# 插件化实现过程

[应用文体综述型文章](https://www.zhihu.com/question/34424881)

王桉树

海南六六六科技有限公司

## 关键词

插件 热部署 热加载 DL框架 [Dynamic-load-apk](https://github.com/singwhatiwanna/dynamic-load-apk)

## 前言

“我去彩票站”运用到的插件化开发过程，是根据一定规范编写出代码，打包成文件，将这个打包的文件称作插件。通过部署插件到管理平台，已安装的应用可以在启动的时候，检查、下载、更新插件。实现特定版本应用能在有wifi的情况下，更新到最新的插件，并运行最新插件里面的代码。既可以使用新功能，也可以修复应用存在的问题。

## 简介

项目采用 2014年底Dynamic-load-apk 开源的插件化实现方案。经过 2015年上半年的改造及适配，正式运用到“我去彩票站”项目中。

## 插件文件及其打包

“我去彩票站”插件化开发中，项目结构分为 Host模块和插件模块（现阶段只用到 Main插件模块）。Host模块用于放置共用的代码。主要包括Android 官方支持库（v4，v7，Design）资源、加载 H5的相关文件和更新插件的文件。插件模块主要是放置及编写业务代码。用户中心、投注、首页的功能都是在插件模块下完成的，大部分需求都是在插件模块下开发。

APK File

x86\_64

/layout

/xml

classes.dex

resources.arsc

/META\_INF

/lib

MANIFEST.MF

CERT.RSA

CERT.SF

armeabi

armeabi-v7a

arm64-v8a

x86\_64

/assets

/drawable

/res

AndroidManifest.xml

图1. APK文件包含的资源

插件中主要包含 res文件夹、Androidmanifest文件、classes.dex文件和resource.arsc文件。

插件打包包括内置插件包和对外发布插件包。

* 内置插件包通过 Gradle脚本命令 assembleRelease 执行，并把打包的插件拷贝到 Host 模块目录下 assets的 plugins文件夹下，并以插件的 ID，命名文件名。内置插件包将随着应用发布到市场。
* 对外发布插件包通过 Gradle脚本命令 assembleProduct 执行，生成的文件放在插件模块下的bin文件夹下，为后续的发布提供插件文件。

插件打包管理系统是负责对外发布插件的打包、管理系统。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 应用名 | 项目名 | 包名 | 主版本 | appId | 插件数 | 创建时间 | 发布时间 | 修改时间 | 下载主机地址 | 发布状态 | 查看插件信息 | 发布操作 |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 插件名 | 插件标识 | 创建时间 | 当前最新版本 | 当前默认版本 | 插件上一个版本 | 微核心版本 | 最低os | 最低微核心 | 插件类型 | 加载顺序 | 查看插件包 信息 | 打包操作 |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 插件路径 | 是否可更新 | 包文件md5值 | 插件包版本 | 创建时间 | 发布时间 | 修改时间 | 发布状态 | svn版本 | 发布策略 | 插件大小(M) | uriConfigPath | uriConfigMd5 | 是否显示 | 更新说明 | 查看 补丁信息 | 添加 渠道插件包 配置 | 插件发布 操作 |

表1.插件打包管理平台系统

## 应用是如何加载代码和资源？

1. 启动运用时，加载插件。
2. DexClassLoader解析classes.dex，AssetManager 解析 resources.arsc。
3. DLProxyFragmentActivity 执行插件的DLBasePluginFragmentActivity的代码

HostEventBus

PluginLoadEvent

PluginPkgDownEvent

图2. 插件加载



图3. Activity生命周期

应用运行时，系统主线程会执行 Activity的生命周期，这里的 Activity已经被封装为 DLProxyFragmentActiviy，主线程调用的 Activity将用 DLProxyFragmentActivity表述。用 DLProxyFragmentActivity 封装了调用类的 DexClassLoader 及调用资源的 AssetManager 及 Resource 对象。

DLProxyFragmentActivity 对象，包含了具体实现的 DLBasePluginFragmentActivity对象。DLBasePluginFragmentActivity 包含了 Activity生命周期的所有方法。DLProxyFragmentActivity 负责执行生命周期，调用DLBasePluginFragmentActivity 相对应的生命周期方法，执行业务代码。

classes.dex

resources.arsc

插件模块开发需要遵守一些开发规则：

1. Activity 需要继承 DLBasePluginFragmentActivity；
2. 跳转页面（Activity）使用封装的方法ActivityTool#forwardTarget
3. 区分 DLBasePluginFragmentActivity 和 DLProxyFragmentActivity 的使用场景

## **插件打包发布管理平台**

## **插件化开发的研究现状与发展前景**

插件化开发目前主要在国内有应用。从 2012年开始，有多个开源的实现（表1）。

表2. 各个插件化项目介绍

|  |  |  |
| --- | --- | --- |
| 发布时间 | 项目名 | 简介 |
| 2012年7月 | AndroidDynamicLoader | 屠毅敏，大众点评。用Fragment来实现，addAssetPath来读取插件中的资源。 |
| 2013年 | 23Code | 自定义控件动态下载。并不是一个开源的项目 |
| 2014年初 | [Altas](http://v.youku.com/v_show/id_XNTMzMjYzMzM2.html) | [阿里。伯奎](http://club.alibabatech.org/resource_detail.htm?topicId=84) |
| 2014年底 | [Dynamic-load-apk](https://github.com/singwhatiwanna/dynamic-load-apk) | [任玉刚。静态代理Activity](http://blog.csdn.net/singwhatiwanna/article/details/40283117) |
| 2015年4月 | [OpenAltas/ACDD](https://github.com/bunnyblue/ACDD) | 淘宝，扩展AAPT来解决插件与宿主的资源id冲突的问题。 |
| 2015年8月 | [DroidPlugin（里程碑技术）](https://github.com/DroidPluginTeam/DroidPlugin) | [张勇 技术实现复杂，改写很多Android系统的底层代码](http://www.infoq.com/cn/presentations/the-realization-principle-and-application-of-droidplugin?utm_campaign=rightbar_v2&utm_source=infoq&utm_medium=presentations_link&utm_content=link_text) |
| 2015年9月 | [AndFix](https://github.com/alibaba/AndFix) | 阿里。黎三平 |
| 2015年10月 | [Nuwa（女娲）](https://github.com/jasonross/Nuwa) | 贾吉鑫。大众点评 ，MultiDex的思路，根据QQ空间理论方案，主要解决的是热修复问题，不维护了 |
| 2015年11月 | [DynamicAPK](https://github.com/CtripMobile/DynamicAPK) | [携程](http://www.infoq.com/cn/articles/ctrip-android-dynamic-loading) |
| 2015年底 | [small框架](https://github.com/cayden/MySmall) | 福建的林广亮，通过脚本方式解决资源冲突问题 |
| 2016年 | VirtualApp | 罗迪 |
| 2016年 | DroidPlugin | 360 ，360手机助手实现的一种插件化框架，可直接运行第三方独立apk，不修改或安装apk. |
| 2016年 | [Tinker](https://github.com/Tencent/tinker) | 微信，热修复 |
| 2017年6月 | [VirtualAPK](https://github.com/didi/VirtualAPK) | 滴滴开源插件框架 ， 支持所有的android特性，四大组件方面 |
| 2017年7月 | [RePlugin](https://github.com/Qihoo360/RePlugin) | 360手机卫士 ，完整稳定全面的插件化方案，提出“全面插件化” |

但由于插件化开发使用的是 Android 非公开的 SDK API，并且Google已经开始这些方法进行管控。在 Android P的版本，非公开的 SDK API 大部分列为灰名单，将在 debug模式下提示开发人员修改。现有在插件化及热加载都影响到，并且暂无提出解决方法。

参考文献

【略】

注解

1. Addon或Extension 遵循一定规范的应用程序接口编写出来的程序，在原有系统增加新功能。
2. 软件模块（Module）是一套一致而互相有紧密关连的软件组织。它分别包含了程序和数据结构两部分。