



+



《Android互联网架构师课程》

直播VIP课

做一家受人尊敬的企业，做一位受人尊敬的老师

APK极限压缩与webp应用

- ◆ 1 webp对jpg png的优势 京东为什么会放弃传统图片
- ◆ 2 webp编码原理 跟哈夫曼压缩的不同之处
- ◆ 3 帧内预测技术详解
- ◆ 4 APK极限压缩7大步

课程安排



01

Webp优势



02

Webp编码原理



03

APK极限压缩



04

WebP在Apk的应用



Webp的优势

优势体现在哪里



jpg原图: 30.7kb



webp(60): 2.9kb
体积减少: 90.55%



webp(80): 3.9kb
体积减少: 87.3%



webp(100): 15kb
体积减少: 51.14%

webp编码对于哈夫曼压缩性能更优异些。

哈夫曼与webp本质上都是从编码来解决图像压缩

哈夫曼是对rgb的元数据进行变频压缩

webp编码是通过预测技术对图片压缩



预测技术 (VP8编码 H264)

WebP 压缩使用的图像编码方式与 VP8 视频编码对关键帧压缩方式相同

换句话说解释：Google将视频编码技术搬到了图片上 形成了Webp编码格式

WebP编码流程

WebP 会将图片划分为两个 8x8 色度像素宏块和一个 16x16 亮度像素宏块。在每个宏块内，编码器基于之前处理的宏块来预测冗余动作和颜色信息。

通过图像关键帧运算，使用宏块中已解码的像素来绘制图像中未知部分

通过预测模式 去除冗余数据，实现更高效的压缩。

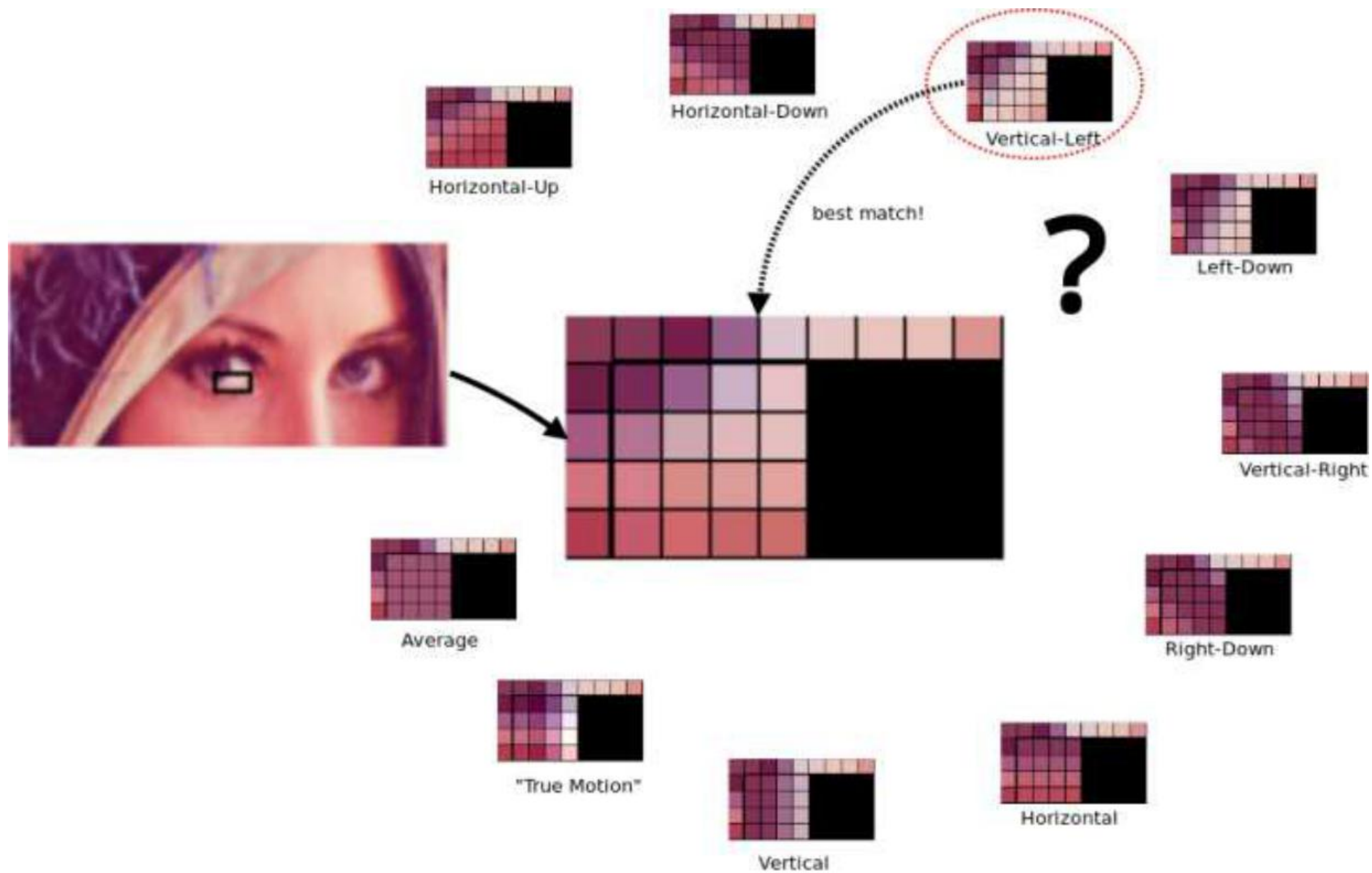


预测模式

- ◆ H_PRED（水平预测）：用宏块左边的列 L 的填充块的每一列；
- ◆ V_PRED（垂直预测）：用宏块上边的行 A 的填充宏块的每一行；
- ◆ DC_PRED（DC预测）：用行 A 和列 L 的像素的平均值作为宏块唯一的值来填充宏块；
- ◆ TM_PRED（TrueMotion预测）：除了行 A 和列 L 之外，用宏块上方和左侧的像素P、A（从P开始）中像素块之间的水平差异以列 L 为基准拓展每一行。



预测技术



WebP的劣势

我们都知道播放视频需要下载播放器，播放器的作用是视频文件进行解码。所以每一个播放器都会包含解码器

WebP使用的是视频编码 难道需要**下载**视频编码器

其实是不需要的，很多厂商已经支持了webp的解码，可以说webp就像png一样普遍。

可是？

在早些年代 那个时候还没有webp，
比如Android4.3之前 他们是不支持webp的



不要怂，一起上！
--- 鲁迅

WebP兼容所有Android系统平台

百度搜索libwebp



下载libwebp

Beyond Linux® From Scratch (systemd edition) - Version 2019-06-12
Chapter 10. Graphics and Font Libraries

[Prev](#)
LibTIFF-4.0.10

[Up](#)
Home

mypai

libwebp-1.0.2

Introduction to libwebp

The libwebp package contains a library and support programs to encode and decode images in WebP format.

This package is known to build and work properly using an LFS-8.4 platform.

Package Information

- Download (HTTP): <http://downloads.webmproject.org/releases/webp/libwebp-1.0.2.tar.gz>
- Download MD5 sum: 02c0c55f1dd8612cd4d462e3409ad35d
- Download size: 3.9 MB
- Estimated disk space required: 45 MB
- Estimated build time: 0.4 SBU



WebP兼容所有Android系统平台

百度搜索libwebp



下载libwebp

Beyond Linux® From Scratch (systemd edition) - Version 2019-06-12

Chapter 10. Graphics and Font Libraries

Prev

LibTIFF-4.0.10

Up

Home

mypai

libwebp-1.0.2

Introduction to libwebp

The libwebp package contains a library and support programs to encode and decode images in WebP format.

This package is known to build and work properly using an LFS-8.4 platform.

Package Information

- Download (HTTP): <http://downloads.webmproject.org/releases/webp/libwebp-1.0.2.tar.gz>
- Download MD5 sum: 02c0c55f1dd8612cd4d462e3409ad35d
- Download size: 3.9 MB
- Estimated disk space required: 45 MB
- Estimated build time: 0.4 SBU

下载地址: <http://downloads.webmproject.org/releases/webp/libwebp-1.0.2.tar.gz>



WebP编译

1 Android.mk增加

ENABLE_SHARED := 1

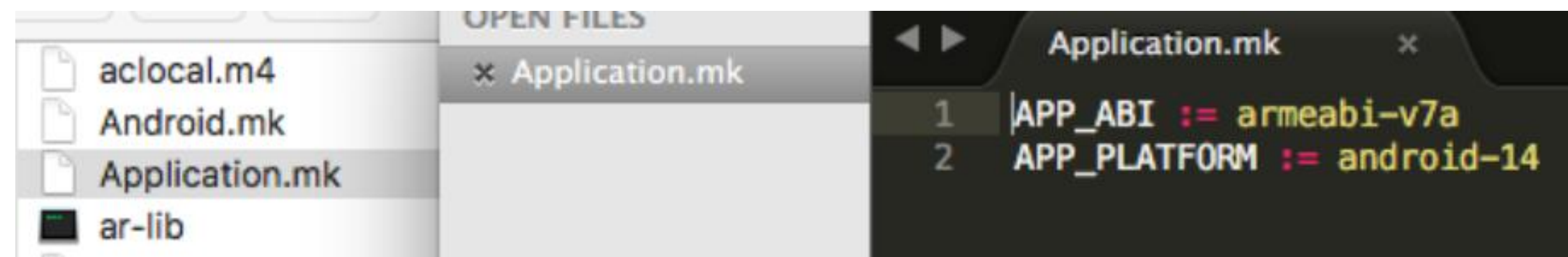
```
1 LOCAL_PATH := $(call my-dir)
2 ENABLE_SHARED := 1
3 WEBP_CFLAGS := -Wall -DANDROID -DHAVE_MALLOC
4 WEBP_CFLAGS += -fvisibility=hidden
5
6 ifeq ($(APP_OPTIM),release)
7     WEBP_CFLAGS += -finline-functions -ffast-math
8     WEBP_CFLAGS += -ffunction-sections -fdata-sections
9     ifeq ($(findstring clang,$(NDK_TOOLCHAIN_PATH))
```

2 添加(jar包调用需要 用到jni接口)

swig/libwebp_java_wrap.c \

```
#####
1 # libwebp
2
3 include $(CLEAR_VARS)
4
5 LOCAL_SRC_FILES := \
6     $(dsp_enc_srcs) \
7     $(enc_srcs) \
8     $(utils_enc_srcs) \
9     swig/libwebp_java_wrap.c \
```

3 创建Application.mk



WebP的劣势

我们都知道播放视频需要下载播放器，播放器的作用是视频文件进行解码。所以每一个播放器都会包含解码器

WebP使用的是视频编码 难道需要**下载**视频编码器

其实是不需要的，很多厂商已经支持了webp的解码，可以说webp就像png一样普遍。

可是？

在早些年代 那个时候还没有webp，
比如Android4.2之前 他们是不支持webp的



不要怂，一起上！
--- 鲁迅

采用libwebp的优势

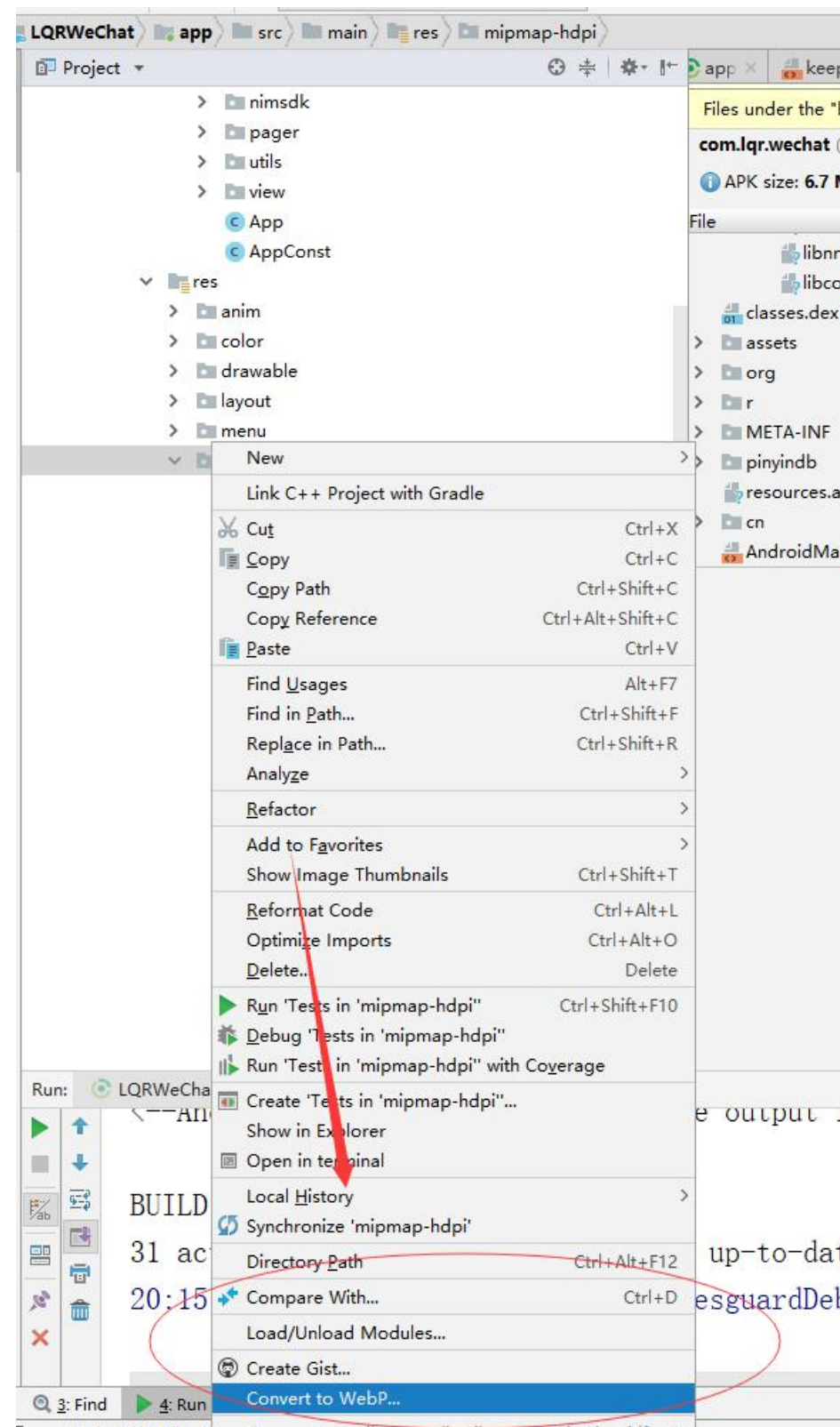
- ◆ 减少webp解码时间 可以超过 jpg 和png解码的时间
- ◆ 降低编码的时间，比原生的编码小号时间更短 30ms
- ◆ 增加兼容性，兼容所有型号的手机，4.3以下也是兼容的 播放器



APK瘦身七大步

第一步：将图片转换成webp

1、点击文件夹 进行全部转换



第二步：去除多语言

1、由于第三方库，如appcompat-v7的引入，库中包含了大量的国际化资源，根据情况通过配置删除。

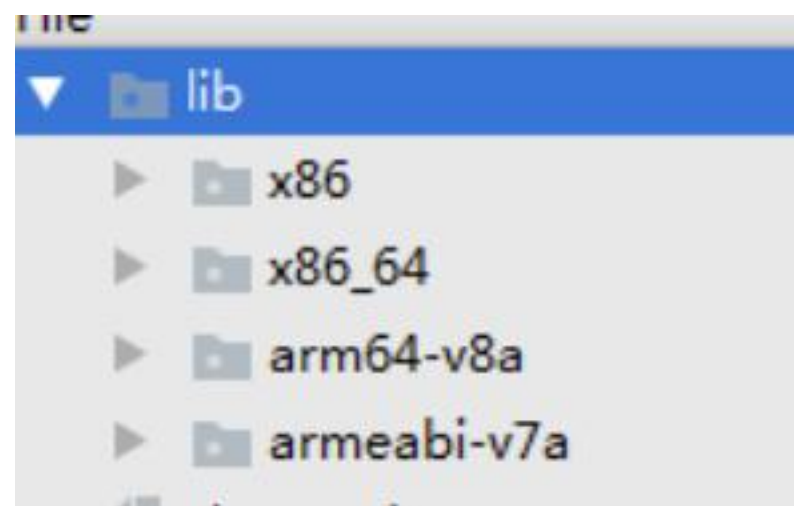
```
defaultConfig {  
    applicationId "com.lqr.wecha  
    minSdkVersion 18  
    targetSdkVersion 26  
    versionCode 1  
    versionName "1.0"  
    testInstrumentationRunner "a  
  
    只保留英文资源  
    resConfigs 'en'|
```

| ID | Name | default |
|------------|---|--------------|
| 0x7f070000 | abc_action_bar_home_description | Navigate ... |
| 0x7f070001 | abc_action_bar_home_description_format | %1\$s, %2... |
| 0x7f070002 | abc_action_bar_home_subtitle_descripti... | %1\$s, %2... |
| 0x7f070003 | abc_action_bar_up_description | Navigate ... |
| 0x7f070004 | abc_action_menu_overflow_description | More opt... |



第三步：只保留armeabi-v7a

如果项目中包含第三方SDK或者自己使用了ndk，
如果不进行配置会打包全cpu架构的动态库进入apk。
对于真机，只需要保留一个armeabi (armeabi-v7a)就可以了。



```
ndk {  
    //设置支持的SO库架构  
    abiFilters "armeabi-v7a"  
}  
javaCompileOptions {
```



APK瘦身

进入源码LayoutInflater源码分析:

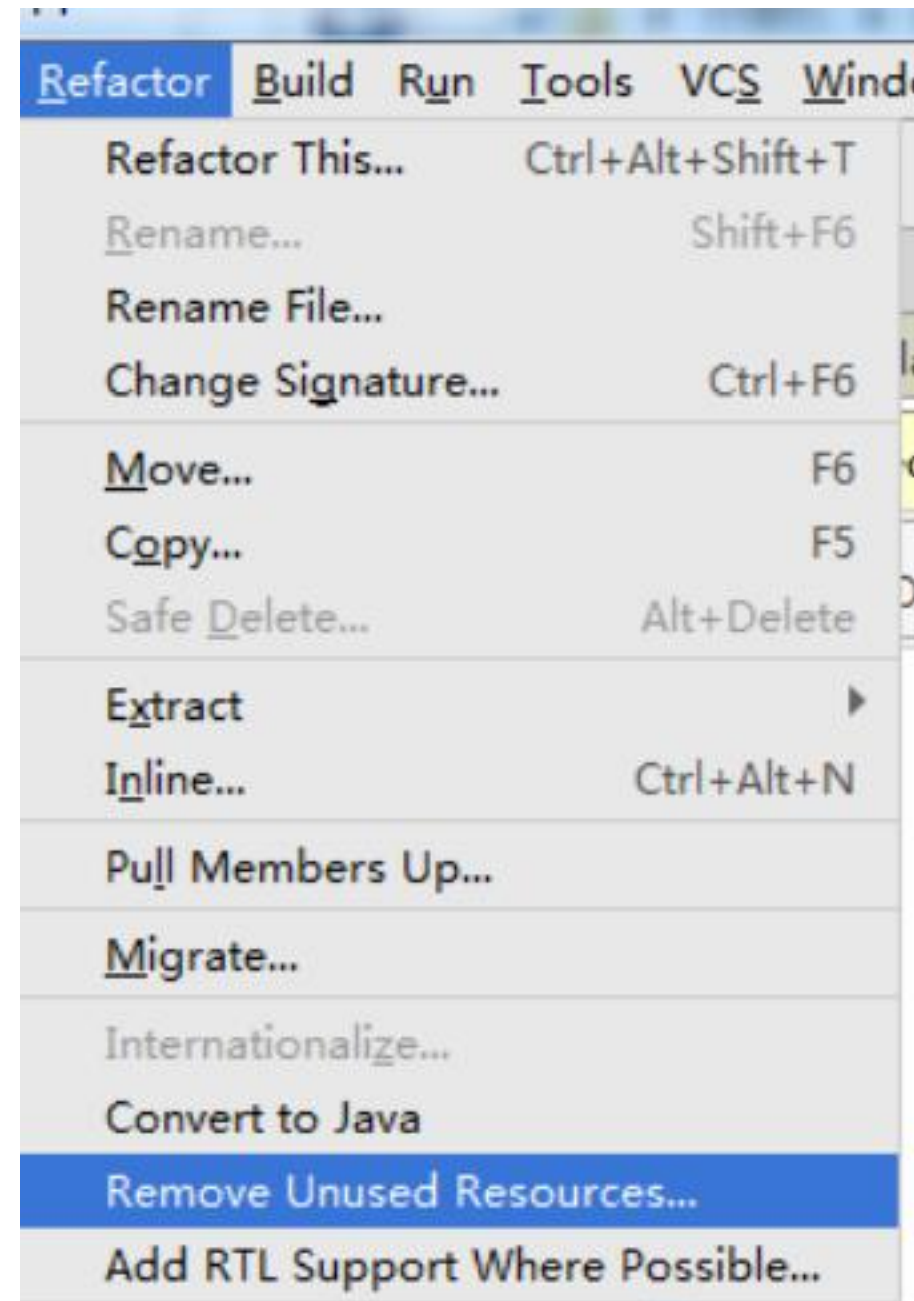
`LayoutInflater.inflate(xml);`

```
void rInflate(XmlPullParser parser, .....) {  
    .....  
    else {  
        final View view = createViewFromTag(parent, name, context, attrs);  
        final ViewGroup viewGroup = (ViewGroup) parent;  
        final ViewGroup.LayoutParams params = viewGroup.generateLayoutParams(attrs);  
        rInflateChildren(parser, view, attrs, true);  
        viewGroup.addView(view, params);  
    }  
}
```



第四步：移除无用资源

1、一键移除, 如果出现使用动态id使用资源会出现问题(不建议)



```
int indetifier =getResources().getIdentifier("img_bubble_receive", "drawable", getPackageName());
getResources().getDrawable(indetifier);
```

动态获取资源id, 未直接使用R. xx. xx , 则这个id代表的资源会被认为没有使用过(类似不能混淆反射类)



第四步：Link检查

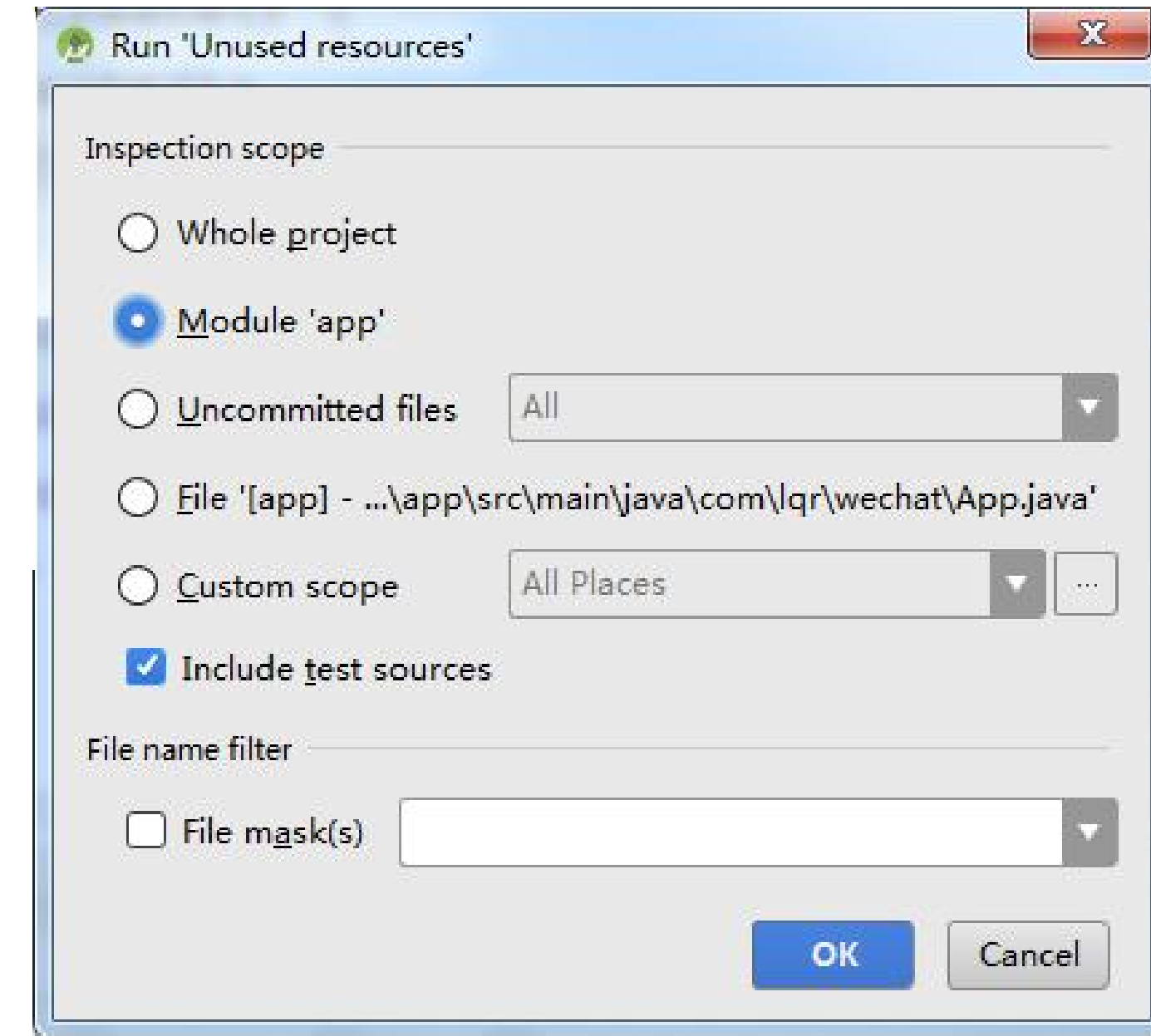
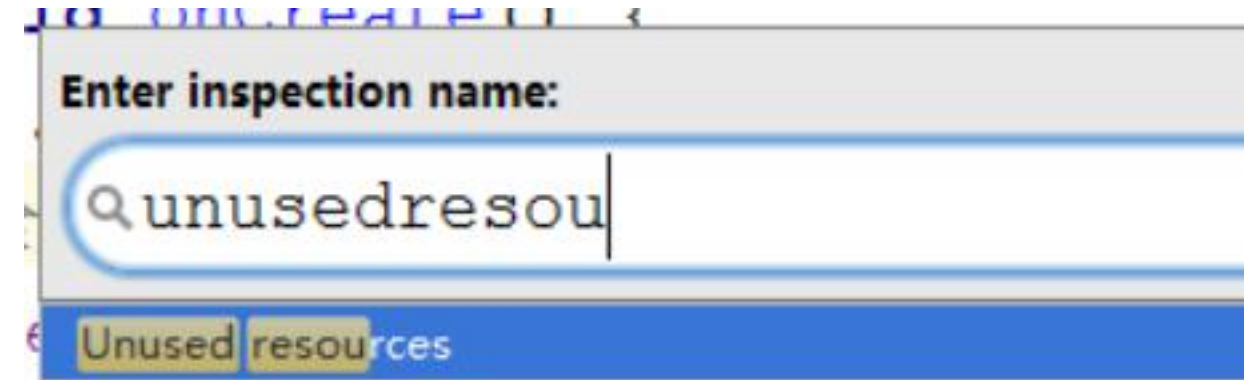
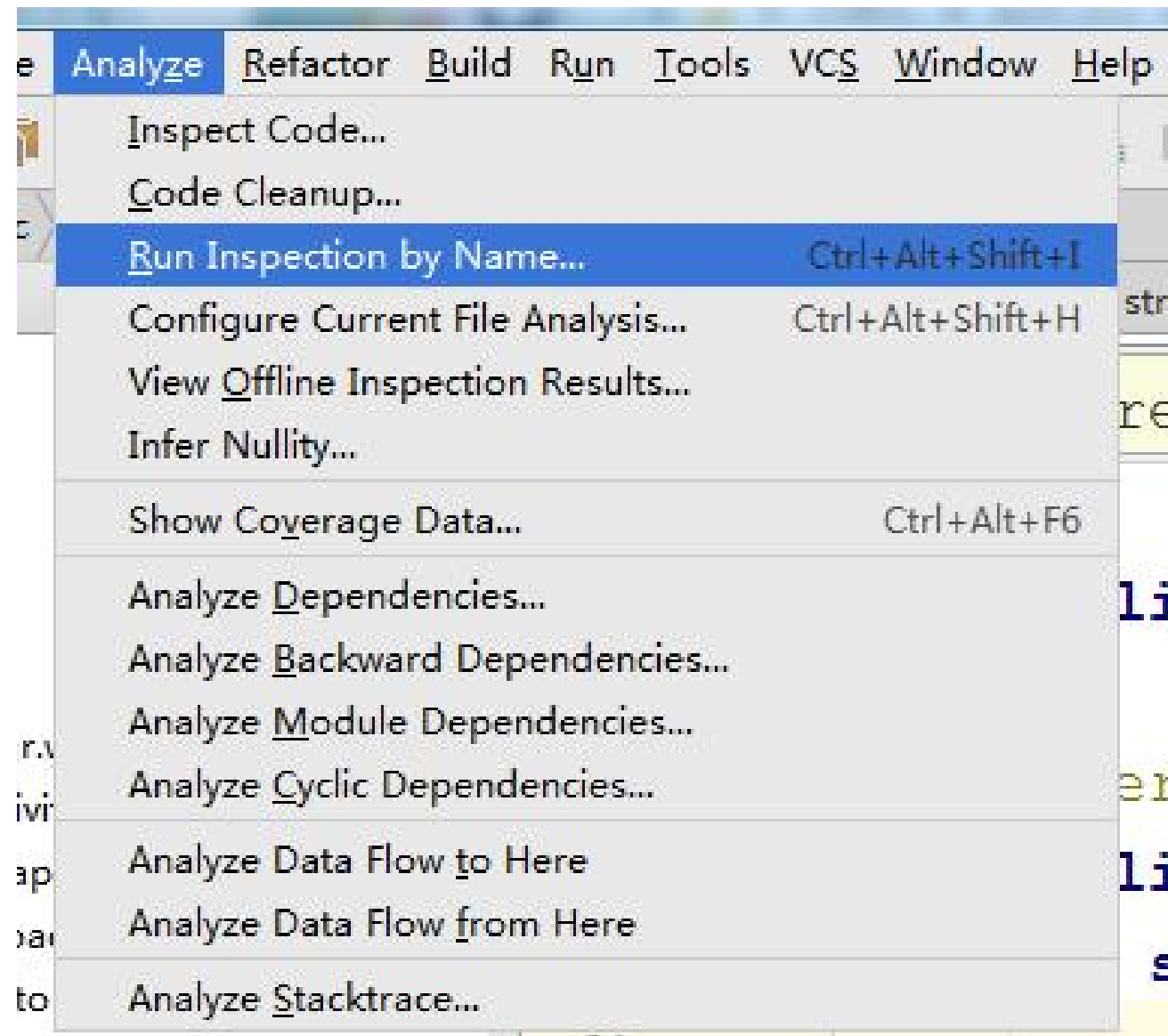
■ 什么是 Lint

Lint 是Android Studio 提供的 代码扫描分析工具，它可以帮助我们发现代码结构/质量问题，同时提供一些解决方案，而且这个过程不需要我们手写测试用例。

代码迭代版本一多，很容易会遗留一些无用的代码、资源文件，我们可以使用 Lint 进行清除。



第四步：移除无用资源(Lint检查)



```
int indetifier =getResources().getIdentifier("img_bubble_receive", "drawable", getPackageName());  
getResources().getDrawable(indetifier);
```

动态获取资源id, 未直接使用R. xx. xx ， 则这个id代表的资源会被认为没有使用过(类似不能混淆反射类)



第五步：开启混淆

■ 什么是 混淆

Android代码混淆，又称Android混淆，是一种Android APP保护技术，用于保护APP不被破解和逆向分析。

ProGuard的三大作用

压缩

移除未被使用的类、属性、方法等，并且会在优化动作执行之后再次执行（因为优化后可能会再次暴露一些未被使用的类和成员。

优化

优化字节码，并删除未使用的结构。

混淆

将类名、属性名、方法名混淆为难以读懂的字母



第六步：开启删除无用资源（与Lint不同）

■ `shrinkResources = true`

`shrinkResources` 用来开启压缩无用资源，也就是没有被引用的文件（经过实测是 `drawable`, `layout`，实际并不是彻底删除，而是保留文件名，但是没有内容，等等），但是因为需要知道是否被引用所以需要配合 `minifyEnabled` 使用，只有当两者都为 `true` 的时候才会起到真正的删除无效代码和无引用资源的目的

与去除无用资源不同的是，比如 某个 `java` 类没有用到，被混淆时删除了，而该类引入了 `layout` 资源。此时会将这个资源也压缩掉



第六步：开启删除无用资源（严格模式与普通模式）

■ 如何开启严格模式

```
<?xml version="1.0" encoding="utf-8"?>  
<resources xmlns:tools="http://schemas.android.com/tools"  
    tools:shrinkMode="strict" />
```

■ 如何手动保留xml

```
<?xml version="1.0" encoding="utf-8"?>  
<resources xmlns:tools="http://schemas.android.com/tools"  
    tools:keep="@layout/base_*"   
    tools:discard="@layout/unused2" />
```



第七步：AndResGuard微信资源压缩方案

■ 什么是 AndResGuard

AndResGuard是一个缩小APK大小的工具，它的原理类似Java Proguard，但是只针对资源。它会将原本冗长的资源路径变短，例如将res/drawable/wechat变为r/d/a。

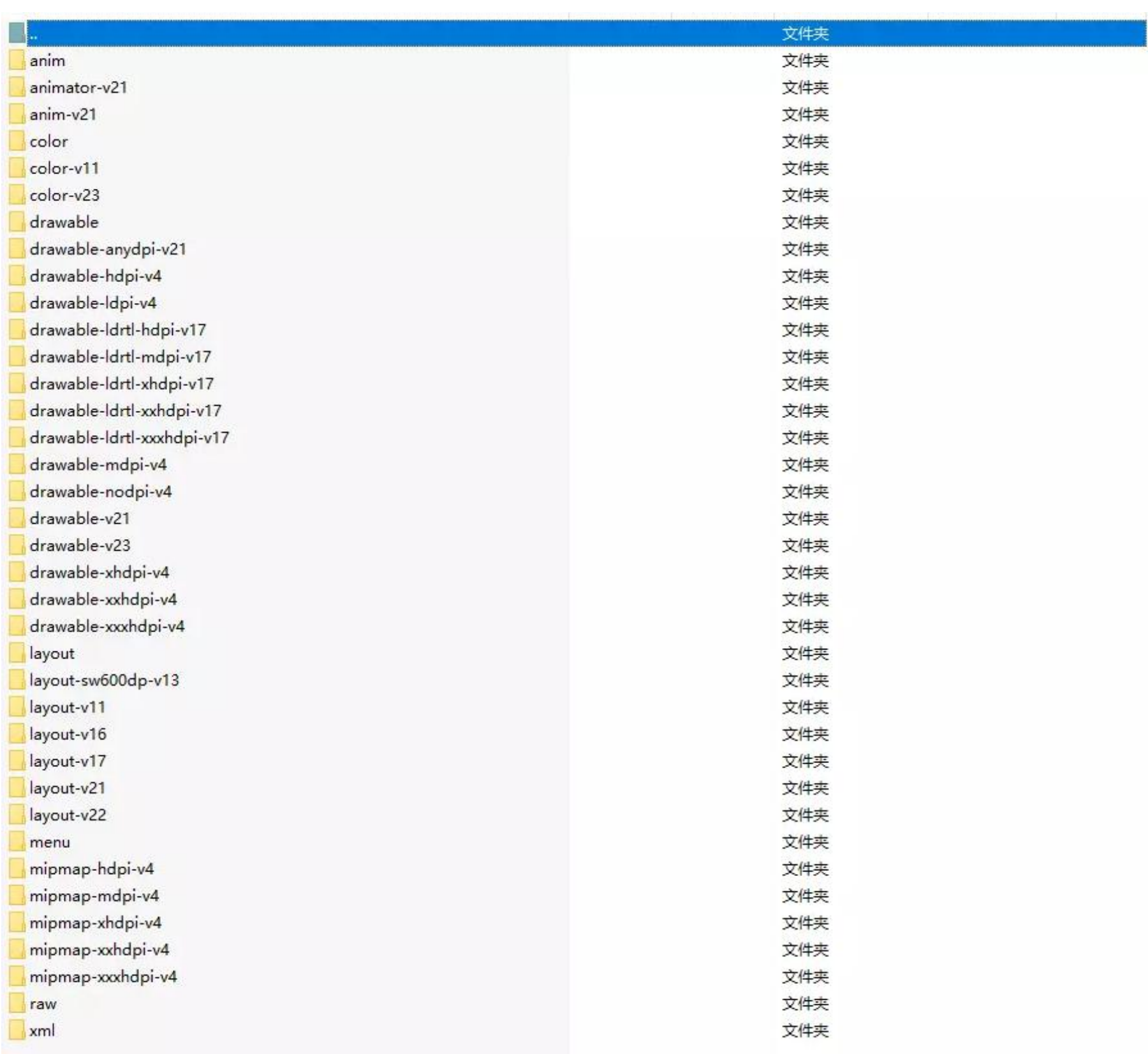
■ 为什么 使用AndResGuard

在以往的开发中，我们通常只混淆了代码，资源文件却暴露在他人面前，res文件夹下所有文件名的可读性过强

微信的开源库AndResGuard 7zip压缩正好解决这种问题，对资源进行混淆，保护res资源文件的可读性，同时，可以减少APP的大小



第七步： AndResGuard微信资源压缩方案



此时，我们根本无法知道哪个文件夹是存放布局文件的，哪些文件夹是存放图片的，即使找到存放xml文件的文件夹，我们也难以知道这些xml是干嘛用的



谢谢观看