

GIT 代码提交规范

文件状态： <input checked="" type="checkbox"/> 草稿 <input type="checkbox"/> 正在修改 <input type="checkbox"/> 正式发布	部门	系统软件组
	版本	0.2
	作者	杨贵平
	完成时间	20170317
	审核	杨贵平
	审核时间	20170315
密级状态：绝密() 秘密() 内部资料(<input checked="" type="checkbox"/>) 公开()		

修改记录:

版本	修订者	时间	说明
0.1	杨贵平	2017/3/17	基本版本, 只支持 ADD/FIX/MOD 的提交
0.2	杨贵平	2017/4/25	多个 projects 修改的关联
0.3	杨贵平	2017/08/14	明确代码提交识别, 及提交文件的要求.

一. 背景及问题:

因为涉及到系统代码, project 多, 维护的人也多, 修改历史也长.

本文档一方面要求:

- 修改中要标识谁/为什么修改
- 提交代码简洁, 去除不需要的修改, 只提交需要的修改;
- 提交信息的简洁明了。

请大家严格按规范执行. 代码审批时, 也要检查提交信息是否符合规范.

二. 思路和方法:

1. 修改的识别:

增加修改的识别标识, 用于告诉其它人, 这个地方是谁, 为什么修改, 修改的开始和结束位置.

这个的目的方便其它的针对公版代码对比查问题时, 方便区分哪些是我们自己做的修改, 哪些是公版更新的. 对比合并时方便有针对性的处理.

建议的识别:

单行:

```
//steven: commens for change.
```

多行:

```
// steven: commens for changes.
```

```
{codes added}
```

```
// steven: end
```

这两条都适用 C/C++, JAVA 代码. 同样也适用于脚本的修改.

其它特殊修改说明:

1. 如果修改的地方是其它人已经修改过的地方, 可以不另外增加识别. 直接修改, 在 git 提交信息中说明即可.
2. 自行修改代码, 识别必需加. 防止别人合并代码时, 把你的修改合并掉了.
3. 合并原厂提供的 patch, 如果是单个问题的 patch, 修改的文件/内容不多, 必需加.

如果是批量的 patch 合并, 修改的文件很多, 内容也很多. 可以不加. 但在提交的 git 信息中要说明清楚.

在提交之后, 用 **git diff** 检查时, 注意是否都有做明显的标识. 如果没有, 需要补上.

2. 代码提交简洁:

代码提交是把针对问题需要的修改提交上去, 不必要的内容, 无关内容不要提交去, 其它问题的修改内容也要分开提交. 不要把好几个问题合并在一个提交中. 后续跟进时不好处理.

典型情况:

整个提交中有只一个词, 提交的修改中, 列表了好几个文件的修改. 但实际上这些文件中都只是文件格式变化了, 空格/空行的变化而已.

还有新开的案子, 内核中调好了. 所有一堆问题全部一个提交搞定. 当时是省事. 但时间久了, 有问题要查当时的修改点时, 就需要把整个提交找出来, 一点点看, 非常麻烦.

所以在提交之前, 用 `git diff` 查看一下修改情况. 如果是无法的修改, 注意处理一下. 不要直接提交无用的修改信息.

3. 提交信息简洁:

`commit` 信息要完整简要, 格式要求包含标题及正文.

标题要求:

1. 开头用大写的关键字开头,
关键字有: `ADD/FIX/MOD`, 但要区分清楚:
`ADD` 代表新加的功能.
`FIX` 代表解决 `BUG`.
`MOD` 其它的基本都可以用这个了.

这个要严格执行, 后续发布软件时生成修改历史就要用这个. 代码审批时, 也要重点检查.

2. 关键字与正文之前用": " 冒号+空格分开
这个要求, 主要是方法阅读.
3. 标题可以用中文, 也可以用英文. 标题描述不要太长. 30 个字母以内.
方便在 `gerrit` 中显示完整. 也方便阅读理解.
3. 结尾如果只有标题, 不要用符号, 如果有正文, 要加": "
方便在 `gerrit` 中显示完整. 也方便阅读理解.

正文要求:

1. 正文和标题之间要空一行. 方便阅读.
2. 正文在 72 个字符处换行. 每行不要太长.
3. 正文要说明是针对什么问题或为什么修改. 而不是如何修改.
4. 如果问题修改涉及多个 `projects`, 则需要在正文中标明.

如果修改的内容少, 可以只要标题.

如果内容比较多, 必需写正文, 说明详细情况.

多 projects 例子:

ADD: 增加 AP6255 模块支持

修改涉及:

common/

device/amlogic/

hardware/amlogic/wifi/

这样提交一目了然, 其它人同步代码时也减少出错机会。