## 安装包性能优化

一个字 删

1 图片压缩·

图片：apk里面的资源 压缩图片（美工）

svg图片（其实就是一个文件 类是xml形式）一些图片的描述（牺牲CPU的计算能力，节省空间）

使用原则：简单的图标

webp：可以在bmp.compress(Bitmap.CompressFormat.WEBP,quality,baos) 压缩成webp格式图片

google现在非常提倡 VPN8派生而来 保存图片比较少 支持有损压缩，无损压缩，比PNG文件小45% FackBook在用，腾讯、淘宝。即使png进行其他压缩荏苒可以 工具：isparta.github.io

缺点：加载相比于png要慢很多，但是配置比较高。

2资源动态加载：比如emoji表情、换肤

动态下载的资源。一些模块的插件化动态添加

3Lint工具 建议优化的点（有些建议不靠谱）

1. 检测没有用的布局 删除
2. 未使用到的资源 比如图片
3. 建议string.xml有一些没有用到的字符

4极限压缩

7zzip工具的使用

5.progurad 混淆

让apk变小

1. 可以删除注释和不用的代码
2. 将java文件名改成短命 a.java、b.java
3. 方法名等 CommonUtil.getDisplayMetrix()->a.a()

常规的安装包的优化之外继续压缩 资源文件在压缩

系统编译完文件以后：

映射关系：res/drawable/ic\_launcher.png –》0x7f020000

在做混淆：要事先将res/drawable/ic)launcher.png图片改成a.png

drawable String layout 文件名字 比如 R.string.description—>R.zstring.a

res/drawbale/ic\_launcher.png图片改成a.png

还可以更夸张：res/drawable🡪r/d

res/value

res/drawable/ic\_launcher.png图片改成r/d/a.png

读取resource.arsc二进制文件，然后修改某一段一段的字节

有一段叫做res/drawable/ic\_launcher.png 在自己当中的第800位-810位

将这段改成R/d/a.png 的字节码