

1、Android中单个 dex 文件所能包含的最大方法数为 65536

2、dexopt 是什么？

1. 一个程序：安装应用时，系统会通过 dexopt 来优化 dex 文件。
2. dexopt 采用一个固定大小的缓冲区来存储应用所有方法的信息(LinearAlloc-缓冲区)
3. 低版本中 dexopt的缓冲区 比较小，就会出现即使方法数量不足 65536 ，仍然会报错。

3、如何解决方法数越界问题？

1. 删除无用的代码和第三方库(局限)
2. 插件化(重量级且兼容性问题较多)
3. Google的 multidex 方案，简单方便。

4、multidex简介

1. Android 5.0以前需要引用 android-support-multidex.jar 包-位于 SDK的目录extras/android/support/multidex/library/libs 。
2. Android 5.0开始默认支持 multidex ，可以从 apk 中加载多个 dex文件

5、multidex的使用

```
//build.gradle
apply plugin: 'com.android.application'

android {
    defaultConfig {
        //1. 重点一
        multiDexEnabled true
    }
    ...
}
dependencies {
    //2. 重点二
    compile 'com.android.support:multidex:1.0.0'
}
```

使用方案一(指定application):

```
<application
    ...
    android:name="android.support.multidex.MultiDexApplication">
    ...
</application>
```

使用方案二(继承MultiDexApplication):

```
public class BaseApplication extends MultiDexApplication{...}
```

使用方案三:

```
public class BaseApplication extends Application{  
    //1. attachBaseContext会在onCreate()前执行  
    protected void attachBaseContext(Context base) {  
        super.attachBaseContext(base);  
        //2. 安装功能  
        MultiDex.install(this);  
    }  
}
```

6、Multidex的缺点

1. 应用启动速度会降低---因为需要加载额外的dex文件。甚至会出现ANR现象，如果其他的dex文件较大的时候，因此要避免生成较大的dex文件。
2. Dalvik linearAlloc的Bug可能会导致multidex的应用无法在部分Android 4.0以前的手机上运行(极少遇到)