

支付宝钱包开发框架

王卫星

目录



- ■APP都是怎么玩的
 - ■以前APP的玩法
 - 当下的玩法
- 支付宝开发框架介绍
 - 支付宝钱包面临的挑战
 - ■基于框架的开发模式
 - ■插件化动态部署
 - 性能优化
 - 稳定性



以前的玩法

APPS



神童



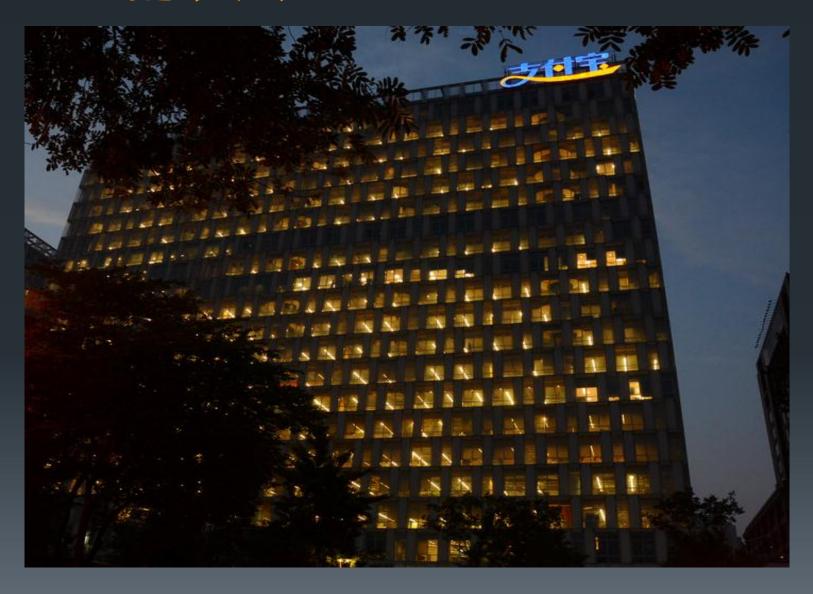
一个人的持久战?





现在的玩法





成熟的APP开发框架



WebApp框架:

- Sencha
- phoneGap
- AppCan

Native框架:

- Parse.framework
- avoscloud

机型款

式多

- OS平台、ROM版本
- 机型的性能 (CPU、RAM、屏幕大小不一)
- 网络能力 (2G、3G、wifi)、电量、流量、内存

部署成

本高

- 版本更新慢
 - •-包大小,网速
 - - 老版本维护成本高
 - •- 小白用户不会升级
- •渠道众多,推广成本高
 - - 厂商预装,菜市场,主站下载

支付宝 Alipsystom

支付宝钱包产品架构

基础技术

应用中心

単规

资产

安全

框架

TODO

账单列表

银行卡

安全等级

基础服务

个人应用中 心

交易详情

鉴权产品

工具

商户应用

对账单

手机

卡券

账户管理

基金

设备管理

航旅

表付宝 Alipayanan

现在面临的挑战



•多团队并行开发 高效

快速上线

- 每月发布一个新版本
- 变化频繁的业务能快速、及时 上线

产品稳定

• 保证快速上线的 同时需要保证产 品的稳定性

表付宝 Alipey.com

框架的技术目标

- 客户端状态及性能监测
- 客户端异常恢复
- ・异常上报

稳定

- ・各个业务高度独立
- ・第三方业务快速接入
 - 动态推送

扩展

质量

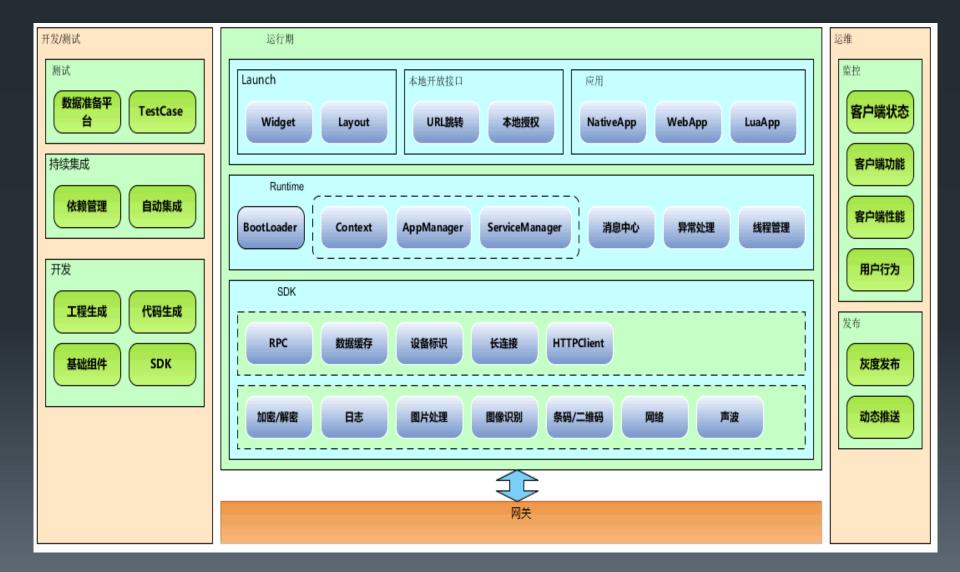
- Athrun
- ・自动化测试框架

效率

- ・业务并行开发
- 提供必要的基础服务

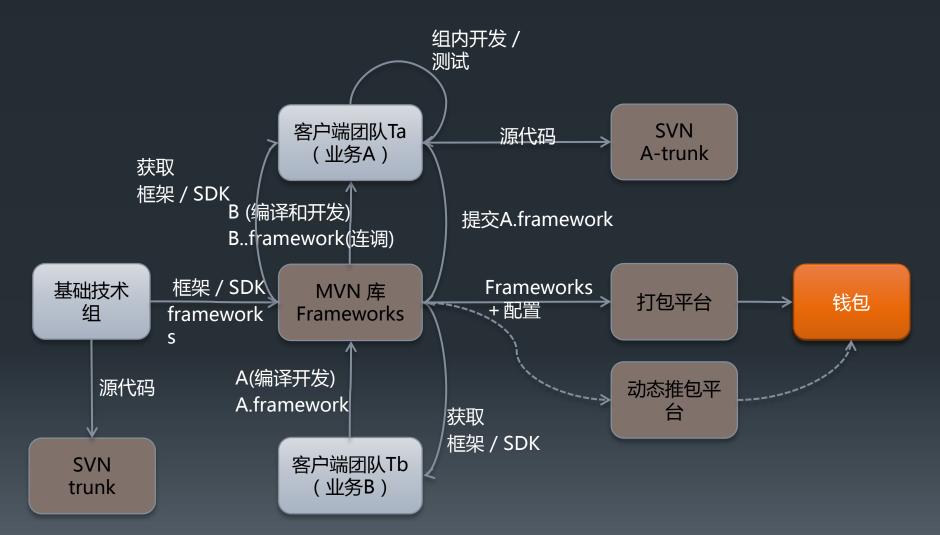
整体的架构





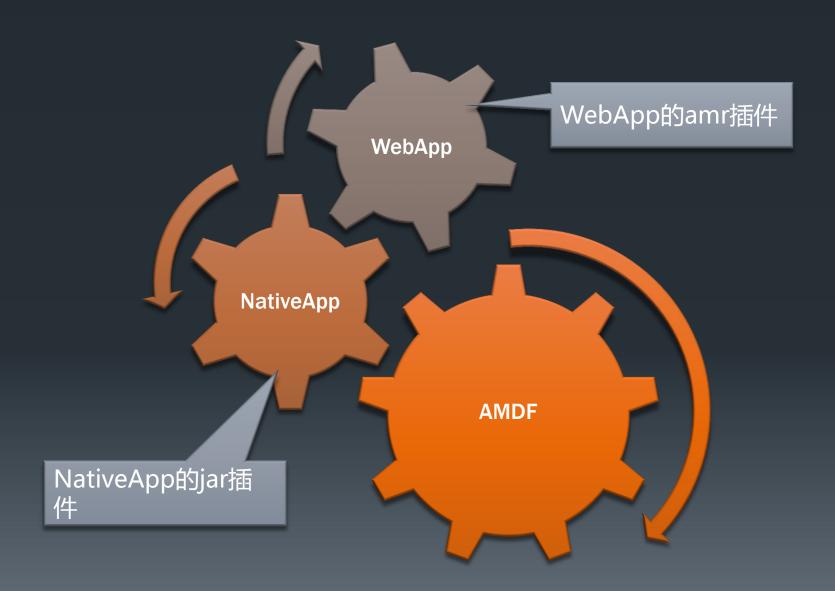
基于框架的开发模式





插件动态部署





webApp





性能优化



数据缓存

•图片加载

网络优化

RPC

性能监控

• 自动埋点





缓存策略:

内存

磁盘

网络

适配策略:

原始大小

特定几种大小

实时生成

RPC













自动代码生成



RPC - 优化



Keep-Alive

• https提升50%

Gzip压缩

• 压缩率:50%+

ETag缓存

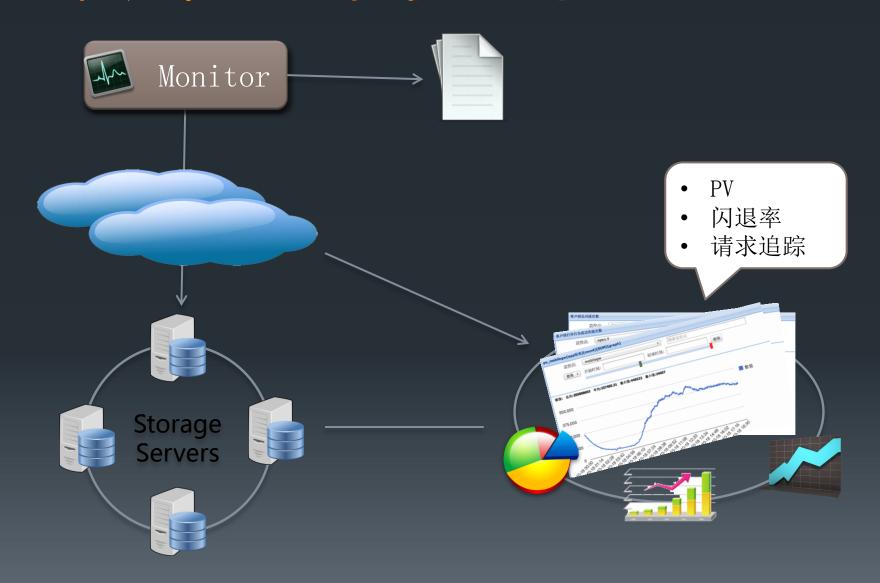
- ETag / If-None-Match / 304
- 命中率: 31.02% / 节省流量: 6096.47MB

批量请求

- 发送时多请求合并
- •接收时分次多次返回

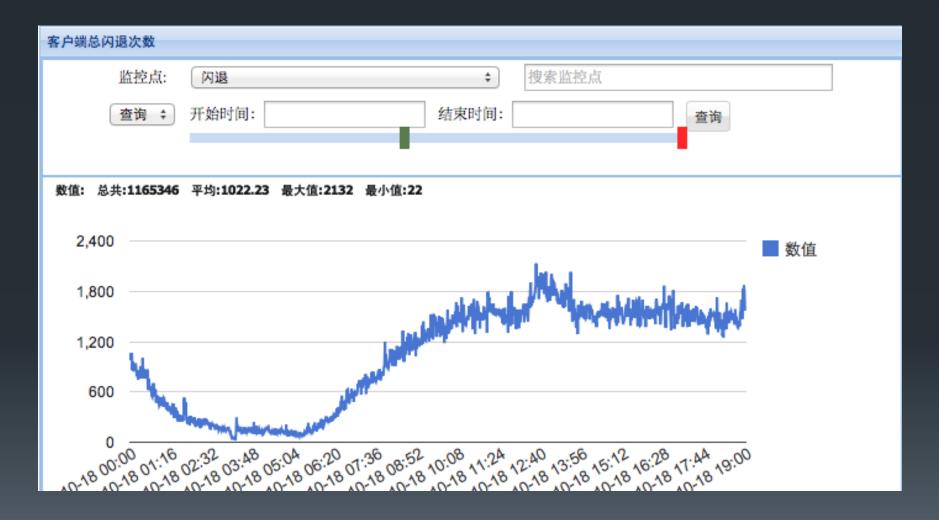
之 Alipsyseom

稳定性 - 运行状态监控



稳定性 - 运行状态监控













控制台

测试环境: 发送界面截图

线上环境:发送点击事件 [UIApplication sendEvent:]



界面截图



界面还原

自动埋点





集点	集点信息		埋点	埋点刷新	
序	seedid	采集点名称	采集配	采集配置	
1	COCAddView00001		○#	关	
2	COCAddView00013	yikatongchengshi	● 开	×	
3	COCAddView00014		○# (•×	
4	COCAddView00015		○# (•¥	
5	COCAddView00016		○# (• ≭	
6	COCAddView00017		О# (•	
7	COCAddView00018		О# (•×	
8	COCAddView00019		○# (• ≭	
9	COCAddView00020		○π (•¥	
10	COCAddView00021		○# (•¥	
11	COCAddView00022		○# (• ≭	



谢!

微博: 支付宝祝枝山

来往: 祝枝山(王卫星)

Csdnid: wangweixing2000