)

[2.2 涉及部门 5](#_Toc5193)

[2.2.1 FAE 5](#_Toc10278)

[2.2.2 测试部 5](#_Toc31175)

[2.2.3 软件部 5](#_Toc32422)

[3 代码维护规则 6](#_Toc24176)

[3.1 代码仓库 6](#_Toc8400)

[3.2 版本号命名规则 6](#_Toc28073)

[3.3 代码上库规则 6](#_Toc24480)

[3.3.1 分支命名规则 6](#_Toc743)

[3.3.2 代码提交规则 6](#_Toc25980)

[3.3.3 代码评审 8](#_Toc28852)

[3.3.4 版本发布规则 8](#_Toc11668)

[4 版本维护频率及实施细则 10](#_Toc13573)

[4.1 重要共性问题 10](#_Toc6004)

[4.2 定期更新 10](#_Toc11542)

[4.3 版本准入准出规则 10](#_Toc32721)

[5 发布流程 11](#_Toc11630)

[5.1 主要流程 11](#_Toc20763)

[5.2 流程说明 11](#_Toc18104)

[5.2.1 需求发起 11](#_Toc20836)

[5.2.1.1 软件需求及发布评审流程 11](#_Toc1674)

[5.2.2 软件预测试 14](#_Toc24716)

[5.2.3 研发内部归档 14](#_Toc15195)

[5.2.4 邮件通知 15](#_Toc12510)

[5.2.5 软件发布EIPD归档 15](#_Toc15669)

[5.3 流程图 16](#_Toc5502)

[6 涉及的表单及模板 17](#_Toc2278)

[6.1 ReleaseNote 17](#_Toc7511)

[6.2 预测试表单 17](#_Toc10680)

# 目的

为了确保RIL版本和开发人员发布的版本一致，不会出现版本混乱，保证测试代码版本的稳定性，以及开发代码版本的可控性，使基线库完全的受控起来。通过版本发布、基线发布报告等规程来保证软件生命过程中所有产品的完整性、一致性、可追溯性，同时也保证测试人员的工作效率。若是要变更必须走变更流程。

# 涉及范围

## 涉及产品

适用于广和通所有RIL版本的发布。

## 涉及部门

FAE、测试部、软件部。

### FAE

主要对客户进行产品的技术引导和技术培训,主导为客户进行方案设计。针对客户提出的问题及需求，向研发发起问题流程，并且对研发解决客户问题进行协助及简单RIL需求修改。

### 测试部

主要根据项目需求制定测试标准和规范，并且制定测试计划。按照测试方案和流程对产品进行功能和性能测试，并且对测试数据进行搜集、统计和分析，从而对产品的质量进行分析和评估。

### 软件部

主要是解决项目需求或者FAE提出的客户需求及问题，在保证软件质量的前提下发布软件。

# 代码维护规则

该章节主要对RIL代码维护规则进行说明。

## 代码仓库

Android4.x通用GIT代码仓库：[ght@192.168.1.38:/home/ght/sourcecode/android4.x-ril-fix](mailto:ght@192.168.1.38:/home/ght/sourcecode/android4.x-ril-fix)

Android5.x通用GIT代码仓库：[ght@192.168.1.38:/home/ght/sourcecode/android5.x-ril-fix](mailto:ght@192.168.1.38:/home/ght/sourcecode/android5.x-ril-fix)

Android4.x旧版本GIT代码仓库：[ght@192.168.1.38:/home/ght/sourcecode/android4.x-ril-old](mailto:ght@192.168.1.38:/home/ght/sourcecode/android4.x-ril-old)

Android5.x旧版本GIT代码仓库：[ght@192.168.1.38:/home/ght/sourcecode/android5.x-ril-old](mailto:ght@192.168.1.38:/home/ght/sourcecode/android5.x-ril-old)

RIL代码维护以通用GIT代码仓库为主，旧版本GIT仓库将逐步被通用GIT仓库代替。

代码目录结构解析：

bin目录：存放编译后的RIL库文件，没特殊声明存放文件格式为arm 32bit文件。

doc目录: 存放ReleaseNotes、测试报告及UserGuide

src目录：存放RIL源码。

## 版本号命名规则



## 代码上库规则

### 分支命名规则

1. 需要定制版本必须从主线拉分支，分支名以客户中文首字母大写全拼\_系统版本\_模块命名。

如客户华阳android4.2使用H330模块，那么分支名为 HuaYang\_Android4.2\_H330

2. 需要从某个标签版本中进行修改必须执行git branch -b branch\_name tag\_name切换标签同时创建分支，修改后版本号以临时版本进行命名。如标签版本为RIL\_V40TO44.01.02,修改后版本应为RIL\_V40TO44.01.02\_T01

### 代码提交规则

1. 每次修改代码前必须与服务器代码同步至最新。

2. 代码提交格式，本地普通电脑均需要配置

[\\192.168.1.140\datashare\00\_软件一部资料库\(C)\_Linux\_RIL\git\_config](\\\\192.168.1.140\\datashare\\00_软件一部资料库\\(C)_Linux_RIL\\git_config)

此目录存放git提交格式的配置，其中.gitconfig内容，红框部分需要根据实际情况填写，/home/ght为电脑用户根目录。

[user]

email = liuqf@fibocom.com

name = LiuQiFeng

[color]

ui = true

[commit]

template = /home/ght/.gitmessage.txt

[core]

editor = vim

而.gitmessage.txt里面存放的是commit的格式：

[CQ/EIPD]:

[Author]:

[Time]:

[Description]:

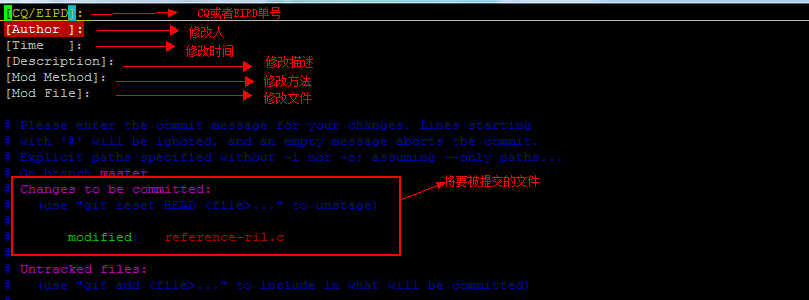
[Mod Method]:

[Mod File]:

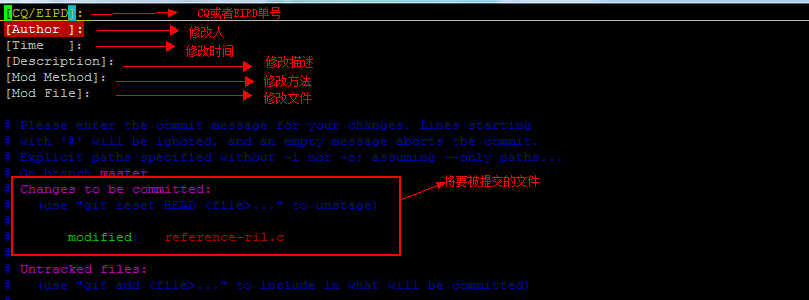
将此两文件放置于Linux电脑用户根目录，本文是ght用户，用户根目录为/home/ght/，无需要重启，立刻生效。下面以modified为例，讲述commit步骤。

modified提交步骤：

1. 执行git status .指令，查看当前修改状态，已经修改文件将显示红色字样，并有前缀modified字样。
2. 执行git add file指令，更新文件，执行git status .指令，再次查看当前修改状态，将绿色字样显示。
3. 执行git commit reference-ril.c指令将显示gitmessage文件内容，以操作vim方式进行文本编译，保存将提交，不保存将不提交。
4. 保存提交后，执行git log指令可查询提交记录

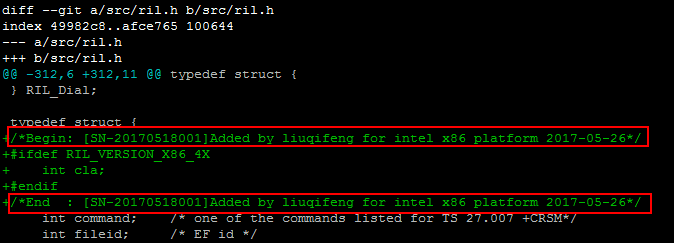


3. 代码修改记录格式（英文）



/\*Begin : [CQ or EIPD number] Added/Modfied/Deleted by who for what date\*/

/\*End : [CQ or EIPD number] Added/Modified/Deleted by who for what date\*/



4. 每修改一个问题，需验证无误后才能提交代码。

5. 一个提交对应一个问题，避免一个提交对应多个问题

### 代码评审

现阶段RIL还未添加review流程，但后续会添加gerrit评审流程，待评审后才将本地代码合入远程服务器。

38服务器(Gerrit)

本地Linux(Git)

(修改上传)

38服务器(Git)

review（合并)

clone/pull

push

### 版本发布规则

1. 主线版本正式发布时，版本号不能带T及S后缀。必须是主线版本号。

2. 版本每次非正式对外发布，必须更新一次临时版本号。

3. 版本发布时需要在doc目录新建以版本号命名目录，并添加releaseNote、测试报告。

4. 版本发布时需要在bin目录新建以版本号命名目录，并添加编译后的lib库。

5. 版本每次正式与非正式发布都必须打标签。标签名以发布版本号命名，标签注释格式：版本号+TAG VERSION

如：[git](http://lib.csdn.net/base/git" \o "Git知识库" \t "http://blog.csdn.net/da_caoyuan/article/details/_blank) tag -a RIL\_V40TO44.01.02 -m “RIL\_V40TO44.01.02 TAG VERSION“

# 版本维护频率及实施细则

本主章节就与FAE、测试经理与会讨论后得出的RIL版本维护更新的重要途径的结论进行细化。

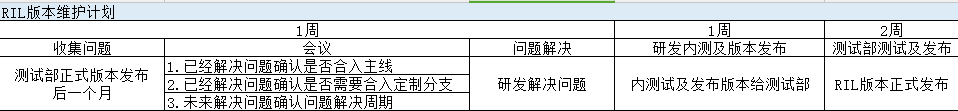
## 版本更新触发条件

### 重要共性问题或需求触发

1. 前端反馈致命的共性bug，对大部分的客户有影响。
2. 增加对新产品的支持。

### 定期更新

1. 定期2个月更新一次RIL版本



2. 待版本稳定后，更新日期将会延长，更新周期需要与前端沟通

## 维护策略

1. 需要合入的问题包括CQ问题，EIPD问题与需求，FAE解决的问题

2. 前一轮主线版本发布一个月之后的第一周，由开发发起评审会议，输出当前遗留的客户问题与需求列表，相关FAE，产品线，测试部与会，确认合入问题作为此轮版本的需求输入。需求确认之后到此轮版本正式发布之前所提的问题和需求均纳入到下一轮主线版本的需求再作评审，不作为此轮版本的输出限制。

3. 重要共性问题或需求触发版本更新之后，定期维护计划相应延后一个更新周期。

4. RIL版本测试完毕之后，以源码的方式发布。应渠道要求，会针对Android 4.x和5.x，ARM32位同步发布一个.so，测试完毕后单独发布给渠道。

备注：RIL所有对外版本都将经测试部测试后发出。

## 临时版本维护

临时版本维护前期：

1. 由FAE提交客户问题处理流程，研发确认问题后，提供解决方案，发布临时版本。
2. 有验证环境由测试部进行针对性测试，无验证环境，如：音频等需客户配套环境的由FAE进行版本验证测试。

临时版本维护后期：

1. 简单问题代码修改验证由FAE进行修改，按照上库规则进行临时版本上库，发布验证由FAE进行。

如：音频AT添加等。

1. 复杂问题由FAE提交客户问题处理流程，按EIPD流程走，有验证环境由测试部进行针对性测试，无验证环境由FAE进行版本验证。

## 版本准入准出规则

参照模块准入准出规则，细节待讨论。

# 发布流程

该章节主要对软件的发布流程进行说明。

## 主要流程

软件发布主要涉及以下过程：

* 需求发起
* 需求评审
* 需求处理
* 软件预测试
* 软件归档
* 邮件通知
* 软件发布EIPD归档

## 流程说明

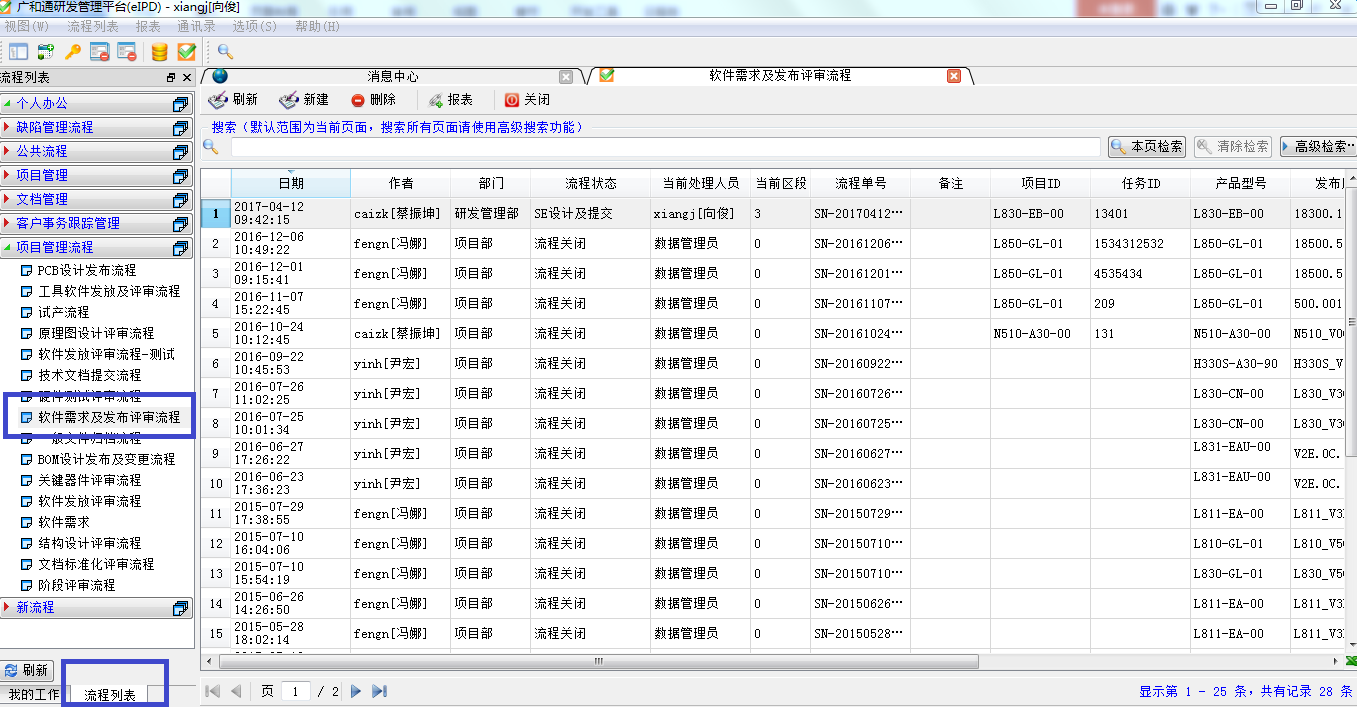
### 需求发起

主要由SWPJM发起，根据项目和客户的需求在EIPD发起流程。

EIPD流程为：软件需求及发布评审流程。

#### 软件需求及发布评审流程

由SWPJM根据项目需求在EIPD发起。如下图：



软件需求及发布评审流程分为8个过程：项目经理软件需求申请、软件需求评审、项目经理确认、SE设计及提交、SE部门经理会审、测试工程师测试及发布、软件总工审批归档、研发CMO。

项目经理软件需求申请阶段，按下图进行填写：



软件需求评审阶段，由评审人员进行评审，交给部门经理审核。评审人员包含但不限于测试人员，FAE人员，软件相关人员，项目经理等。



项目经理确认阶段，产品型号需要选择Linux Driver, 测试完成时间需与对应测试经理商量后填入。



当流程处于SE设计及提交时，当前处理人会变为软件工程师，我们需要处理该需求。在需求解决后先进行5.2.1节及5.2.2节，再按下图要求填写对应资料及软件，并且转交给部门经理审核。



### 软件预测试

需求完成后，在版本正式发布前，软件责任人需要模块的基本功能进行预测试，并且提供预测试报告。保证软件的质量。测试项及模版见第6章节。

### 研发内部归档

预测试结束后，需要根据项目的类型把相应的文件归档到研发服务器，并且在代码管理服务器上面做记录，例如：GIT上面需要打标签，标注当前版本从哪条修改之后发布的版本。

归档服务器：[\\192.168.1.140\50\RD\_Main\_Documents\RIL版本管理](\\\\192.168.1.140\\50\\RD_Main_Documents\\RIL版本管理)

在归档服务器新建以发布版本名命名的文件夹，将服务器版本git clone下载，并打包成zip格式文件防止加密，zip文件名以发布版本名字命名。并将zip拷贝于对应归档服务器的版本文件夹目录，于当前目录解压zip压缩包。

归档服务器发布版本归档文件：

doc/版本ReleaseNote

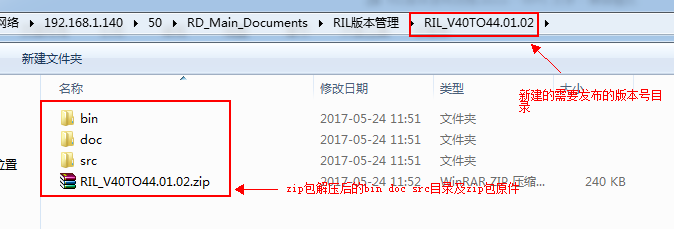
doc/版本预测试报告

src/代码

bin/版本号/xxx.so

上述文件ZIP压缩包

如图：



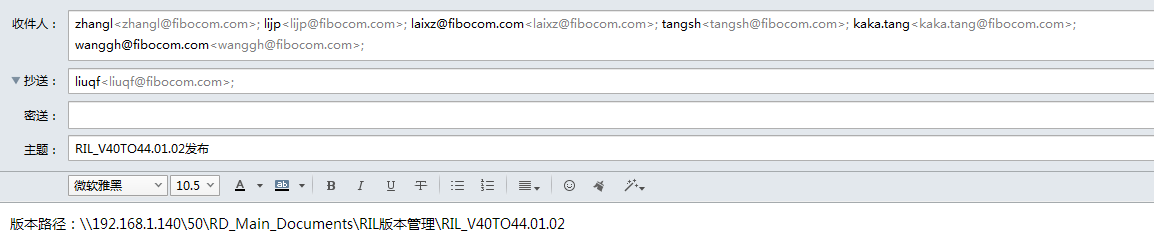
### 邮件通知

文件归档后，首先通过邮件通知研发总监、测试相关人员和部门相关人员。

邮件格式如下：

|  |  |
| --- | --- |
| 邮件接收者 | 研发总监、测试相关人员、部门相关人员 |
| 邮件标题 | XXX.XXX.XXX.XXX版本发布 |
| 邮件内容 | XX工：  XXX.XXX.XXX.XXX版本基于 XXX.XXX.XXX.XXX版本做了如下修改：  1，......  2，......  3，......  路径：\\192.168.1.140\50\RD\_Main\_Documents\RIL版本管理\XXX版本 |

如图：



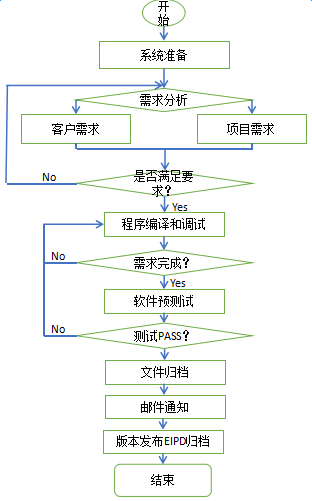
### 软件发布EIPD归档

经测试部测试通过后，将由软件总工审批归档，EIPD归档路径分别为：

abf://192.168.1.8/Soft/对外标准版本/Linux Driver

abf://192.168.1.8/Soft/对外定制版本/Linux Driver

## 流程图



# 涉及的表单及模板

## ReleaseNote



## 预测试表单

