

# 从分层复用到自动化测试——看美团客户端架构的演变

梁士兴

美团点评 研究员





# QCon

## 全球软件开发大会

# 成为软件技术专家 的必经之路

### [北京站] 2018

2018年4月20-22日 北京·国际会议中心

# 7折

购票中, 每张立减2040元

团购享受更多优惠



识别二维码了解更多



# AiCon

全球人工智能与机器学习技术大会

助力人工智能落地

2018.1.13 - 1.14 北京国际会议中心



扫描关注大会官网





# 极客时间

重拾极客精神·提升技术认知

## 下载极客时间App

获取有声IT新闻、技术产品专栏，每日更新



扫一扫下载极客时间App



# SPEAKER INTRODUCE



## 梁士兴

美团点评 研究员

梁士兴，2009年毕业于北京航空航天大学。毕业后在IBM中国研发中心工作了5年。2014年7月份加入到美团点评，现任职为技术研究员。

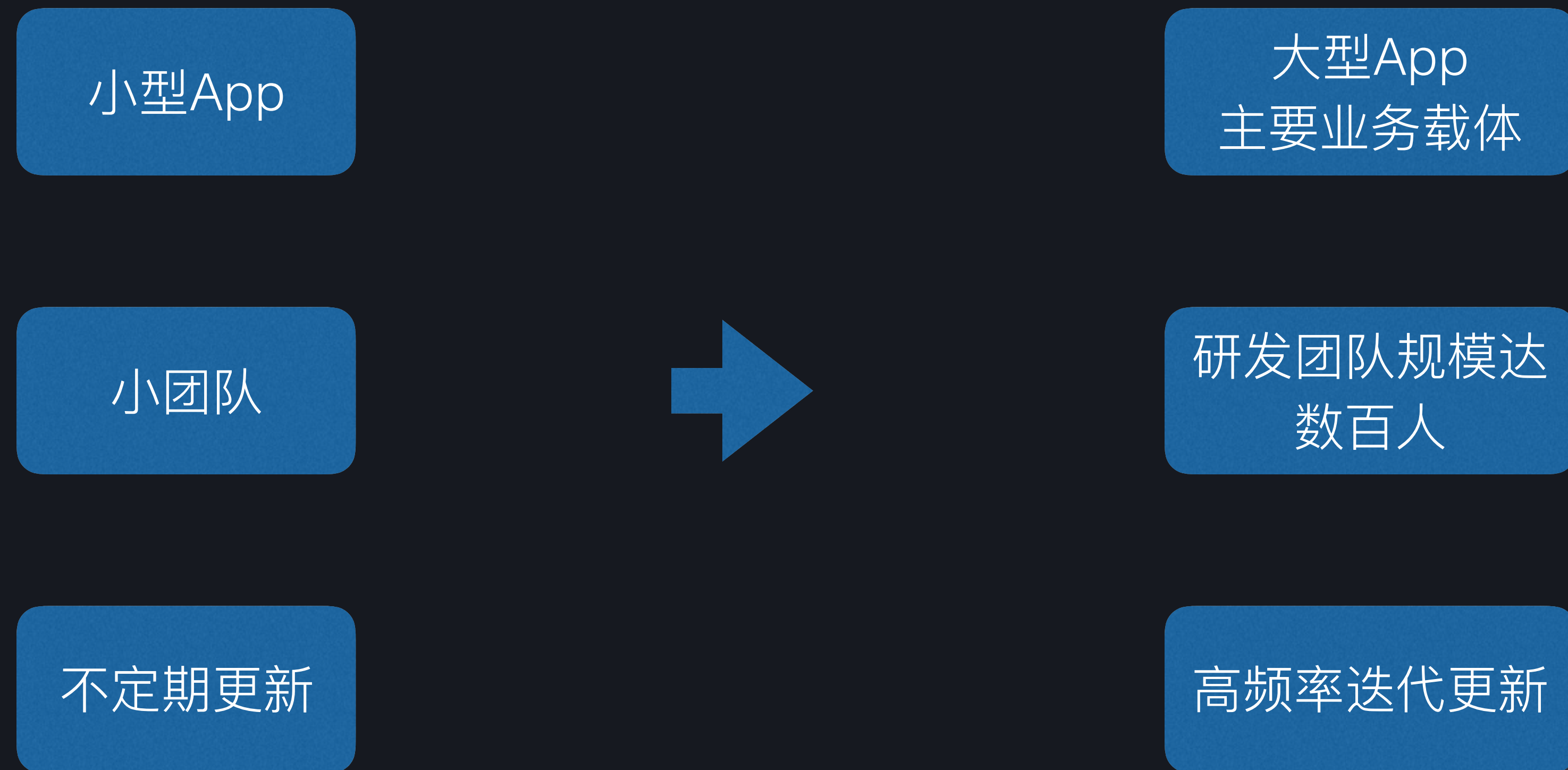
美旅基础服务平台负责人，经历了美团客户端业务架构演变的全部过程。因而对客户端的业务架构有了进一步的理解和思考，愿意与大家进行分享。

# TABLE OF CONTENTS 大纲

---

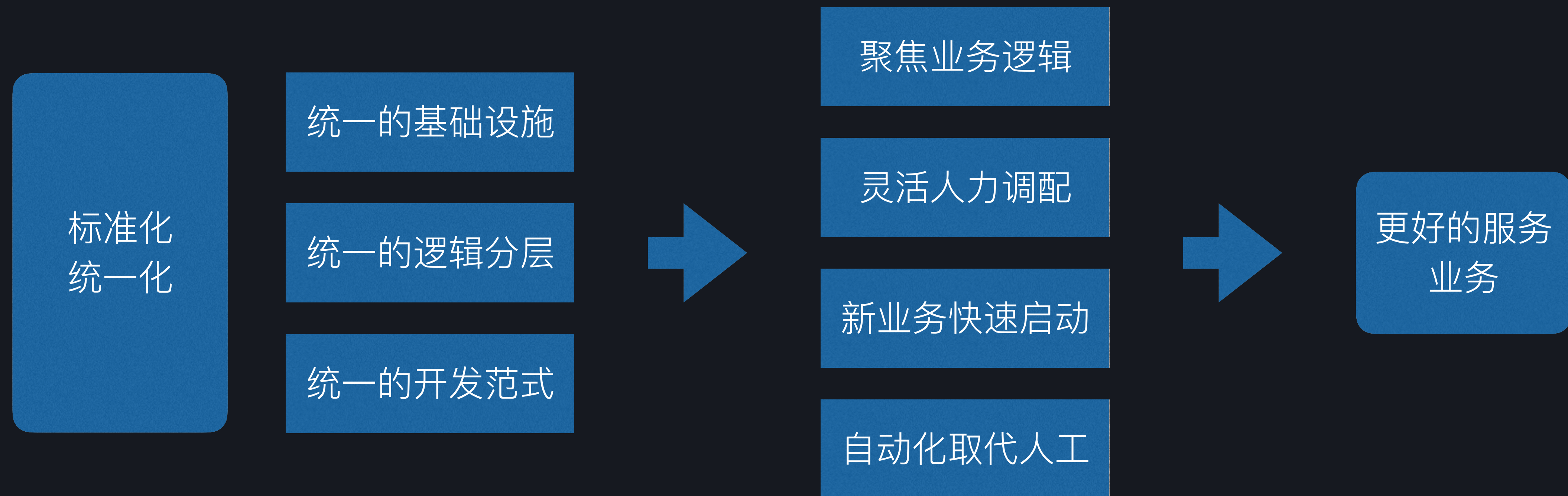
- 为什么要建设客户端架构
- 美团客户端架构演变的三个阶段
  - 原始状态—大团购模式
  - 业务隔离 & 基础设施、开发范式、多应用复用
  - 工程自动化
- 未来的发展方向

# 客户端业务架构的背景





# 客户端业务架构的收益





# 美团客户端的技术体系

## 研发阶段

## 线上阶段

业务逻辑

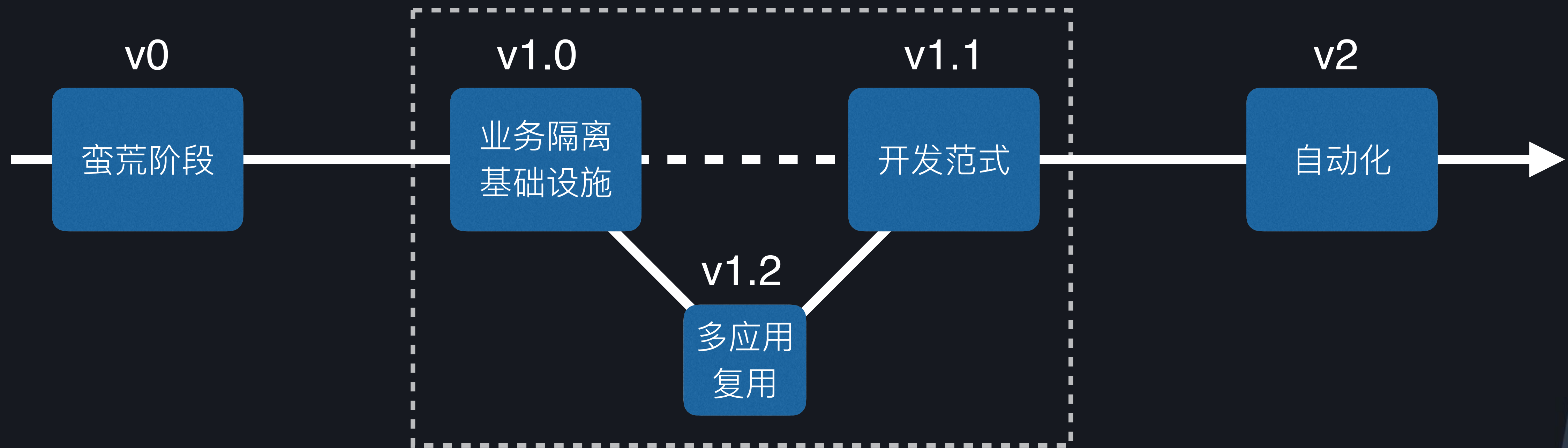
开发范式

技术栈





# 美团客户端架构发展的3个阶段

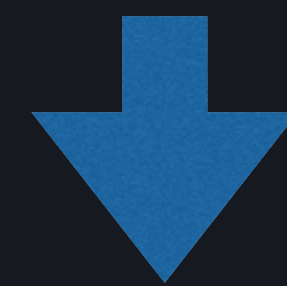




# v0. 蛮荒阶段



- 万物皆为团购
- 最“经典”的架构形式
- 单一研发团队



- 业务遍地开花
- 移动支付占比超过 90%



# 面临的挑战

业务

业务差异化—需要深度定制

团队规模急速扩张

协同效率

研发团队拆分到各个业务

不同业务间干扰严重



# v1.0. 解决问题—业务隔离 & 基础设施



- 提供公共基础设施
- 各个业务之间解耦、物理隔离
- 对新业务的接入保持开放

# 阶段性困难与收益

## 痛点

业务解耦 工作量大

开发同时进行重构

## 收益

业务间消除干扰

各业务独立编译

App二进制集成

## 建议

提供标准业务模板

新业务做小白鼠

短期可以接受代码拷贝行为



# v1.0. 复盘

## 进步的地方

**业务层面：**业务间实现了隔离，消除了干扰

**工程层面：**实现了工程标准化，新业务接入方式实现了标准化

**技术层面：**独立编译 + 二进制集成，大幅提升App的构建效率

## 仍需改进

开发层面没有实现标准化，代码复用依然困难

一线团队研发效率并没有显著提升

## 问题本质

要想开发层面的标准化  
需要制定**开发范式**

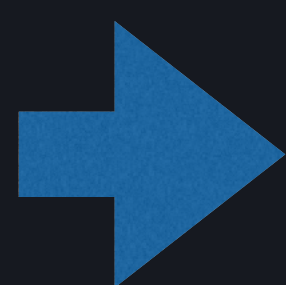
# v1.1. 统一开发范式背景

茴： ①回 ②回 ③回 ④廻 ⑤廻 ⑥廻

茴有六种写法，好像孔乙己也没学全！  
同一需求在客户端的实现方法，恐怕不止六种！



业务模块的接口设计，恐怕要更加混乱！！



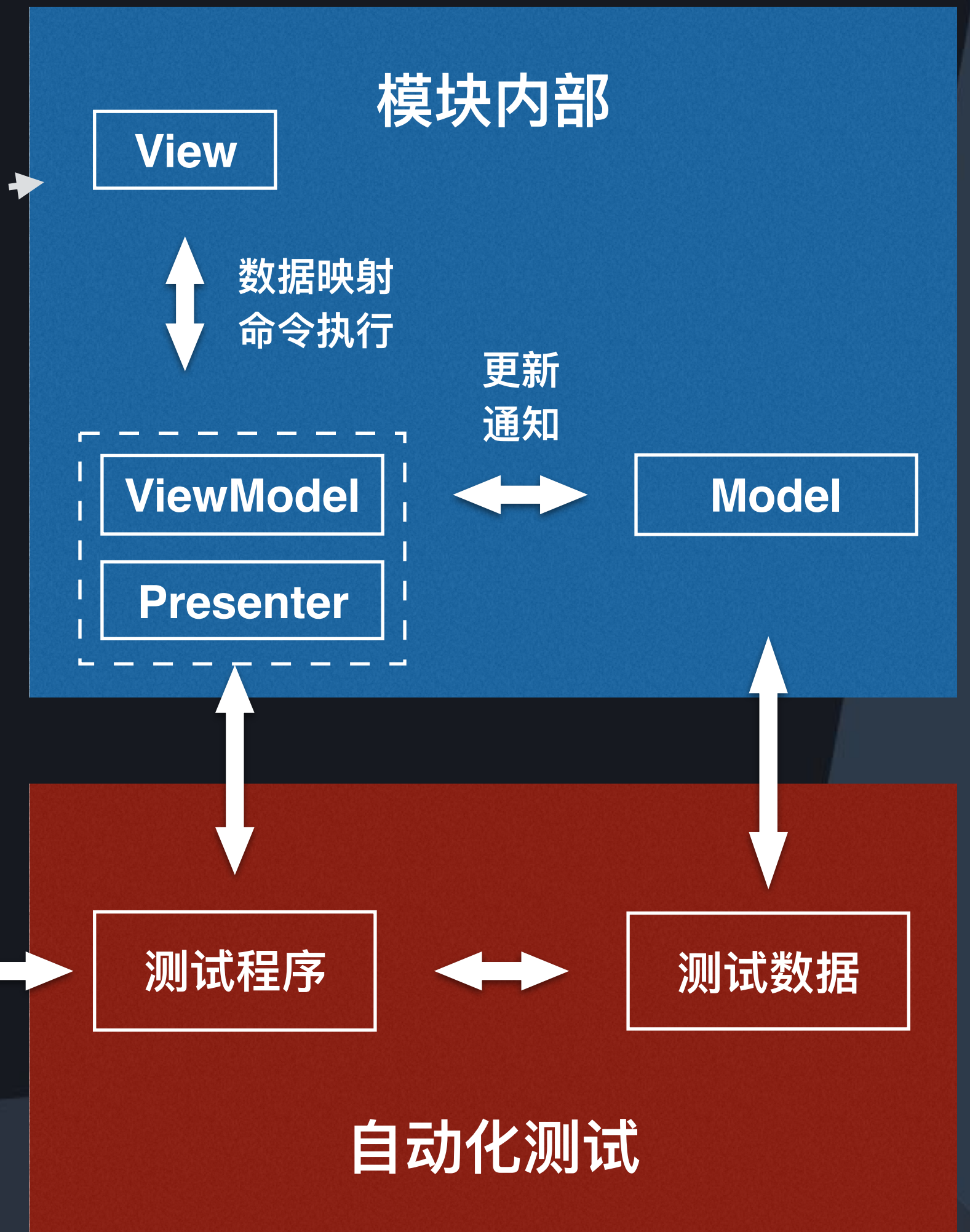
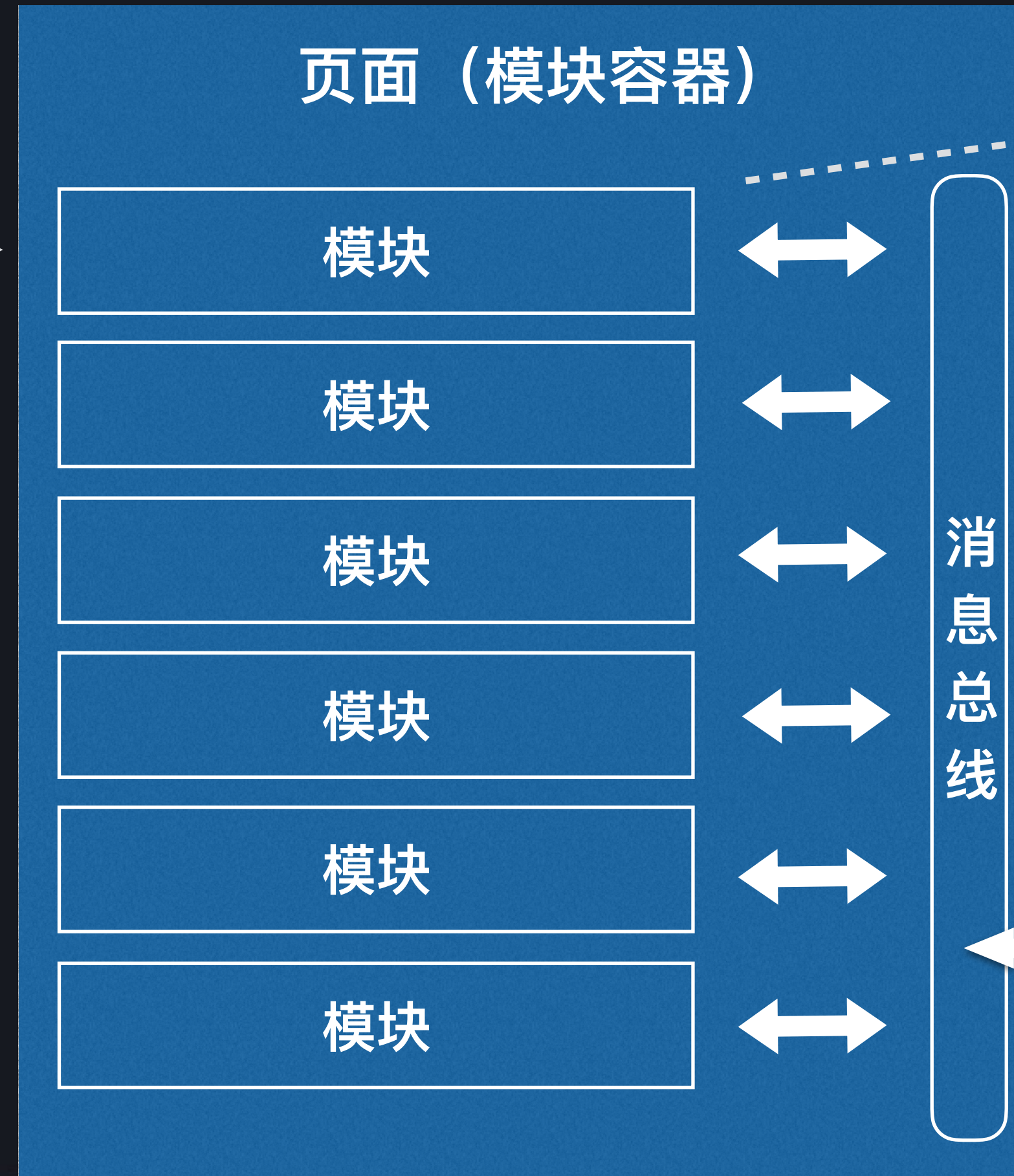
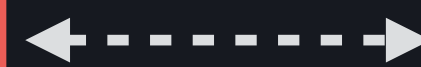
开发范式目标：书同文，车同轨  
实现开发方式统一化，业务模块接口标准化



# v1.1. 统一开发范式实践






UI与模块  
相对应





# v1.1. 统一开发范式收益

对比不同阶段，效率、高阶工程师的变化：

	业务线1 2014.7	业务线2 2016.12	独立App 2017.8
标准工作量/标准人力	X/Y	2X/1.5Y	8X/2Y
效率对比	100%	133%	400%
高阶工程比例			

● 高阶      ● 低阶

高阶工程师需求降低

开发效率明显提升：  
模块复用成为常态



# v1.1. 统一开发范式的困难

## 困难

代码重构成本高

开发同时进行重构

## 建议

新需求涉及的页面同步重构

# v1.2. 与点评合并带来的全新挑战

## 技术栈不统一

业务  
组件

扫码

账户

城市

分享

支付

UGC

基础  
设施

桥协议

布局

图片

离线化

定位

网络

地图

存储

埋点

风控

## 后端服务不统一

公共参数

数据格式

## 交互样式不统一

主题

交互

...

把XX功能整体从美团  
迁移到点评，就改改样式，  
应该很快吧！！



产品经理



# v1.2. 多应用复用



抽象基础服务  
隔离基础设施差异带来的影响



构建过程隔离差异化部分

二进制接入



点评

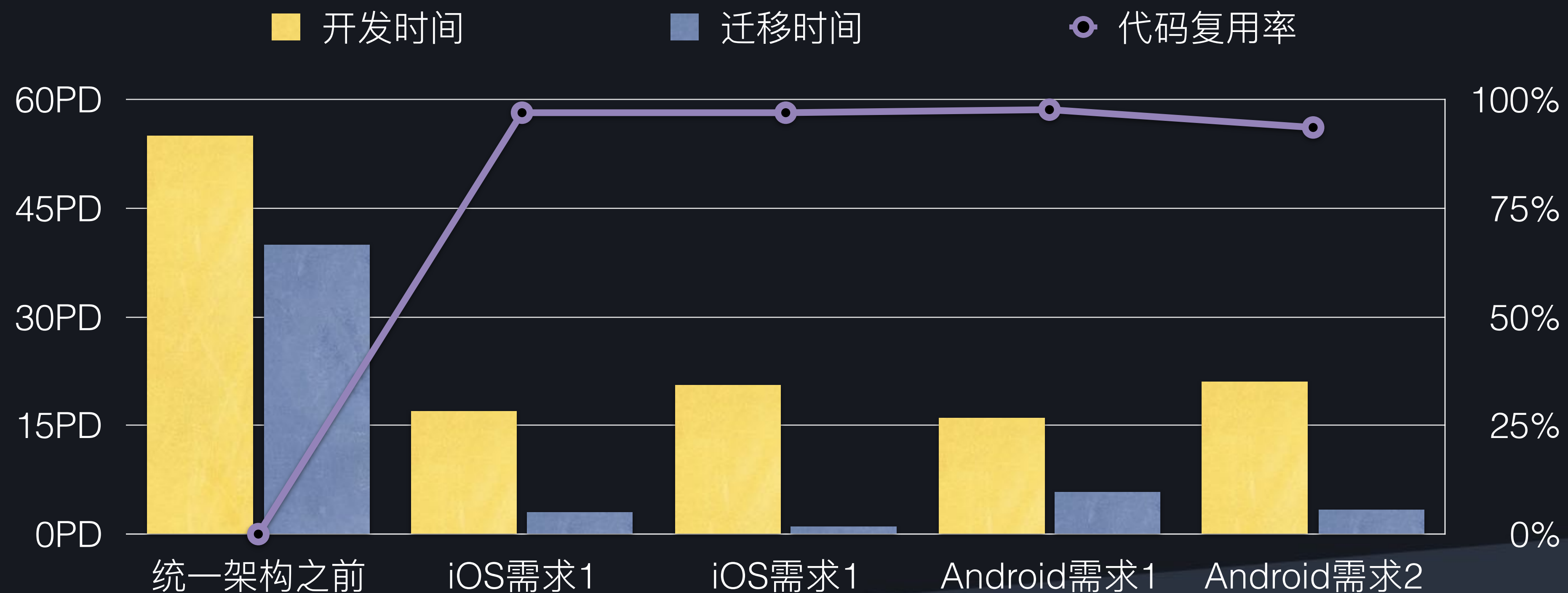


二进制接入



# v1.2. 多应用复用收益

多APP复用需求呈现常态化





# 如何进一步提升研发效率？



# 针对什么做自动化?

迭代周期

测试阶段

新功能测试

**回归测试**

发布前验证

- 新功能测试
- 回归测试
- 发布前验证

测试周期分解



开发阶段

UI开发

业务逻辑开发

网络开发

API联调

- 业务逻辑开发
- UI开发
- 网络开发
- API联调

开发阶段时间占比



- 自动化测试
- 代码脚手架



# 自动化测试方法—业务特点

## 多种入口

首页

搜索

运营活动

...

主流程

列表

详情

购买

售后

## 信息增值服务

攻略

地图

相册

...

## 两种 页面 类型

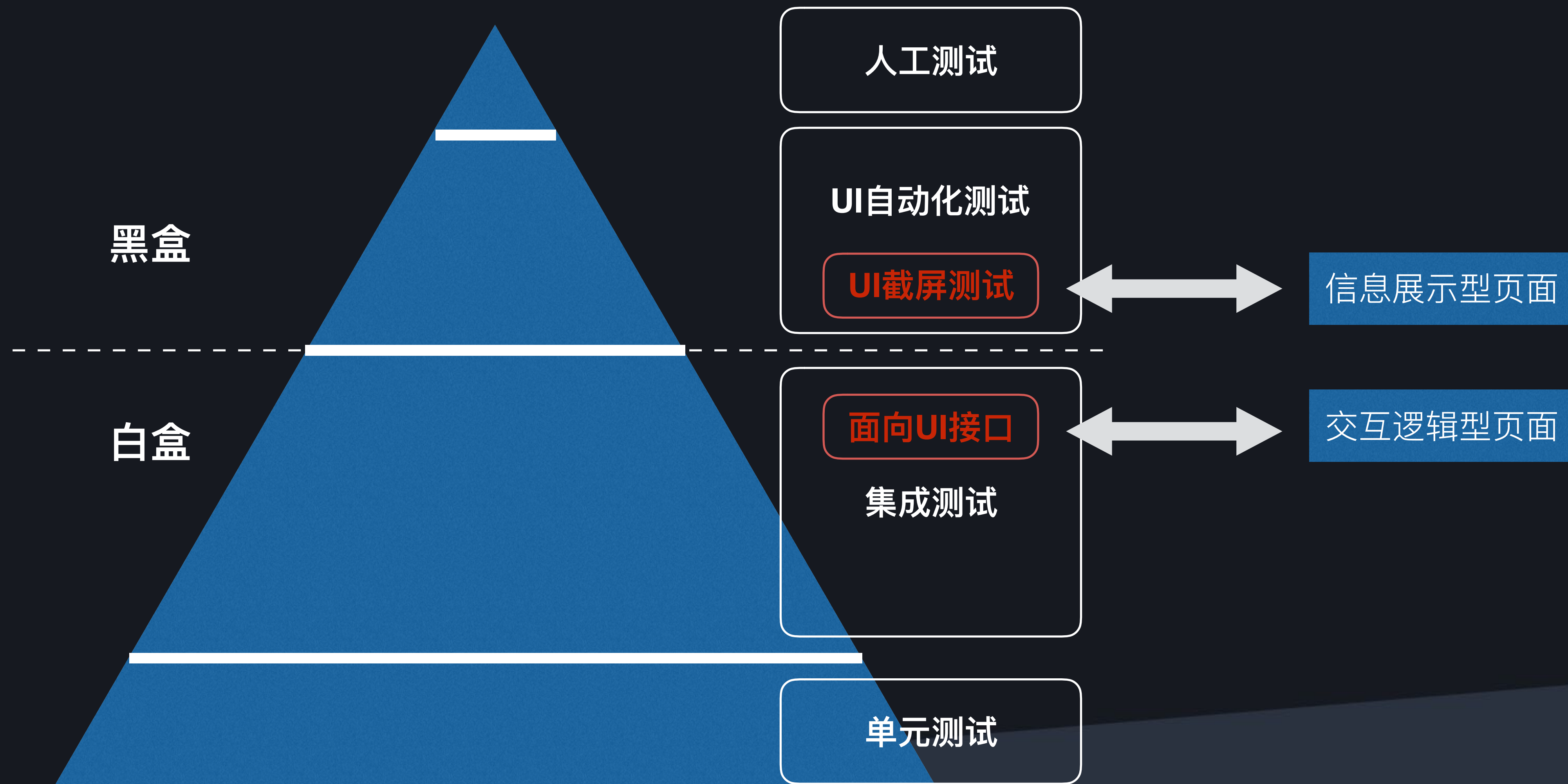
信息展示型页面

占比80%  
测试关注展示行为

交互逻辑型页面

占比20% 通常对质量要求更高  
测试关注程序逻辑

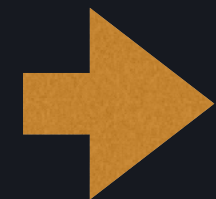
# 自动化测试的方法



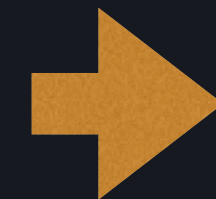


# 自动化测试三部曲

设置上下文环境



模拟用户操作



校验结果

# 截屏测试方案

参考图



实际效果图

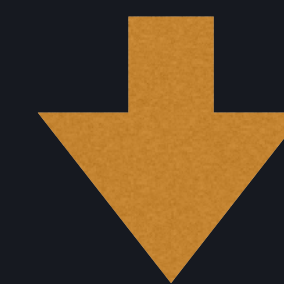


Diff分析



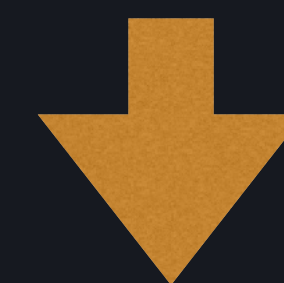
设置上下文环境：

数据打桩，模拟登录层出，  
设置时间.....



模拟用户操作：

展示UI，滚动屏幕，点击按钮.....

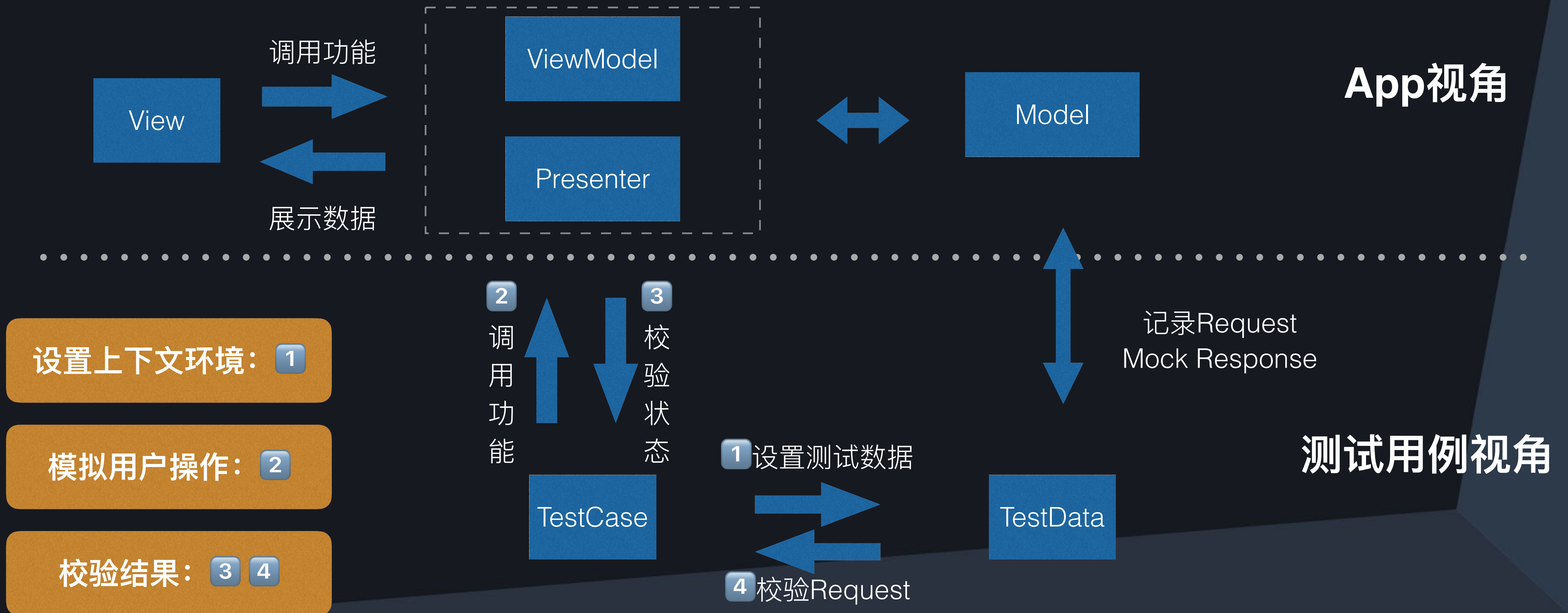


校验结果：

截屏 & 图片比较



# 面向UI接口的集成测试



# 自动化测试的挑战

成本

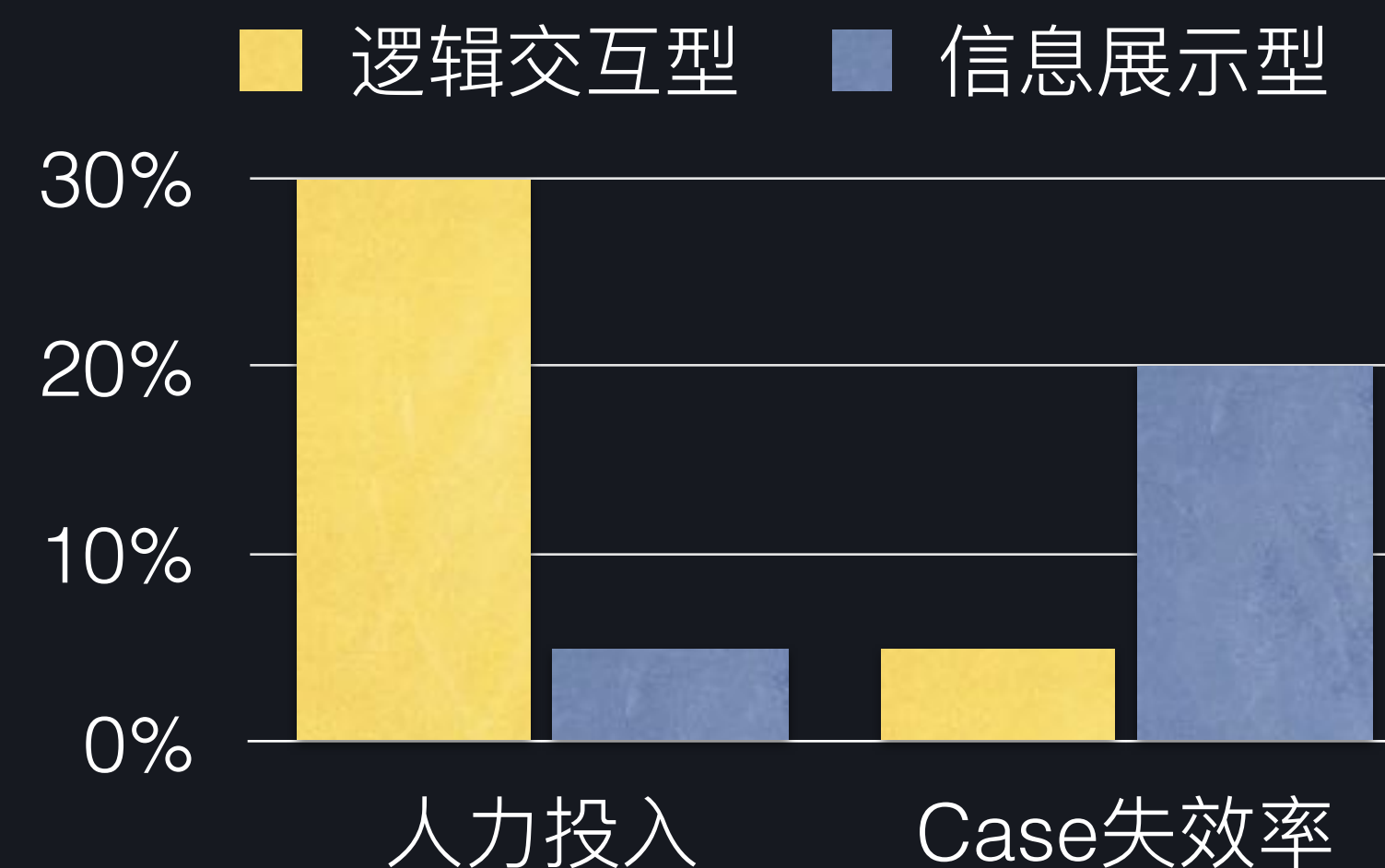
开发测试用例人力投入

测试用例的维护成本

执行效率

CI单次执行时间

CI执行成功率



问题：构建 + 执行 约 30分钟

优化方案：复用二进制构建结果

效果：6分钟

问题：15%概率失败

优化方案：优化CI执行逻辑，引入重试机制



# 自动化测试带来的收益

线上质量提升

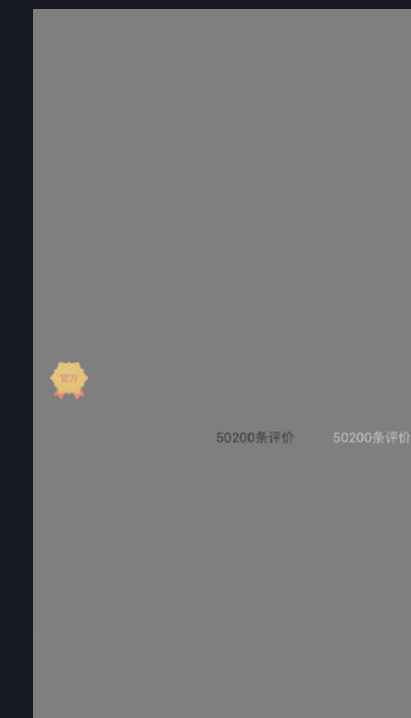
故障数量下降，故障等级降低

研发效率提升

减少了人工测试成本，进而提升迭代频率

# 未来展望

## 场景1： 日常测试



报Bug

更新Case

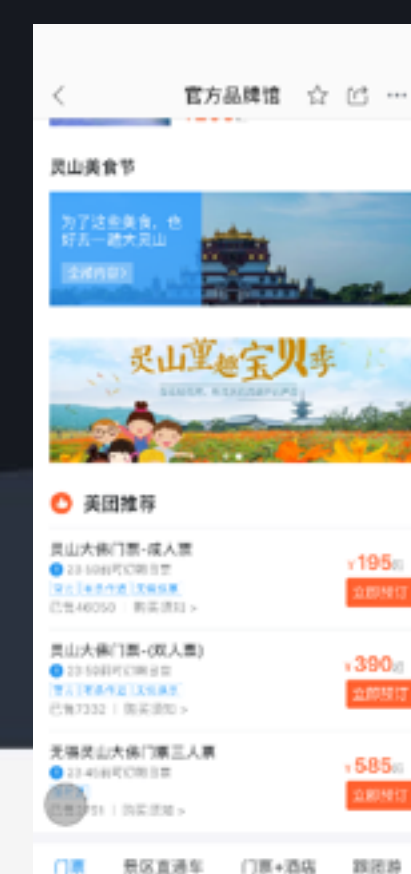
## 场景2： 新需求测试 多机型验证



报Bug

确认Case

## 场景3： 集中测试 新机型



报Bug

确认Case

测试管理平台

代码脚手架



# THANK YOU

如有需求，欢迎至 [ 讲师交流会议室 ] 与我们的讲师进一步交流

邮箱: [liangshixing@meituan.com](mailto:liangshixing@meituan.com)

微信号: liangshixing



梁士兴

北京 海淀



扫一扫上面的二维码图案，加我微信