

# 定制版本 manifest.xml 的管理

<b>文件状态：</b> <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	0.1
	作者	杨贵平
	完成时间	<b>20170523</b>
	审核	杨贵平
	审核时间	<b>20170523</b>
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		2012 / 6 / 13	Initial draft

## 一. 背景及问题:

之前的代码使用单独的 manifest.xml, 因为与代码分开, 不能及时更新。缺少维护。造成很多定制版本的代码在后期取不出来或是取出来运行有问题。

现在把 manifest.xml 放进代码中, 与代码一起维护。避免 manifest.xml 与代码脱节。

## 二. 思路和方法:

### 1. 生成定制版本的 manifest.xml:

在已经编译准备好的 android 代码目录下运行:

```
repo manifest -r --suppress-upstream-revision -o 001_3G.xml
```

生成的文件内容每一行就是一个 project, 如:

```
<project name="L_gx/platform/frameworks/wilhelm" path="frameworks/wilhelm"
revision="8ab36ee4e279148987ed3f6d2f3de3711aee8afb"/>
<project name="L_gx/platform/hardware/amlogic/audio" path="hardware/amlogic/audio"
revision="6acb77396d8b03703f6704ecf82d446b5451bc66"/>
```

这个就是此定制版本的代码 manifest.xml。 用这个可以取出此本地一样的代码。

但前提是本地代码全部都有提交, 如果没有提交, 其它人取代码时会报错。

### 2. 定制分支的修改:

前面生成的文件, 可以取出对应的代码, 但如果后续有更新时, 此文件就到不到最新的修改。

所以要做修改, 修改方法就是把对应的 revision="xxxxxx" 里面的 ID, 修改成对应的分支名字。

如:

```
<project name="L_gx/platform/system/core" path="system/core" revision="001_3G"/>
```

详细的修改之后的文件可以参考 S905 5.1 上 001\_3G.xml. 及 S905X 上的 007\_BenQ\_4\_3.xml.

只要包含了相关定制分支的，都要加进来

查找包含指定分支的代码，可用命令：

```
repo forall -c "git remote -v; git branch -av | grep 001_3G"    ( 001_3G 为分支名字，根据实际换成其它的 )
```

输出的列表中，会包含全部 projects 的信息，如果有包含定制分支的，也会显示出来。

### 3. manifest.xml 的提交：

生成的 manifest.xml 要提交到服务器，其它人才能更新下载。文件要放到 .repo/manifests 目录下，并提交到服务器 master 分支。注意：.repo/manifests 本地有一个 default 分支，这个是本地分支，别人同步代码时，取的是 master 分支。所以一定要提交到 master 分支。

### 3. 定制 manifest.xml 的代码同步：

在取得 master 分支的 manifests 目录之后 就可以指定 manifest.xml 并取定制的代码 如取 001\_3G 的定制代码：

在 repo init -u manifests -b master 之后，

```
repo init -m 001_3G.xml
```

```
repo sync -j4
```

代码取出之后就可以编译了。

### 4. 后期的维护：

因为定制代码的 manifest.xml 只是当前的代码，如果后续分支有变化，如 framework/base 在当前代码中没有定制，也没有定制的分支，后续更新时，加了定制，同时也加了定制的分支。此时就需要更新 manifest.xml。

修改提交的方法同上。只是再修改一行而已。

### 5. 注意事项：

a. 在提交 manifest.xml 之后，最好自己先取代码验证一下。防止本地有代码没有提供，造成同步不了。

b. 在 `repo init -m manifest.xml` 之后，要 `ls -al .repo/` 看一下，`manifest.xml` 是否已经指向正确的文件。