# Android 面试整理

## 一.View的绘制

1）[setContentView()的源码分析](http://www.jianshu.com/p/1c1f871cca1f)  
2）[SnackBar的源码分析](http://www.jianshu.com/p/989295fe43f9)  
3）[利用decorView机制实现底部弹出框](http://www.jianshu.com/p/9225d30615a4)  
4）[换肤框架(一)之Support v7库解析](http://www.jianshu.com/p/c5c752f26b12)  
5）[小红书欢迎页的视差效果实现](http://www.jianshu.com/p/7c6660da0f10)  
6）[换肤框架(二)之换肤框架实现解析](http://www.jianshu.com/p/7fa79553bf13)  
7）[View和ViewGroup的绘制原理源码分析](http://www.jianshu.com/p/48601578c9a8)  
8）[自定义ViewGroup(一)之卡牌](http://www.jianshu.com/p/9759a1666494)

## 二.事件的分发

1）[从场景到源码分析事件分发](http://www.jianshu.com/p/b85d2d85ff9d)  
2）[事件分发实例之侧滑菜单](http://www.jianshu.com/p/76f616011e72)  
3）自定义behavior的源码分析  
4）自定义behavior的例子分析

## 三.动画框架

1）属性动画的高级使用方法  
2）模仿京东的弹性加载例子分析  
3）共享元素转场动画

## 四.Handler机制[will be written by [Ticoo](http://www.jianshu.com/u/17b184013928)]

1）[Handler,MessageQueue,与Looper三者关系分析](http://www.jianshu.com/p/0b0fcae16c59)  
2）HandlerThread源码分析  
3）IntentService源码分析  
4）Handler常见应用场景和常见问题分析

## 五.Binder机制

1）从framework分析AIDL生成文件

## 六.流行框架源码分析

1）AsynTask源码分析  
2）Butterknife源码分析  
3）编译期注解的使用例子分析  
4）AOP在代码中的使用分析  
5）设计模式在安卓中的使用

6）Rxjava2源码分析

## 七.性能优化[will be written by [无心追求](http://www.jianshu.com/u/61b0362f5329)]

1）冷启动优化  
  1.1.APP启动分析工具（TraceView，Systrace）  
  1.2.布局优化（扁平化，Merge的使用，ViewStub的使用）  
  1.3.各种布局的性能对比（LinerLayout，RelativeLayout，ConstraintLayout）  
  1.4.APP启动速度优化实例解析  
  1.5.Activity启动优化，启动流程解析  
2）内存优化  
  2.1.内存泄露分析工具（MAT，LeakCanary，AndroidStudio Monitor）  
  2.2.LeakCanary原理分析  
  2.3.Android三级缓存策略，LruCache源码分析，DiskLruCache源码分析  
  2.4.[内存泄漏的一些坑](http://www.jianshu.com/p/3184b0a8fc4b)  
  2.5.[MAT内存泄漏分析(一)](http://www.jianshu.com/p/5f261b470a19)  
 2.6.[MAT内存泄漏分析(二)](http://www.jianshu.com/p/95b72f7c9969)  
3）网络优化  
4）图片压缩性能优化  
5）数据传输性能优化  
6）安装包性能优化  
7）渲染优化  
  7.1.UI卡顿原理分析  
  7.2.UI卡顿检测分析  
  7.3.[BlockCanary原理分析](http://www.jianshu.com/p/768e21ce76b5)  
  7.4.卡顿优化实例解析  
8）电量优化  
  8.1.APP耗电分析方法与工具（第三方，Google官方Battery Historian）  
  8.2.Android Doze源码分析  
  8.3.Android Alarm与JobSchedule机制分析  
  8.4.耗电优化实例解析

## 八.网络编程[will be written by [无心追求](http://www.jianshu.com/u/61b0362f5329)]

1.TCP，UDP，Http协议  
2.Okhttp源码架构分析  
3.网络传输数据加密策略（对称加密AES，DES，非对称加密RSA）  
4.二进制协议（mqtt协议），协议编解码（tlv编解码）  
5.Java Socket编程（IO，NIO）  
6.Java多线程编程（队列，同步锁，线程池）

## 九.Android推送SDK[will be written by [无心追求](http://www.jianshu.com/u/61b0362f5329)]

1.SDK架构设计  
2.[心跳策略](http://www.jianshu.com/p/517166082bd8)  
3.重连策略  
4.tcpdump抓包分析  
5.进程保活  
6.多进程通信  
7.TCP连接多路复用  
8.Android系统休眠  
9.[TCP网络问题](http://www.jianshu.com/p/75d2e2969b3b)  
10.[DHCP租约到期自动续租问题导致TCP连接异常](http://www.jianshu.com/p/fabff1b9cf6a)

## 十.FrameWork层源码的分析

1）FrameWork层源码的分析  
2）InstantRun源码分析  
3）热修复或者插件化框架源码分析

Android动态加载dex技术初步了解 <http://www.jianshu.com/p/e835d1d280c4>

Android支持动态加载的方式有两种：DexClassLoader和PathCLassLoder

DexClassLoader可加载jar/apk/dex,且支持从SD卡加载；

PathClassLoader只支持加载已经安装到Android系统中的apk文件

## 十一.android发布及软件相关

adb shell 获取package信息

修改android项目报名·：（层级相同）把module里的build.gradle文件里application对应包名换成你即将要改的。把androidManifest.xml包名换成你将要换的

## 十二.android系统组件

Android使用Fragment的懒加载：但是我们希望在actviity可见的时候,只显示当前的fragment,这个时候呢,第一个可见的fragment数据加载了,但是其他fragment此时是不可见的,那么就不是很希望将其他几个fragment的数据也加载了;此时我们就希望用到懒加载,当界面可见的时候fragment才会加载数据;主要是通过setUserVisibleHint

Android广播那些事：

IV. 安全性与最佳实践

如果你的广播不需要发送给本应用以外的组件，使用LocalBroadcastManager来发送广播，这样安全性和效率都比较高

静态注册有可能造成大量的App启动，这将会影响系统的性能，所以尽量使用动态注册来替代静态注册。这一点Android系统就做出了很好的示范，比如 CONNECTIVITY\_ACTION 这个广播只发送给动态注册的广播接收者。

不在广播的Intent中包含敏感的信息，因为只要注册了这个广播就能读取到这些信息。你可以通过以下3中方式来获得一定的安全性。

通过使用权限来发送广播，这样只有声明了该权限的应用才能收到广播。但是你很难确保你的权限不被泄漏。

Android4.0及以上版本，在发送广播的时候可以通过setPackage来指定package（可以指定多个），这样只有匹配的package能接受到。

使用LocalBroadcastManager来发送本地广播。

当你注册了一个广播，意味着任何App都可以给你发送广播，以下有三点可以限制接收者：

注册的时候增加权限。

在AndroidManifest.xml注册receivers时，将android:exported属性设为false。

使用LocalBroadcastManager来注册。

action的命名空间是全局的，这意味着action有可能会与其他App冲突，所以最好是有一个自己的命名空间。

因为广播接收者是运行在主线程，它应该快速地被执行并且return，所以不要在onReveive方法中做比较耗时的操作。

不要在广播接收者中启动activitys，这违背了用户的使用习惯，特别是不止一个接收者时。这种情况下可以考虑展示一个notification来替代。

## 十三.android高级动画

Android 制作逐渐显示动画(描边动画、矢量动画VectorDrawble)：<http://www.jianshu.com/p/700ccbd3796c> 传统的andorid提供的动画只有平移、缩放、显示等效果，那么怎么才能逐渐绘制动画？答案是实用矢量动画

## 十三.android安全以及加密

浅谈android数据存储加密 <http://www.jianshu.com/p/4f4a927339a9>

按可逆性：加密可分为可逆算法和不可逆算法

按对称性：加密可分为对称算法和非对称算法

Base64编码算法 （可逆）-MD5加密 （不可逆）（还有一个sha1值，可能做过支付宝的会比较熟悉）

-Des加密 （对称，可逆）-Aes加密 （对称，可逆）

-Rsa加密（非对称，可逆）

Android代码混淆干货：<http://www.jianshu.com/p/09688a5852bf>

* 压缩: 移除无效的类、属性、方法等
* 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，比如a,b,c

### 1.不能被混淆

* 在AndroidManifest中配置的类，比如四大组件
* JNI调用的方法
* 反射用到的类
* WebView中JavaScript调用的方法
* Layout文件引用到的自定义View
* 一些引入的第三方库（一般都会有混淆说明的）

## 十四，枚举

java中的Enum是包含固定常量集的数据类型.当我们需要预定义一组代表某种数据的值时一般都会使用枚举, 而当要保证类型安全时, 我们经常会使用Enum。

比如, 当我们要保证常量使用正常时, 我们经常使用Enum在编译时校验确保类型安全

Enum中的每个值都是一个对象,每个声明都将使用一些运行时内存来简单引用该对象,所以Enum相较于static int会**占用更多的内存**.

另外添加单个Enum将**增加最终DEX文件的大小**（是static int的**13**倍）

<http://www.jianshu.com/p/8ce7c2eeef0b> 使用枚举正确姿势