# KJI2Home

悬浮窗修复原理，图像平滑原理

判断相机打开优雅的方法：CameraManager.java AvailabilityCallback

学习flycamera设计hal

找出修改：

fw.base下的19.1.1-3.19

如何少监听一次

扫描app的最佳位置。

避免

adb shell dumpsys meminfo com.android.settings

无法查询到内存信息

# Anr监控

05-09 09:51:48.335 D/BlockMonitor( 761): this binder call cost 5128 ms

05-09 09:51:48.335 D/BlockMonitor( 761): java.lang.Throwable

05-09 09:51:48.335 D/BlockMonitor( 761): at android.os.BlockMonitor.checkBinderTime(BlockMonitor.java:22)

05-09 09:51:48.335 D/BlockMonitor( 761): at android.os.BinderProxy.transact(Binder.java:622)

05-09 09:51:48.335 D/BlockMonitor( 761): at com.dji.protocol.IProtocolManager$Stub$Proxy.addPackListener(IProtocolManager.java:162)

05-09 09:51:48.335 D/BlockMonitor( 761): at com.dji.protocol.ProtocolManager.addPackListener(ProtocolManager.java:58)

05-09 09:51:48.335 D/BlockMonitor( 761): at com.dji.systemui.flycamera.FlyCameraUI.updateSkyTalkChangedView(FlyCameraUI.java:81)

05-09 09:51:48.335 D/BlockMonitor( 761): at com.dji.systemui.flycamera.FlyCameraUI.access$100(FlyCameraUI.java:37)

05-09 09:51:48.335 D/BlockMonitor( 761): at com.dji.systemui.flycamera.FlyCameraUI$2$2.run(FlyCameraUI.java:130)

05-09 09:51:48.335 D/BlockMonitor( 761): at android.os.Handler.handleCallback(Handler.java:755)

05-09 09:51:48.335 D/BlockMonitor( 761): at android.os.Handler.dispatchMessage(Handler.java:95)

05-09 09:51:48.335 D/BlockMonitor( 761): at android.os.Looper.loop(Looper.java:167)

05-09 09:51:48.335 D/BlockMonitor( 761): at android.app.ActivityThread.main(ActivityThread.java:6240)

05-09 09:51:48.335 D/BlockMonitor( 761): at java.lang.reflect.Method.invoke(Native Method)

05-09 09:51:48.335 D/BlockMonitor( 761): at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:996)

05-09 09:51:48.335 D/BlockMonitor( 761): at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:864)

05-09 09:51:48.335 W/BlockMonitor( 761): msg { when=-31s91ms callback=com.dji.systemui.flycamera.FlyCameraUI$2$2 target=android.os.Handler }call cost 5129 ms

05-09 09:51:48.335 W/BlockMonitor( 761): java.lang.Throwable

05-09 09:51:48.335 W/BlockMonitor( 761): at android.os.BlockMonitor.checkMsgTime(BlockMonitor.java:31)

05-09 09:51:48.335 W/BlockMonitor( 761): at android.os.Looper.loop(Looper.java:175)

05-09 09:51:48.335 W/BlockMonitor( 761): at android.app.ActivityThread.main(ActivityThread.java:6240)

05-09 09:51:48.335 W/BlockMonitor( 761): at java.lang.reflect.Method.invoke(Native Method)

05-09 09:51:48.335 W/BlockMonitor( 761): at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:996)

05-09 09:51:48.335 W/BlockMonitor( 761): at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:864)

Subject: [PATCH] Add blockMonitor to monitor main thread block, which only

works in userdebug image

Change-Id: Ic7d0b2cb2fb2404f7792f306e28c93d9a3bd10a7

---

core/java/android/os/Binder.java | 11 ++++++++-

core/java/android/os/BlockMonitor.java | 43 ++++++++++++++++++++++++++++++++++

core/java/android/os/Looper.java | 15 ++++++++++++

6 files changed, 80 insertions(+), 1 deletion(-)

create mode 100755 core/java/android/os/BlockMonitor.java

diff --git a/core/java/android/os/Binder.java b/core/java/android/os/Binder.java

index ea8ba2f..7fc94a2 100644

--- a/core/java/android/os/Binder.java

+++ b/core/java/android/os/Binder.java

@@ -612,7 +612,16 @@ final class BinderProxy implements IBinder {

public boolean transact(int code, Parcel data, Parcel reply, int flags) throws RemoteException {

Binder.checkParcel(this, code, data, "Unreasonably large binder buffer");

if (Binder.isTracingEnabled()) { Binder.getTransactionTracker().addTrace(); }

- return transactNative(code, data, reply, flags);

+ boolean enableMonitor = BlockMonitor.checkBinderEnabled();

+ long before = 0;

+ if (enableMonitor)

+ before = SystemClock.uptimeMillis();

+

+ boolean ret = transactNative(code, data, reply, flags);

+ if (enableMonitor)

+ BlockMonitor.checkBinderTime(before);

+

+ return ret;

}

public native String getInterfaceDescriptor() throws RemoteException;

diff --git a/core/java/android/os/BlockMonitor.java b/core/java/android/os/BlockMonitor.java

new file mode 100755

index 0000000..8af1afc

--- /dev/null

+++ b/core/java/android/os/BlockMonitor.java

@@ -0,0 +1,43 @@

+/\*\*

+ \* add a file for debug slow

+ \* action of main thread

+ \* \*/

+

+package android.os;

+

+import android.util.Log;

+import android.os.SystemProperties;

+public class BlockMonitor {

+ private static final String TAG = "BlockMonitor";

+ private static final String DEBUG\_ENABLE = "ro.debuggable";

+ private static boolean MONITOR\_BINDER\_ENABLE = "1".equals(SystemProperties.get(DEBUG\_ENABLE, "0"));

+ private static boolean MONITOR\_MSG\_ENABLE = MONITOR\_BINDER\_ENABLE;

+ private static final long BINDER\_CALL\_TIMEOUT = SystemProperties.getLong("persist.sys.binder.timeout", 5000);

+ private static final long MSG\_CALL\_TIMEOUT = SystemProperties.getLong("persist.sys.msg.timeout", 5000);

+

+ static void checkBinderTime(long before) {

+ try {

+ long cost = SystemClock.uptimeMillis() - before;

+ if(cost >= BINDER\_CALL\_TIMEOUT) {

+ Log.d(TAG, "this binder call cost " + cost + " ms", new Throwable());

+ }

+ } catch (Exception e) {}

+ }

+

+ static void checkMsgTime(long before, String msg) {

+ try {

+ long cost = SystemClock.uptimeMillis() - before;

+ if(cost >= MSG\_CALL\_TIMEOUT) {

+ Log.w(TAG, " msg " + msg + "call cost " + cost + " ms", new Throwable());

+ }

+ } catch (Exception e) {}

+ }

+

+ static boolean checkMsgEnabled() {

+ return MONITOR\_MSG\_ENABLE && Looper.myLooper() == Looper.getMainLooper();

+ }

+

+ static boolean checkBinderEnabled() {

+ return MONITOR\_BINDER\_ENABLE && Looper.myLooper() == Looper.getMainLooper();

+ }

+}

diff --git a/core/java/android/os/Looper.java b/core/java/android/os/Looper.java

index d299672..f9b8a9b 100644

--- a/core/java/android/os/Looper.java

+++ b/core/java/android/os/Looper.java

@@ -150,6 +150,19 @@ public final class Looper {

if (traceTag != 0 && Trace.isTagEnabled(traceTag)) {

Trace.traceBegin(traceTag, msg.target.getTraceName(msg));

}

+

+ long before = 0;

+ boolean enableMonitor = BlockMonitor.checkMsgEnabled();

+ String checkMsg = "";

+ if (enableMonitor) {

+ before = SystemClock.uptimeMillis();

+ try {

+ checkMsg = msg.toString();

+ } catch (Exception e) {

+ checkMsg = "msg :what = " + msg.what;

+ }

+ }

+

try {

msg.target.dispatchMessage(msg);

} finally {

@@ -158,6 +171,8 @@ public final class Looper {

}

}

+ if (enableMonitor)

+ BlockMonitor.checkMsgTime(before, checkMsg);

if (logging != null) {

logging.println("<<<<< Finished to " + msg.target + " " + msg.callback);

}

--

1.9.1

# OIO

<https://juejin.im/post/5c8b1bd56fb9a049e12b1692?utm_source=gold_browser_extension%5C>

键盘原理控制

https://stackoverflow.com/questions/4336762/disabling-the-fullscreen-editing-view-for-soft-keyboard-input-in-landscape

frameworks

命令：frameworks\cmds:settings.input.wm,am,添加自己的

风格：这里研究吧，问问睿哥，frameworks\core\res\res\values

debug:用android stadio 开发framework

https://blog.csdn.net/u012455213/article/details/54691340

https://juejin.im/entry/5934cfd8fe88c20061ccc63e

https://cloud.tencent.com/developer/article/1145378

https://zhuanlan.zhihu.com/p/30892074

ReceiverCallNotAllowedException

https://blog.csdn.net/earbao/article/details/9981241

https://blog.csdn.net/A771642/article/details/70284335

Preference详解之Preference系的基本应用和管理(

https://blog.csdn.net/CrazyMo\_/article/details/51496516

Android下PreferenceScreen 加载流程

https://blog.csdn.net/Happy\_Develop\_/article/details/73498447

DatePickerDialog 修改主题颜色

https://blog.csdn.net/wxx\_csdn/article/details/54344388

Tcpdump命令+Wireshark

https://www.cnblogs.com/findyou/p/3491035.html

Android ActionBar学习(二) -- 自定义ActionBar Style

https://blog.csdn.net/yegongheng/article/details/38659147

https://juejin.im/entry/58da1683128fe1006cd7592d

https://juejin.im/entry/58da1683128fe1006cd7592d

https://blog.csdn.net/yegongheng/article/details/38659147

https://github.com/kesenhoo/android-training-course-in-chinese/blob/master/basics/actionbar/styling.md

https://blog.csdn.net/xy\_nyle/article/details/18970211

settings7架构详解(menifest架构，解决目前问题未导向)

系统应用：Systemui! 解剖学习

第三方模块架构了解

Make apk 基本原理

硬编码/make规则扫描gradle，编译，externalNativeBuild，

https://blog.csdn.net/xiaosayidao/article/details/75096276

LOCAL\_USE\_AAPT2

AAPT2：

兼容：‘’：但是找不到资源！！！

不兼容：

？：双引号呢？

https://blog.csdn.net/hj2drf/article/details/53893900

https://fucknmb.com/2017/10/31/aapt2%E8%B5%84%E6%BA%90compile%E8%BF%87%E7%A8%8B/

AAPT1：

不兼容：‘’：

兼容：/$

https://blog.csdn.net/hj2drf/article/details/53893900

NoSuchFieldError: black\_list\_entries Lcom/android/settingslib/R$array;

or its superclasses (declaration of 'com.android.settingslib.R$array' appears in /system/priv-app/Settings/Settings.apk)

http://www.txtlxg.com/228/xiaosayidao/article/details/75096276

Ninja编译原理

https://blog.csdn.net/ksjay\_1943/article/details/53579176

https://blog.csdn.net/kc58236582/article/details/46458765

android系统开发规范

AS环境搭建：settings-logtags原理，如何借鉴架构

mk ;BUILD\_STATIC\_JAVA\_LIBRARY 基本知识

bouncycastle core-oj telephony-common ims-common包的源码目录

编译环境

编译Android时禁用Jack（）odex如何禁止

preferencesupport 的了解

39. LOCAL\_CERTIFICATE := platform

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

60. ifeq (,$(ONE\_SHOT\_MAKEFILE))

9. LOCAL\_AAPT\_FLAGS

所有mk语法

所有问题原因

package.apk韩语？

jsr305 ，如何借鉴架构

andorid系统命名规范？intent，action

权限问题分析（系统原生如何绕过？）。

Sending non-protected broadcast kpad.intent.action.RECORDER\_STOP from system 4543:com.XX.launcher/1000 pkg com.XX.launcher

------------------------------------

Android event日志打印原理

https://blog.csdn.net/yaowei514473839/article/details/53513435

https://blog.csdn.net/darkengine/article/details/8477502

http://www.voidcn.com/article/p-uqlsbgsq-bbs.html

https://blog.csdn.net/yaowei514473839/article/details/53513435

分析源码

分析remote的系统为何慢，干掉有什么隐患

freamwork端更停下。服务端解决问题吧。不用在go端修改代码！因为更新麻烦！

Logger LOG = Logger.getLogger(ShareServer.class.getName());

如何到Response serve(IHTTPSession session)

下载最新版本试试？

网络修复原理

清理C盘

3288开发：

PCBA-test原理：

摘除一个电阻吧@小马

cto进阶之路

davic技术实现

runtime实现原理

build/debug

bugodex原理：crash

Cpp:奔溃调试tombstones，@debugc整理

LOG

定制日志打印逻辑

快速指定某个进程是否可以打印数据

rom修改，设置log的level，对开发开放，对用户关闭，不影响用户自己的程序

AS

脚本研究

https://blog.csdn.net/topgun\_chenlingyun/article/details/7701538

as的ragle，py的，dji-go的

root@zs600a:/data/system 目录解读

多网络源码分析

https://redmine.rockchip.com.cn/issues/158079

ro.wifi.channels= 默认wifi评点，是否可以优化自动选择噪声低的

SurfaceComposerClient 设置翻转

https://blog.csdn.net/androiddeveloper\_lee/article/details/9496005

https://blog.csdn.net/myarrow/article/details/7180561

https://blog.csdn.net/luoshengyang/article/details/7691321

bugs:

https://jira.djicorp.com/browse/SAT-1645

http://read.pudn.com/downloads86/sourcecode/comm/330501/MD5/MD5Checksum.cpp\_\_.htm

F:\K\4-bugs\sessionk0001

1860原理：F:\K\out\gl300k\ota\system\bin

F:\K\4-bugs\ag405

F:\K\4-bugs\gl300k-tiny

F:\K\4-bugs\binder

otg,Usb电压

https://blog.csdn.net/uranus\_wm/article/details/9838847

privison/go/devicetest/pc 优化工艺流程，device密码，至少可以缩短3步,售后

从efuse到hello:https://confluence.djicorp.com/pages/viewpage.action?pageId=36415729

3e错误码方案，loader没挂就好。。。

做好 stop 3288 不后悔的准备

3399：

link.json原理

libdji\_link\_jni原理，协议分析，mk原理@jni/Android.mk

JNI实战收藏，https://developer.android.com/preview/behavior-changes.html#ndk

device 系统模块打包原理

rm500:/ # dmesg | grep avc原理@rm500\_grep\_vc

C:\Users\key.guan\Desktop\lib64了解

launcher3原理拖拽原理，学习研究！不能一直搬家额

白屏原理，install

SystemUI编译环境搭建

系统应用AS下的settigns！;https://www.jianshu.com/p/3a4b5f1c09c1

settings7详解

https://juejin.im/entry/57e8d525bf22ec00587a5848

原理

https://blog.csdn.net/cloud\_jianxi/article/details/45126739原理aapt2和aapt1，修改实战

<path 原理，系统设置实战

https://confluence.djicorp.com/pages/viewpage.action?pageId=34209838

dex原理，如何屏蔽编译优化，为何编译可以优化？

--------------------------

GIT

repo 实战的坑

http://www.cauchy.me/2015/09/27/recovery-from-a-disastrous-git-rebase-mistake/

https://blog.csdn.net/shift\_wwx/article/details/41678773

-------------------------

3288开发：

PCBA-test原理：sessionk0001-reboot.log;各个项目测试原理，bug修复原理;gpio修复原理

recovery:metadata分区原理，挂载原理

# 分享

只有恨少一部分人能做技术大牛，大部分都是要做管理的

厚度很薄

人数增加没有提高效率

isLowPowerModeInternal原理

AutomaticBrightnessController原理，分段调试

Art虚拟机

areDefaultRuntimePermissionsGrantedLPr原理

*// Layout debugging***boolean** layout = SystemProperties.getBoolean(View.DEBUG\_LAYOUT\_PROPERTY, **false**);

android-7.1.1\_r1/frameworks/av/media/libmediaplayerservice/MediaPlayerService.cpp

android-7.1.1\_r1/frameworks/base/services/core/java/com/android/server/pm/DefaultPermissionGrantPolicy.java

分享要实用为主

LOCAL\_PATH:= $(call my-dir)

include $(CLEAR\_VARS)

LOCAL\_SRC\_FILES :=$(call all-java-files-under, src)

LOCAL\_PACKAGE\_NAME := FirstSystemApp

LOCAL\_CERTIFICATE := platform

include $(BUILD\_PACKAGE)

所有的系统周期

编码

编译

安装（升级）

运行

DEBUG

书写文档

Android分享：

0.什么叫系统应用？

1.RM500系统应用AS环境搭建方案/实现

3.系统app家族图

附件技术：

hide暴露方法

-------------------------------------------

图传分享：2）服务设计；input子系统+组合键设计原理

通讯原理；AMS；四大组件原理

1）APP编译原理（APK如何打包，虚拟机如何加载到内存中，生命周期）；

主要分享业务吧，技术作为附件，一张图就好了

------------------

1.分析按键抖动问题，查看内核修改原理

1.完成selinux，数据备份机制

1.分析时间问题

2.分析

--------------

系统FW特性，config化！

1.

Merge:rd-39:18.2.25

自动化测试控件ID

getActiveStreamType原理

WallpaperManagerService AMS

**WallpaperManagerService**

# LOCAL\_PROGUARD\_ENABLED

1.INCREMENTAL\_BUILDS定义/LOCAL\_JACK\_FLAGS

ifneq ($(INCREMENTAL\_BUILDS),)

LOCAL\_PROGUARD\_ENABLED := disabled

LOCAL\_JACK\_ENABLED := incremental

LOCAL\_DX\_FLAGS := --multi-dex

LOCAL\_JACK\_FLAGS := --multi-dex native

endif

2.混淆的原理：LOCAL\_PROGUARD\_ENABLED禁用逻辑

解释下面这两个的问题

LOCAL\_PROGUARD\_ENABLED := disabled

LOCAL\_JACK\_FLAGS := --multi-dex native

3.混淆的语法

ok的实例

##ok

#-keep class org.msgpack.core.buffer.\*\*{\*;}

-keep class org.msgpack.core.buffer.MessageBuffer{\*;}

-keep class org.msgpack.core.buffer.MessageBufferU{\*;}

-keep class org.msgpack.core.buffer.MessageBufferBE{\*;}

BUTTTTT----------------------------------

下面为何不生效呢？

#-keep class org.msgpack.core.buffer.\*

###XXXX111

###-keep class org.msgpack.core.buffer.MessageBuffer

###-keep class org.msgpack.core.buffer.MessageBufferU

###-keep class org.msgpack.core.buffer.MessageBufferBE

###XXXX222

# Keep class (extends) MessageBuffer which are used by reflection.

# -keepclassmembers class org.msgpack.core.buffer.MessageBuffer{

# public <init>(byte[], int, int);

# }

# -keepclassmembers class \*\* extends org.msgpack.core.buffer.MessageBuffer {

# public <init>(byte[], int, int);

# }

###XXXX33333

#-keepclassmembers class

###-keep class org.msgpack.core.buffer.MessageBuffer {

### public <init>(byte[], int, int);

###}

###-keep class org.msgpack.core.buffer.MessageBufferU {

### public <init>(byte[], int, int);

###}

###-keep class org.msgpack.core.buffer.MessageBufferBE {

### public <init>(byte[], int, int);

###}

###XXXX555

###-keepclassmembers class \* {

### @org.msgpack.core.buffer.MessageBuffer <methods>;

### @org.msgpack.core.buffer.MessageBufferU <methods>;

### @org.msgpack.core.buffer.MessageBufferBE <methods>;

###}

# 4.模块重复包含

including ./frameworks/opt/kpad-core/Android.mk ...

build/core/base\_rules.mk:183: \*\*\* frameworks/opt/kpad-core: MODULE.TARGET.JAVA\_LIBRARIES.kpad-core already defined by frameworks/opt/kpad-core.

make: \*\*\* [out/build-rm500.ninja] Error 1

#### make failed to build some targets (46 seconds) ####

TARGET\_BOARD\_PLATFORM=rk3399

TARGET\_PRODUCT=rm500

原因：#include frameworks/opt/kpad-core/Common.mk

--------------------------------------------------------------------------

5.classes.jack'找不到？

build/core/base\_rules.mk:319: warning: ignoring old commands for target `out/target/product/rm500/system/lib64/libjpeghwdec.so'

Starting build with ninja

ninja: Entering directory `.'

ninja: error: 'out/target/common/obj/JAVA\_LIBRARIES/kpad-core\_intermediates/classes.jack', needed by 'out/target/common/obj/JAVA\_LIBRARIES/services.core\_intermediates/classes.jack', missing and no known rule to make it

make: \*\*\* [ninja\_wrapper] Error 1

#### make failed to build some targets (46 seconds) ####

原因：jar模块未定义

-------------

service包含:

Starting build with ninja

ninja: Entering directory `.'

ninja: error: 'frameworks/opt/kpad-core/java/com/android/server/EventLogTags.logtags', needed by 'out/target/common/obj/all-event-log-tags.txt', missing and no known rule to make it

make: \*\*\* [ninja\_wrapper] Error 1

---06-02 20:12:39.265 2531-2839/system\_process D/KJIA: update from reportid in ('57167c8cd2cf7d4956da55fdbb54c1a9c6ca','f85cbe14df05cd41dad9e12deba1d86cb8d2','9a099c5ed8a90d42f5d92e0ded23166c70e5','938d810fd27a2d4a85dacbed8aa04c3f243f','86dc83cdd0b96d4e33d81bbde27856728cd5') null into kji\_analytics\_reports state=1

06-02 20:12:39.265 2531-2839/system\_process D/KJIA: SenderService handler id is 0 data len is:5

--------- beginning of crash

06-02 20:12:42.356 4355-4355/system\_process D/KJIA: KJIA initialised.

06-02 20:12:42.356 4355-4355/system\_process D/KJIA: DebugUtils is start init.

06-02 20:12:42.357 4355-4355/system\_process E/Environment: Path requests must specify a user by using UserEnvironment

java.lang.Throwable

at android.os.Environment.throwIfUserRequired(Environment.java:913)

at android.os.Environment.getExternalStorageDirectory(Environment.java:449)

at com.kpad.core.analytics.util.DebugUtils.init(DebugUtils.java:41)

at com.kpad.core.analytics.KJIA.enableDebug(KJIA.java:57)

at com.kpad.core.KpadReportUtil.initKJIReport(KpadReportUtil.java:27)

at com.kji.KjiServicesHelper.initKJIReport(KjiServicesHelper.java:259)

at com.android.server.SystemServer.run(SystemServer.java:349)

at com.android.server.SystemServer.main(SystemServer.java:226)

at java.lang.reflect.Method.invoke(Native Method)

at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:957)

at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:825)

06-02 20:12:42.364 4355-4355/system\_process D/KJIA: getExternalStorageDirectory = /sdcard

06-02 20:12:42.365 4355-4355/system\_process W/System.err: java.io.FileNotFoundException: /sdcard/kjia.test.data.txt (Permission denied)

06-02 20:12:42.365 4355-4355/system\_process W/System.err: at com.kpad.core.analytics.KJIA.enableDebug(KJIA.java:57)

----------------------------------

out\target\common\obj\APPS\framework-res\_intermediates

proguard\_options 是哪里定义的，作用是什么？

APK打包原理一定要懂得哦

public\_resources.xml

私有资源的ID：src\com\android\internal.R.kji\_lable其实是有的啊

拿到是要重新连接才能编译生效？？

1.INCREMENTAL\_BUILDS定义/LOCAL\_JACK\_FLAGS

ifneq ($(INCREMENTAL\_BUILDS),)

LOCAL\_PROGUARD\_ENABLED := disabled

LOCAL\_JACK\_ENABLED := incremental

LOCAL\_DX\_FLAGS := --multi-dex

LOCAL\_JACK\_FLAGS := --multi-dex native

endif

2.混淆的原理：LOCAL\_PROGUARD\_ENABLED禁用逻辑

解释下面这两个的问题

LOCAL\_PROGUARD\_ENABLED := disabled

LOCAL\_JACK\_FLAGS := --multi-dex native

3.混淆的语法

ok的实例

##ok

#-keep class org.msgpack.core.buffer.\*\*{\*;}

-keep class org.msgpack.core.buffer.MessageBuffer{\*;}

-keep class org.msgpack.core.buffer.MessageBufferU{\*;}

-keep class org.msgpack.core.buffer.MessageBufferBE{\*;}

BUTTTTT----------------------------------

下面为何不生效呢？

#-keep class org.msgpack.core.buffer.\*

###XXXX111

###-keep class org.msgpack.core.buffer.MessageBuffer

###-keep class org.msgpack.core.buffer.MessageBufferU

###-keep class org.msgpack.core.buffer.MessageBufferBE

###XXXX222

# Keep class (extends) MessageBuffer which are used by reflection.

# -keepclassmembers class org.msgpack.core.buffer.MessageBuffer{

# public <init>(byte[], int, int);

# }

# -keepclassmembers class \*\* extends org.msgpack.core.buffer.MessageBuffer {

# public <init>(byte[], int, int);

# }

###XXXX33333

#-keepclassmembers class

###-keep class org.msgpack.core.buffer.MessageBuffer {

### public <init>(byte[], int, int);

###}

###-keep class org.msgpack.core.buffer.MessageBufferU {

### public <init>(byte[], int, int);

###}

###-keep class org.msgpack.core.buffer.MessageBufferBE {

### public <init>(byte[], int, int);

###}

###XXXX555

###-keepclassmembers class \* {

### @org.msgpack.core.buffer.MessageBuffer <methods>;

### @org.msgpack.core.buffer.MessageBufferU <methods>;

### @org.msgpack.core.buffer.MessageBufferBE <methods>;

###}

4.模块重复包含

including ./frameworks/opt/kpad-core/Android.mk ...

build/core/base\_rules.mk:183: \*\*\* frameworks/opt/kpad-core: MODULE.TARGET.JAVA\_LIBRARIES.kpad-core already defined by frameworks/opt/kpad-core.

make: \*\*\* [out/build-rm500.ninja] Error 1

#### make failed to build some targets (46 seconds) ####

TARGET\_BOARD\_PLATFORM=rk3399

TARGET\_PRODUCT=rm500

原因：#include frameworks/opt/kpad-core/Common.mk

--------------------------------------------------------------------------

# 5.classes.jack'找不到

build/core/base\_rules.mk:319: warning: ignoring old commands for target `out/target/product/rm500/system/lib64/libjpeghwdec.so'

Starting build with ninja

ninja: Entering directory `.'

ninja: error: 'out/target/common/obj/JAVA\_LIBRARIES/kpad-core\_intermediates/classes.jack', needed by 'out/target/common/obj/JAVA\_LIBRARIES/services.core\_intermediates/classes.jack', missing and no known rule to make it

make: \*\*\* [ninja\_wrapper] Error 1

#### make failed to build some targets (46 seconds) ####

原因：jar模块未定义

-------------

service包含:

Starting build with ninja

ninja: Entering directory `.'

ninja: error: 'frameworks/opt/kpad-core/java/com/android/server/EventLogTags.logtags', needed by 'out/target/common/obj/all-event-log-tags.txt', missing and no known rule to make it

make: \*\*\* [ninja\_wrapper] Error 1

# 为何aar依赖需要中间变量res等文件，并且只能通过mkanroid来更新

# 编译bug

之前为何编译没有问题

Error 101 Unresolved link/see tag “原理”

## 日志

cannot be resolved or is not a field

1.原理

2.修改overlay为何都是，server.jar寻址不到资源，解压framework-res.apk其实是有的

资源编译原理，怎么去排查问题，佳节资源编译原理，怎么去排查问题，佳节

[ 22% 1035/4563] Building with Jack: out/target/common/obj/JAVA\_LIBRARIES/framework\_intermediates/with-local/classes.dex

FAILED: /bin/bash out/target/common/obj/JAVA\_LIBRARIES/framework\_intermediates/with-local/classes.dex.rsp

ERROR: /home/key/work/rk3399\_android7.1\_v1.00/frameworks/base/core/java/com/kji/KjiCoreHelper.java:120: kji\_browser\_lable cannot be resolved or is not a field

ERROR: /home/key/work/rk3399\_android7.1\_v1.00/frameworks/base/core/java/com/kji/KjiCoreHelper.java:140: kji\_browser\_lable cannot be resolved or is not a field

[ 22% 1035/4563] target Prebuilt: webview (out/target/product/rk3399\_excavator/obj/APPS/webview\_intermediates/package.apk)

Archive: out/target/product/rk3399\_excavator/obj/APPS/webview\_intermediates/package.apk

inflating: out/target/product/rk3399\_excavator/obj/APPS/webview\_intermediates/uncompressedlibs/lib/arm64-v8a/libwebviewchromium.so

inflating: out/target/product/rk3399\_excavator/obj/APPS/webview\_intermediates/uncompressedlibs/lib/armeabi-v7a/libwebviewchromium.so

deleting: lib/arm64-v8a/libwebviewchromium.so

deleting: lib/armeabi-v7a/libwebviewchromium.so

adding: lib/arm64-v8a/libwebviewchromium.so (stored 0%)

adding: lib/armeabi-v7a/libwebviewchromium.so (stored 0%)

[ 22% 1035/4563] target Java: framework (out/target/common/obj/JAVA\_LIBRARIES/framework\_intermediates/classes)

FAILED: /bin/bash out/target/common/obj/JAVA\_LIBRARIES/framework\_intermediates/classes-full-debug.jar.rsp

frameworks/base/core/java/com/kji/KjiCoreHelper.java:120: error: cannot find symbol

appInfo.labelRes = com.android.internal.R.string.kji\_browser\_lable;

^

symbol: variable kji\_browser\_lable

location: class string

frameworks/base/core/java/com/kji/KjiCoreHelper.java:140: error: cannot find symbol

info.labelRes = com.android.internal.R.string.kji\_browser\_lable;

^

symbol: variable kji\_browser\_lable

location: class string

Note: Some input files use or override a deprecated API.

Note: Recompile with -Xlint:deprecation for details.

Note: Some input files use unchecked or unsafe operations.

Note: Recompile with -Xlint:unchecked for details.

2 errors

ninja: build stopped: subcommand failed.

分析：回退版本都不行，显然是缓存导致的

模块编译时正常的呢

主要是Jenkins脚本机制有问题吧，忽然不支持overlay

但是此时copy到framework也不行啊。。删除kji\_browser\_lable，hdmi之类的新增文案也报fial

就是Jenkins脚本改了东西！！！非常影响效率了现在

## 分析

### 报错点

Building with Jack:

out/target/common/obj/JAVA\_LIBRARIES/framework\_intermediates/with-local/classes.dex FAILED:

/bin/bash out/target/common/obj/JAVA\_LIBRARIES/framework\_intermediates/with-local/classes.dex.rsp

ERROR: /home/key/work/rk3399\_android7.1\_v1.00/frameworks/base/core/java/com/kji/KjiCoreHelper.java:120

用Jack来编译framework\_intermediates/with-local/classes.dex，无法引用到新加的资源

com.android.internal.R.string.kji\_browser\_lable

3.编译失败原理

ninja: build stopped: subcommand failed.

make: \*\*\* [ninja\_wrapper] Error 1

#### make failed to build some targets (01:07 (mm:ss)) ####

Build android failed!

4.misc现在通过了，之前为何没有通过

com.android.settings

不混淆为何类的数量变多了

LOCAL\_DX\_FLAGS := --multi-dex --main-dex-list=$(mainDexList) --minimal-main-dex

LOCAL\_JACK\_FLAGS += --multi-dex native