# 系统应用IDE方案

关键字：系统应用 系统设置 android7.1 android stdio，settings

## 概述

### 行业现状

芯片公司：MTK/RK/intel, ui定制非常少，并且主要是处理系统底层，一般用source insight等文本工具

手机厂商：hw目前只是用AS当文本工具看。Oppo/小米/酷派目前一般都用AS，各种导入包/依赖/AAPT编译问题，如何导入和解决相关的编译问题/网上参考资料较少。一般是专门的sdk团队负责搭建，主要编译framework各种jar，暴露hide方法。

Settings的依赖要比SystemUI复杂度类似，以Settings为例

### 开发环境方案对比

直接把文件夹拖入eclipse/android studio，本质和记事本效率其实差别不大, eclipse目前已经很少用了，因此我们主要考虑AS下的搭建

|  |  |  |
| --- | --- | --- |
| 生命周期 | 记事本环境方案 | AS环境方案 |
| 1.编码 | 工具：vi/sourceinsight/notepad/  文档结构不合理：res和src文档结构不合理，修改资源文件和代码文件非常不方便  语法助手：几乎没有 | 文档结构：  语法助手：自动：import，layout实时预览，代码自动补全，代码语法检查，重构变量/方法等。 |
| 2.编译 | linux的mm命令  语法问题出错：回到1 | 自动 |
| 3. 安装（升级） | com控制台的adb install | 自动 |
| 4.运行 | 手动点击/或者脚本启动 | 自动 |
| 5.调试 | 业务逻辑问题：回到1，插入Log.d代码，然后2->3->4  Ui问题：回到1->2->3->4->5 | 直接依附进程，断点调试 |
| 效率指数 | 1 | 3 |
| 环境配置难度 | 0  不用任何配置 | 3d  1.比较复杂，Settings为例，依赖10个左右的jar包，3个其他工程，需要编译系统sdk，消除as的aapt和系统源码的aapt差异；系统签名 |
| 系统覆盖率 | 100%  应用层，系统层，底层都ok | 20%  只支持部分apk，framwork/native/kernel/driver无法支持 |

目前我们对settings和systemui ui定制多，采用as符合当前业务和效率的需要

### 方案目标

目前我们对settings和systemui ui定制多，采用as符合当前业务和效率的需要。结合目前的协作和工作环境，并且都是远程登陆到服务器工作。该方案需要实现以下目标。

* [as](http://1.as/)环境：在as环境下，可以正常编译，充分利用ide的高效功能，提取系统class.jar，系统签名
* 兼容多种编译方式：支持mk和as模块编译，支持系统全量编译，尽量减少两种编译方式代码的差异性
* 便捷的git操作：团队协作，合理gitignore，可以从服务器代码git pull到本地编写代码
* 可移植性强：便于从android7.1直接迁移到android8.0，由于android系统更新频繁，应用定制必须考虑兼容性问题，做到强移植性，减少如android7.0迁移到8.0的工作量。

### 问题

迁移到AS环境下有一定的工作量：比如系统设置在android7.0又多依赖了settingslib，然后product字段/资源重载机制as的aapt又不支持

要解决如何系统签名，不同的系统应用本身模块依赖负责，如何导入系统隐藏的方法？

举例一个最简单的应用：【系统设置】

product字段/资源重载机制as的aapt又不支持

### AS说明

Android Studio 3.1.2

Build #AI-173.4720617, built on April 14, 2018

JRE: 1.8.0\_152-release-1024-b02 amd64

JVM: OpenJDK 64-Bit Server VM by JetBrains s.r.o

Windows 7 6.1

## AS本地源码环境搭建-Settings依赖分析

基本思想：mk分析为线索，导入jar包和对应的源码工程

### 查找Settings依赖库(Android.mk)

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/79765186) [copy](https://blog.csdn.net/qq1028850792/article/details/79765186)

1. LOCAL\_PATH:= $(call my-dir)
2. include $(CLEAR\_VARS)
4. LOCAL\_SRC\_FILES := \
5. $(call all-logtags-files-under, src)
7. LOCAL\_MODULE := settings-logtags
9. include $(BUILD\_STATIC\_JAVA\_LIBRARY)
11. # Build the Settings APK
12. include $(CLEAR\_VARS)
14. LOCAL\_JAVA\_LIBRARIES := bouncycastle core-oj telephony-common ims-common
15. LOCAL\_STATIC\_JAVA\_LIBRARIES := \
16. android-support-v4 \
17. android-support-v13 \
18. android-support-v7-recyclerview \
19. android-support-v7-preference \
20. android-support-v7-appcompat \
21. android-support-v14-preference \
22. jsr305 \
23. settings-logtags
25. LOCAL\_MODULE\_TAGS := optional
27. LOCAL\_SRC\_FILES := \
28. $(call all-java-files-under, src) \
29. $(call all-java-files-under, src-kpad)
31. LOCAL\_RESOURCE\_DIR := $(LOCAL\_PATH)/res \
32. $(LOCAL\_PATH)/res-kpad \
33. frameworks/support/v7/preference/res \
34. frameworks/support/v14/preference/res \
35. frameworks/support/v7/appcompat/res \
36. frameworks/support/v7/recyclerview/res
38. LOCAL\_PACKAGE\_NAME := Settings
39. LOCAL\_CERTIFICATE := platform
40. LOCAL\_PRIVILEGED\_MODULE := **true**
42. LOCAL\_PROGUARD\_FLAG\_FILES := proguard.flags
44. LOCAL\_AAPT\_FLAGS := --auto-add-overlay \
45. --extra-packages android.support.v7.preference:android.support.v14.preference:android.support.v17.preference:android.support.v7.appcompat:android.support.v7.recyclerview
47. ifneq ($(INCREMENTAL\_BUILDS),)
48. LOCAL\_PROGUARD\_ENABLED := disabled
49. LOCAL\_JACK\_ENABLED := incremental
50. LOCAL\_DX\_FLAGS := --multi-dex
51. LOCAL\_JACK\_FLAGS := --multi-dex **native**
52. endif
54. include frameworks/opt/setupwizard/library/common-full-support.mk
55. include frameworks/base/packages/SettingsLib/common.mk
57. include $(BUILD\_PACKAGE)
59. # Use the following include to make our test apk.
60. ifeq (,$(ONE\_SHOT\_MAKEFILE))
61. include $(call all-makefiles-under,$(LOCAL\_PATH))
62. endif

#### LOCAL\_STATIC\_JAVA\_LIBRARIES

静态库是需要编译进apk的

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/79765186) [copy](https://blog.csdn.net/qq1028850792/article/details/79765186)

1. LOCAL\_STATIC\_JAVA\_LIBRARIES := \
2. android-support-v4 \
3. android-support-v13 \
4. android-support-v7-recyclerview \
5. android-support-v7-preference \
6. android-support-v7-appcompat \
7. android-support-v14-preference \
8. jsr305 \
9. settings-logtags

#### LOCAL\_JAVA\_LIBRARIES

非静态库(非静态库是Android系统自带的库)

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/79765186) [copy](https://blog.csdn.net/qq1028850792/article/details/79765186)

1. LOCAL\_JAVA\_LIBRARIES := bouncycastle core-oj telephony-common ims-common

#### Include文件夹

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/79765186) [copy](https://blog.csdn.net/qq1028850792/article/details/79765186)

1. include frameworks/opt/setupwizard/library/common-full-support.mk
2. include frameworks/base/packages/SettingsLib/common.mk

其中setupwizard/navigationbar和setupwizard/library/和SettingsLib是包含res的工程库,因此不能导入jar包

#### 小结

从Android.mk文件得出结论是:

Settings依赖上述的三个工程setupwizard/navigationbar和setupwizard/library/和SettingsLib (带有res的工程)

静态库

1. android-support-v4 \
2. android-support-v13 \
3. android-support-v7-recyclerview \
4. android-support-v7-preference \
5. android-support-v7-appcompat \
6. android-support-v14-preference \
7. jsr305 \
8. settings-logtags

动态库:

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/79765186) [copy](https://blog.csdn.net/qq1028850792/article/details/79765186)

1. bouncycastle core-oj telephony-common ims-common

### SettingsLib

这个模块比较简单

#### comom

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/80287021) [copy](https://blog.csdn.net/qq1028850792/article/details/80287021) @commom.mk

1. LOCAL\_DEX\_PREOPT := **true**
2. ifeq ($(LOCAL\_USE\_AAPT2),**true**)
3. LOCAL\_STATIC\_ANDROID\_LIBRARIES += \
4. android-support-annotations \
5. android-support-v4 \
6. SettingsLib
7. **else**
8. LOCAL\_RESOURCE\_DIR += $(call my-dir)/res
9. LOCAL\_AAPT\_FLAGS += --auto-add-overlay --extra-packages com.android.settingslib
10. LOCAL\_STATIC\_JAVA\_LIBRARIES += \
11. android-support-annotations \
12. android-support-v4 \
13. SettingsLib
14. endif

#### android.mk

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/80287021) [copy](https://blog.csdn.net/qq1028850792/article/details/80287021)

1. LOCAL\_PATH := $(call my-dir)
2. include $(CLEAR\_VARS)
3. LOCAL\_USE\_AAPT2 := **true**
4. LOCAL\_MODULE := SettingsLib
5. LOCAL\_DEX\_PREOPT := **false**
6. ANDROID\_COMPILE\_WITH\_JACK := **false**
7. #WITH\_DEXPREOPT := false
8. LOCAL\_SHARED\_ANDROID\_LIBRARIES := \
9. android-support-v4 \
10. android-support-v7-recyclerview \
11. android-support-v7-preference \
12. android-support-v7-appcompat \
13. android-support-v14-preference
14. LOCAL\_RESOURCE\_DIR := $(LOCAL\_PATH)/res
15. LOCAL\_JAR\_EXCLUDE\_FILES := none
16. LOCAL\_SRC\_FILES := $(call all-java-files-under, src)
17. include $(BUILD\_STATIC\_JAVA\_LIBRARY)

### 查找Setupwizard依赖库(Android.mk)

上文分析到还依赖于setupwizard

1. include frameworks/opt/setupwizard/library/common-full-support.mk

具体文件如下：

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/80287021) [copy](https://blog.csdn.net/qq1028850792/article/details/80287021)

1. #
2. # Include this make file to build your application against this module.
3. #
4. # Make sure to include it after you've set all your desired LOCAL variables.
5. # Note that you must explicitly set your LOCAL\_RESOURCE\_DIR before including this file.
6. #
7. # For example:
8. #
9. #   LOCAL\_RESOURCE\_DIR := \
10. #        $(LOCAL\_PATH)/res
11. #
12. #   include frameworks/opt/setupwizard/library/common-full-support.mk
13. #
15. # Check that LOCAL\_RESOURCE\_DIR is defined
16. ifeq (,$(LOCAL\_RESOURCE\_DIR))
17. $(error LOCAL\_RESOURCE\_DIR must be defined)
18. endif
20. # Add --auto-add-overlay flag if not present
21. ifeq (,$(findstring --auto-add-overlay, $(LOCAL\_AAPT\_FLAGS)))
22. LOCAL\_AAPT\_FLAGS += --auto-add-overlay
23. endif
25. # Include setup wizard library, if not already included
26. ifeq (,$(findstring setup-wizard-lib-full-support,$(LOCAL\_STATIC\_JAVA\_LIBRARIES)))
27. LOCAL\_RESOURCE\_DIR += \
28. $(call my-dir)/main/res \
29. $(call my-dir)/eclair-mr1/res \
30. $(call my-dir)/full-support/res
31. LOCAL\_AAPT\_FLAGS += --extra-packages com.android.setupwizardlib
32. LOCAL\_STATIC\_JAVA\_LIBRARIES += setup-wizard-lib-full-support
33. endif
35. ## Include transitive dependencies below
37. # Include support-v7-appcompat, if not already included
38. ifeq (,$(findstring android-support-v7-appcompat,$(LOCAL\_STATIC\_JAVA\_LIBRARIES)))
39. LOCAL\_RESOURCE\_DIR += frameworks/support/v7/appcompat/res
40. LOCAL\_AAPT\_FLAGS += --extra-packages android.support.v7.appcompat
41. LOCAL\_STATIC\_JAVA\_LIBRARIES += android-support-v7-appcompat
42. endif
44. # Include support-v7-recyclerview, if not already included
45. ifeq (,$(findstring android-support-v7-recyclerview,$(LOCAL\_STATIC\_JAVA\_LIBRARIES)))
46. LOCAL\_RESOURCE\_DIR += frameworks/support/v7/recyclerview/res
47. LOCAL\_AAPT\_FLAGS += --extra-packages android.support.v7.recyclerview
48. LOCAL\_STATIC\_JAVA\_LIBRARIES += android-support-v7-recyclerview
49. endif

#### 资源LOCAL\_RESOURCE\_DIR

依赖的资源文件如下

1. LOCAL\_RESOURCE\_DIR += \
2. $(call my-dir)/main/res \
3. $(call my-dir)/eclair-mr1/res \
4. $(call my-dir)/full-support/res
5. = frameworks/support/v7/appcompat/res
6. frameworks/support/v7/recyclerview/res

#### LOCAL\_STATIC\_JAVA\_LIBRARIES

setup-wizard-lib-full-support

android-support-v7-appcompat

android-support-v7-recyclerview

#### setup-wizard-lib-full-support模块分析

setup-wizard-本身又依赖setup-wizard-lib-full-support，这个在android,mk定义了该模块

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/80287021) [copy](https://blog.csdn.net/qq1028850792/article/details/80287021)

1. ##
2. # Build the full-support library, which includes RecyclerView and any other support libraries as
3. # they are integrated.
4. #
5. include $(CLEAR\_VARS)
7. LOCAL\_AAPT\_FLAGS := --auto-add-overlay \
8. --extra-packages android.support.v7.appcompat \
9. --extra-packages android.support.v7.recyclerview
10. LOCAL\_MANIFEST\_FILE := main/AndroidManifest.xml
11. LOCAL\_MODULE := setup-wizard-lib-full-support
12. LOCAL\_RESOURCE\_DIR := \
13. $(LOCAL\_PATH)/main/res \
14. $(LOCAL\_PATH)/eclair-mr1/res \
15. $(LOCAL\_PATH)/full-support/res \
16. frameworks/support/v7/appcompat/res \
17. frameworks/support/v7/recyclerview/res
18. LOCAL\_SDK\_VERSION := current
19. LOCAL\_SRC\_FILES := $(call all-java-files-under, main/src eclair-mr1/src full-support/src)
20. LOCAL\_STATIC\_JAVA\_LIBRARIES := \
21. android-support-v4 \
22. android-support-v7-appcompat \
23. android-support-v7-recyclerview
25. include $(BUILD\_STATIC\_JAVA\_LIBRARY)

综上：setup-wizard-lib-full-support的AndroidManifest 是main/AndroidManifest.xml，源码目录为 main/src eclair-mr1/src full-support/src，资源文件为main/res /eclair-mr1/res full-support/res

，依赖jar为android-support-v4/android-support-v7-appcompat/android-support-v7-recyclerview

#### 小结

common-full-support依赖setup-wizard-lib-full-support

### 依赖关系图



## 导入对应的jar和源码

### 关于support.jar

*/\* implementation files('framework-libs/android-support-design.jar')*

*implementation files('framework-libs/android-support-v4.jar')*

*implementation files('framework-libs/android-support-v7-appcompat.jar')*

*implementation files('framework-libs/android-support-v7-recyclerview.jar')*

*implementation files('framework-libs/android-support-v13.jar')\*/*

*//for we need res ,must*

compile **'com.android.support:design:25.4.0'**

compile **'com.android.support:support-v4:25.4.0'**

compile **'com.android.support:appcompat-v7:25.4.0'**

compile **'com.android.support:recyclerview-v7:25.4.0'**

compile **'com.android.support:support-v13:25.4.0'**

### **导入SettingsLib源码**

源码位置：frameworks/base/packages/SettingsLib/

然后设置jdk1.7

根据import报错解决问题：

1. framework方法缺失：需要framework.jar,通过Use Libraries方式导入framework.jar(同时需要配置优先使用这个库)。

2.根据报错libcore/icu：找到对应源码,导入core-libart-classes.jar （也是通过系统make模块编译方式得到）

配置Gradle脚本

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/79765186) [copy](https://blog.csdn.net/qq1028850792/article/details/79765186)

1. provided files('../framework\_libs/core-libart-classes.jar')

### 导入setupwizard/navigationbar源码

源码路径: /frameworks/opt/setupwizard/navigationbar

并设置成Is Library

### 导入setupwizard/library源码

源码路径: frameworks/opt/setupwizard/library

### **Settings**

#### **导入framework.jar**

同上

#### **导入静态库**

android-support直接在AS下用远程仓库获得，注意版本号和系统保持一致

1. LOCAL\_STATIC\_JAVA\_LIBRARIES := \
2. android-support-v4 \
3. android-support-v13 \
4. android-support-v7-recyclerview \
5. android-support-v7-preference \
6. android-support-v7-appcompat \
7. android-support-v14-preference \
8. jsr305 \

jsr305.jar通过系统make得到，

settings-logtags

#### **导入动态库**

以下库都需要通过系统编译得到，源码目录不做展开了

**[javascript]** [view plain](https://blog.csdn.net/qq1028850792/article/details/79765186) [copy](https://blog.csdn.net/qq1028850792/article/details/79765186)

1. LOCAL\_JAVA\_LIBRARIES := bouncycastle core-oj telephony-common ims-common

telephony-common.jar主要是基带相关的包，导入包含了hide的jar，否则会如下报错：

错误: 找不到符号符号: 变量 BEARER\_BITMASK位置: 类 Carriers

错误: 找不到符号符号: 变量 EXTRA\_KEY\_ALERT\_TITLE位置: 接口 Phone

## 编译

这里描述AS编译过程产生的问题

### AS下的特殊配置

#### Found item String more than one time

字符串资源不支持product字段，根据错误提示的字段，使用py脚本自动扫描删除

**html]** [view plain](https://blog.csdn.net/qq1028850792/article/details/80287021) [copy](https://blog.csdn.net/qq1028850792/article/details/80287021)

1. **//dpad-settings\fixStringRes.py**
2. # -\*- coding: utf-8 -\*-
3. import glob
4. import os,shutil
5. '''
6. line='<string name="imei\_information\_title">IMEI information</string>'
7. key = 'storage\_type\_external'
8. pos = line.find('<string name="')
9. v = RepeatString.get(key, -1)
10. '''
11. def fixStringRes(f):
12. reg0 = 'product="'
13. regs1=['product="tablet"','product="emulator"','product="nosdcard"','product="device"']
14. endTag = "</string>"
15. filename= "strings.xml"
16. RepeatString={'imei\_information\_title':0
17. ,'storage\_summary\_format':0
18. ,'storage\_type\_internal':0
19. ,'storage\_type\_external':0}
20. tmp = open(filename, "w")
21. file = open(f)
22. needWrite=True
23. for line in file:
24. pos = line.find('<string name="')
25. if pos >= 0:
26. key = line[pos + len('<string name="') : line.find('" ')]
27. v = RepeatString.get(key, -1)
28. if v >= 0:
29. print "find RepeatString obj"
30. RepeatString[key] = v + 1
31. if v >= 1:
32. print "rm RepeatString"
33. continue
34. if needWrite == False:
35. print "rm cur line=%s,[needWrite]=" % line, needWrite
36. needWrite = line.find(endTag) >= 0
37. continue
38. needWrite = True
39. if line.find(reg0) >= 0:
40. for reg in regs1:
41. if line.find(reg) >= 0:
42. print "find it!!"
43. needWrite = False
44. break
45. if needWrite == False:
46. needWrite = line.find(endTag) >= 0
47. print "rm cur line=%s,[needWrite]=" % line, needWrite
48. continue
49. '''print "save line=%s[needWrite]=" % line, needWrite '''
50. tmp.write(line)
51. tmp.close()
52. file.close()
53. if os.path.isfile(f):
54. os.remove(f)
55. else:
56. print "error: %s" %f
57. f = f[:len(f)-len(filename)]
58. shutil.move(filename, f)
59. files = glob.glob(r'.\settings\res\values\*\\strings.xml')
60. for f in files:
61. print "handle File at %s" % f
62. fixStringRes(f);

#### gesture\_fingerprint\_swipe是空文件异常

这个是手势定义，都是空文件，另外还有[gesture\_ambient\_lift](https://android.googlesource.com/platform/packages/apps/Settings/+/12e4beb%5E%21/)等

在make环境ok，但是as不行。

解决办法：重名为.png

#### 如何保证系统自带的api优先识别

比如要添加framework.jar 和core.jar。在项目的build.gradle 最后添加上面这段代码，options.compilerArgs.add （），在括号中填写真实的路径，有多个jar包就用分号隔开。

**[html]** [view plain](https://blog.csdn.net/qq1028850792/article/details/80287021) [copy](https://blog.csdn.net/qq1028850792/article/details/80287021)

1. allprojects {
2. repositories {
3. jcenter()
4. }
5. gradle.projectsEvaluated {
6. tasks.withType(JavaCompile) {
7. options.compilerArgs **<<** "-Xlint:unchecked" **<<** "-Xlint:deprecation"
8. options.compilerArgs.add('-Xbootclasspath/p:framework\_libs/telephony-common.jar;framework\_libs/core.jar')
9. }
10. }
11. }

#### android自带的属性规范

AS不支持@\*,会提示

<item name="@\*android: not found

'@\*android:attr/regularColor' not found 等

因为已经不支持@开头使用android自带的属性，我们只要把@符号删掉就可以了

<https://www.niwoxuexi.com/blog/android/article/310.html>

<https://stackoverflow.com/questions/47033594/android-studio-3-0-error-style-attribute-androidattr-windowenteranimation-n>

### Jdk8格式

Android7采用的java1.8编译，不少代码是1.8规范的，AS默认是1.7的。

修改有两个办法

方法一：让AS支持java1.8（推荐）：在gradle下配置：

compileOptions {

sourceCompatibility JavaVersion.VERSION\_1\_8

targetCompatibility JavaVersion.VERSION\_1\_8

}

方法二：把1.8的代码改为1.7规范

### buildToolsVersion 异常

报错：Execution failed for task ':app:transformClassesWithDesugarForDebug'

解决办法：Open your build.gradle and use below

android {

compileSdkVersion 26

buildToolsVersion "26.0.1"

Then **Clean-Rebuild-Run** [.MORE](https://stackoverflow.com/questions/45833937/execution-failed-for-task-apptransformclasseswithdesugarfordebug-error-on-andro?utm_medium=organic&utm_source=google_rich_qa&utm_campaign=google_rich_qa)

### preferenceStyle不支持

删除

<**item name="@android:preferenceStyle"**>@style/Preference</**item**>

<**item name="@android:editTextPreferenceStyle"**>@style/EditTextPreference</**item**>

<**item name="@android:dropdownPreferenceStyle"**>@style/Preference.DropDown.Material</**item**>

<**item name="@android:preferenceFragmentStyle"**>@style/PreferenceFragmentStyle</**item**>

### EventLogTags找不到

Make 编译这个class吧

### Gradle异常

异常打印：DexArchiveMergerException: r8.errors.CompilationError: Program type already present:android.Manifest

解决办法：clean your project；then rebuild。

<https://stackoverflow.com/questions/47079710/dexarchivemergerexception-with-android-studio-3-0>

provided files(**'../framework\_libs/classes-jarjar.jar'**)

**classes-jarjar.jar 和supprot-v14重合了？**

## 安装

### PARSE\_FAILED\_MANIFEST\_MALFORMED

可以编译出apk了，但是此时无法安装。配置文件有异常

C:\Users\key.guan\Desktop>adb install -r -d F:\K\1-src\3399\test\dpad-settings\build\outputs\apk\debug\dpad-settings-debug.apk

adb: failed to install F:\K\1-src\3399\test\dpad-settings\build\outputs\apk\debug\dpad-settings-debug.apk: Failure [INSTALL\_PARSE\_FAILED\_MANIFEST\_MALFORMED: Fai

led parse during installPackageLI: /data/app/vmdl360143393.tmp/base.apk (at Binary XML file line #153): <meta-data> requires an android:value or android:resource attribute]

Failure [INSTALL\_PARSE\_FAILED\_MANIFEST\_MALFORMED: Failed parse during installPackageLI: /data/app/vmdl566395868.tmp/base.apk (at Binary XML file line #3421): <meta-data> requires an android:value or android:resource attribute]

INSTALL\_PARSE\_FAILED\_MANIFEST\_MALFORMED meta-data android:resource

### signatures do not match

问题：Failure [INSTALL\_FAILED\_UPDATE\_INCOMPATIBLE: Package com.android.settings signatures do not match the previously installed version; ignoring!]

分析：没有系统签名！

解决：系统签名

### Adb Failure [INSTALL\_FAILED\_TEST\_ONLY

adb install –t –r

### base.apk code is missing

问题：Failure [INSTALL\_FAILED\_INVALID\_APK: Package couldn't be installed in /data/app/com.android.settings-1: Package /data/app/com.android.settings-1/base.apk code is missing]

分析：AS的编译机制，系统可能PMS对于apk的解析处于一个错误的状态

解决：$ adb shell pm uninstall com.android.settings

DELETE\_FAILED\_INTERNAL\_ERROR

Error while Installing APK

重启系统就好！

## 运行

### DrawerLayout MeasureSpec.EXACTLY

DrawerLayout must be measured with MeasureSpec.EXACTLY.错误

引用android.support.v4包中的drawerlayout时经常出现这种错误

[drawerlayout必须给一个确定的大小不能用wrap\_content](https://blog.csdn.net/mikogodzd/article/details/50164669)

触发方法： 系统设置->关于手机

类似地方：**Settings$SimStatusActivity/Settings$ImeiInformationActivity**

#### 问题分析

发生在DeviceInfoSettingsActivity，其他Activity不会发生该问题原因在于风格，之前由于AS下面有些风格不支持，删除了之后，引起的bug。

解决办法：

**android:layout\_width="match\_parent"**

**android:layout\_height="match\_parent"**

1. 设死长宽，修改为1920\*1080可以规避，不会报错，但是异常
2. DeviceInfoSettingsActivity成弹窗风格，**android:theme="@style/Theme.SubSettingsDialogWhenLarge"**

在使用ToolBar+DrawerLayout的时候，总是报如下的错误：

**android:layout\_width="match\_parent"**

**android:layout\_height="match\_parent"**

**https://blog.csdn.net/qq\_38859786/article/details/72852950**

#### 问题解决

直接删除修改风格

，但是此时

### SeekBar$OnSeekBarChangeListener)' on a null object reference

为何原生的没有？NPE

### SecurityException: Package com.android.settings is currently frozen!

adb shell am broadcast -a android.intent.action.MASTER\_CLEAR

显示异常。

### Must specify preferenceTheme in theme

### @android:attr/preferenceFragmentStyle' not found

getActivity().getTheme().resolveAttribute(

android.support.v7.preference.R.attr.***preferenceTheme***, tv, **true**);