

LIANJIA.com

# 链家网移动端敏捷之术



LIANJIA 链家 .com



促进软件开发领域知识与创新的传播



关注InfoQ官方信息  
及时获取移动大会演讲  
视频信息



[深圳站] 2016年07月15-16日  
咨询热线: 010-89880682



[上海站] 2016年10月20-22日  
咨询热线: 010-64738142



郭晓铭

链家网移动端架构师

一枚成长中的程序媛

对复杂业务下的架构设计和研发效率提升有浓厚兴趣

愿始终做一个不端不装的技术人

# 内容摘要

业务现状

刀耕火种

开疆拓土

与时偕行

更进一步

# 业务现状



# 链家网移动端业务介绍



连接 \* 数据 \* 服务

买方服务

二手房

新房

学区房

租房

地图找房

百科

...

卖方服务

估价

委托

钥匙托管

销售周报

房屋实堪

房源动态

...

经纪人

房源管理

客源管理

签约

签后

商机获取

金融

...

# 链家网移动端业务特点

## ● 目标用户角色丰富

服务于房产领域的各个角色：用户、业主、经纪人

## ● 业务地域范围广且差异性大

北京、天津、青岛、成都等17个城市，各个城市的业务范围不同

## ● 业务和产品多元化

覆盖二手房、新房、租房、估价、业主委托、钥匙托管、房源管理、客源管理等业务，包含链家App，Link，新房Link，案场等多个App

## ● 线上业务需快速响应线下业务的部署

房产O2O平台，强调线上线下的配合

# 刀耕火种





# 业务简单

- 业务覆盖城市范围小

仅支持北京等少数城市

- 产品用户角色单一

产品仅服务于买方，卖方、经纪人角色非目标用户

- 业务单一，未形成线上线下的闭环

早期仅支持二手房业务，且线上的浏览行为未和线下的经纪人带看等行为打通



1+1

# 团队工作方式传统

- 团队规模小

iOS和Android端各2位，人员密切配合，不存在业务线的分工

- 瀑布式开发

版本需求比较稳定，在开发过程中很少调整

- 手工打包上线

产品投放渠道较少，发版节奏平缓，测试和上线均靠手工打包



# 架构简单

lianjia.com

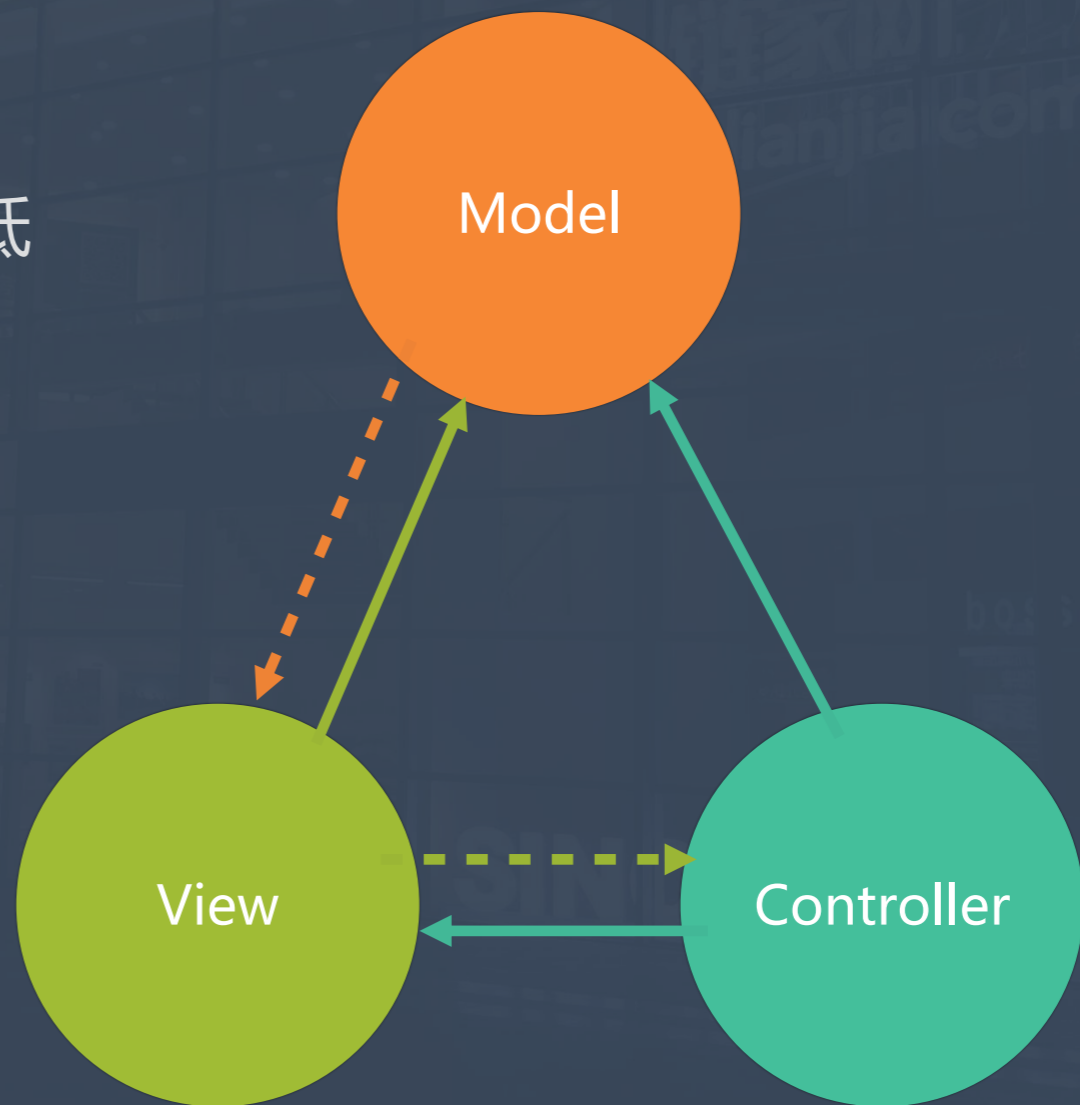
## MVC架构

- 通用设计，学习和维护成本低

MVC架构功能划分清晰；不容易产生误解，开发人员之间沟通无障碍

- 对复杂业务不适应

业务逻辑变复杂时，Controller将变得越来越大



# 开疆拓土



# 需求与挑战

- 地域性业务差异

支持北京、青岛、成都、南京等9个城市，各城市支持的业务范围需根据公司战略随时调整

- 业务增多，内容型业务的形态多变

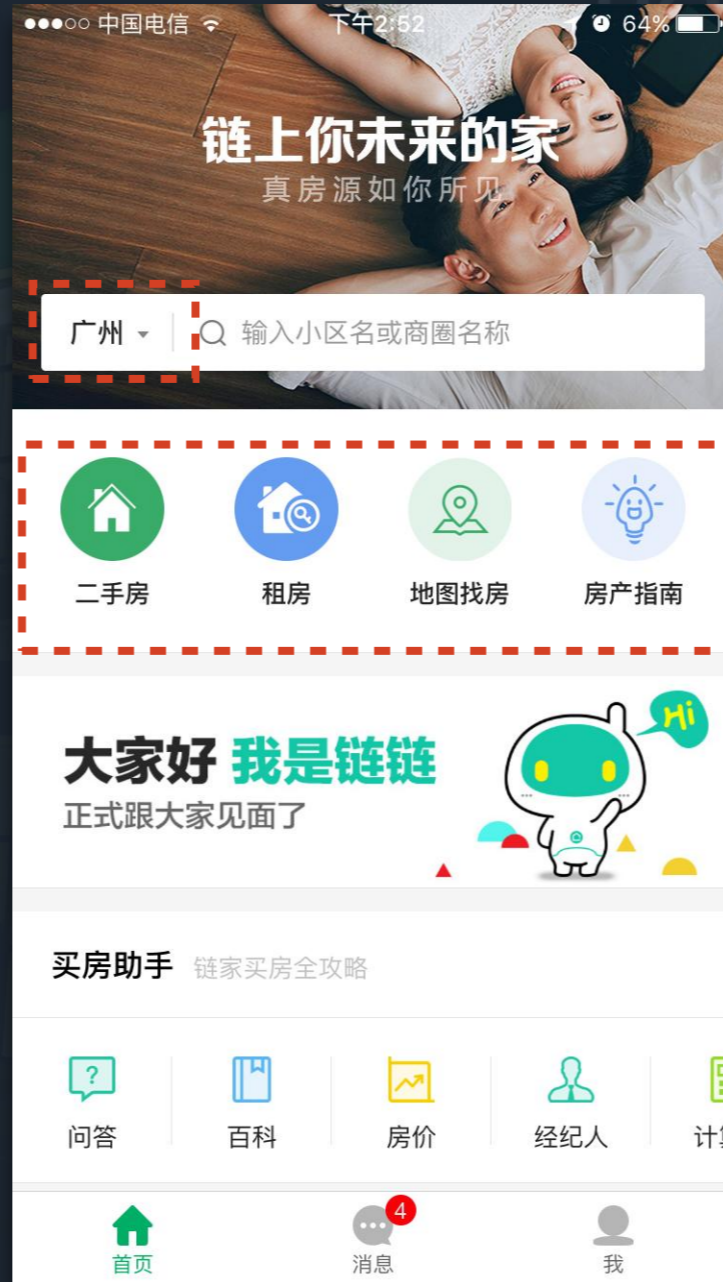
由单一二手房买卖业务增加到新房、租房、百科、问答等业务群，且热点、百科等内容型业务的具体形态多变

- 人员增多，人均产出下降，代码质量堪忧

团队人员增长到3倍，技术水平参差不齐，规范缺失，项目质量缺乏保障



# 地域性业务差异



# 基于短链的配置化

lianjia.com

## 服务端下发城市配置

配置信息包含业务入口icon、标题、跳转短链，功能开关等

## 跳转短链注册表

保存短链pattern与页面的类别、类参数名、短链参数名、默认参数值、跳转方式等的对应信息

## 短链解析和页面跳转

使用注册表中的短链pattern做正则匹配，根据匹配到的信息创建页面并用对应的跳转方式打开

## 配置化

## 更统一

各个业务的解析和跳转逻辑由跳转引擎统一处理

## 更灵活

城市配置由服务端下发，城市业务范围的调整不依赖发版

## 扩展性更强

快速支持新增城市，且对新业务的支持不影响旧功能

# 业务快速上线和调整





# Hybrid App

LIANJIA.COM

## Native



重点业务体验保证

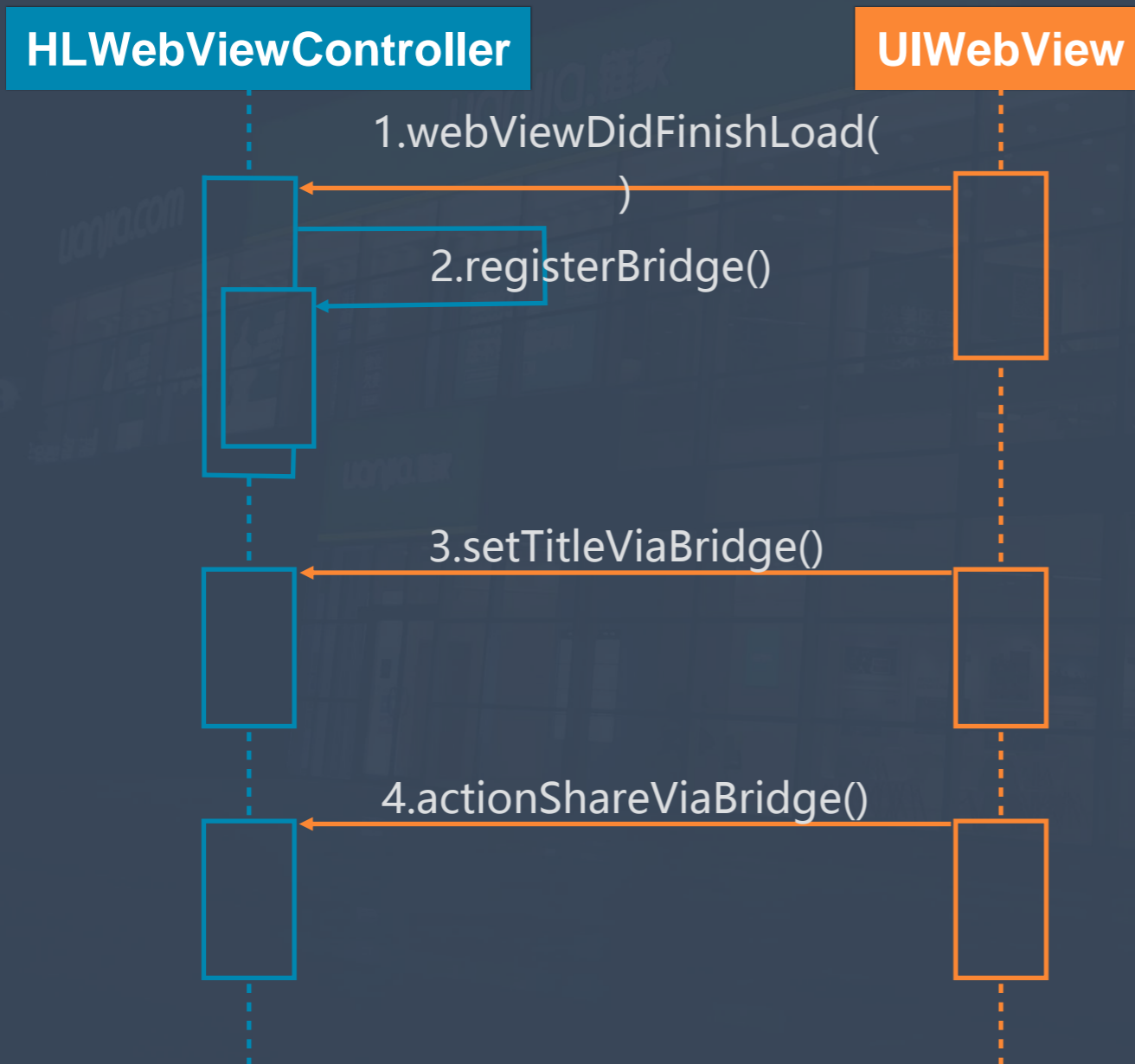
## H5



内容型业务、运营活动



# HLHybridBridge——统一交互方案



iOS平台Native与H5交互时序图

## 很小巧很清晰

不依赖第三方解决方案，交互流程更清晰，学习成本更低

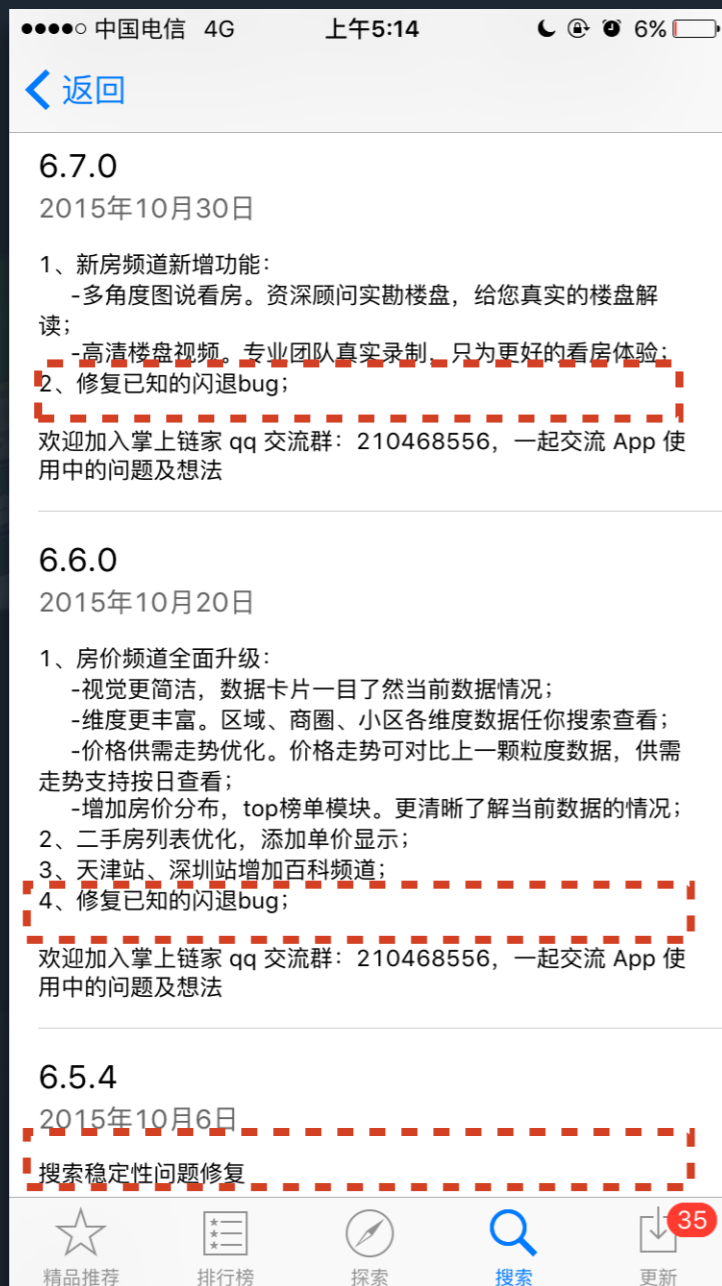
## 信息传递安全

通过Bridge注入参数信息，而不是放在url里

## 两端方案统一

iOS & Android采用统一方案，H5页面开发更加快速

# 项目质量缺乏保障



1 % crash率

# 开发流程优化

## 代码规范形成

代码风格一致，提高可读性；  
统一的入口参数校验、异常处理等，提高健壮性；



## Code Review

同步开发人员对代码和设计的理解；提前发现问题

## 敏捷开发模式

随时交付，提早反馈

# 收益 (一)



0 成本增加新城市



15年3月上线 3 条业务线

# 收益 (二)



0.3% crash率



千行代码bug数 9

# 与时偕行



# 需求与挑战

- **多个业务团队并行开发**

随业务复杂性提升，链家App演变成为多业务团队并行开发模式，需要相应架构支撑

- **多个新产品需快速上线**

计划推出Link、新房Link产品，需要能复制已有功能快速上线

- **对接后端团队越来越多**

沟通成本高，发版风险大；不同团队接口数据格式差异大，客户端数据解析和校验逻辑复杂





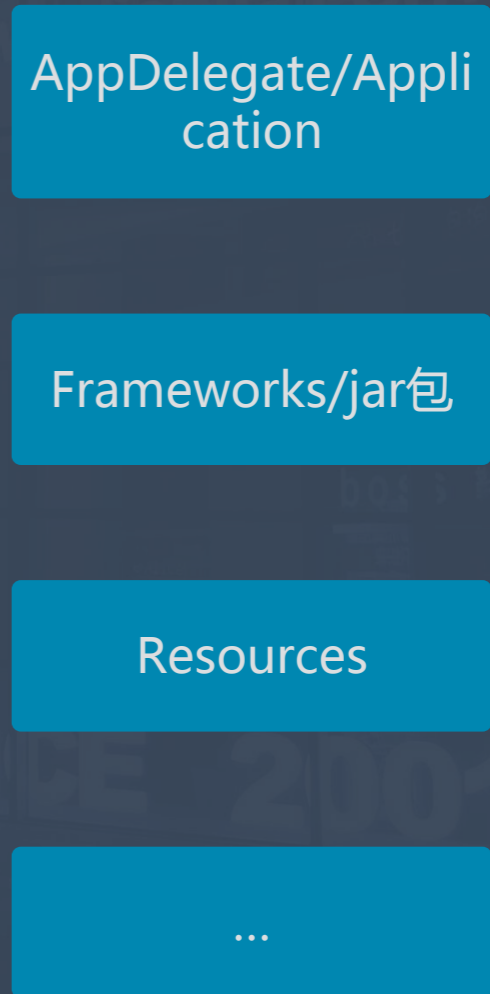
# 组件化

LIANJIA.COM

业务



主工程

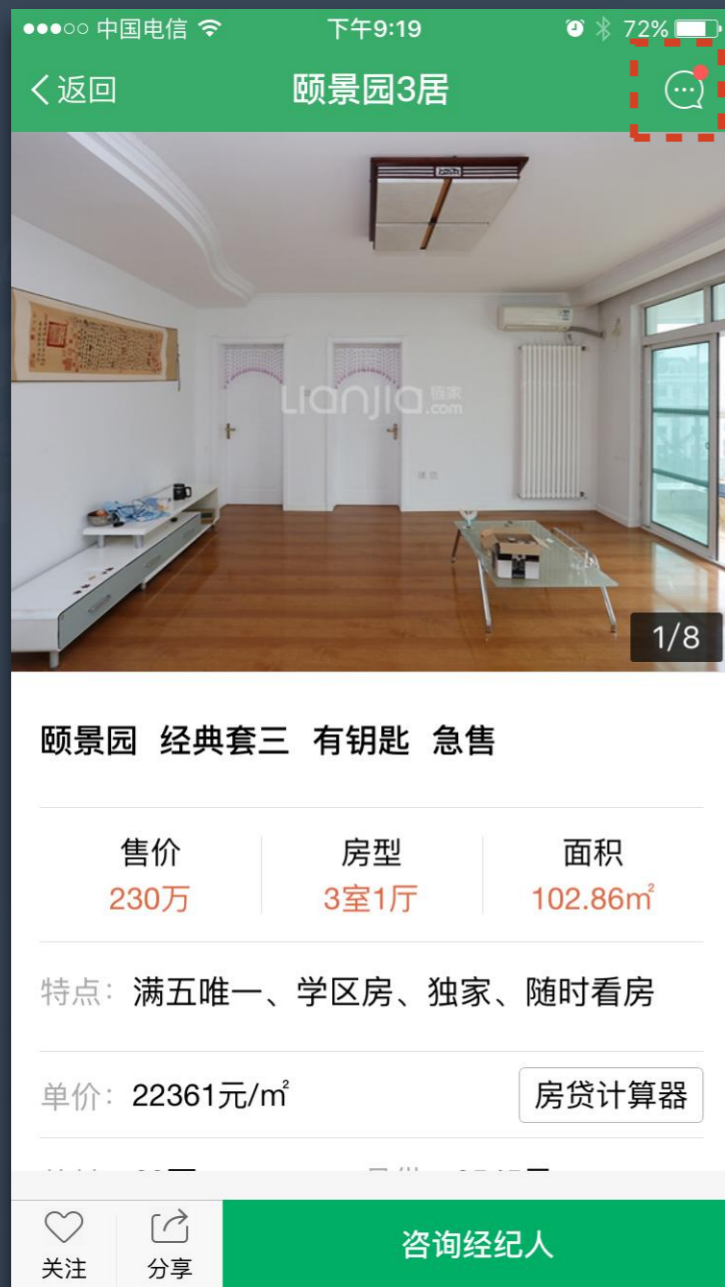


公共模块



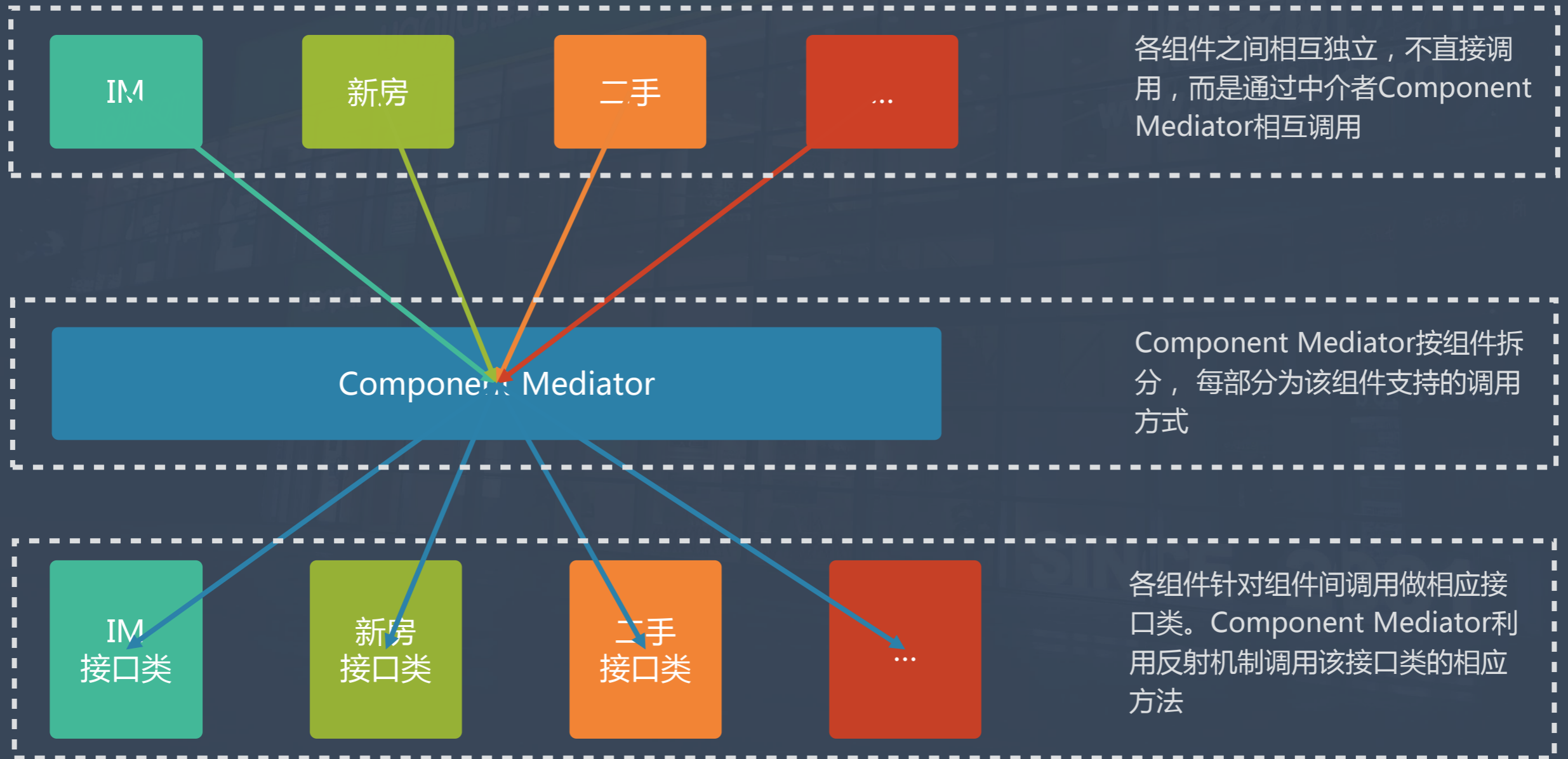
CM:Component Mediator

# 组件间的相互调用



# 组件间的调用方案

lianjia.com



# 组件化过程中的风险控制

lianjia.com

## 代码仓库分离

主工程代码、公共模块代码、以及各业务组件代码仓库分离

## 权限控制

为单个业务团队配置公共模块代码以及其他业务代码的只读权限

## 建立接口类的命名规范

对组件接口类名以及接口方法的命名规范统一，降低开发成本

## 配置化

## 接口类的Code Review

接口类出错的影响范围相对较大，需要业务负责人对接口类做重点review

## 小流量

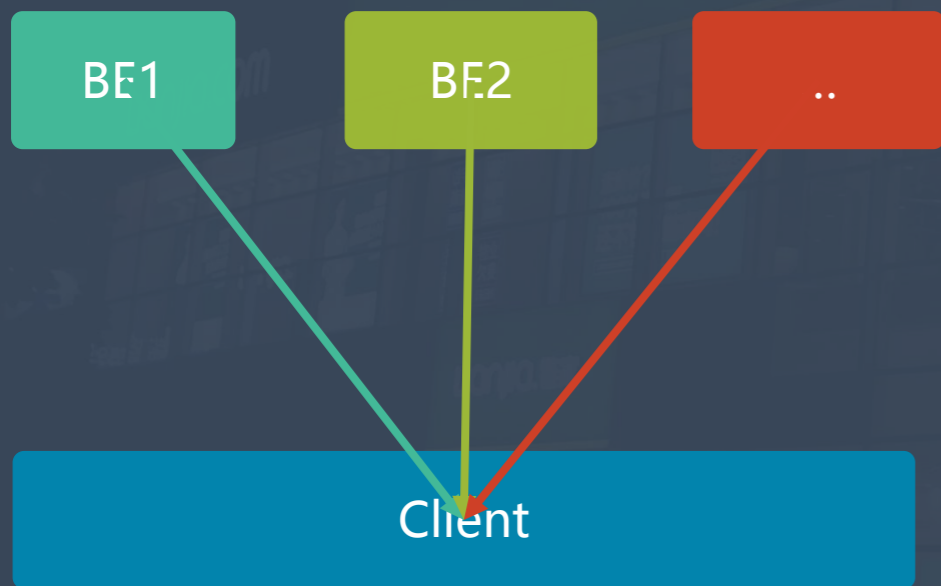
Android端链家App  
iOS 端Link (企业应用)

## 热修复

紧急修复组件化过程中造成的线上问题；每个补丁不允许超过1个版本

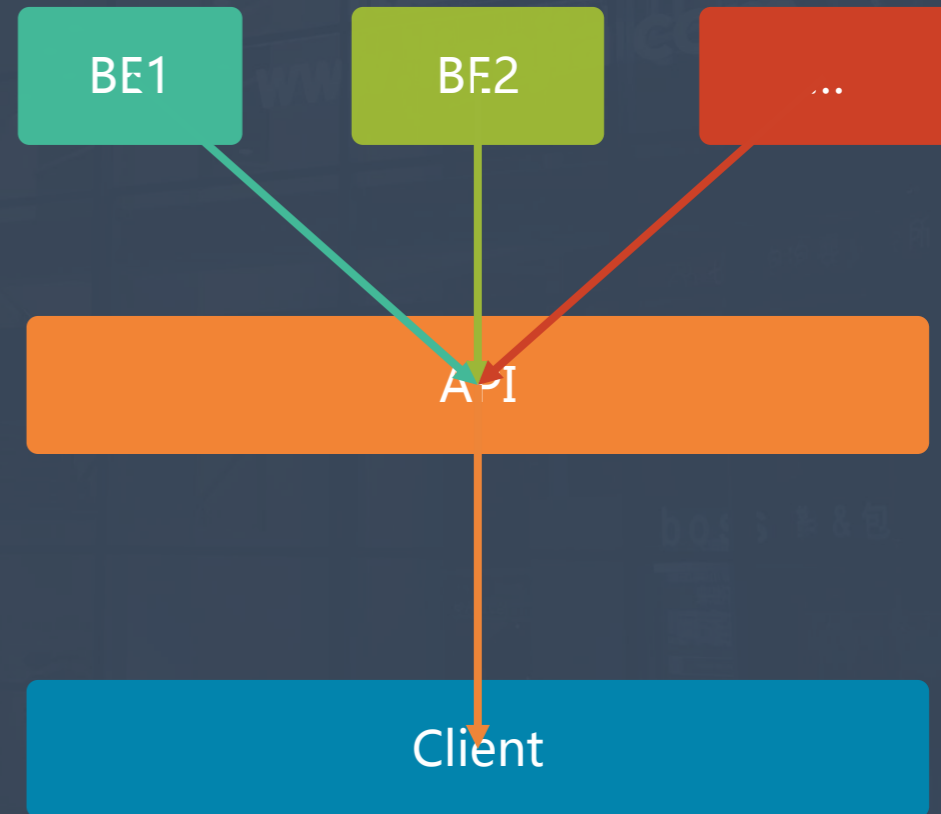
# API团队引入

LIANJIA.COM



Before

沟通成本降低  
发版风险降低  
业务逻辑简化



After

# 移动端与后端配合开发流程



# 收益



版本周期平均 **2** 周 Link app 用时 **1.5** 月 **<0.1** % crash率

注：Link app 代码规模10w+(不含第三方库)

更进一步





# 未来展望

- 插件化

用户对产品功能做个性化定制；减少安装包体积；降低发版成本

- 跨平台技术

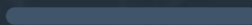
最小的成本覆盖到两个平台，避免重复开发

- 安全性

房屋交易环节的线上化



# Q&A



# Thanks!

