**Android Launch启动过程**

参考：https://blog.csdn.net/Luoshengyang/article/details/6767736

1. SystemServer组件是由Zygote进程负责启动的，启动时会调用SystemServer.main()，创建一个ServerThread线程；
2. ServerThread.run()中启动PackageManagerService和ActivityManagerService等很多服务，并执行他们的main()方法；
3. ActivityManagerService.main()创建了一个ActivityManagerService实例；
4. PackageManagerService.main()创建了一个PackageManagerService服务实例，并添加到ServiceManager中；
5. PackageManagerService类的构造函数中开始执行安装应用程序的过程，调用scanDirLI()来扫描移动设备上/system/framework、/system/app、/vendor/app、/data/app和/data/app-private目录下的apk文件；通过PackageManagerService.scanPackageLI()对以Apk作为后缀名的文件进行解析和安装，并将得到的应用程序信息（package、provider、service、receiver和activity等信息）保存在PackageManagerService中；
6. PackageParser.parsePackage()对AndroidManifest.xml文件中的各个标签进行解析；
7. ActivityManagerService.setSystemProcess()将ActivityManagerService实例托管到ServiceManager中;
8. 在ActivityManagerService.systemReady()中调用ActivityStack.resumeTop- ActivityLocked()；再调用ActivityManagerService.startHomeActivityLocked()；
9. ActivityManagerService.startHomeActivityLocked()首先创建一个CATEGORY\_HOME类型的Intent，然后通过Intent.resolveActivityInfo函数向PackageManagerService查询Category类型为HOME的Activity（这里我们假设只有系统自带的Launcher应用程序注册了HOME类型的Activity)，便能找到com.android.launcher2.Launcher，并通过ActivityStack.startActivityLocked()把Launcher启动起来；
10. Launcher.onCreate()调用LauncherModel.startLoader()，已发送消息的形式，让 LoaderTask.run()来处理；
11. LoaderTask.run()中调用LoaderTask.loadAllAppsByBatch()，通过PackageManagerService.queryIntentActivities()找到所有Action类型为Intent.ACTION\_MAIN，并且Category类型为Intent.CATEGORY\_LAUNCHER的Activity；并通过消息的形式，进入到Launcher.bindAllApplications()，然后通过调用AllApps2D.setApps()为PackageManagerService解析得到的所有应用程序创建一个ApplicationInfo实例了，有了这些ApplicationInfo实例之后，就可以在桌面上展示系统中所有的应用程序了。