滴滴出行-DoKit-开源开发实践

金台

滴滴资深研发工程师

自我介绍

姓名：金台

家乡：浙江台州

职级：资深研发工程师

角色：DoKit项目负责人

爱好：电影&爬山&美食

目录

CONTENTS

你心目中的开源

Part 01 项目&功能介绍

DoKit是什么?

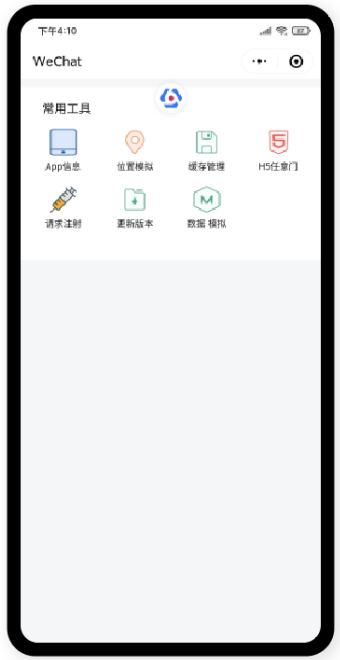
北京大学 PEKING UNIVERSITY

dokit.cn

DoKit For iOS

DoKit For Android

DoKit For Flutter



DoKit For 小程序

思想自由 兼容并包

DoKit项目背景

github: https://github.com/didi/DoraemonKit

官　网:https://www.dokit.cn/

DoKit的由来

痛　点：工具繁杂、业务耦合、研发迁移成本

解决方案：统一管理、内置通用工具、创新、业务代码零侵入

DoKit终端功能总览

北京大学 PEKING UNIVERSITY

基础功能

DoKit插件、悬浮窗、更多页面

平台工具

数据Mock、健康体检、文件同步助手、一机多控

常用工具

App信息、三方库信息、开发者选项、本地语言、沙盒浏览、位置模拟、H5任意门、缓存清理、日志、H5助手

Weex工具

日志、缓存、信息、DevTool

性能监控

帧率、CPU、内存、网络、Crash监控、卡顿、大图检测、模拟弱网、启动耗时、UI层级、函数耗时

视觉工具

取色器、对齐标尺、控件检查、布局边框

DoKit FT团队

北京大学 PEKING UNIVERSITY

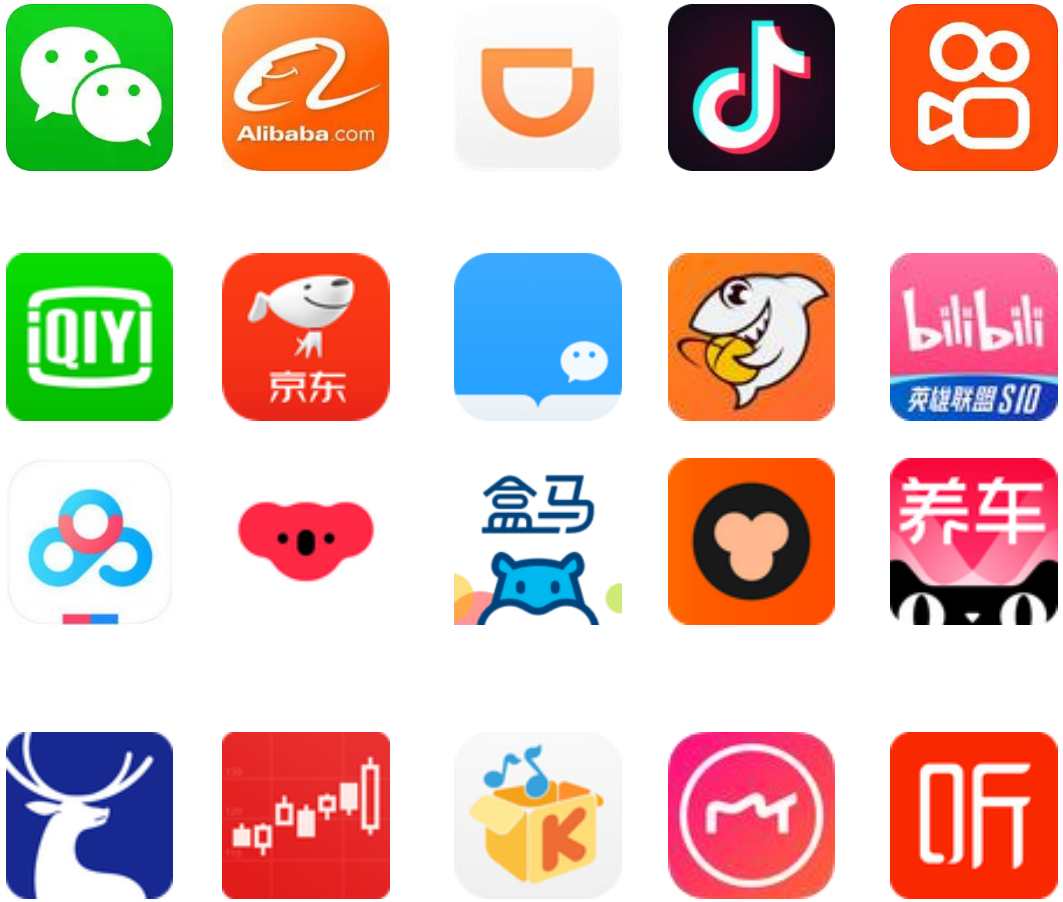
北京大学
PEKING UNIVERSITY

DoKit 效果演示

DoKit社区成绩

北京大学 PEKING UNIVERSITY

指标	数据	备注
github star	16635	滴滴第一 全球前800
github Fork	2362	--
github PR	262	90%以上来自于外部
DoKit活跃用户	10000+	2020.7重新统计
DoKit平台用户	5287	手机号
DoKit平台产品数	4008	Android&iOS



DoKit业务洞察--悬浮窗

痛点

1.需要系统权限，阻碍自动化测试

2.没有统一的API，开发者自定义成本高

解决方案

1.页面自管理，全局记录控件位置信息

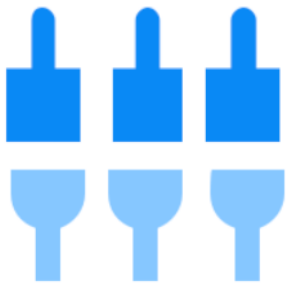
2.重构悬浮窗架构，统一对外API并提供模板方法

业务价值

1.极大的提升了用户自定义悬浮窗的效率

2.减少操作路径，提升自动化测试的效率

DoKit业务洞察--接口Mock



接口Mock

基于App网络拦截方案，无需修改代码即可完成接口Mock

痛点

1.不支持多人协同、数据无法持久化保存

2.操作繁琐

3.业务代码侵入

解决方案

1.统一多个网络库

2.AOP插装，业务代码零侵入

业务价值

1.统一工具，降低学习成本

2.多人协同，提升研发效率100%

3.业务代码无污染，保证线上代码交付质量

DoKit业务洞察--健康体检

健康体检

一键式操作，整合dokit多项工具，数据可视化，
快速准确定位问题,让你对app的性能了如指掌。

痛点

1.性能数据杂乱，产生的数据没有统一的备份和记录

2.操作繁琐、定位问题成本太高

解决方案

1.打通DoKit所有性能工具，提供一键式操作

2.打通平台端，支持数据持久化和图表化并给出优化建议

业务价值

1.一键式操作，提升你的性能数据采集效率

2.数据全面，全方位提现你的每一个性能指标

3.数据可视化，方便展现和定位你的性能瓶颈

DoKit业务洞察--文件同步助手

终端文件同步

通过终端服务，让你的终端空间在平台端完整的
展现并提供强大的文件以及数据库操作能力

痛点

1.终端屏幕尺寸限制，操作不便

2.无法针对手机空间进行多人协同操作

3.终端和PC无法直接交互，需要第三方工具同步(微信)

解决方案

1.通过直接在终端开启http服务并制定相关文件交互接口

2.打通平台端，支持直接在DoKit平台上操作终端空间

业务价值

1.终端空间透明，大屏操作更便捷

2.终端控件多人共享，协同操作更便捷

3.终端接口开放，方便社区自定义平台

DoKit业务洞察--一机多控

一机多控

主从同步，释放人力，让效率提升看得见

痛点

1.跨端技术落地，测试效率无法跟进

2.传统方案成本和门槛较高

解决方案

1.基于局域网WebSocket建立连接

2.捕捉主机手势信息，从机及时响应

业务价值

1.无需任何配置，打开即用

2.低延迟，真实，体验效果极佳

3.主从同步，提升兼容性测试效率

DoKit Part01思考

北京大学 PEKING UNIVERSITY

1、如何选择开源项目?

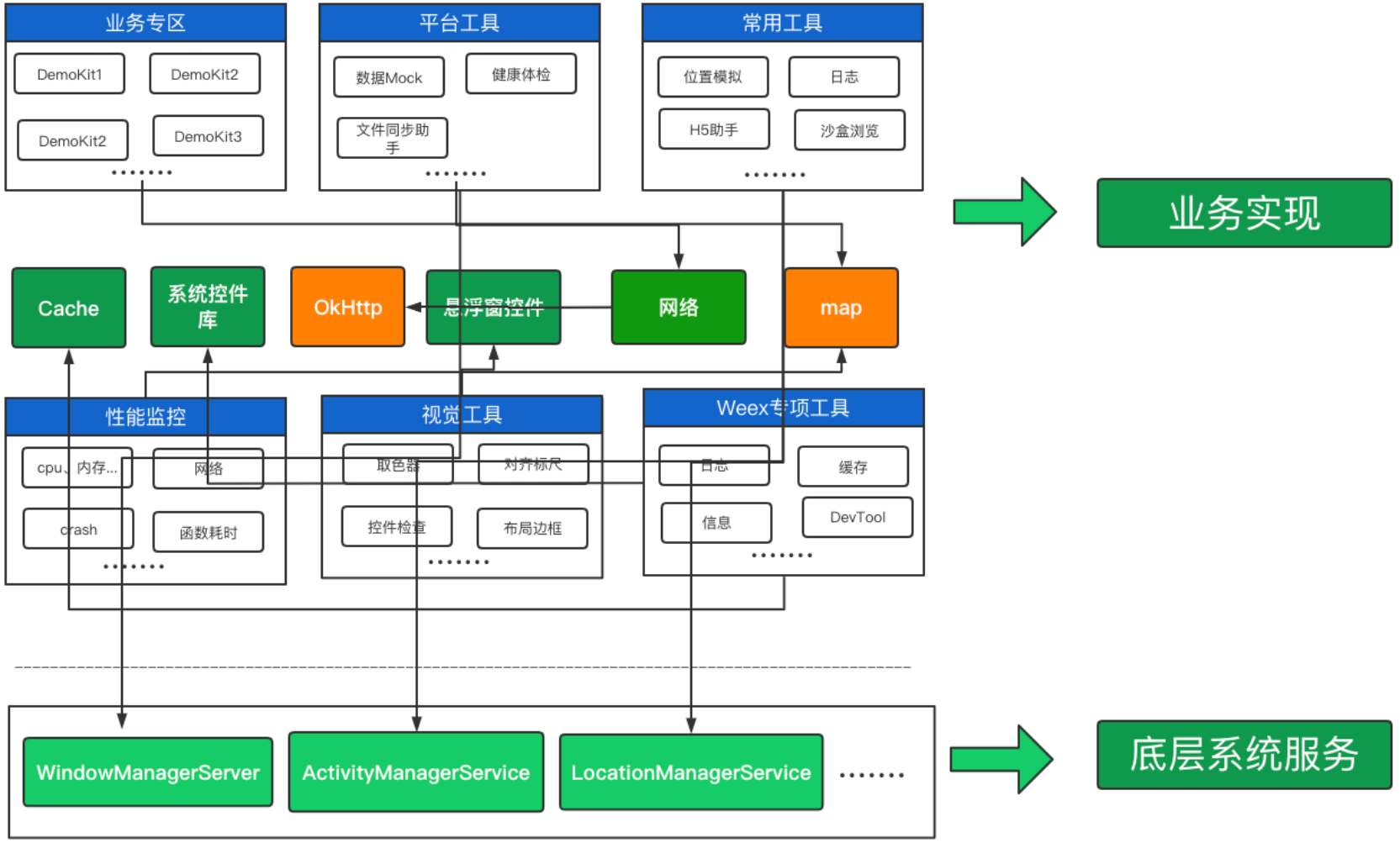
2、优秀的开源项目应该具备哪些特性

3、项目功能如何规划迭代?

北京大学
PEKING UNIVERSITY

Part 02 功能原理分析

DoKit 老架构



稳定性差

- 架构分层设计不足
- 组件间耦合严重

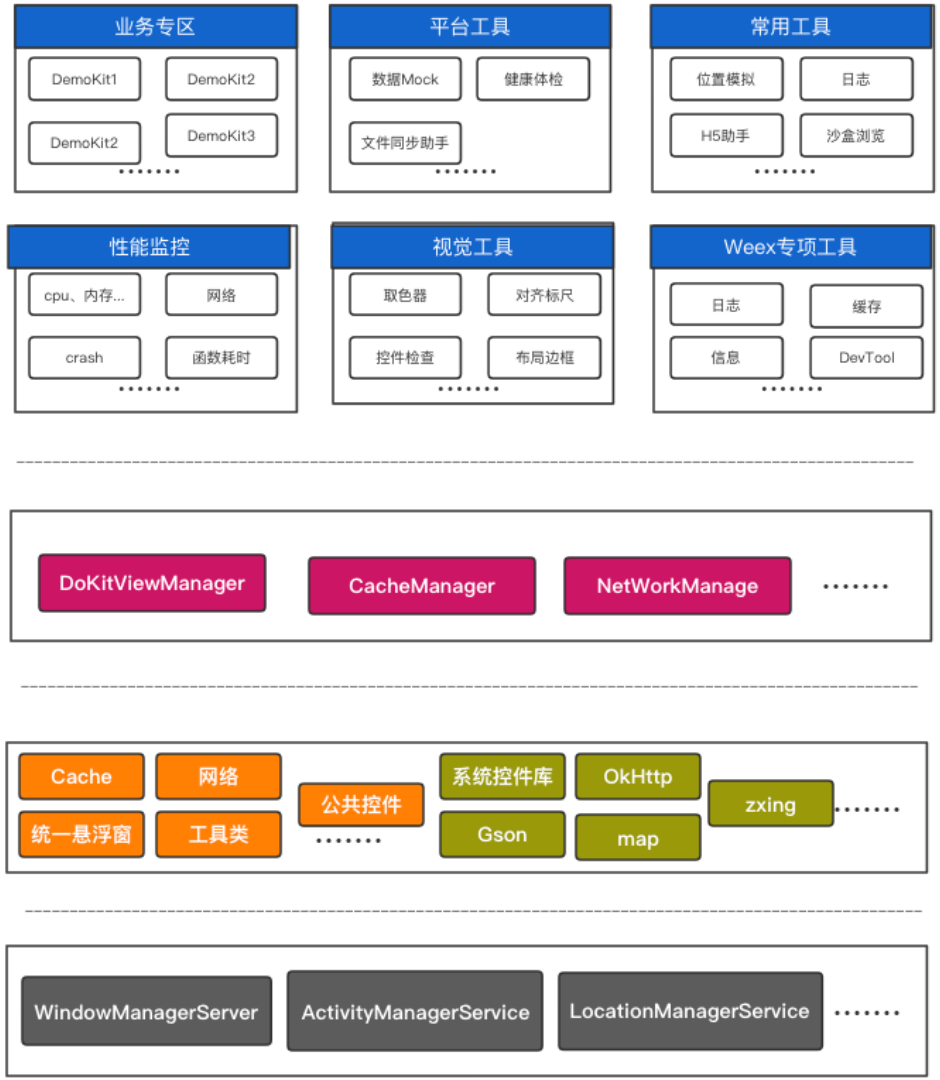
可扩展性差

- 基础模块能力缺失
- 重复造轮子

可维护性差

- 代码风格不统一
- 重复代码多

DoKit 新架构



难点

- 历史包袱重、冗余代码多
- 重构与需求并行
- 自测用例无法全面覆盖

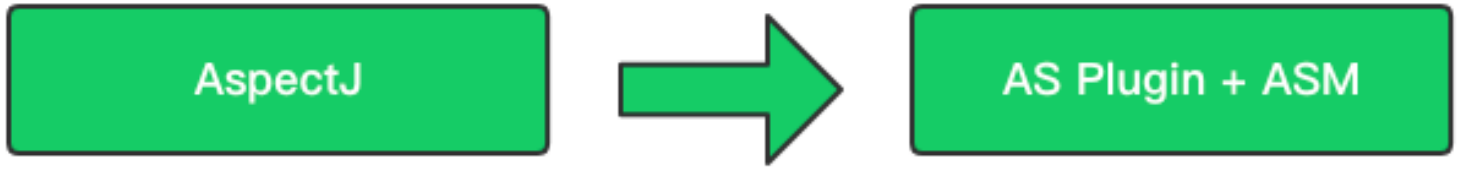
成果

- 代码量减少8%，包体积减少10%
- crash率降低10%
- 社区业务定制效率提升30%

DoKit Android AOP 方案选型

北京大学 PEKING UNIVERSITY

解决业务代码零侵入，避免代码污染



优点

- API丰富、功能强大
- 文档完备
- 项目中广泛使用、稳定性高

缺点

- 集成冲突

优点

- 贴近底层、性能无影响、功能更加强大
- 集成冲突大幅度减少
- 用户配置，操作度高

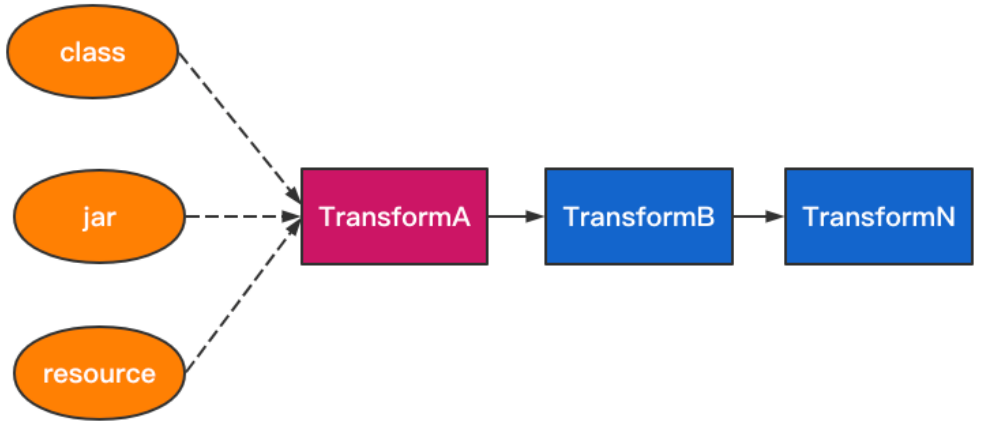
缺点

- JVM字节码的掌握
- 最优的Hook点、主流三方源码阅读

DoKit Android AOP 原理

北京大学 PEKING UNIVERSITY

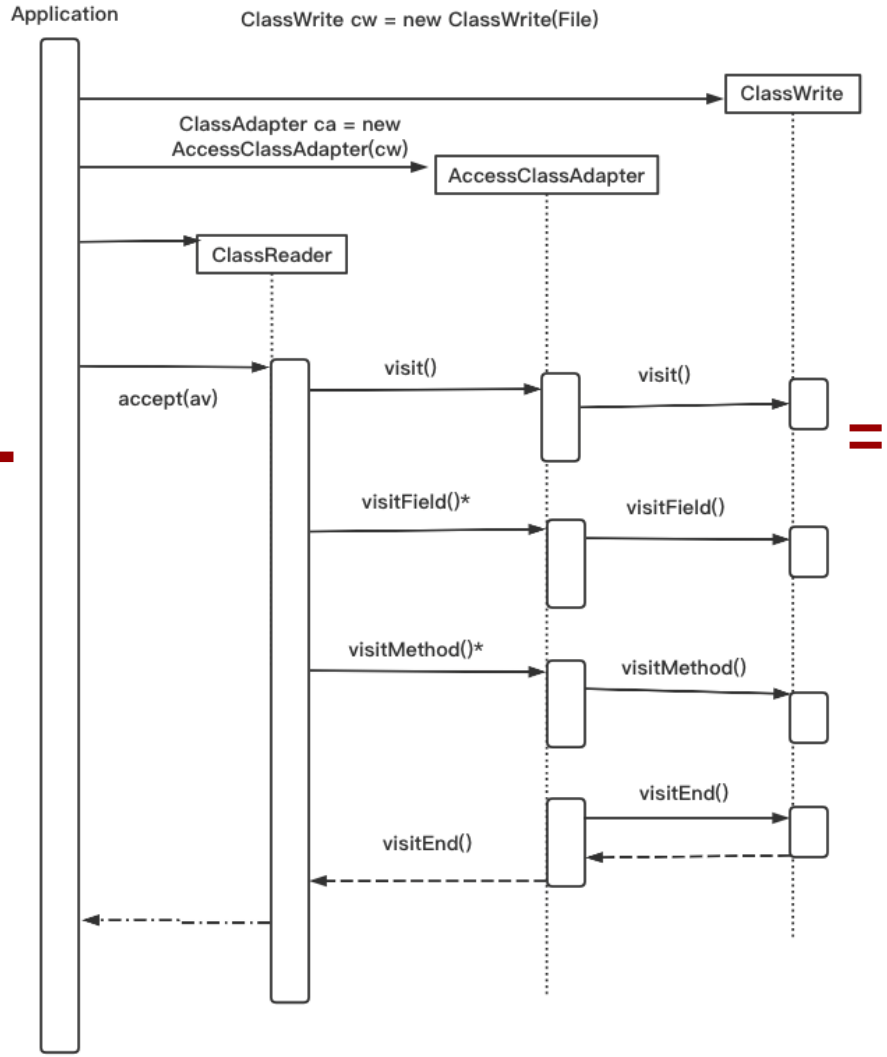
AS Gradle打包流程



紫色:自定义Transform

蓝色:系统Transform

ASM时序图

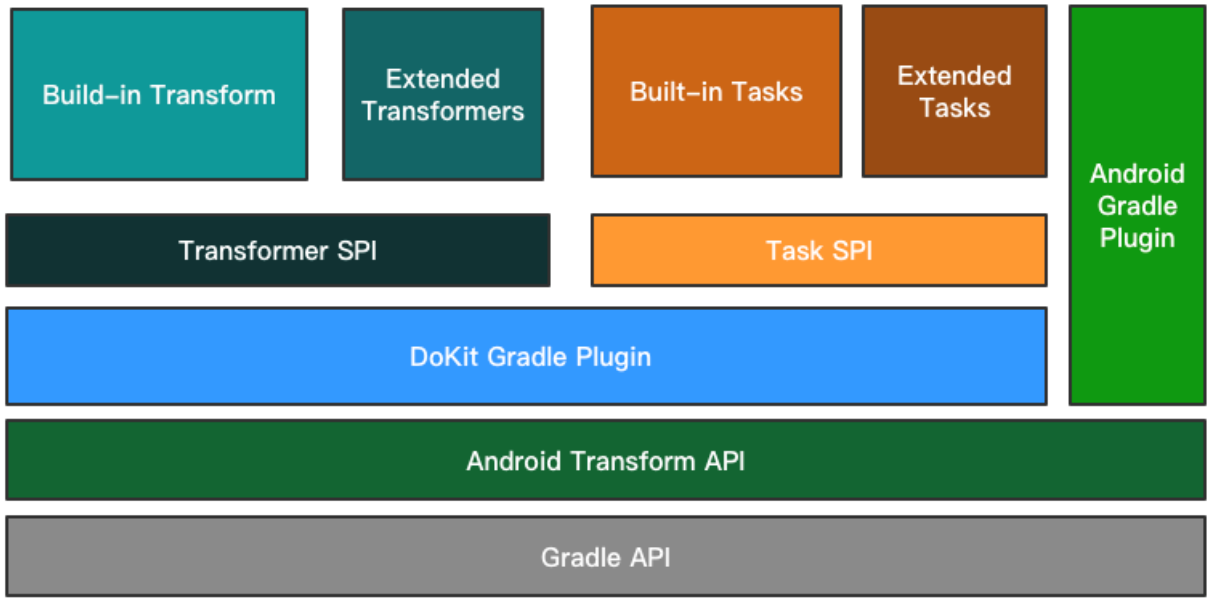


= New JVM Class

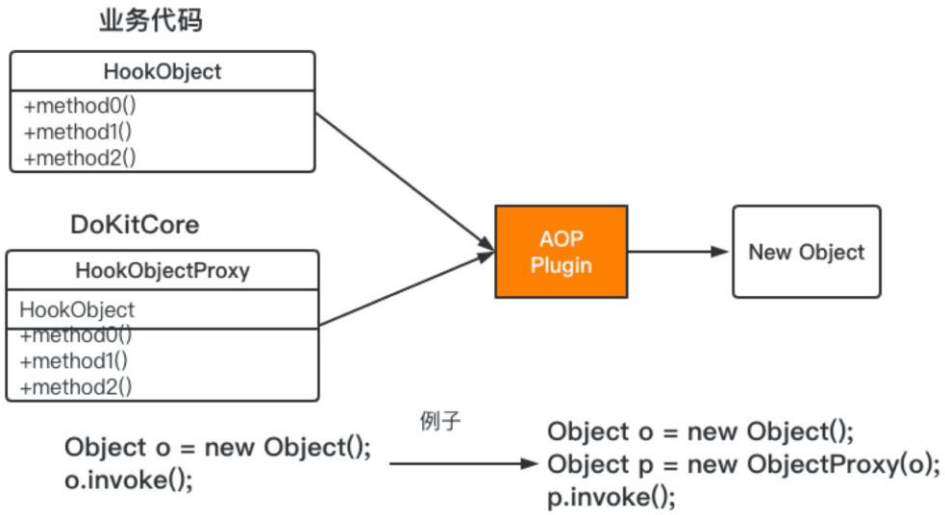
DoKit Android AOP 实现

北京大学 PEKING UNIVERSITY

插件架构



例子



思想自由 兼容并包

场景落地

- 位置模拟
- 网络抓包
- 大图检测
- 函数耗时
- 启动耗时
- 数据Mock
- 三方库信息

现有接口Mock解决方案

北京大学 PEKING UNIVERSITY

抓包工具

Fildder

目标

打造一款面向全平台的数据Mock方案

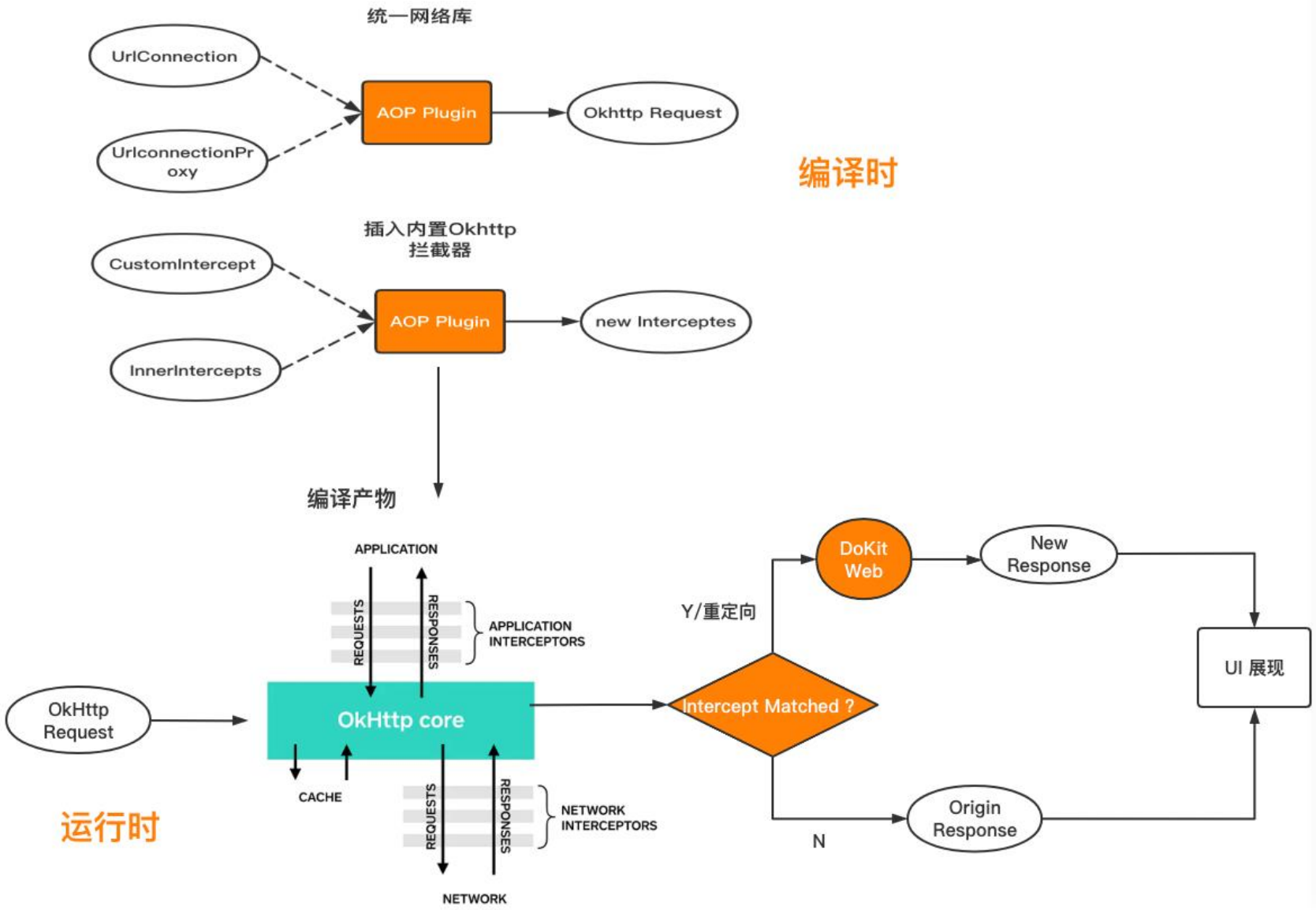
问题

1. 不支持多人协同
2. 不支持场景化
3. 不支持数据持久化
4. 抓包工具繁多，有操作门槛

难点

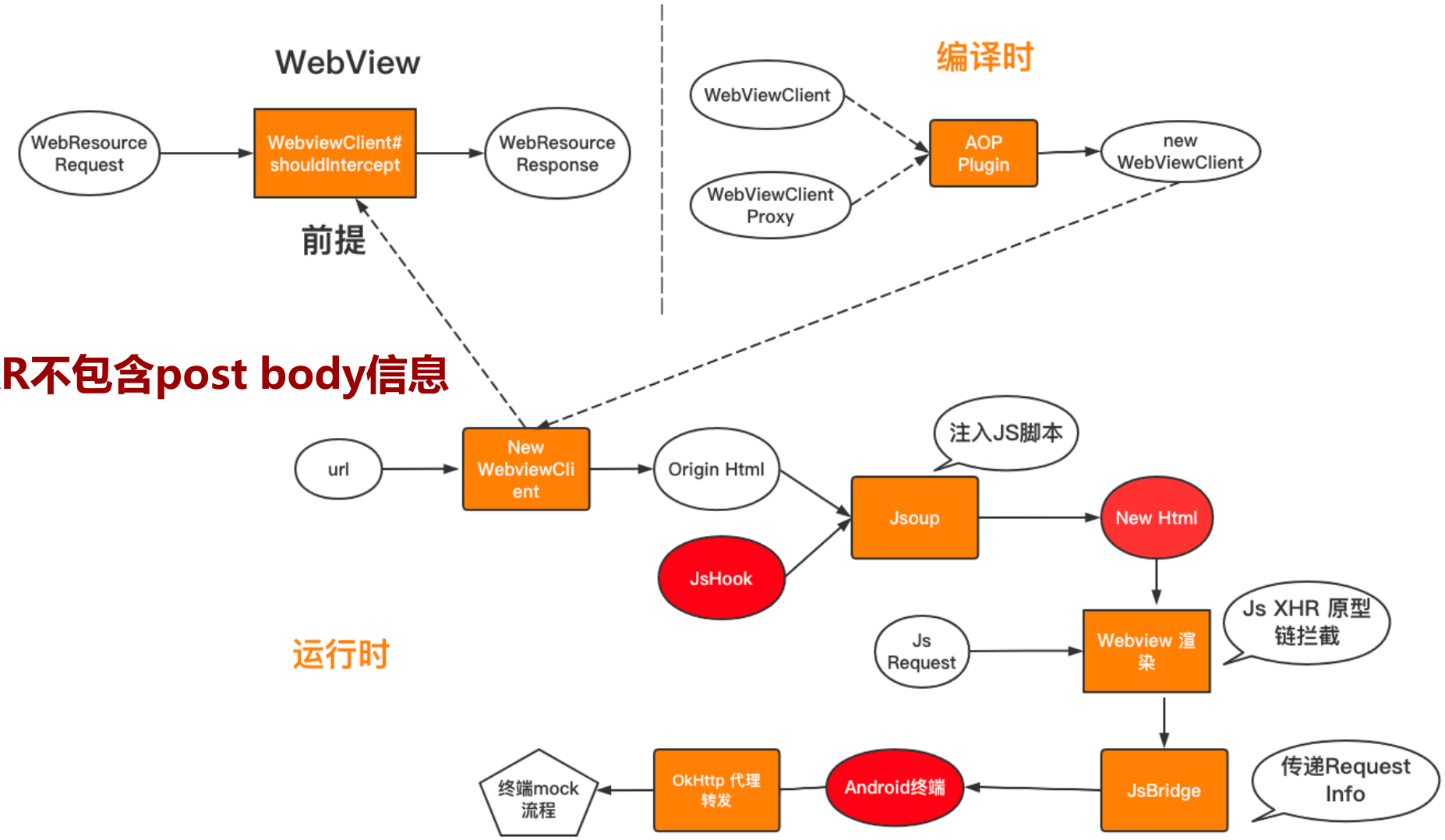
1. Android端网络框架繁多
2. 保证业务代码零侵入
3. 终端无法拦截Ajax请求

落地场景--接口Mock（终端）

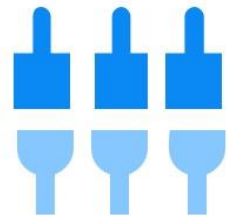


思想

落地场景--接口Mock（H5）



落地场景--接口Mock&业务赋能



打造一款面向全平台的接口Mock方案

收益

1.统一工具，降低学习成本

2.多人协同，提升研发效率100%

3.代码无污染，保证线上交付质量

DoKit Part02思考

北京大学 PEKING UNIVERSITY

1、技术方案如何选型?

2、技术方案可行性调研

3、投入产出比如何抉择

北京大学
PEKING UNIVERSITY

Part 03 社 区 运 营

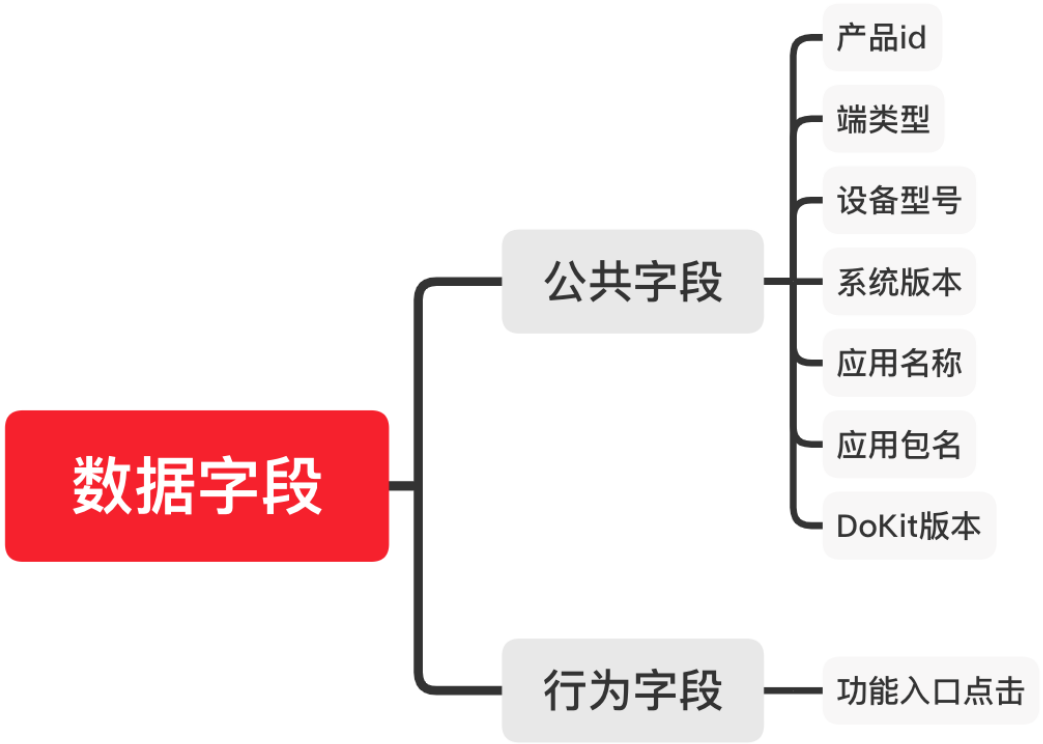
互联网运营框架


用户触达
运营策略
数据指标
数据信息
PUSH/个性化推荐
渠道运营
内容运营
用户运营
活动策略
拉新指标
活跃指标
留存指标
转化指标
传播指标
用户信息
行为数据
消费数据
外部数据

DoKit运营框架


用户触达
运营策略
数据指标
数据信息
产品完善度&口碑&影响力
渠道运营
内容运营
用户运营
活动策划
拉新指标
活跃指标
留存指标
转化指标
传播指标
App信息
行为数据

DoKit运营框架--数据字段

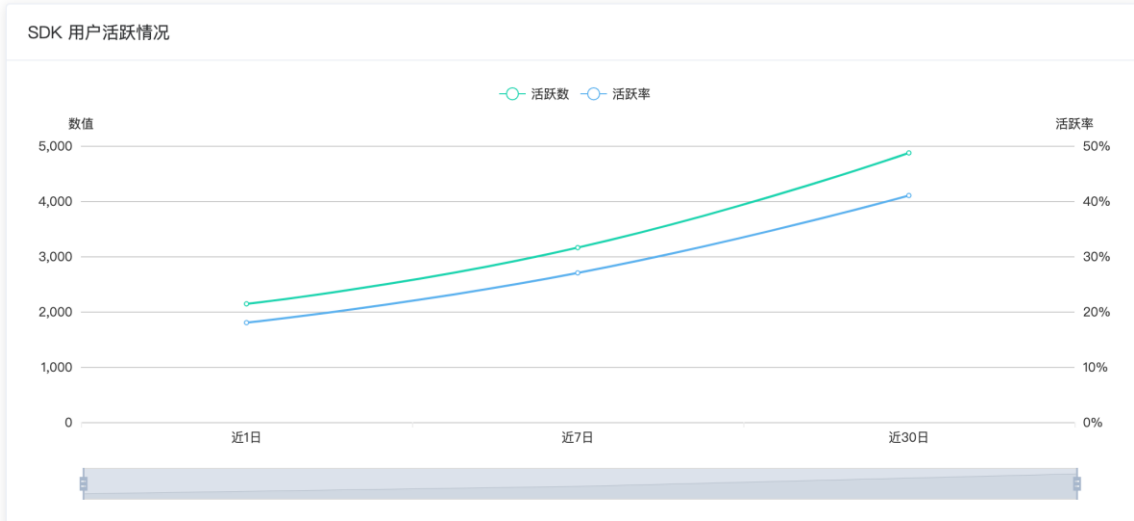


原则：公开、透明

DoKit运营框架--数据指标

北京大学 PEKING UNIVERSITY

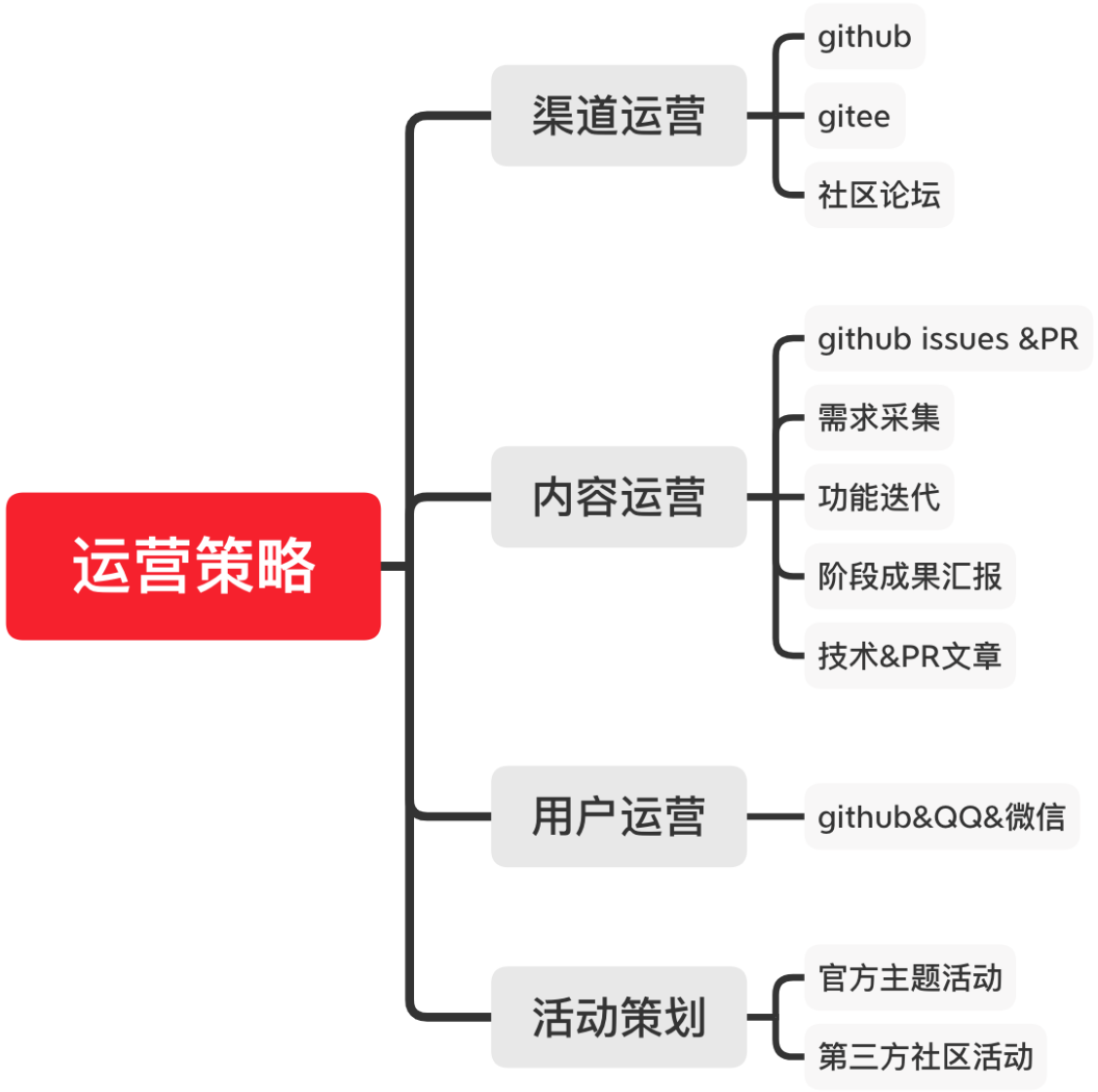
终端SDK活跃



热门功能

eventname	_c1
dokit_sdk_home_ck_entry	827390
dokit_sdk_business_ck	318477
dokit_sdk_comm_ck_h5	28762
dokit_sdk_performance_ck_network	26856
dokit_sdk_comm_ck_log	22962
dokit_sdk_comm_ck_sandbox	15972
dokit_sdk_comm_ck_cache	14678
dokit_sdk_platform_ck_mock	10845
dokit_sdk_comm_ck_appinfo	10367
dokit_sdk_ui_ck_border	9885
dokit_sdk_comm_ck_weaknetwork	7674
dokit_sdk_comm_ck_userdefault	7573
dokit_sdk_performance_ck_fps	7210
dokit_sdk_comm_ck_crash	7169
dokit_sdk_performance_ck_arm	7137
dokit_sdk_performance_ck_block	7041

DoKit运营框架--运营策略



DoKit运营框架--运营示例

北京大学 PEKING UNIVERSITY

PR稿

社区合作

活动策划

DoKit Kotlin&Swift社区改造

DoKit社区分享季

DoKit社区活动:提PR赢奖品

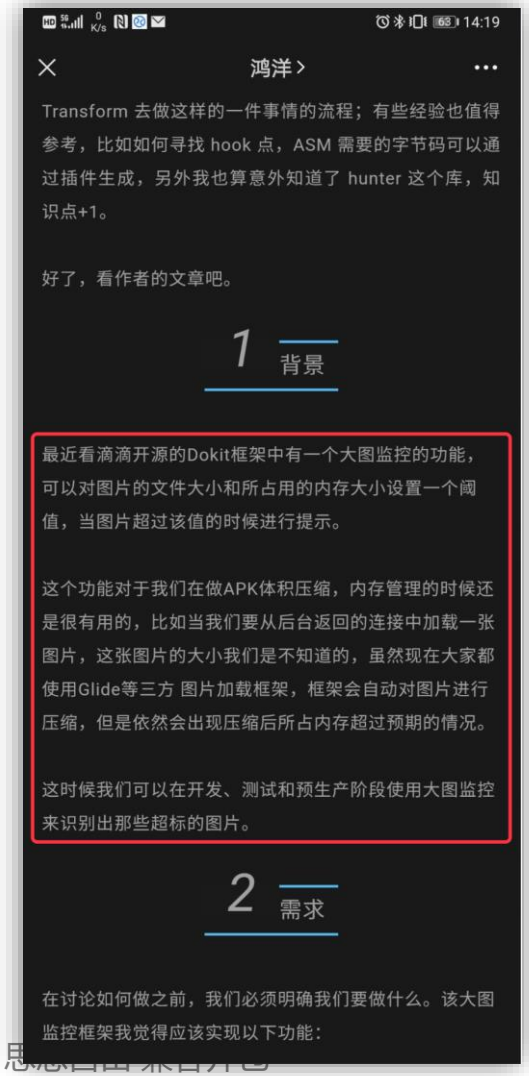
社区维护

QQ、微信群维护

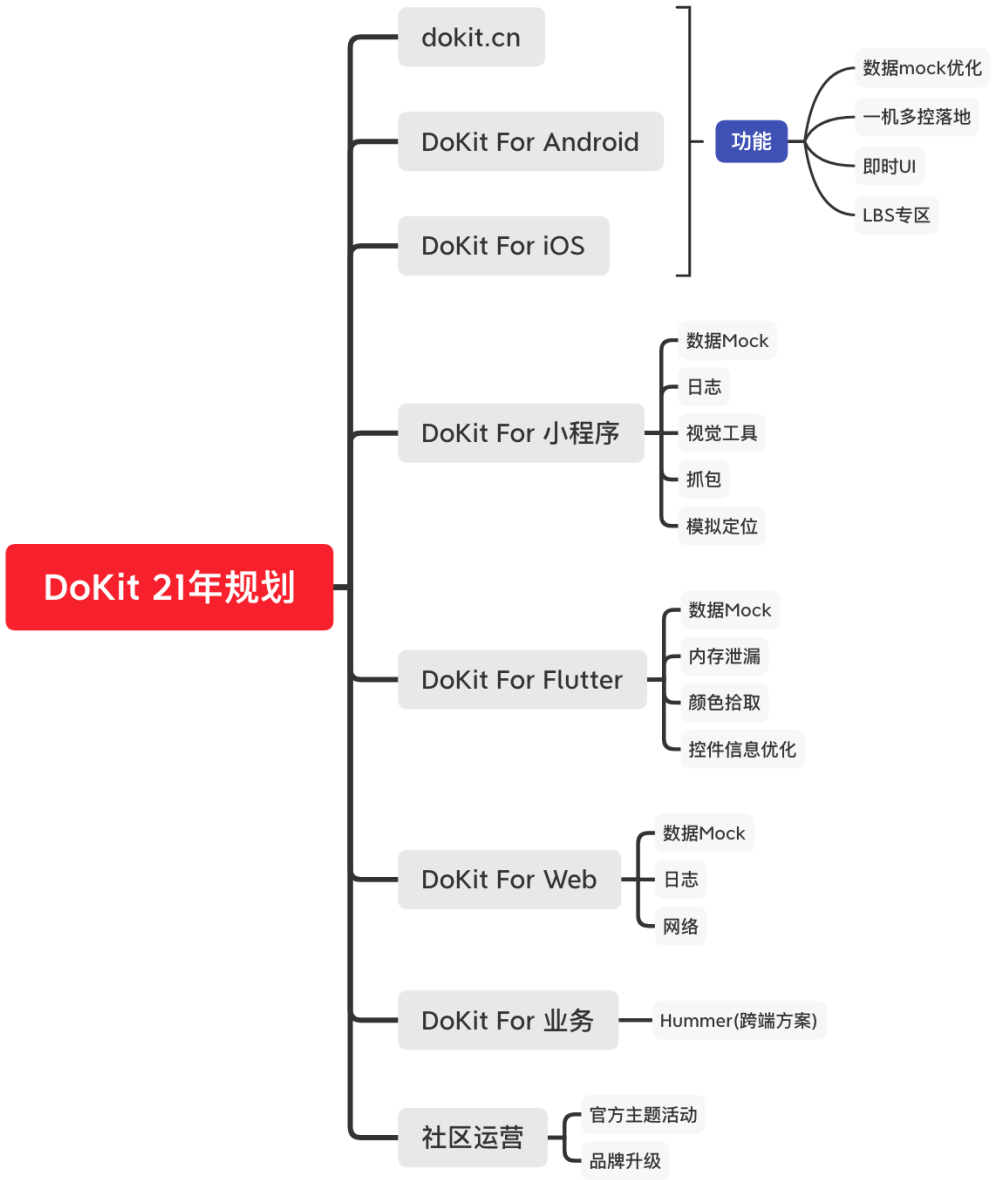
Github Issues & PR处理

DoKit运营框架--成绩

影响力&口碑



DoKit 21年规划


DoKit 21年规划
dokit.cn
DoKit For Android
DoKit For iOS
功能
数据mock优化
一机多控落地
即时UI
LBS专区
DoKit For 小程序
数据Mock
日志
视觉工具
抓包
模拟定位
DoKit For Flutter
数据Mock
内存泄漏
颜色拾取
控件信息优化
DoKit For Web
数据Mock
日志
网络
DoKit For 业务
Hummer(跨端方案)
社区运营
官方主题活动
品牌升级

DoKit开源经验总结

北京大学 PEKING UNIVERSITY

个人

- 是否真的热爱开源这件事?
- 你希望通过开源获得什么?

项目

- 尽量把它当做工作，而不只是兴趣爱好
- 做好规划、评估价值
- 持续迭代优化、及时响应
- 社区维护和运营

技术

- 请尽量少的依赖三方库
- 做好基础架构、高扩展、目录结构清晰
- 做好代码审核保证质量
- 保持技术好奇心、扩展自身技术边界

Part 04 交流 & 提问