主题：Lipland在360搜索中的实践

团队：360搜索客户端
讲师：许澎湃

个人介绍

姓名：许澎湃

职位：360搜索Android架构师

主要负责360搜索App的架构设计、插件框架研发、技术难点攻关、反作弊/DNS防劫持、性能调优等。

课程提纲/内容

- Lipland介绍
- 与同类项目对比
- 基本架构
- 四大组件实现原理
- 在360搜索中的实践
- Lipland为360搜索做了什么

Lipland 介绍

Lipland 介绍

Lipland为近期刚在360官方github上开源的一款Android轻量级插件框架，为A lightweight plug-in framework for android的简称，目前主要在360搜索、360语音助手、360地图等app中应用。

关于名称

Lipland由以下三个元素组成：

- **li** - lightweight
- **pl** - plugin
- **and** - android

为什么还要开发一个

- 历史原因

2014年，360搜索就有迫切的插件化需求，那时DroidPlugin、RePlugin还未开放。

- 侧重点和目标

轻量级，易维护，最简实现

瘦身减流

插件本身独立

聚焦核心需求

什么是核心需求？

- 仅支持android 4.0.0及以上的版本。
- 轻便易用，插件完全免修改。
- 支持四大组件及其基本特性。
- 支持共享宿主的jar包、so库。
- 与主进程隔离，独立插件进程。
- hook少量的几个关键函数。
- 完整的升级/更新管理。

什么是核心需求？

为了降低复杂度、便于统一管理，有了以下特点：

- 放弃android 4.0.0以下。
- 没有多进程、没有插件隔离。
- 没有代码隔离、没有权限隔离。
- 插件共用进程、共用内存、共用代码。

与同类框架比较

\	DynamicAPK	DroidPlugin	RePlugin	Lipland
插件免修改	支持	支持	不支持 (编译期修改)	支持
共享代码	不支持	不支持	支持jar共享	支持jar、so共享
四大组件	部分支持	支持	支持	支持
特点	改造aapt实现，需依赖自身的Bundle机制。	深度hook，全面接管，开发插件无任何依赖	极少hook，只有一处。开发插件需依赖gradle插件	轻量级，少量hook，体积小，开发插件无任何依赖
接入成本	中	无	较低 (编译期)	无
进程	单进程	多进程	单进程&进程坑位	单进程
版本支持	Android 2.3及以上	Android 2.3及以上	Android 2.3及以上	Android 4.0及以上
适用场景	简单插件	应用免安装	全面插件化 支持UI插件化	瘦身减流 独立模块插件化
经典案例	略	360手机助手	360手机卫士	360搜索

与同类框架比较

DroidPlugin

全面hook接管，追求完美插件化，任何一个apk都能直接作为插件运行。

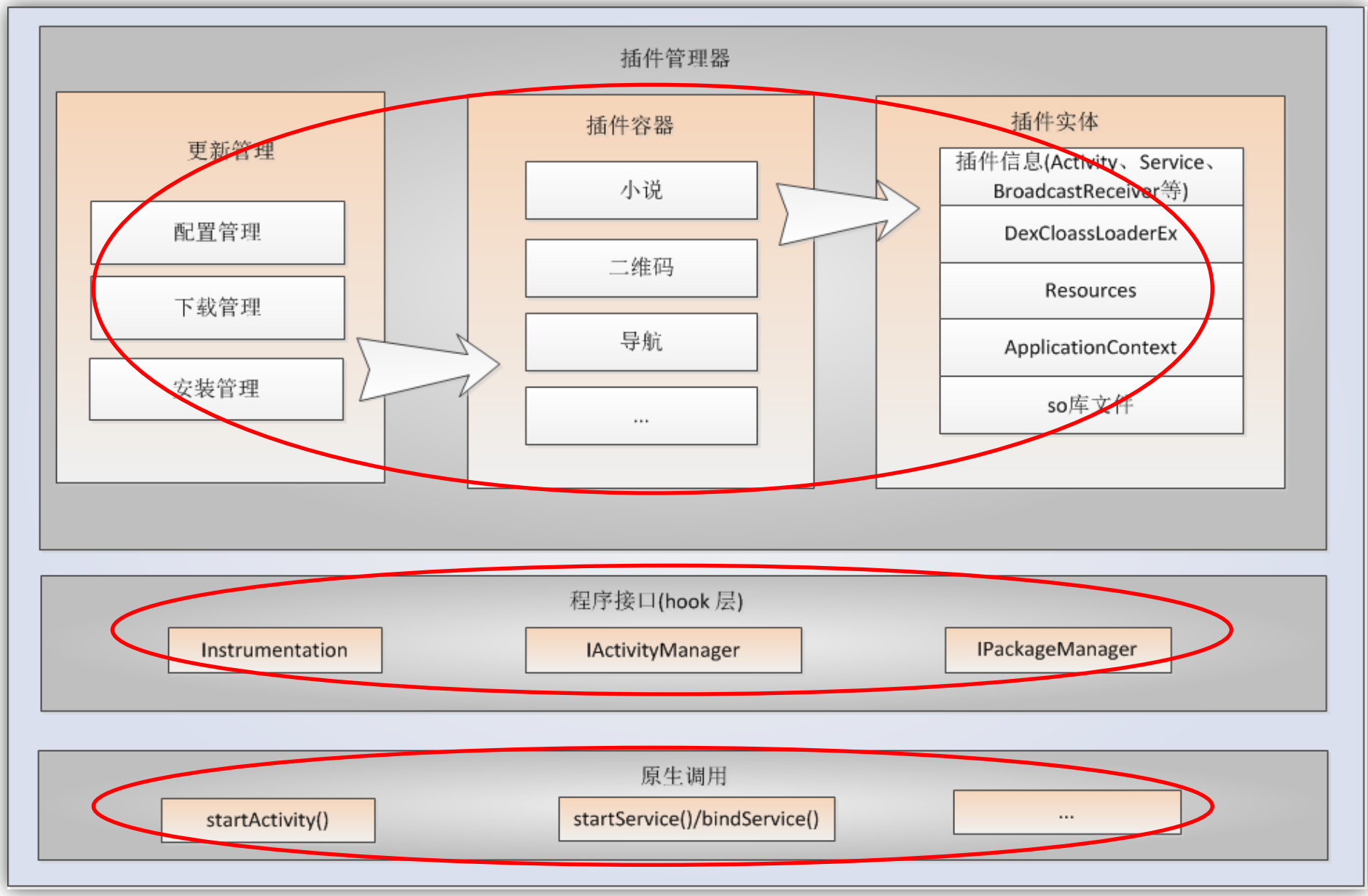
RePlugin

坚持一个hook点、稳定性原则，实现全面插件化，包括UI组件插件化。

Lipland

以瘦身减流为目的，实践场景为参考，用尽量简的方式实现免修改插件框架。

整体架构


插件管理器
更新管理
配置管理
下载管理
安装管理
插件容器
小说
二维码
导航
...
插件实体
插件信息(Activity、Service、BroadcastReceiver等)
DexCloassLoaderEx
Resources
ApplicationContext
so库文件
程序接口(hook 层)
Instrumentation
IActivityManager
IPackageManager
原生调用
startActivity()
startService()/bindService()
...

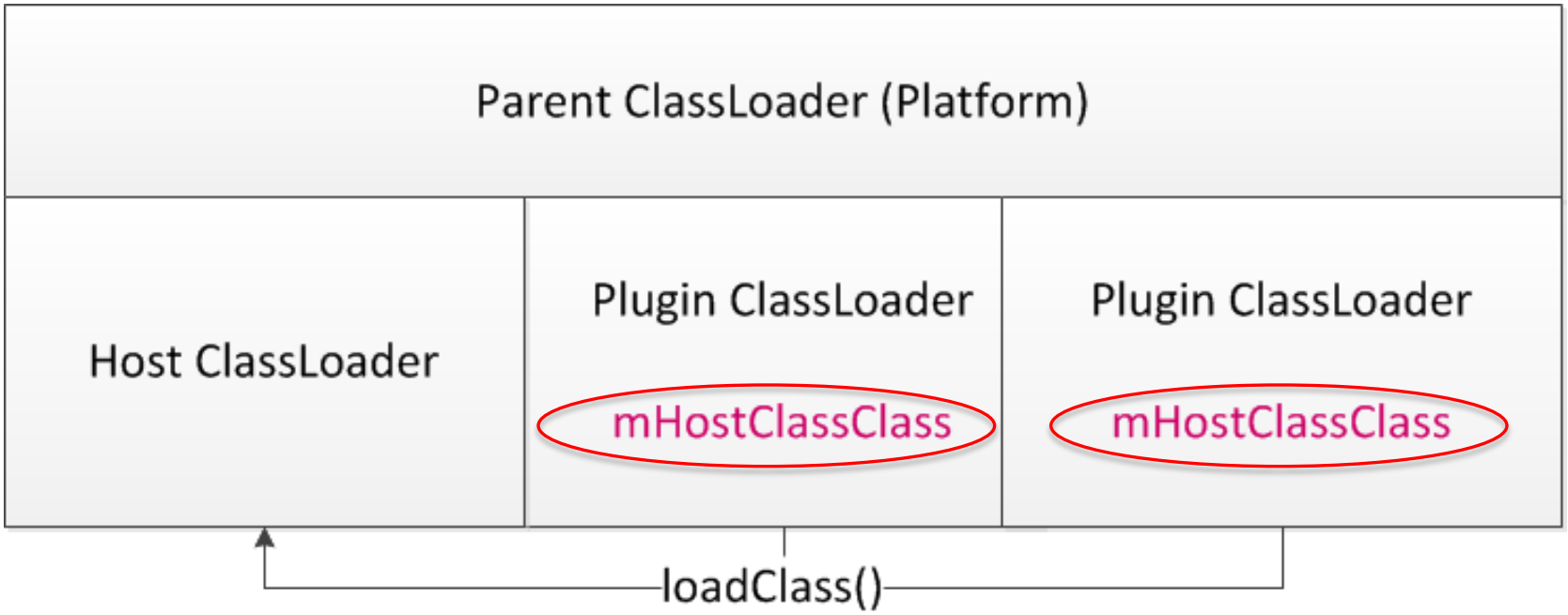
代码共享机制

代码共享机制

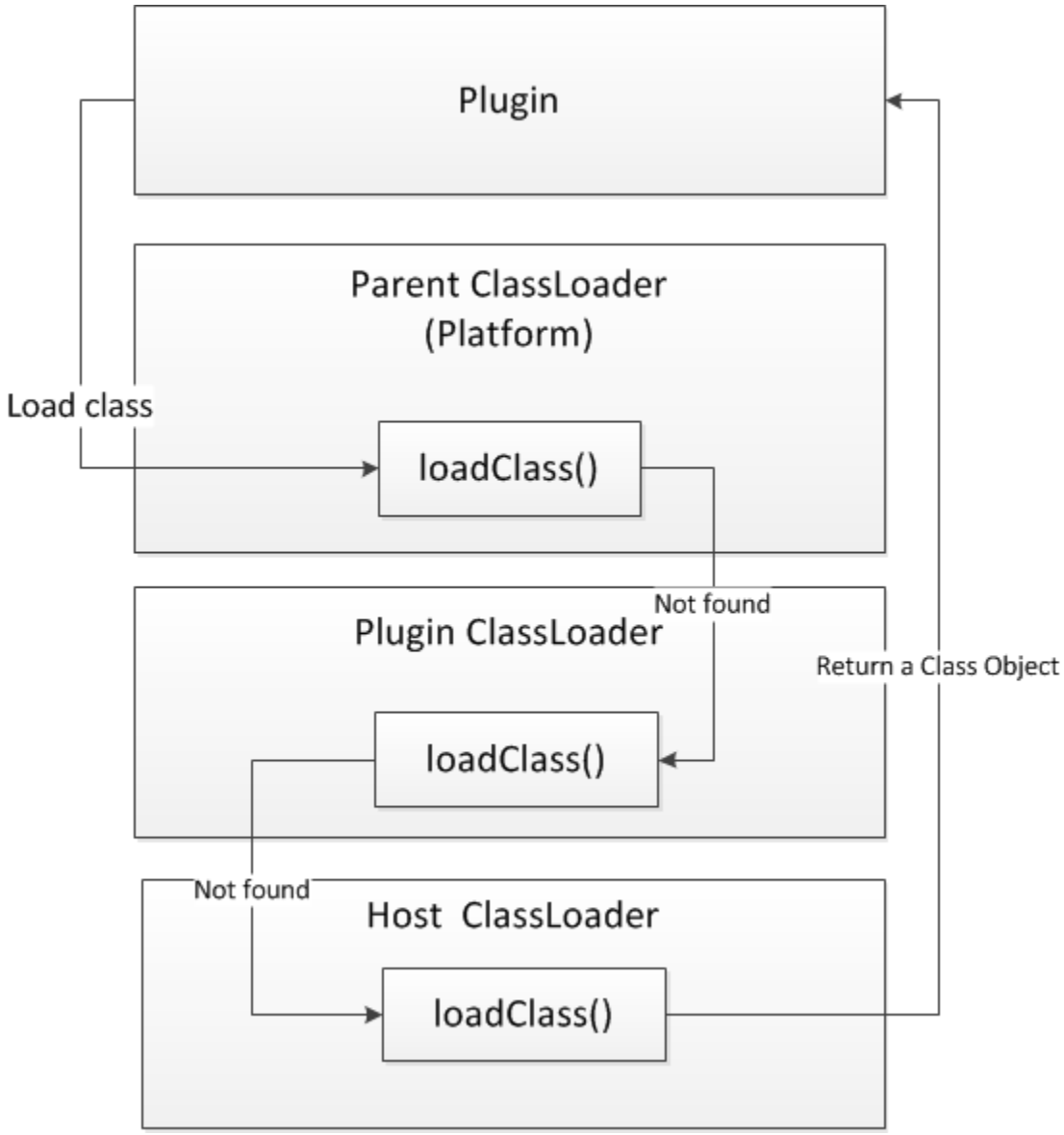
可共享的代码：

- 宿主本身的功能代码。
- 宿主的第三方jar包。
- 宿主的so库。

ClassLoader模型



插件类加载过程


Plugin
Parent ClassLoader
(Platform)
Load class
loadClass()
Not found
Plugin ClassLoader
Return a Class Object
loadClass()
Not found
Host ClassLoader
loadClass()

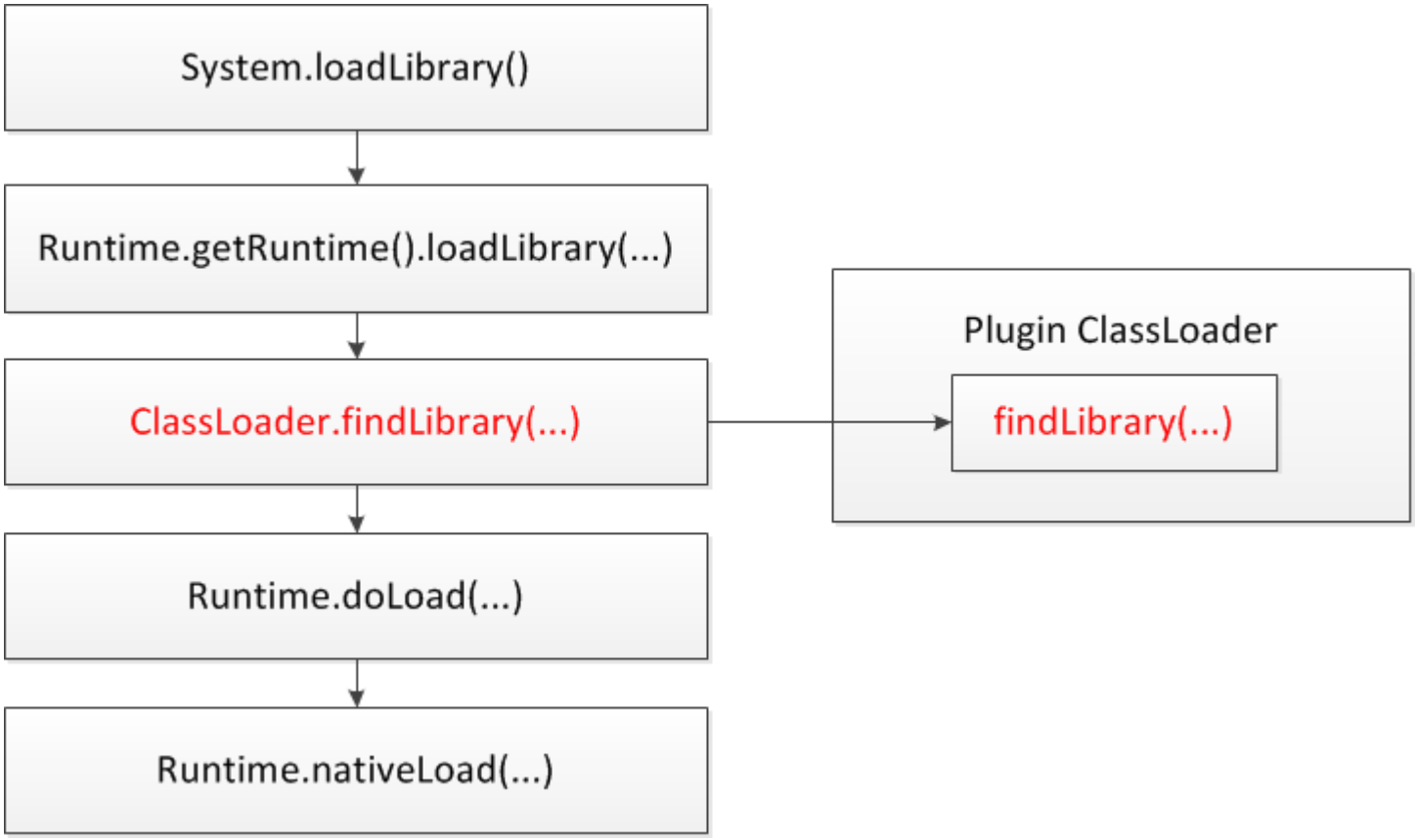
Jar共享

Jar共享的几个需注意的点：

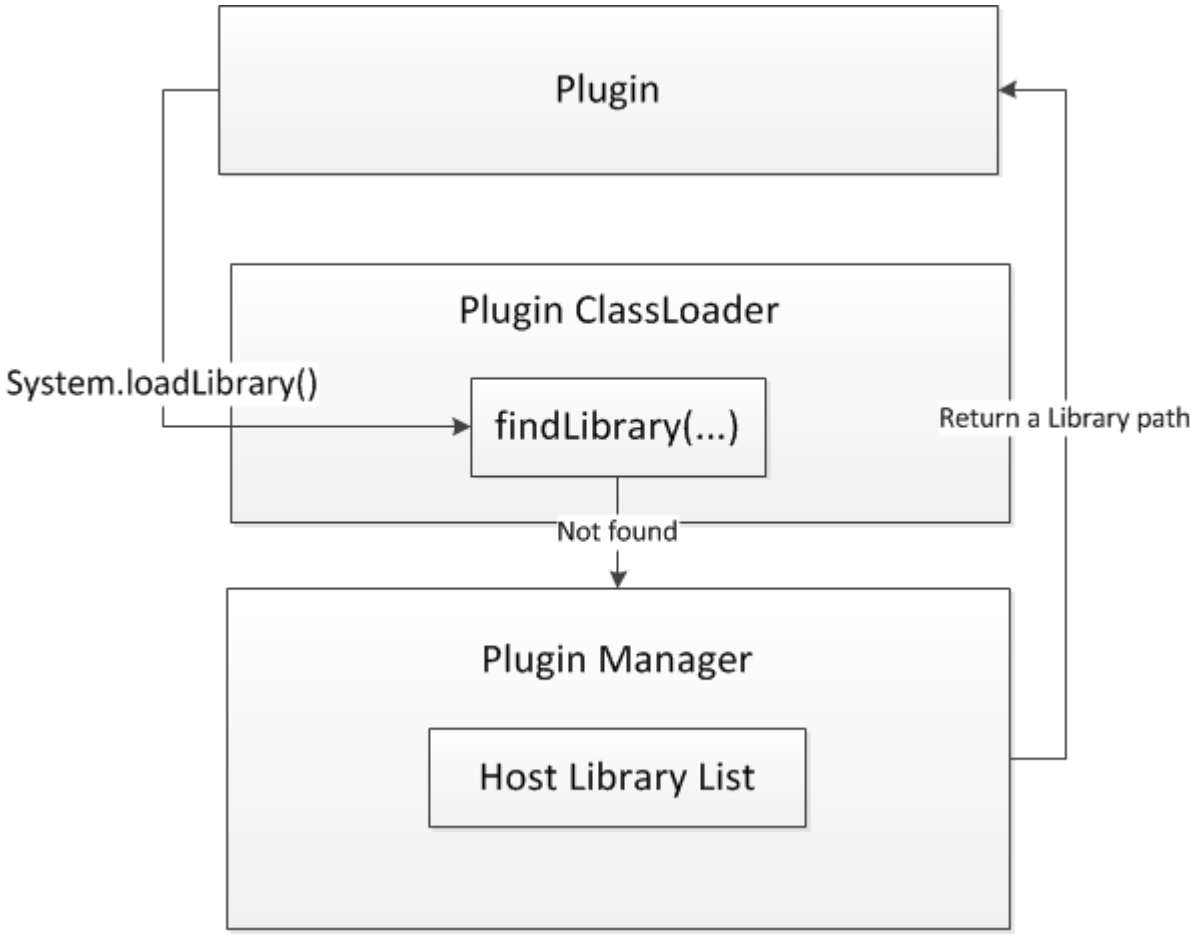
- 插件用Provided的方式编译jar包
- 共享的jar包不能混淆
- 如果换版本应确保jar包的调用接口没变化
- 插件也可以独立使用某个版本的jar包

代码共享机制

so加载过程



so共享实现原理


Plugin
Plugin ClassLoader
System.loadLibrary()
findLibrary(...)
Return a Library path
Not found
Plugin Manager
Host Library List

进程间通信

支持Android所有的原生进程通信方式：

- AIDL通信
- BroadcastReceiver
- ContentProvider
- 组件间传递Intent

插件Activity实现原理

插件Activity实现原理

要启动插件Activity，我们需要解决三个问题：

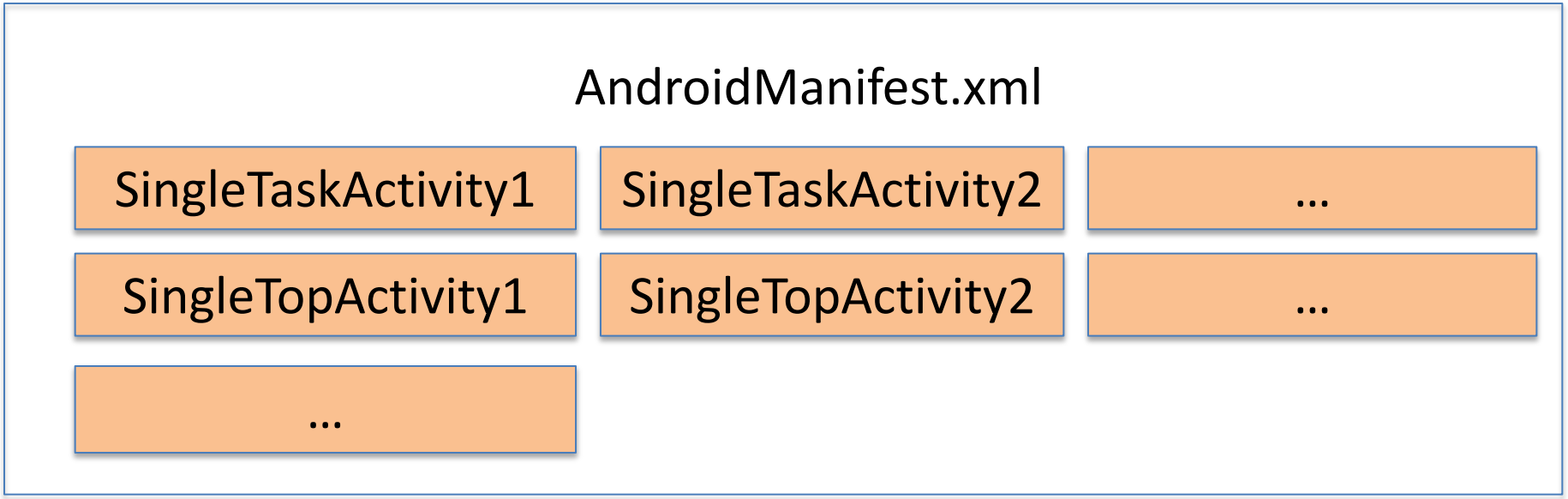
1、插件Activity的身份认证。

2、怎么知道应用正在启动一个Activity。

3、启动插件的Activity，并保持它完整的生命周期和事件回调

插件Activity的身份认证

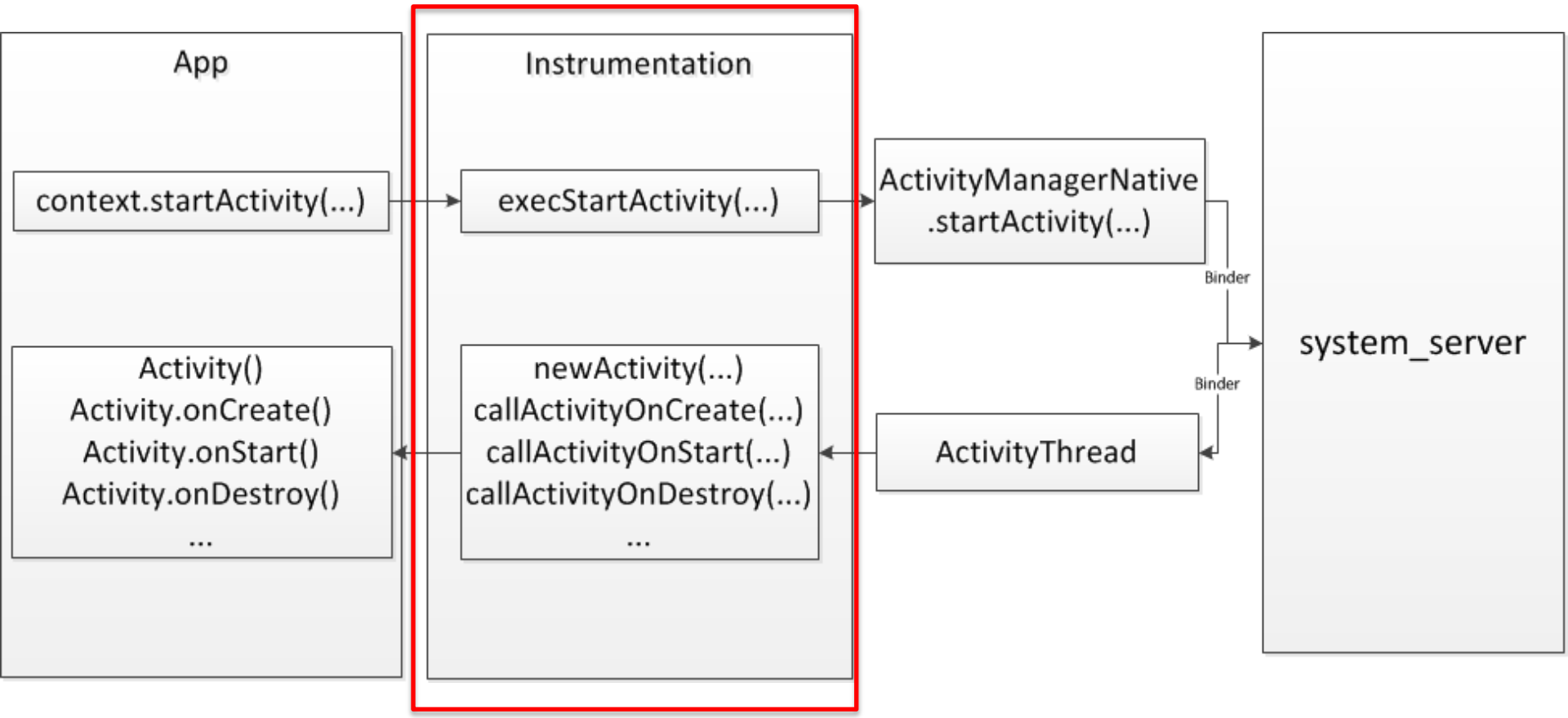
Activity池



由Gradle根据配置自动注册，插件框架自动选取最合适的Activity

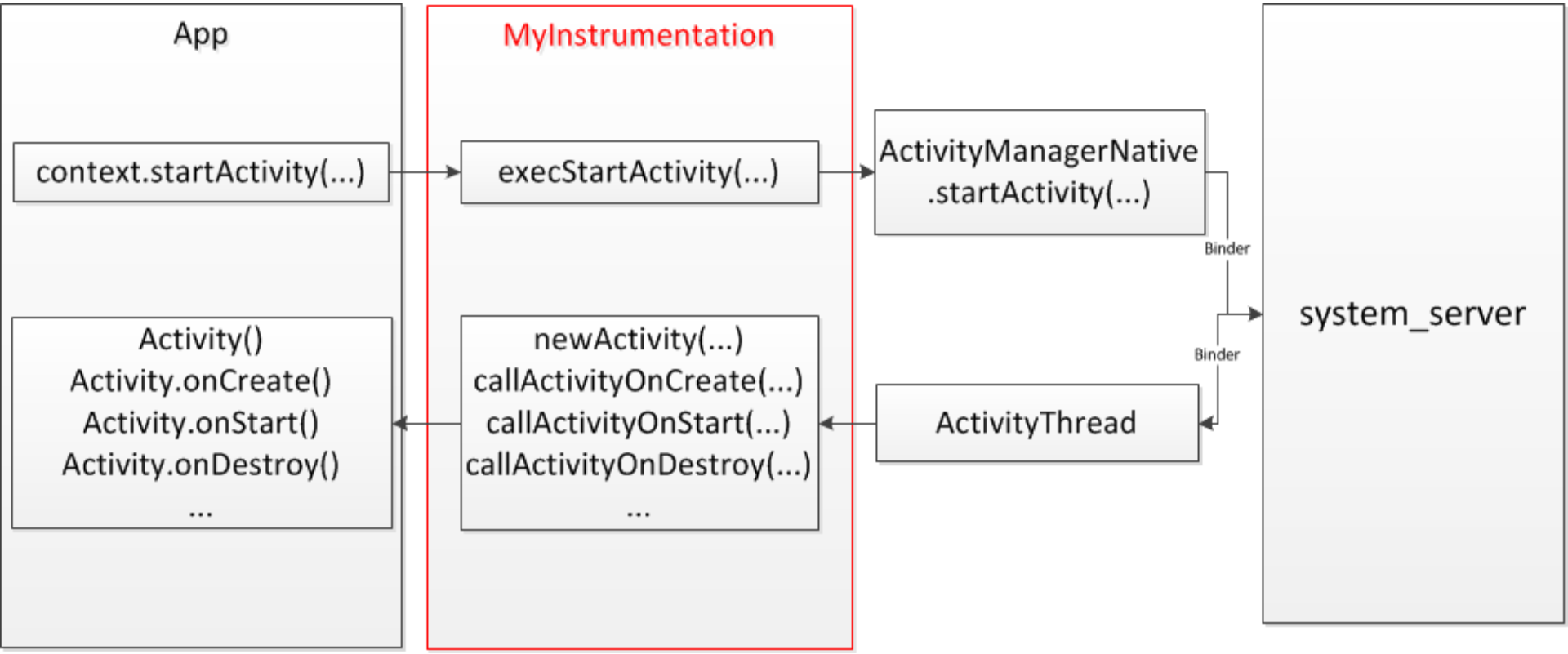
插件Activity实现原理

Activity启动过程



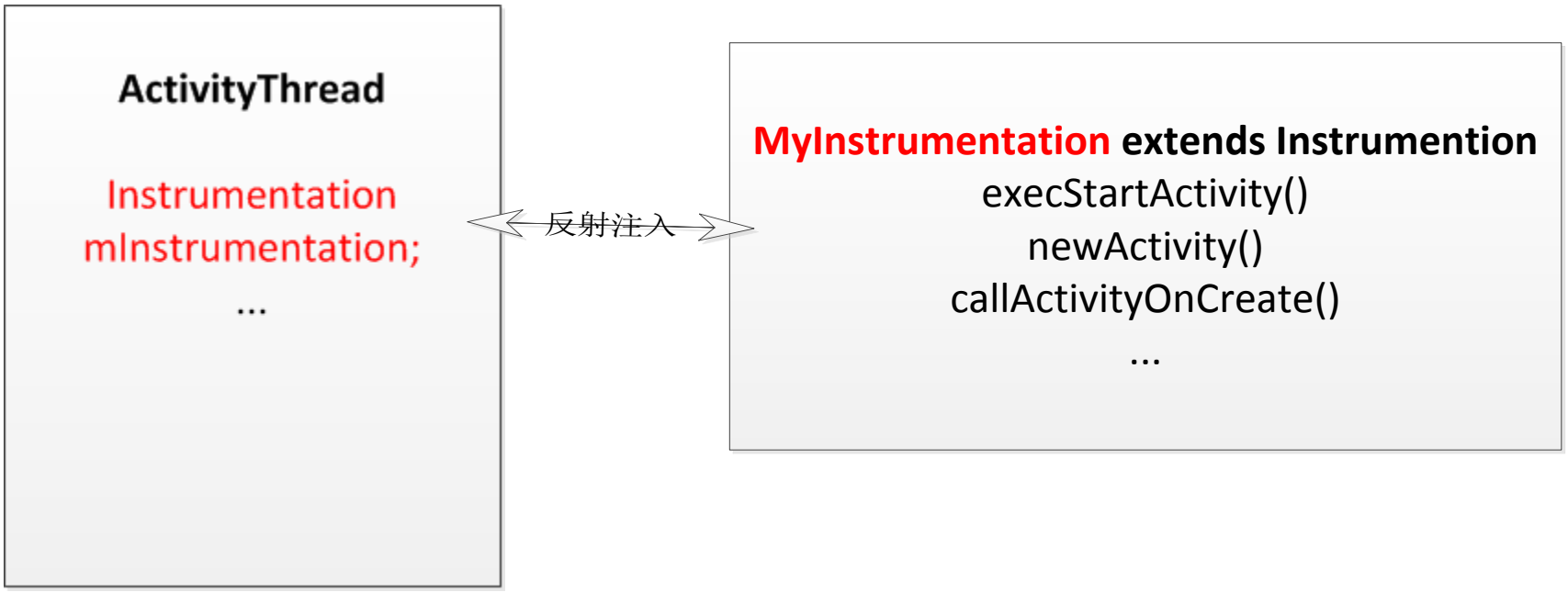
插件Activity实现原理

Activity启动过程



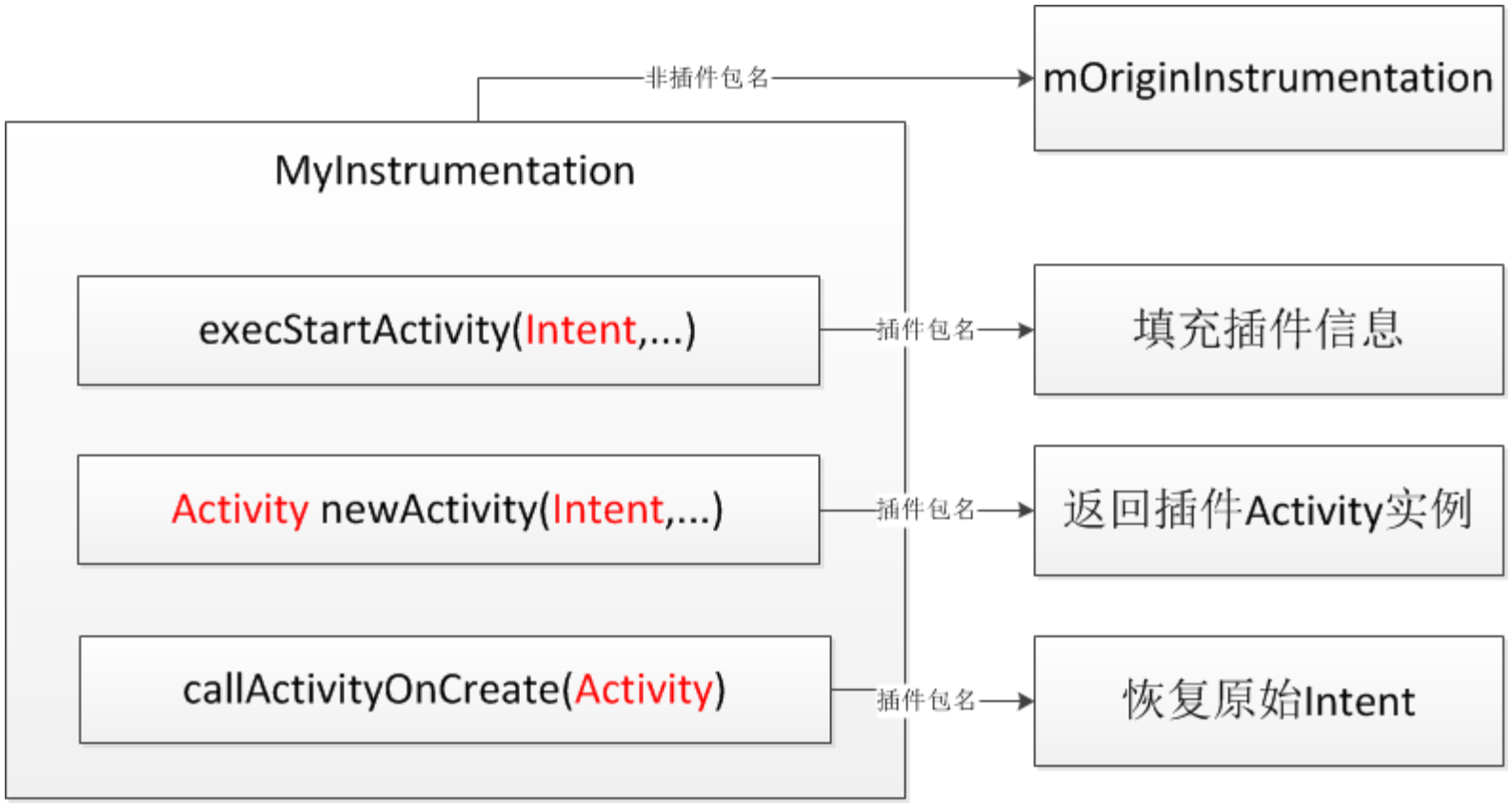
插件Activity实现原理

替换Instrumentation



插件Activity实现原理

启动插件Activity



插件Service实现原理

插件Service实现原理

插件Service调用过程

1、存在一个真实注册的Service，叫做ServiceWrapper

2、 ServiceWrapper维护所有的插件Service的实例列表

3、 ServiceWrapper给每个插件Service代理转发Service事件和生命周期回调。

插件Service实现原理

Hook位置

IActivityManager接口

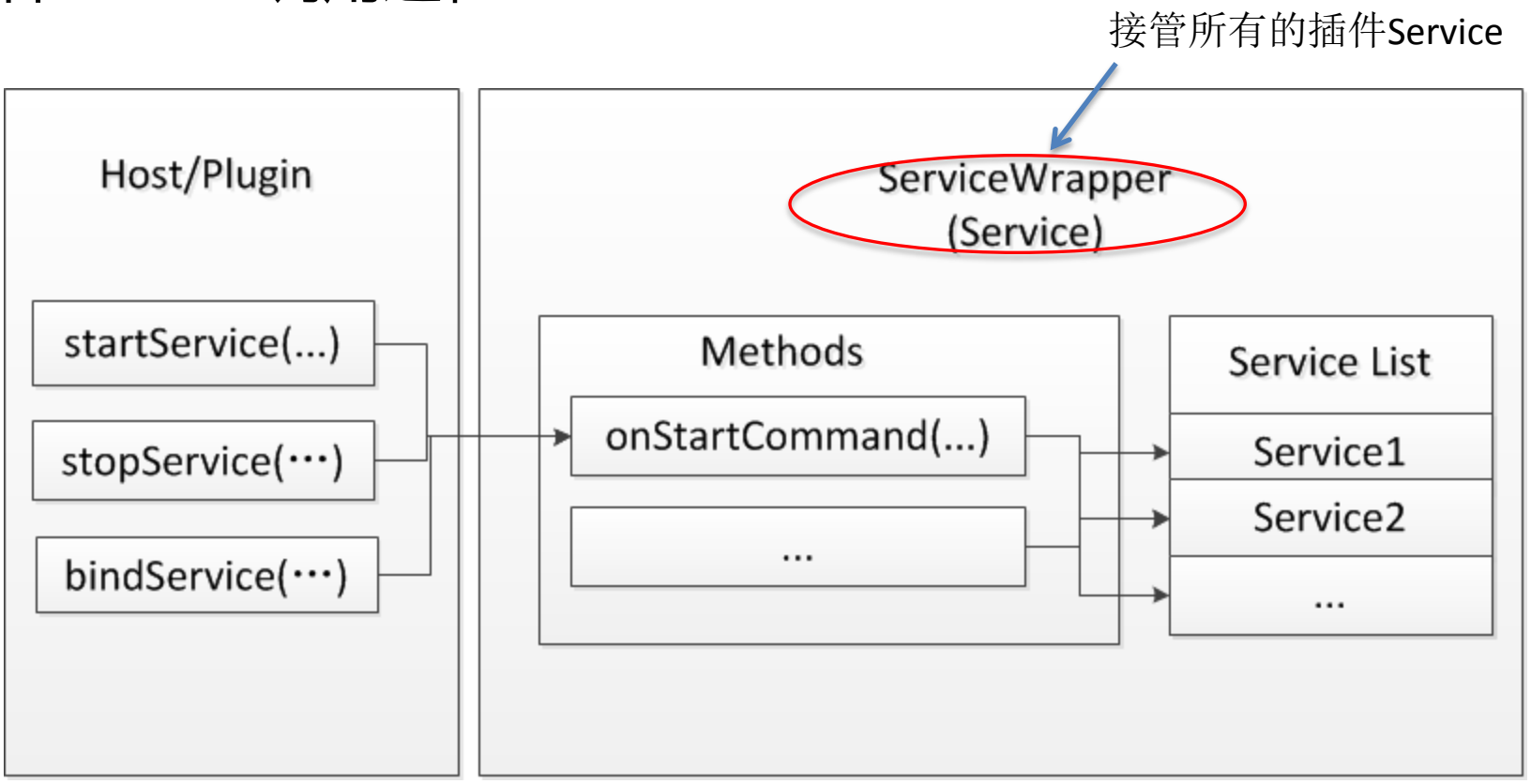
startService()
stopService()

stopServiceToken()

bindService()
unbindService()

插件Service实现原理

插件Service调用过程



插件BroadcastReceiver实现原理

插件BroadcastReceiver的注册

- 动态Receiver

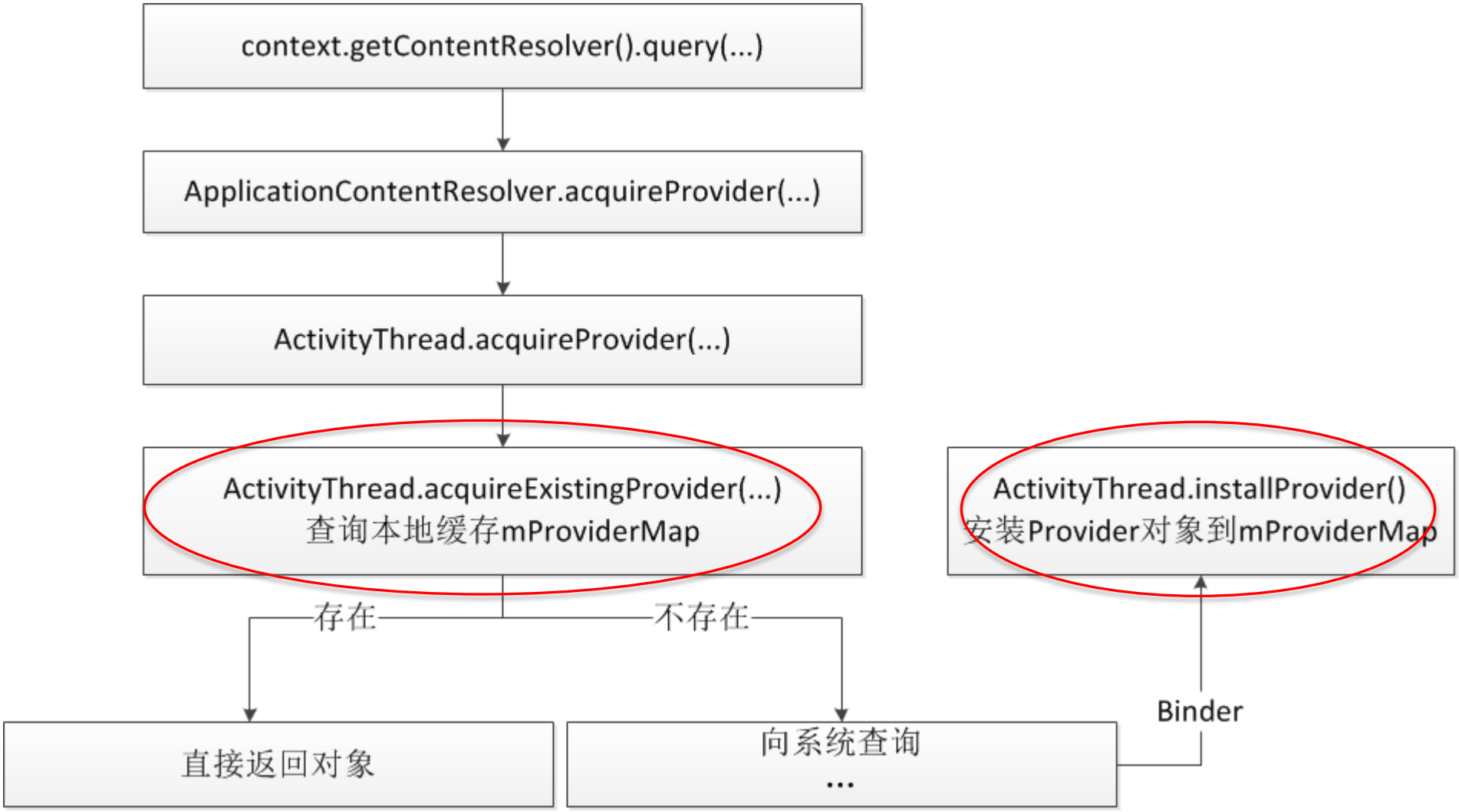
  直接通过插件context对象注册，无需特殊处理

- 静态Receiver

  在解析完插件信息后，直接将所有的静态Receiver用registerReceiver()动态注册进系统

插件ContentProvider实现原理

ContentProvider查询过程



插件ContentProvider实现原理

实现思路

在解析完插件apk后，将注册的Provider直接通过ActivityThread.installProvider()方法安装到本地缓存中。

在360搜索中的实践

面临的问题

- 随着版本迭代，应用体积越来越大。
- 应用模块越来越多，修改一个小的模块需要更新整个应用。
- 不同团队通用模块复用困难，版本难以统一。
- 进行功能控量下发时，需要发布新版本，未命中的用户也带了不需要的新功能。

360搜索面临的问题

为使传统搜索H5页面native体验化，客户端需要实现大量的独立的功能，这就将导致应用规模急速膨胀。

小说 5M

看图 1M

视频 1M

地图导航 14M

语音助手 10M

拍图扫码 4M

360清理大师 5M

…

面临的问题

体验问题

作为一个搜索入口，我们可以明确判断用户当下的需求，但是我们有时候却不能为他们真正解决。

面临的问题

- 用户实际上提出了明确需求：解决我的手机卡顿问题

- 得到这个结果，用户的心里也许有一万头神兽在奔腾。

面临的问题

一万头神兽

如何解决这些问题

怎么解决？

体积和业务规模膨胀的问题

- 所有独立功能模块插件化，由专门独立的团队维护，共享给多个部门。
- 统一更新/控量下发插件机制。
- 宿主本身只负责搜索业务，并作为一个插件容器而存在。

怎么解决？

360搜索
12M
独立发布

动态下发

小说
扫码
美图
地图
清理大师
...

插件
(大小不定)

怎么解决？

体验问题

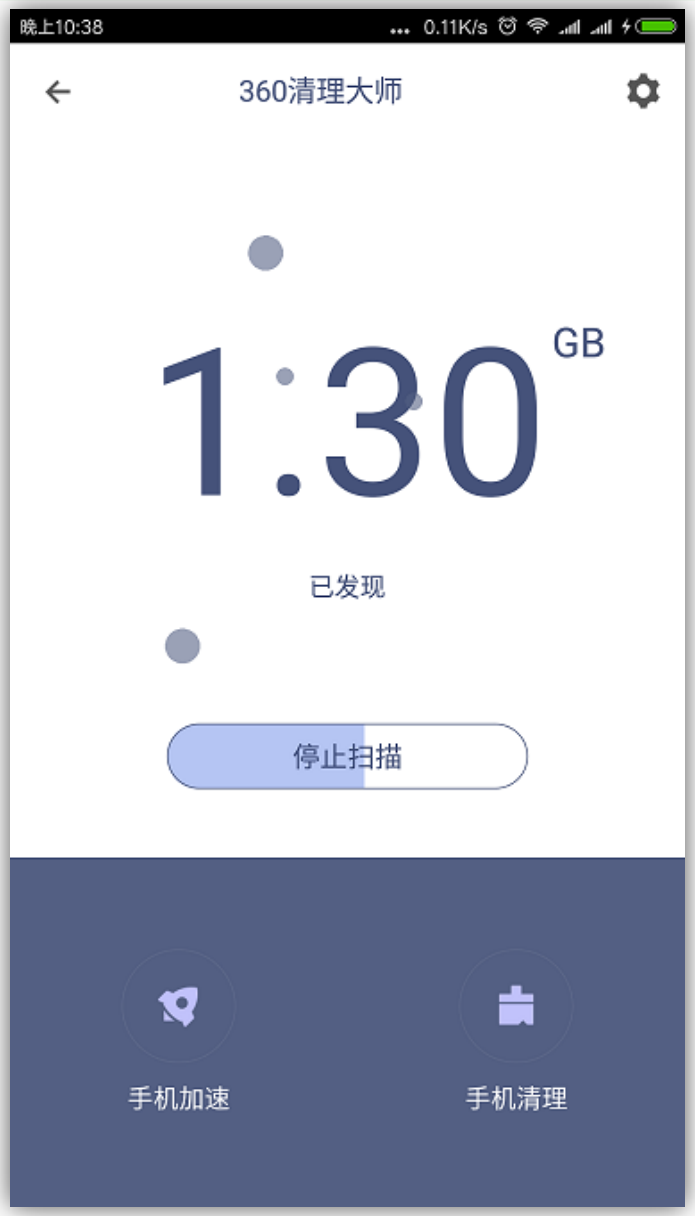
怎么解决？

专门的事由专门的团队来解决。
聚焦核心业务。

怎么解决？

智能推荐插件，发现用户的潜在需求

怎么解决？

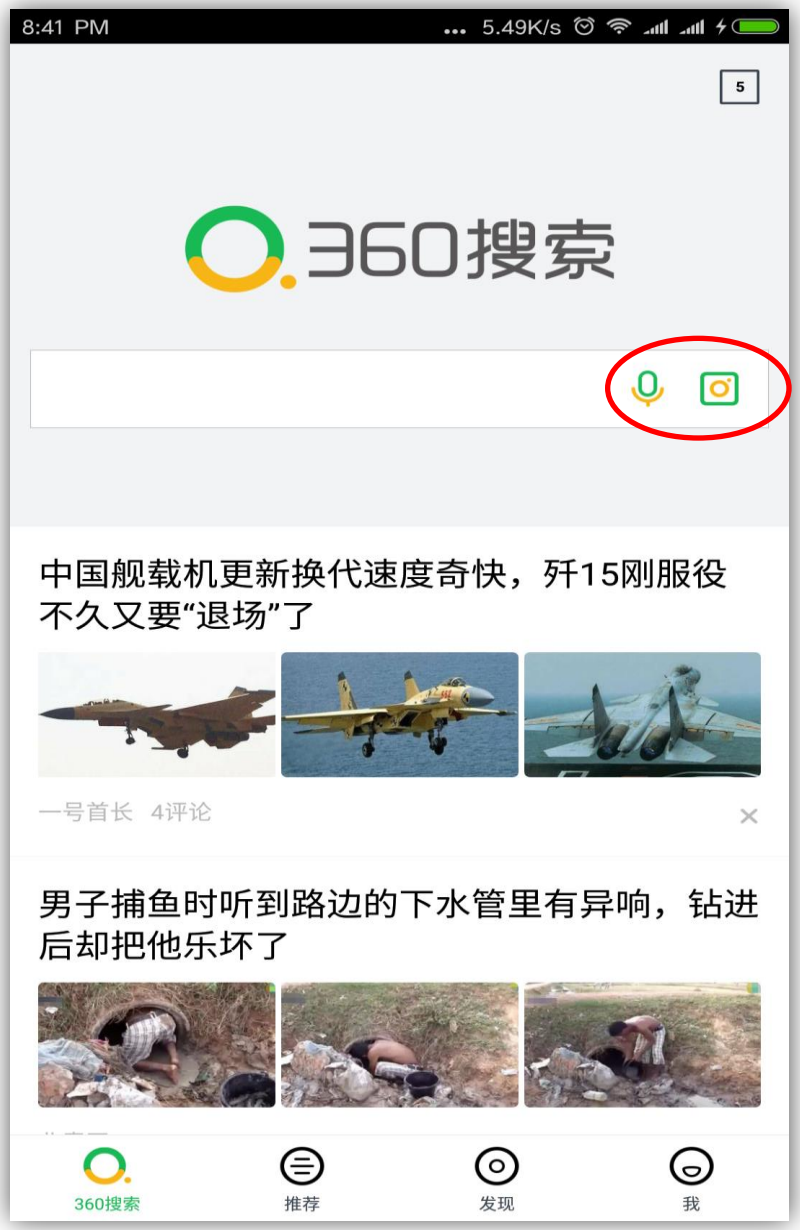


完整的功能，免安装、用户无感知。

满足用户的真实需求，老少皆宜，门槛低。

时间成本、人力成本极低，产品复用率高

更多插件



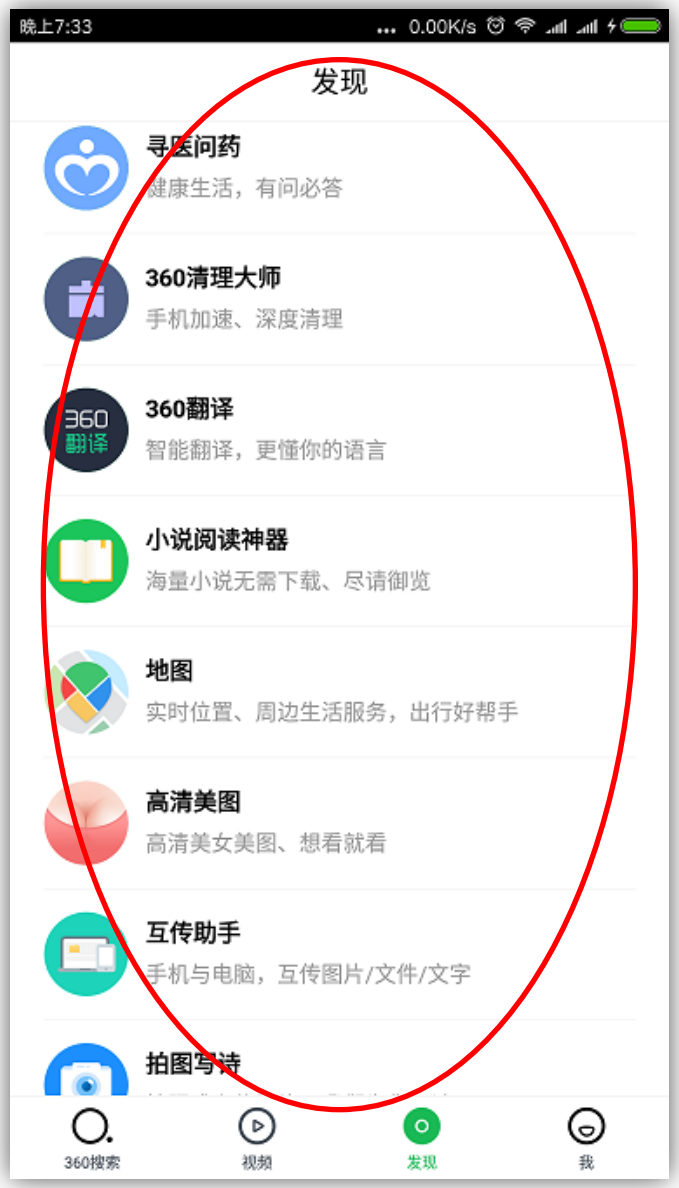
语音助手插件和扫码插件。

更多插件


8:36 PM
5.85K/s
798
公交
驾车
步行
骑行
荧屏里小区
北京民生现代美术馆
格兰维森商务楼
大千食
798创意广场
797路
酒仙酒店公寓
中国当代设计博物馆
亚洲艺
重庆美食川菜家常菜
宏源大厦
终
798艺术区
宏源公寓
大山子社区
东方之光国际中心
张一元
缘分新媒艺术空间
起
蜂巢当代艺术中心
星巴克
朝鲜
星巴克
万红里甲31号院
高家园
万红路
星城亮马国际公寓
酒仙桥
导航
11分钟
677米
详情>

地图导航插件

更多插件



以搜索框query词为关键字

确定用户潜在需求，唤起相应插件

实现native化体验，满足用户真实需求

更多插件化案例

360语音助手

360身边生活

360清理大师

360地图

360小说

360影视

360视频播放器

滴滴打车(sdk)

360看图插件

360美图

360扫码

360天气

拍题插件

饭补插件

…

Lipland为360搜索做了什么

- 减少360搜索体积60%，大幅度减少用户升级成本。
- 拆分出多个独立的团队，例如：小说团队、地图团队等，开发的产品既可复用又独立发布。
- 解放了搜索团队，使人力聚焦在搜索业务开发，大幅度减少了人力成本。

谢 谢!

安全第一

Lipland项目360官方github地址：
https://github.com/Qihoo360/Lipland

技术交流（干货）：奇卓社（360移动技术微信公众号）