**SDK开发实战**

**一、SDK概述**

SDK是Software Development Kit的缩写，译为"软件开发工具包"，通常是为辅助开发某类软件而编写的特定软件包或框架集合等，SDK一般包含相关文档、范例和工具。

SDK可以分为系统SDK和应用SDK。

（1）系统SDK是为特定的软件包，软件框架，硬件平台，操作系统等建立应用时所使 用的开发工具集合。

（2）应用SDK则是基于系统SDK开发的独立于具体业务而具有特定功能的集合 （jar\so\aar）。

在进行Android 应用开发时，我们使用Google提供的系统SDK(Android SDK)，而我们经常使用的支付SDK、分享SDK、地图SDK、友盟SDK、极光SDK则是基于系统SDK开发的。

1. **开发SDK**
2. .jar开发
3. .aar开发

打包aar时会出现SDK引入的第三方问题，如OKhttp等。

1. .so开发

一般情况下，加密问题多在这里处理。

1. **混淆问题**
2. **第三方引入SDK的步骤：**

第一种方式：使用gradle脚本直接导入（比较旧）

1、在项目build.gradle文件内android{}标签中加入如下代码申明本地仓库：

|  |
| --- |
| repositories {  flatDir {  dirs 'libs'  }  }  } |

2、然后将aar包放入libs目录（没有就在app目录下创建一个），随后加入dependencies ：

|  |
| --- |
| dependencies {  compile fileTree(dir: 'libs', include: ['\*.jar'])  ...  compile(name: 'xxxxxx-xxxx', ext: 'aar') //name为aar文件全名  } |

第二种方式：implementation

|  |
| --- |
| implementation fileTree(dir: 'libs', include: ['\*.jar','\*.aar'])  或者  implementation files('libs/xxx.jar')  implementation files('libs/xxx.aar') |

1. **继承Application问题**

如果aar中定义了Application，theme，icon，那么编译时就要抛出如下异常：

编译错误AndroidManifest合并异常

由于项目与aar都定义了自己的Application，编译器不知道用谁的，这就产生了上图的异常，按照提示，在Application标签下添加如下代码：

|  |
| --- |
| tools:replace="android:name" |

如果使用了上述的解决报错的方法，会出现被取代方的Application对象丢失而出现未知的问题。如果让应用在使用SDk时，强制继承SDK中的Application，会出现人为的矛盾。所以

通过接口实现 + 反射 的方式来创建代理Application对象，曲线实现Application的多继承，由。。。。