

客户端多端复用架构

郑金光





郑金光
美团高级工程师

2017年4月加入美团外卖，现主要负责美团外卖App
和美团App外卖频道多端代码复用工作，以及美团外
卖平台化基础服务相关工作

多端复用背景



外卖频道

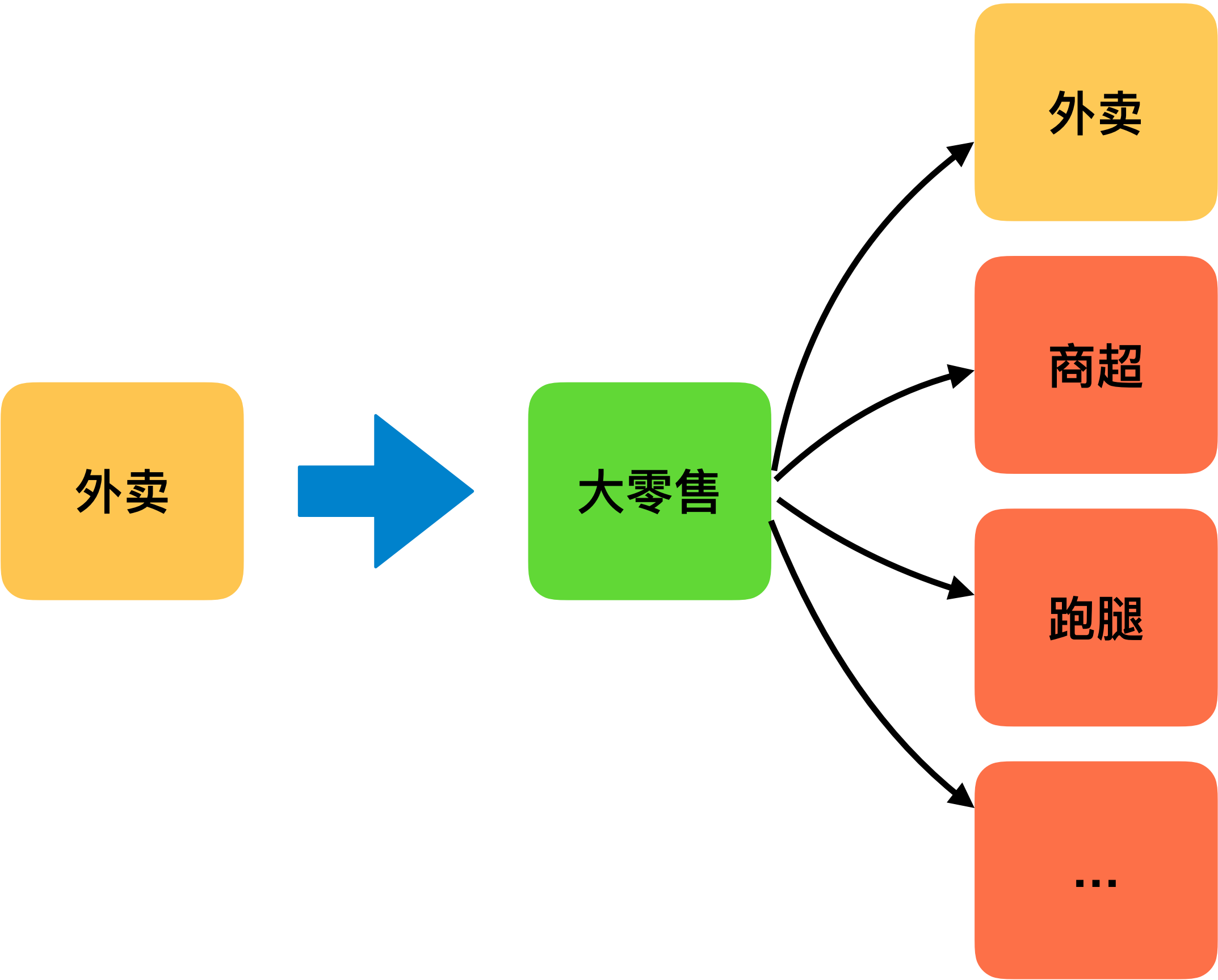


外卖App



业务范围

多端复用背景

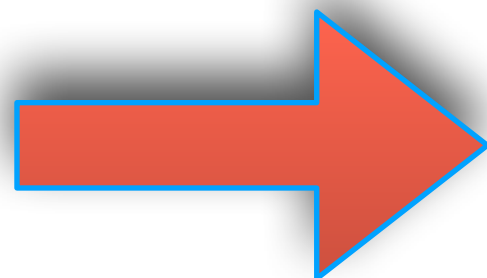


团队变化

多端复用背景



外卖团队



上海搜索团队

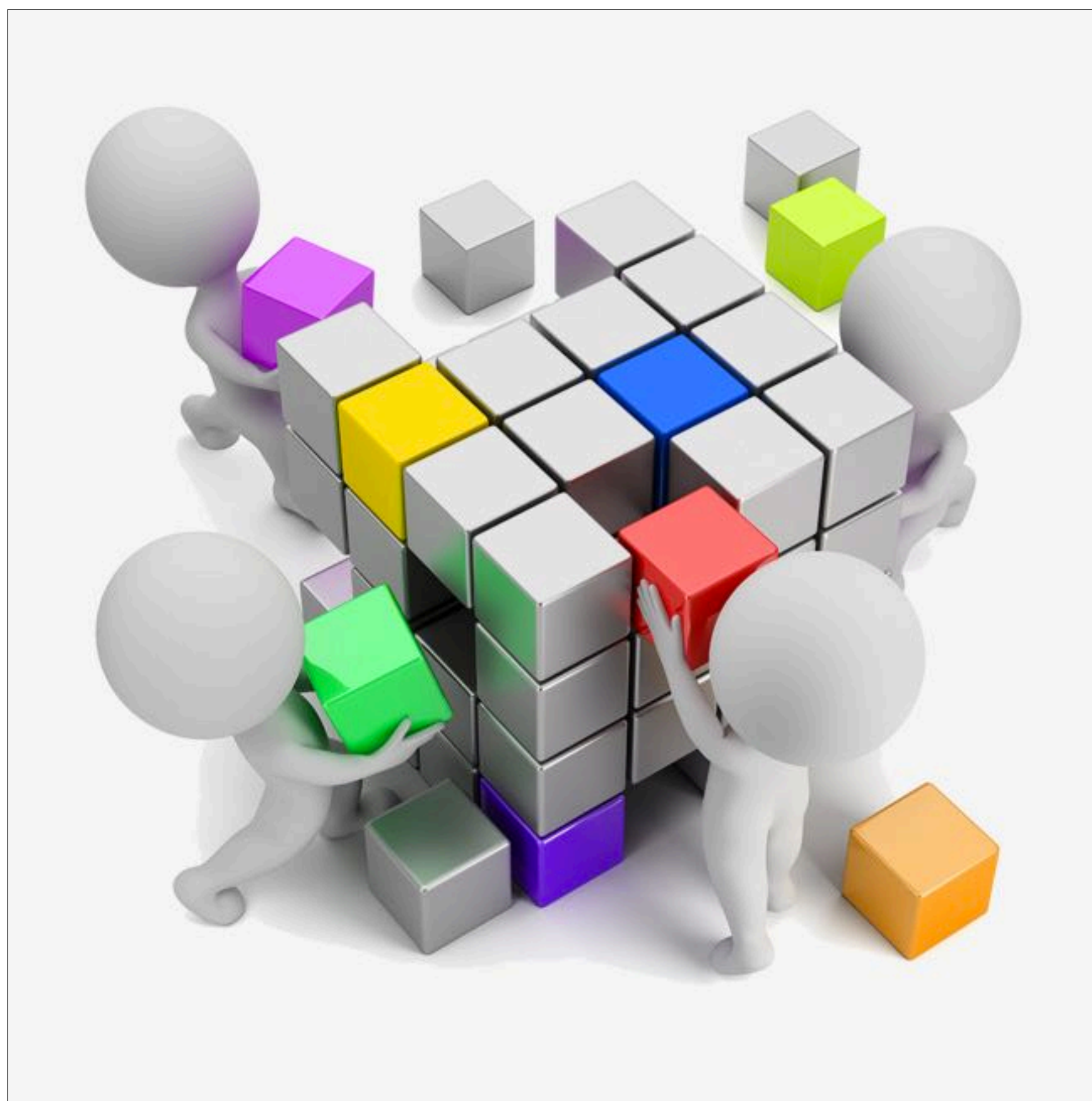


北京外卖团队



北京商超团队

多端复用背景



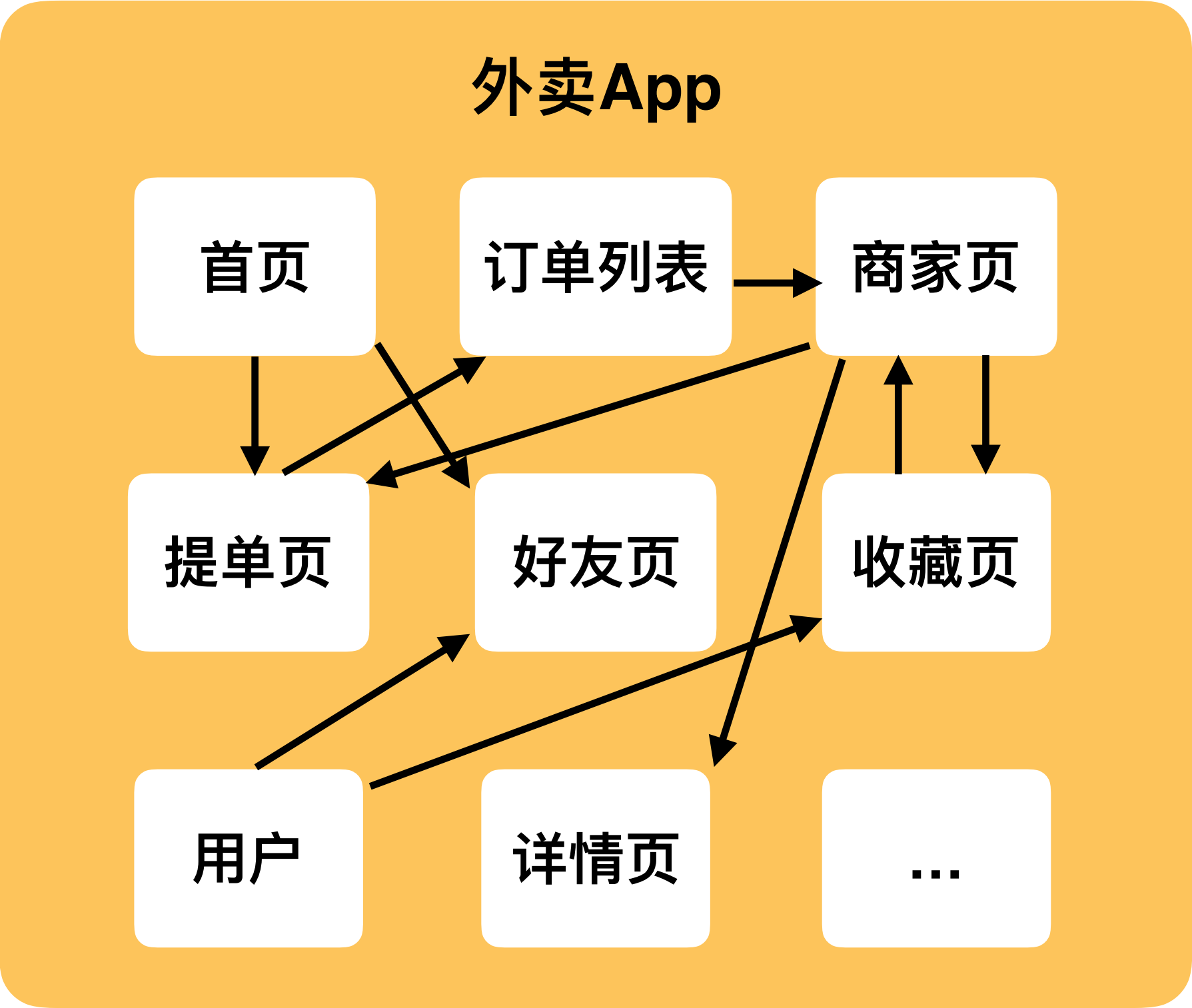
两端需求重复开发

团队内部分工问题

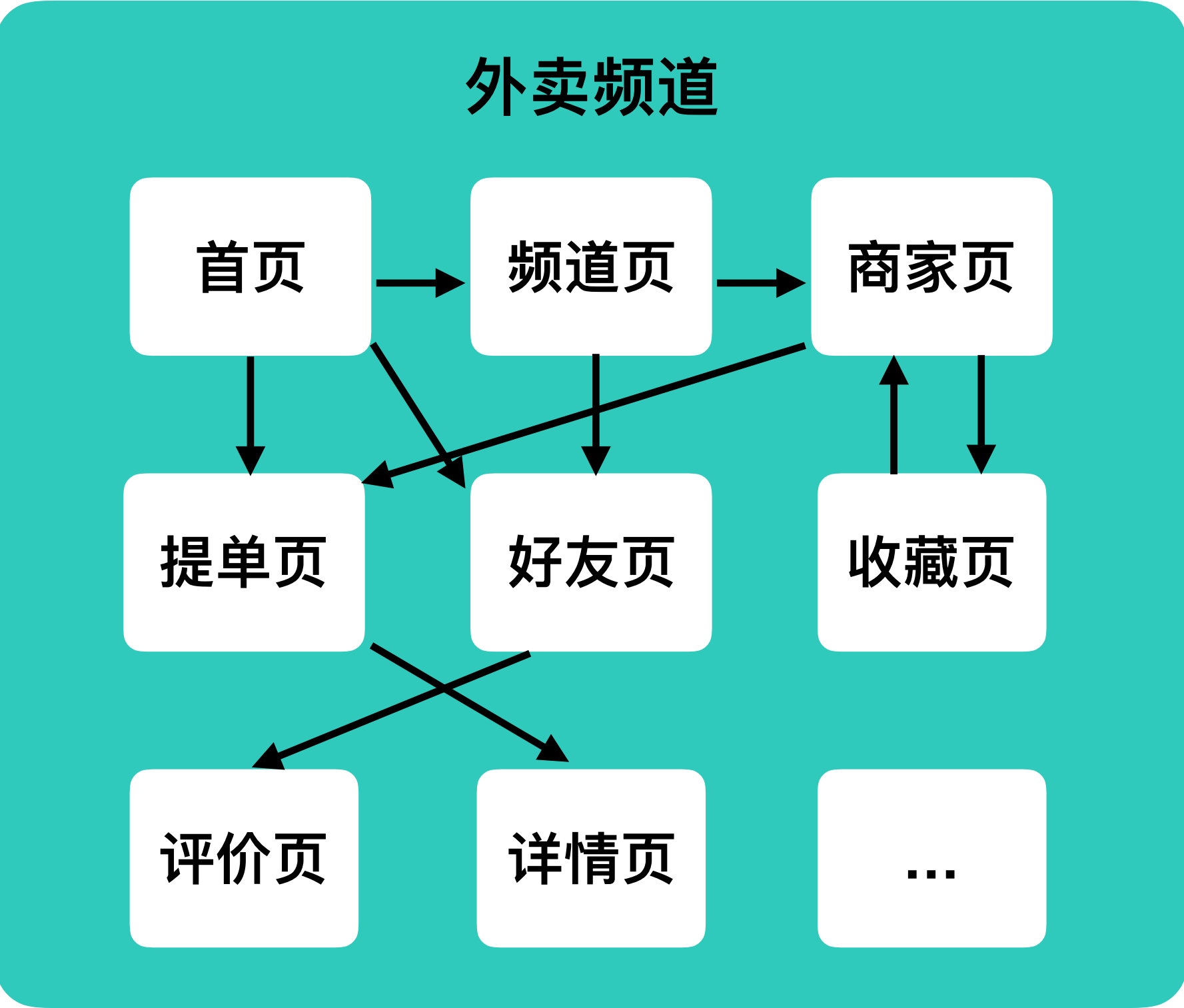
兄弟部门协作问题

客户端代码质量问题

早期客户端架构

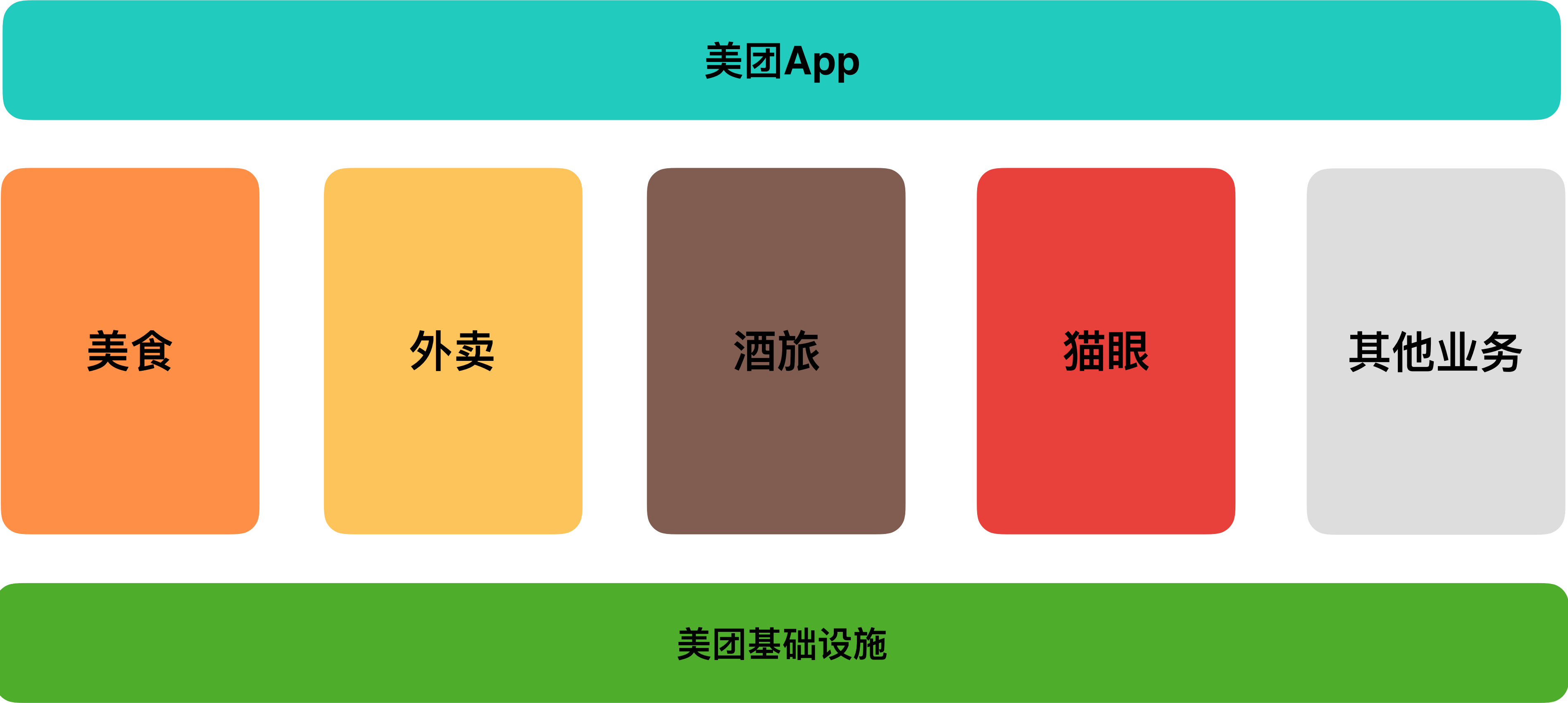


美团App



外卖基础库

美团平台架构

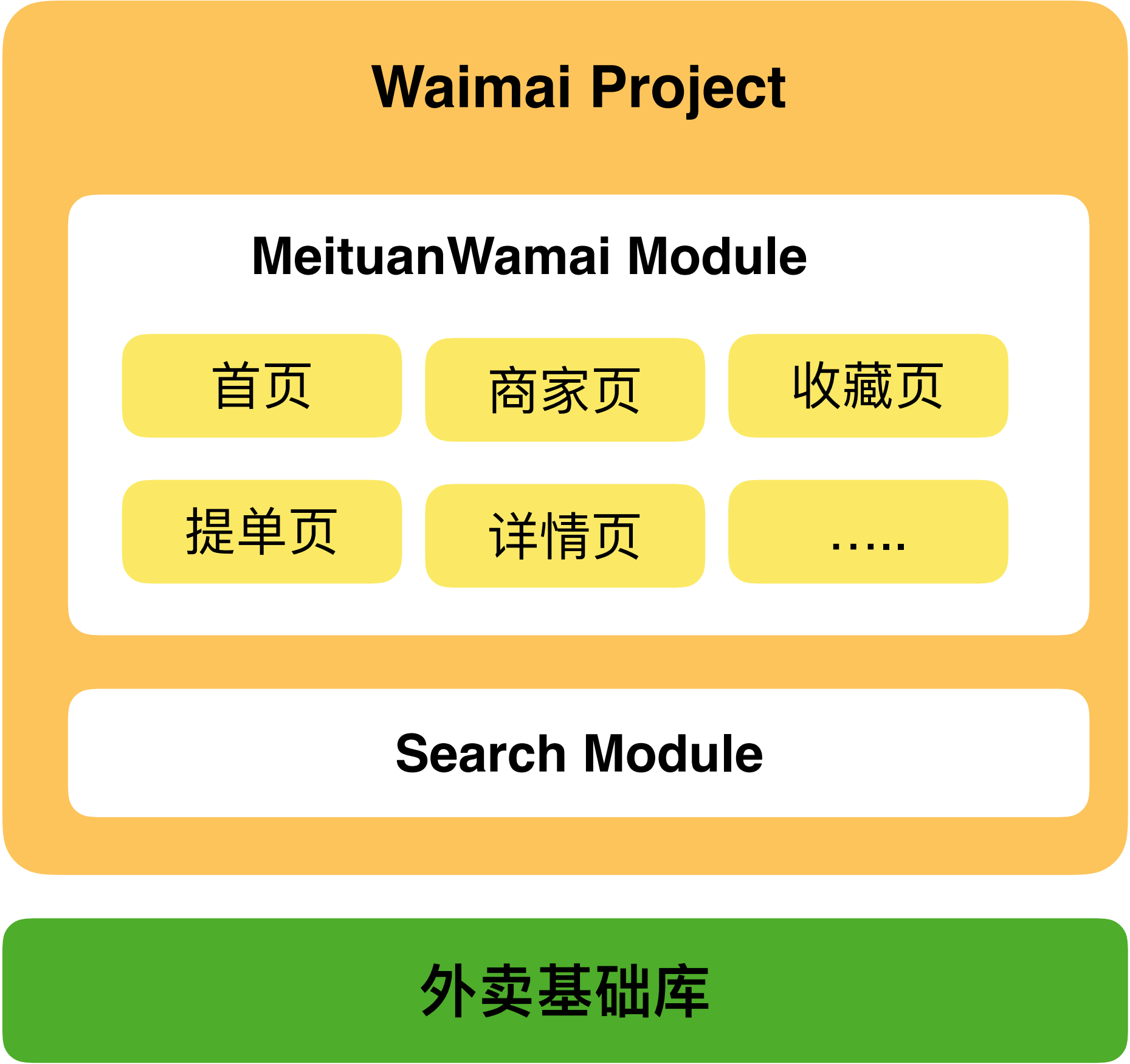


工程结构

代码复用

棘手问题

早期工程结构

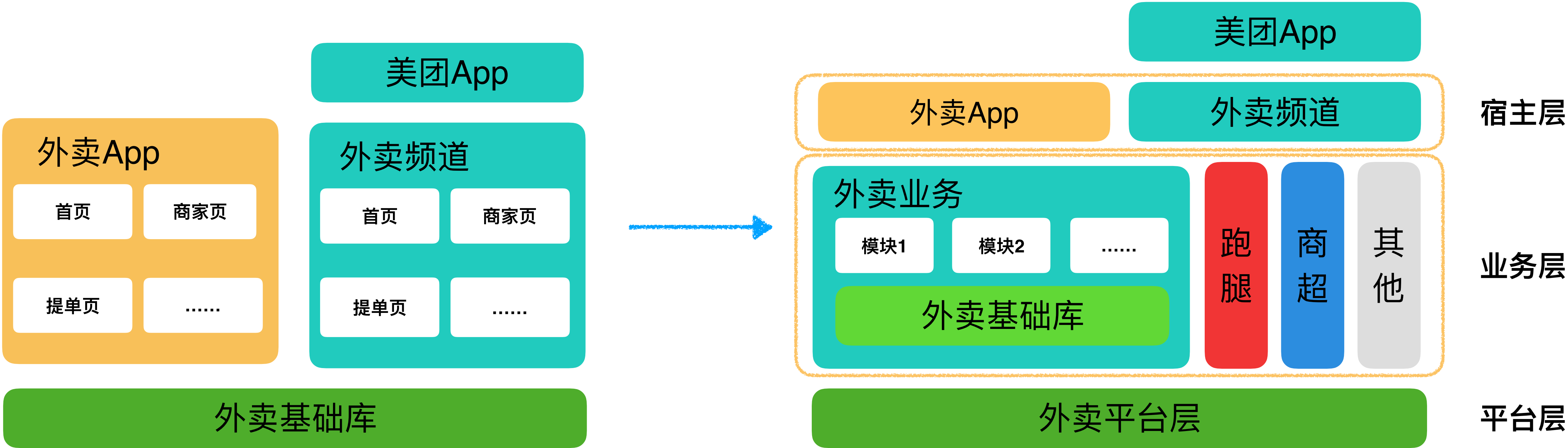


结构单一，复用较难

代码耦合严重

模块粒度太大

工程结构调整设计



康威定律



Melvin Conway

Organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations.

设计系统的组织，其产生的设计等同于组织之内、组织之间的沟通结构——梅尔文·康威

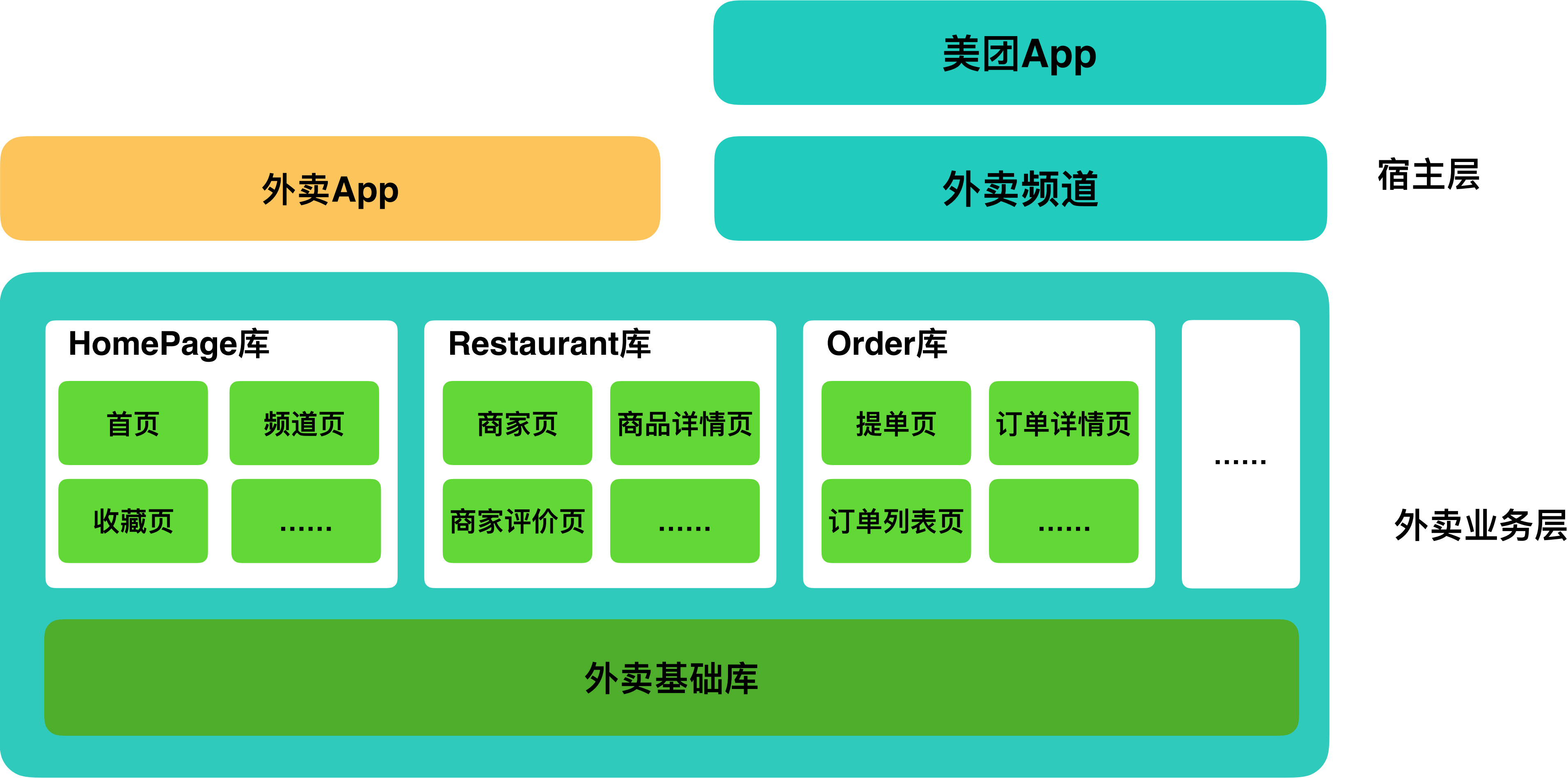


系统设计=组织结构

外卖下单主流程



复用架构



美团外卖复用



美团外卖App

导航区

推荐区

广告区

频道区

好友头条区



导航区

广告区

频道区

好友头条区

推荐区

美团App

涉及问题

外卖首页工程结构调整

- 网络库、图片库
- 基础Activity、Fragment
- 基础数据结构
- 数据统计
- 其他

外卖首页复用设计

美团App

外卖App

外卖频道

HomePage库

首页

头部区

Header区

商家列表

.....

其他模块

外卖基础库

Model

网络库

图片库

.....

module内部代码隔离

```
sourceSets {  
    def dirs = ['base', 'p_home', 'p_kingkong', 'p_citydelivery',  
                'p_collect', 'p_passport', 'p_today_recommend']  
    main {  
        manifest.srcFile 'src/base/main/AndroidManifest.xml'  
        dirs.each { dir ->  
            java.srcDir("src/$dir/main/java")  
            res.srcDir("src/$dir/main/res")  
        }  
    }  
}
```



```
▼ src  
  > base  
  > p_citydelivery  
  > p_collect  
  > p_home  
  ▼ p_kingkong  
    ▼ main  
      > java  
      > res  
  > p_passport  
  > p_today_recommend  
  .gitignore  
  build.gradle
```


pins工程代码检查

定义pins工程name及packageName映射，定义白名单（base）



检查非该目录下的Java文件名引用



输出被非该目录下引用的Java文件代码及行数

```
p_page.json
p_home:com.waimai.bussiness.page.home
p_collect:com.waimai.bussiness.page.collect
```

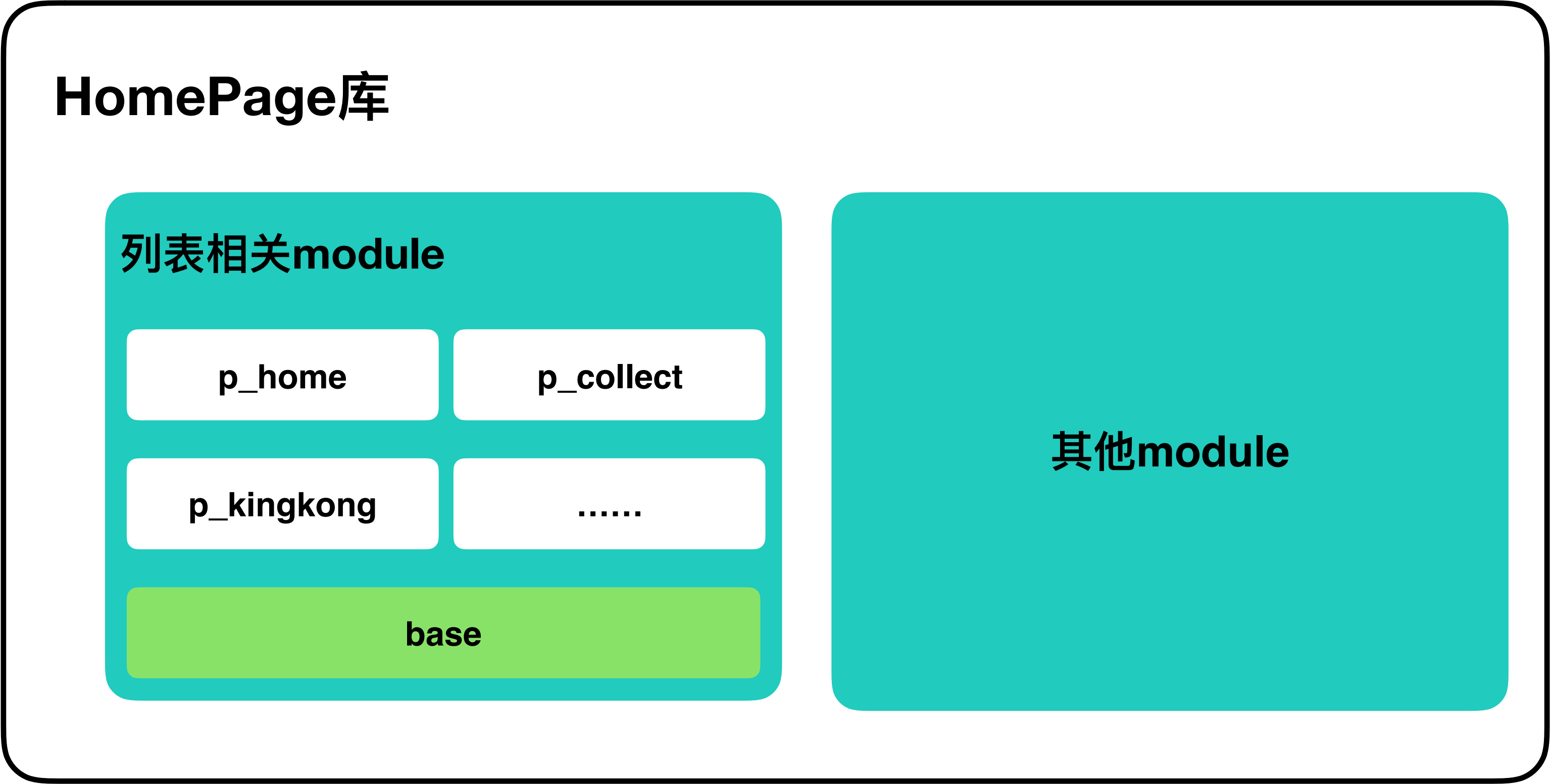


```
> p_collect
> p_foot
> p_home
```



```
CollectionUtils
1 package com.waimai.business.page.collect;
2
3 import com.waimai.business.page.home.CollectionUtils;
4
5 import java.util.Collection;
6
7 public class CollectionUtils {
8
9     public static <T> boolean isEmpty(Collection<T> collection) {
10         return CollectionUtils.isEmpty(collection);
11     }
12 }
13
```

三级工程结构



研发支撑工具

- 代码规范
- 代码检查 WMLint
- 准入检查 WMCI
- 插件化编译 Hydra
- 构建插件

测试支撑工具

- 单元测试
- UI自动化测试
- 性能测试 云测、Metrics
- 自测覆盖率检查 MTJacoco
- 测试工具 AppMock、接单工具 ...

宿主层

外卖App

美团App（外卖频道）

外卖业务层

- 首页 page
- 商家 restaurant
- 订单 order
- RN wmrn
- 搜索 nox
- 商超 store

三方业务层

- 跑腿 legwork
- 扫码购 store
- 闪付 quickpass
- ...

外卖领域模型、基础服务、基础组件（platform）

平台兼容层

跨终端标准化接口

外卖App定制化基础业务组件（WMKit，仅外卖App）

- 登录
- 定位
- 地图
- 分享
- 通知

基础能力层

平台化架构组件

- WMArch
- WMRouter
- ServideLoader
- Mainboard

基础业务组件

- 支付
- 地图
- 登录
- 分享
- 升级
- 安全&加固
- 定位
- 指纹&UUID

基础技术组件

- MT Retrofit
- MT Flexbox
- MT Picasso
- Picasso
- MTRN
- Rxjava
- Hydra
- KNB

监控容灾运维能力

- Metrics
- Robust
- CAT
- Holmes
- Sniffer
- Logan
- Horn
- 限流SDK

发布支撑工具

- Jenkins ACI
- 灰度、全量发布 Eva
- 插件发布 Eva
- 补丁发布 Eva
- RN发布 Eva

运维支撑工具

- 网络监控 CAT
- 稳定性和性能监控 Crash工作台、Metrics
- 业务监控 Sniffer
- 线上追查 Sniffer、Holmes、Logan
- 降级限流 Loadinfo、Horn

工程结构

代码复用

棘手问题

代码复用的背景

两端需求高度重合

代码差异较大，维护困难

页面实现不规范

多端代码复用涉及四个方面

差异化管理

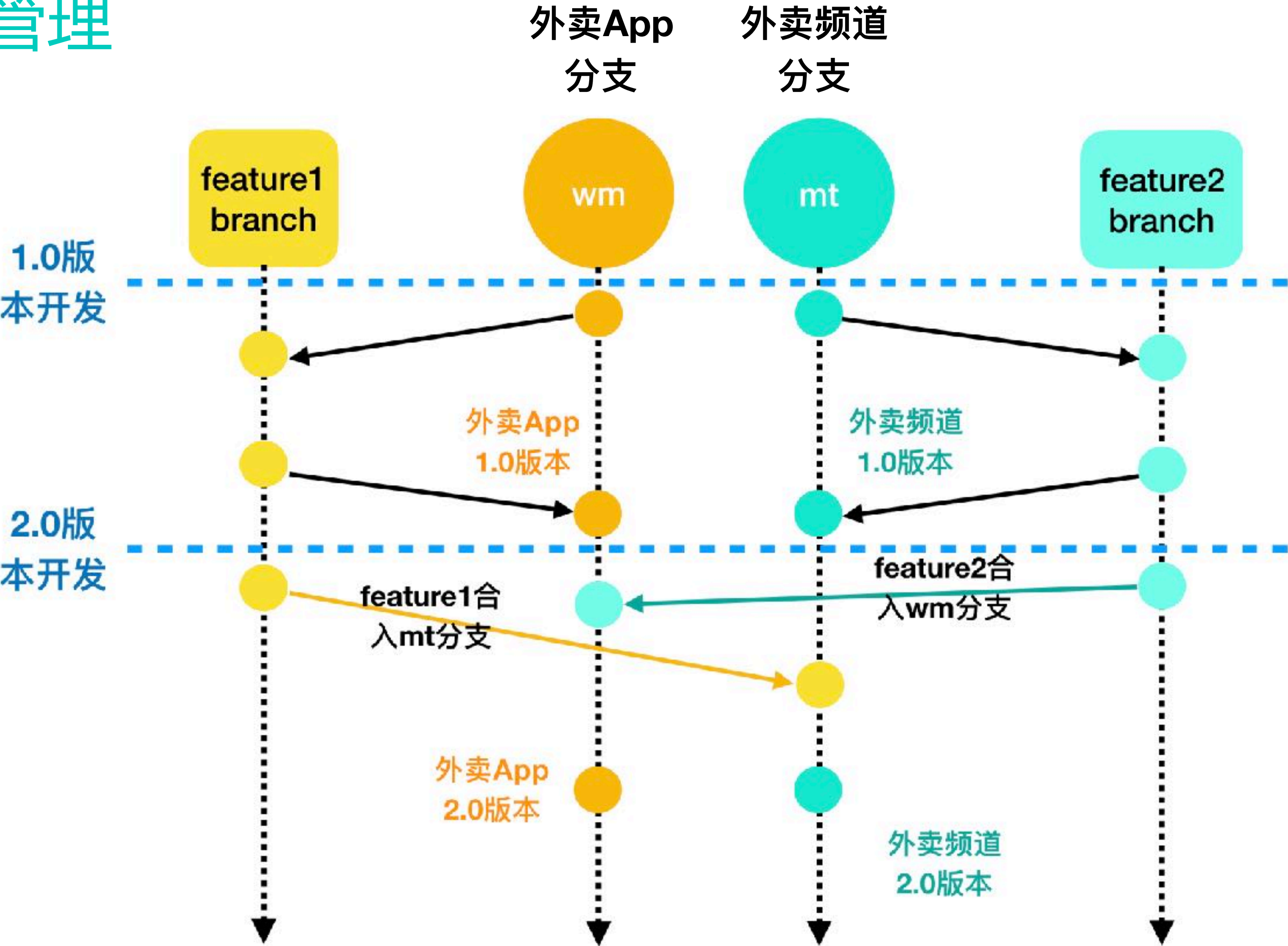
基础服务复用

组件复用

页面复用

分支管理

差异化管理



Product Flavor

差异化管理

```
android {  
    ...  
    defaultConfig {...}  
    buildTypes {...}  
    productFlavors {  
        demo {  
            applicationIdSuffix ".demo"  
            versionNameSuffix "-demo"  
        }  
        full {  
            applicationIdSuffix ".full"  
            versionNameSuffix "-full"  
        }  
    }  
}
```

业务
模块

DiffHandlerImpl
wm-flavor

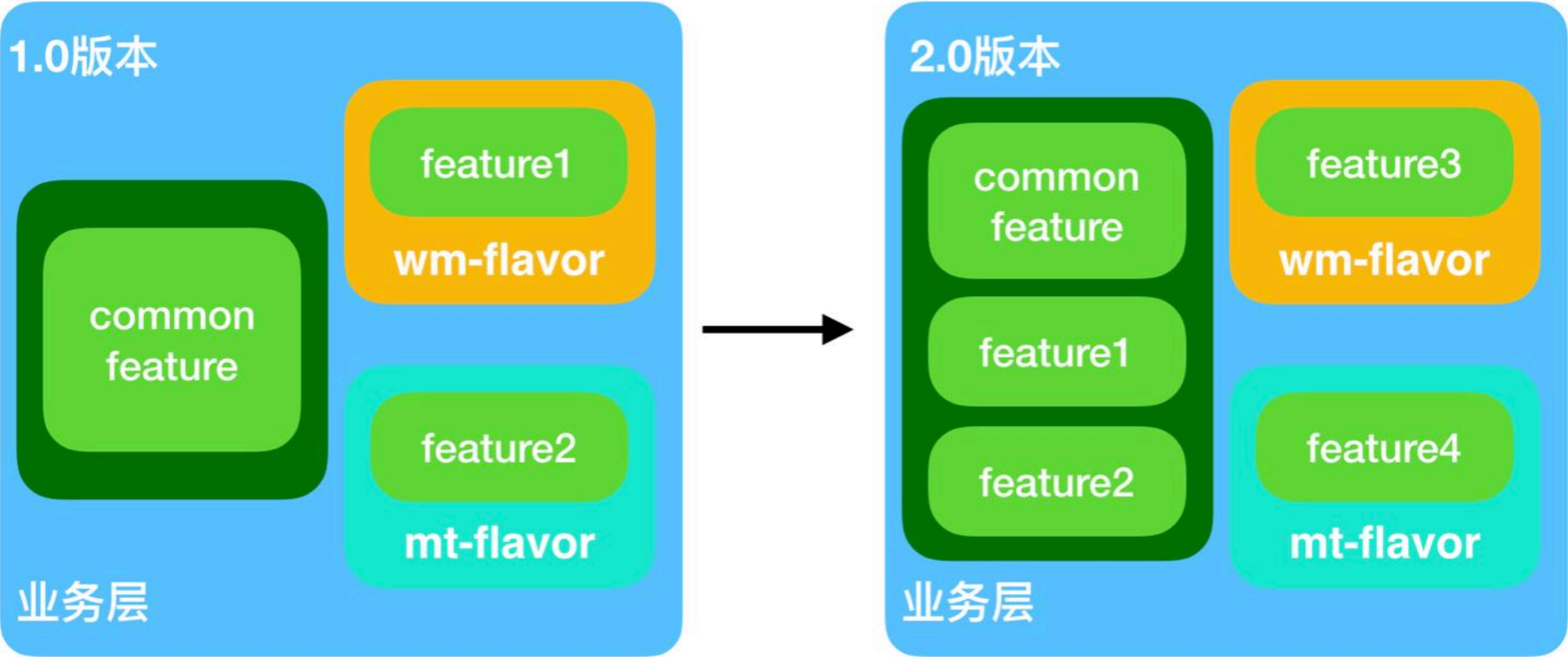
DiffHandlerImpl
mt-flavor

Common

DiffHandler

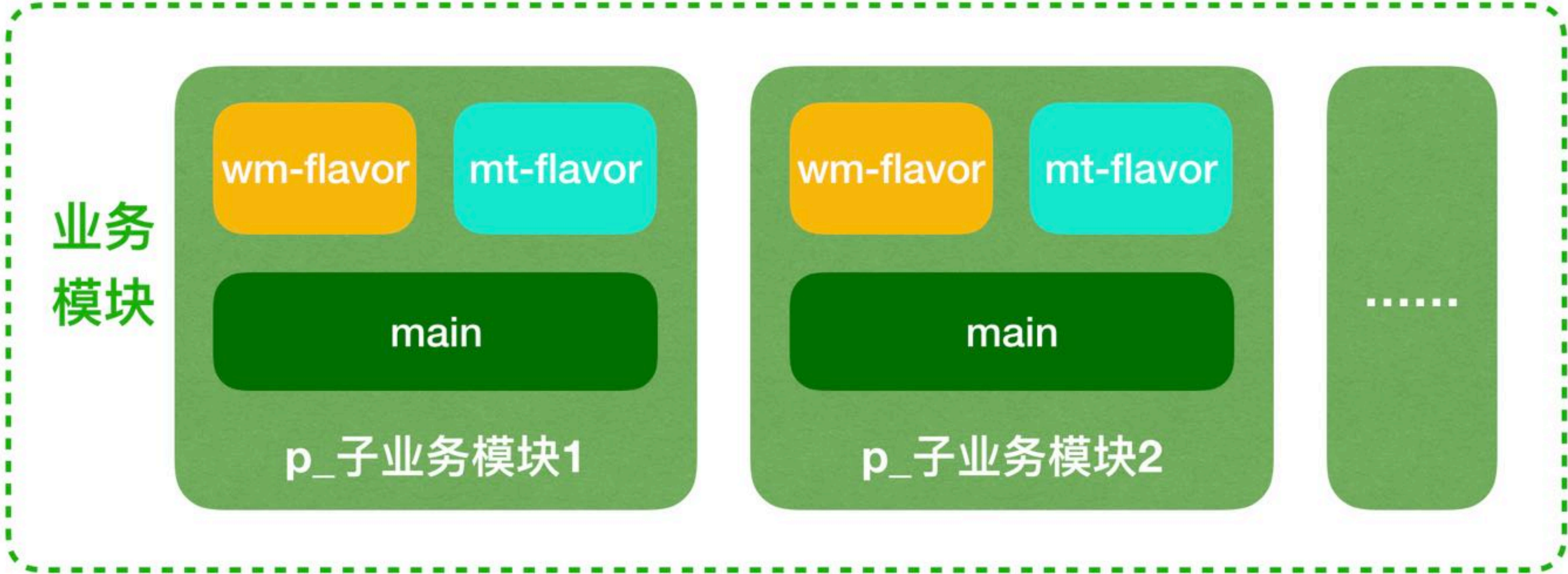
Product Flavor

差异化管理



pins工程+Flavor方案

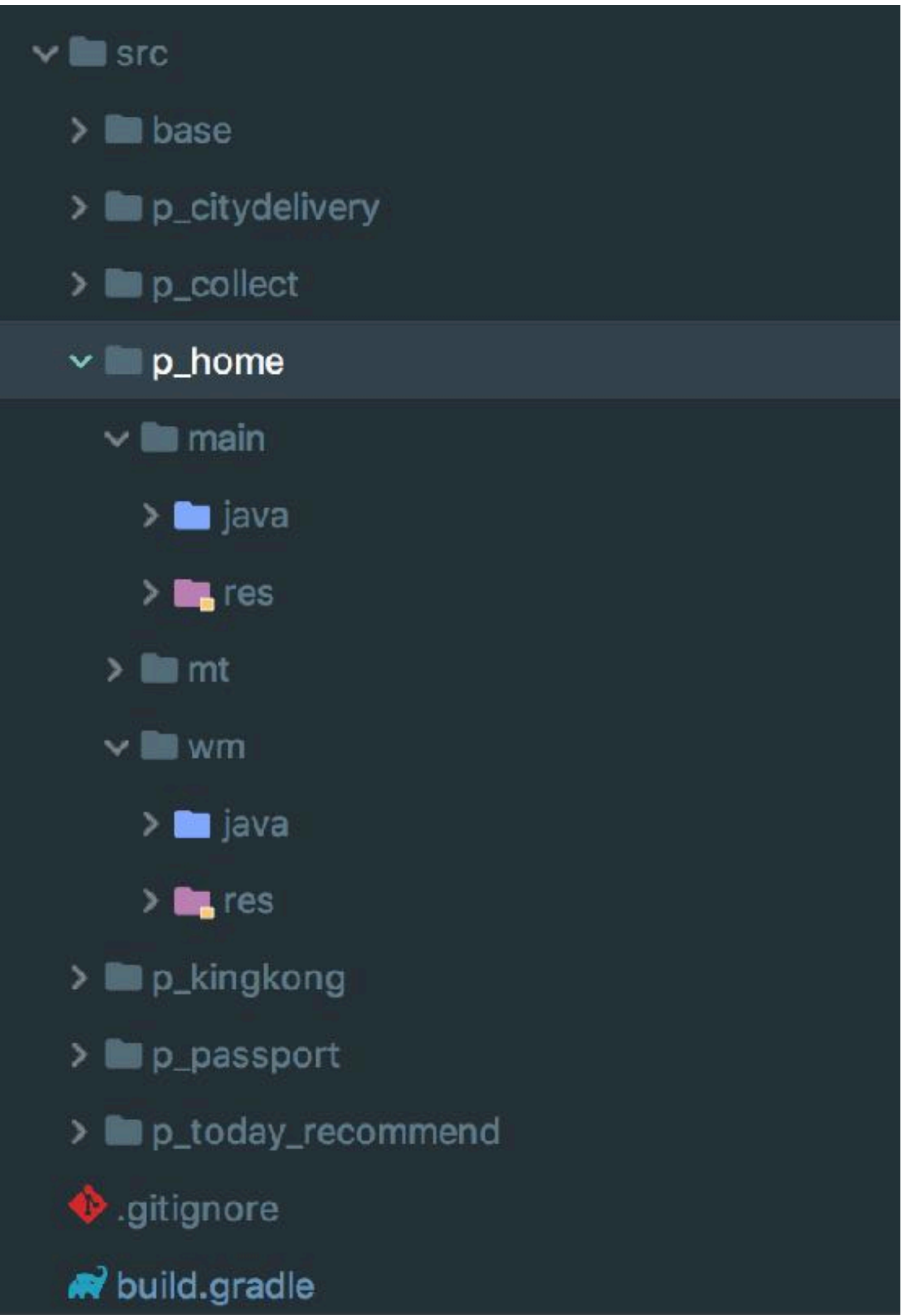
差异化管理



pins工程+Flavor方案

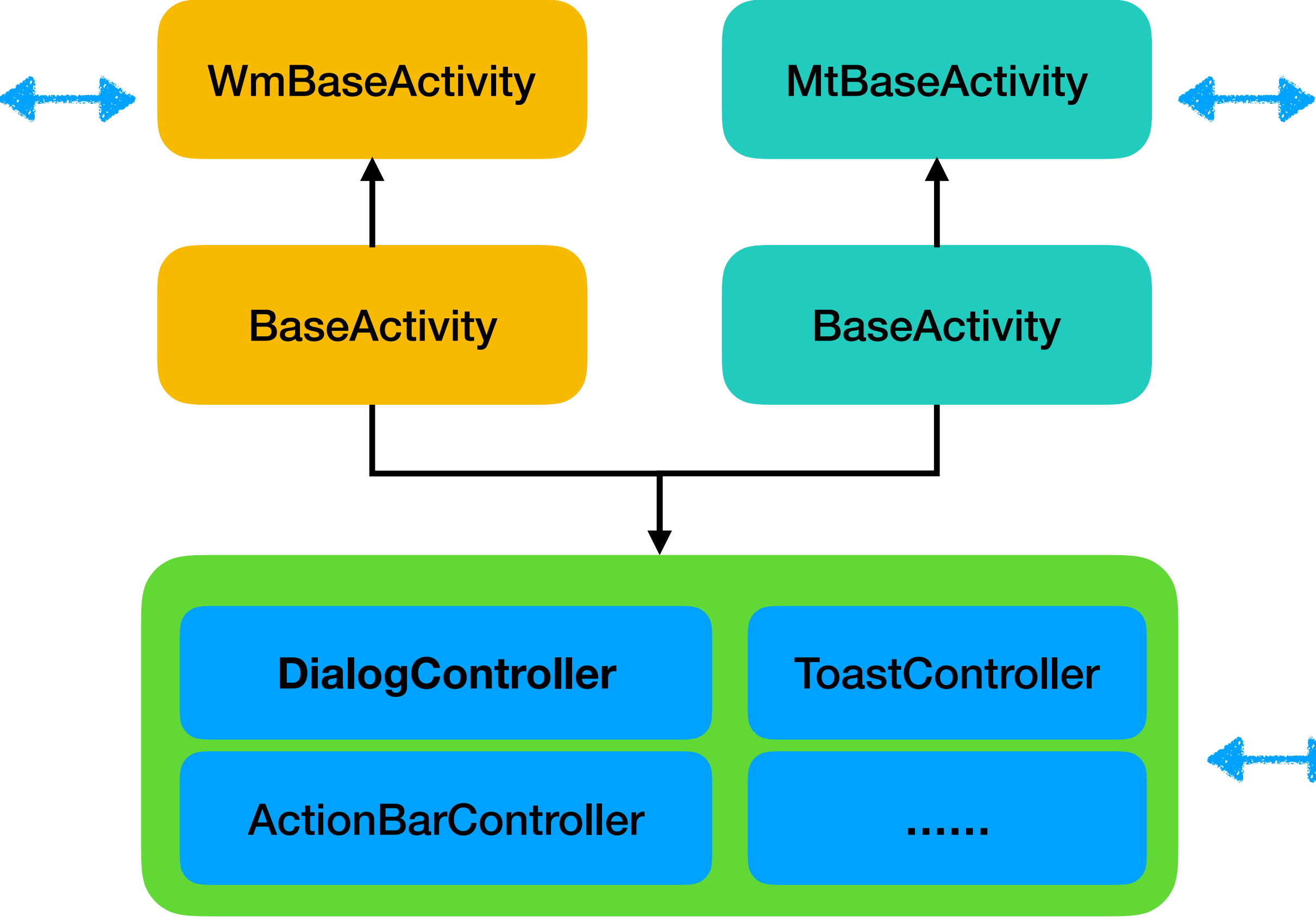
差异化管理

```
sourceSets {
    def dirs = ['base', 'p_home', 'p_kingkong', 'p_citydelivery',
                'p_collect', 'p_passport', 'p_today_recommend']
    main {
        manifest.srcFile 'src/base/main/AndroidManifest.xml'
        dirs.each { dir ->
            java.srcDir("src/$dir/main/java")
            res.srcDir("src/$dir/main/res")
        }
    }
    wm {
        dirs.each { dir ->
            java.srcDir("src/$dir/wm/java")
            res.srcDir("src/$dir/wm/res")
        }
    }
    mt {
        manifest.srcFile 'src/mt/AndroidManifest.xml'
        dirs.each { dir ->
            java.srcDir("src/$dir/mt/java")
            res.srcDir("src/$dir/mt/res")
        }
    }
}
```



基础服务复用

```
BaseActivity
├── ActionBarClickListener
├── addActionBarRightButton(CharSequence, OnClickListener): void
├── addActionBarRightButton(int, OnClickListener): void
├── checkPermissions(int, String, String[], PermissionCallback): void
├── checkPermissions(int, String[], PermissionCallback): void
├── clearAllRunnable(): void ↑WeakHandlerInterface
├── configContextConfiguration(Context): void
├── dismissProgressDialog(): boolean ↑DialogController
├── getActivity(): Activity
├── getGSource(): String
├── getMeterTask(): MetricsSpeedMeterTask
├── getProgressDialog(): Dialog ↑DialogController
├── getProgressOnCancelListener(): OnCancelListener
```



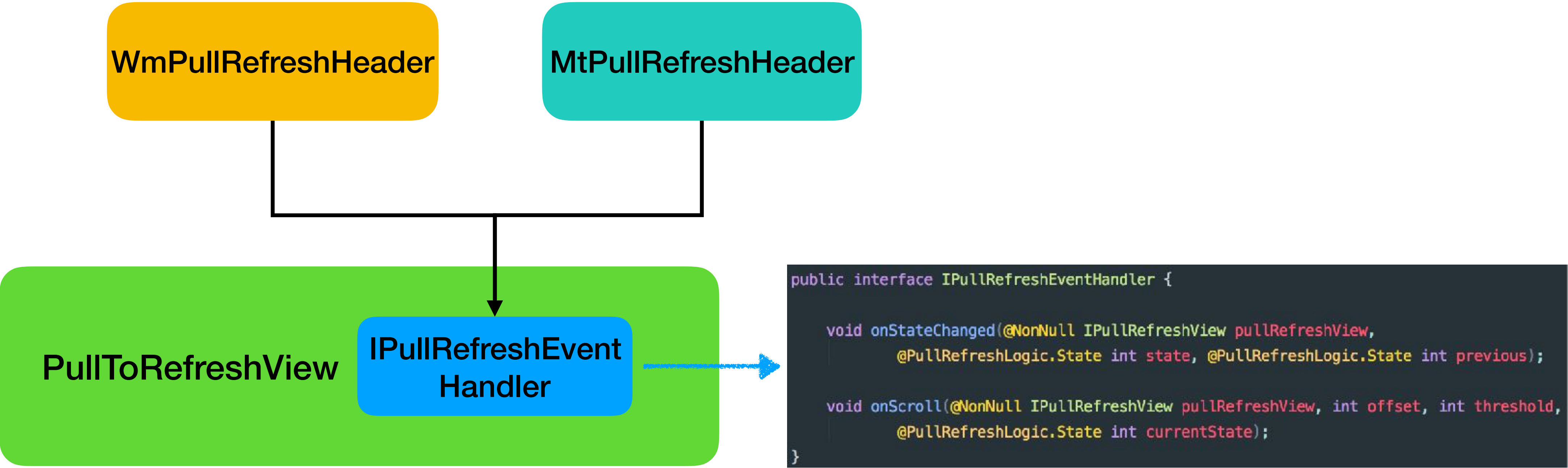
```
BaseActivity
├── onKeyDown(int, KeyEvent): boolean ↑AppCompatActivity
├── onOptionsItemSelected(MenuItem): boolean ↑Activity
├── addActionBarRightButton(CharSequence, OnClickListener): void
├── addActionBarRightButton(int, OnClickListener): void
├── handleException(Exception): void
├── hideProgressDialog(): void
├── onCreate(Bundle): void ↑AppCompatActivity
├── onNewIntent(Intent): void ↑FragmentActivity
├── onRestoreInstanceState(Bundle): void ↑Activity
├── onResume(): void ↑FragmentActivity
├── onSaveInstanceState(Bundle): void ↑AppCompatActivity
├── onStart(): void ↑AppCompatActivity
```

```
BaseActivity
├── ActionBarClickListener
├── ActionBarController
├── Activity
├── DialogController
├── FragmentActivity
├── PermissionContract
├── ServiceErrorObserver
├── ToastController
├── WeakHandlerInterface
```


下拉刷新组件
组件复用



下拉刷新组件
组件复用

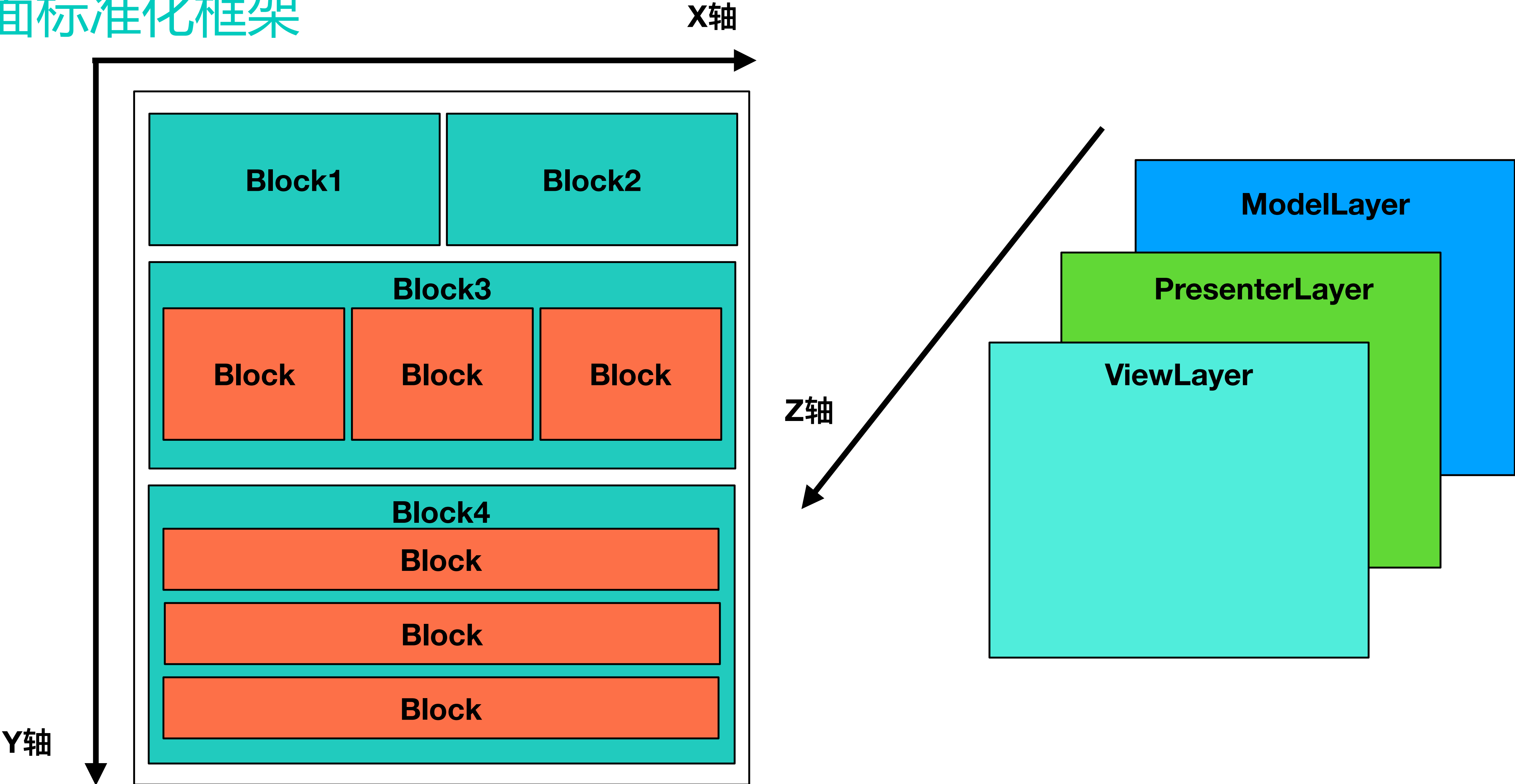


页面复用

页面模块化

逻辑分层

页面标准化框架



美团外卖首页



美团外卖App

导航区



推荐区



广告区



频道区



好友头条区



导航区

广告区

频道区

好友头条区

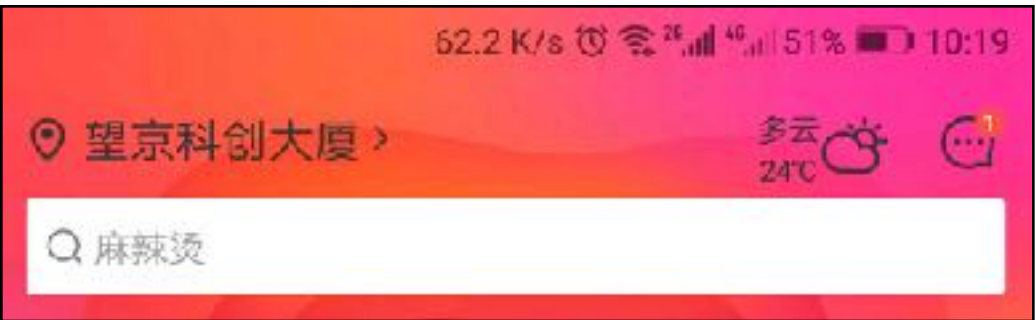
推荐区

美团App

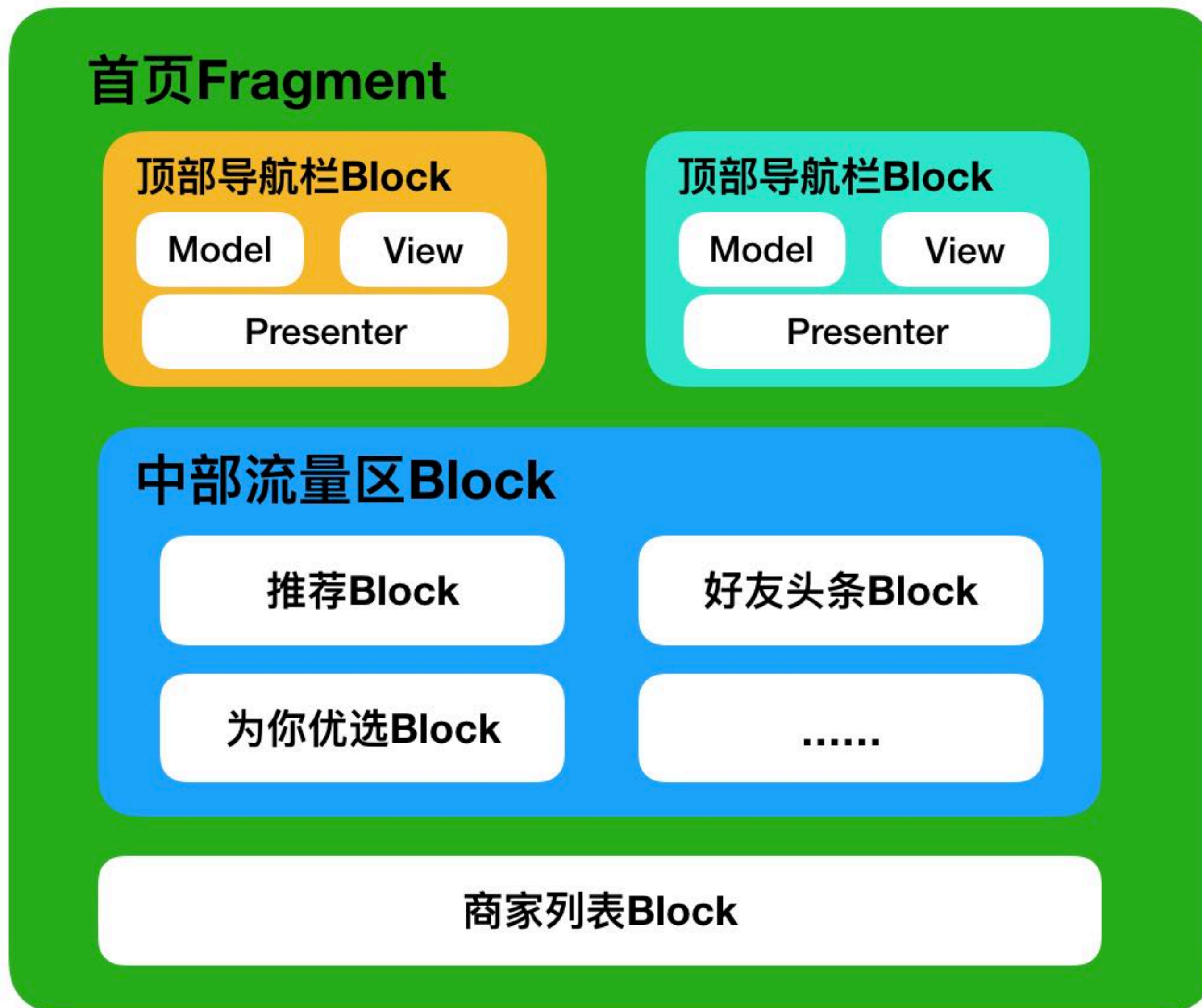
页面复用



拆分模块



页面复用



多端复用整体设计



工程结构

代码复用

棘手问题

此处键入副标题内容

棘手问题

业务库页面跳转

业务库之间代码调用

业务库初始化

页面跳转

某PM：外部跳转Uri能不能先定位？iOS说可以



某RD：Android做不了，系统控制的，我也很绝望啊



友谊的小船说翻就翻



功能难以实现—>PM不开心，后果很严重



RD新人：这个标志位啥意思？看不懂.....



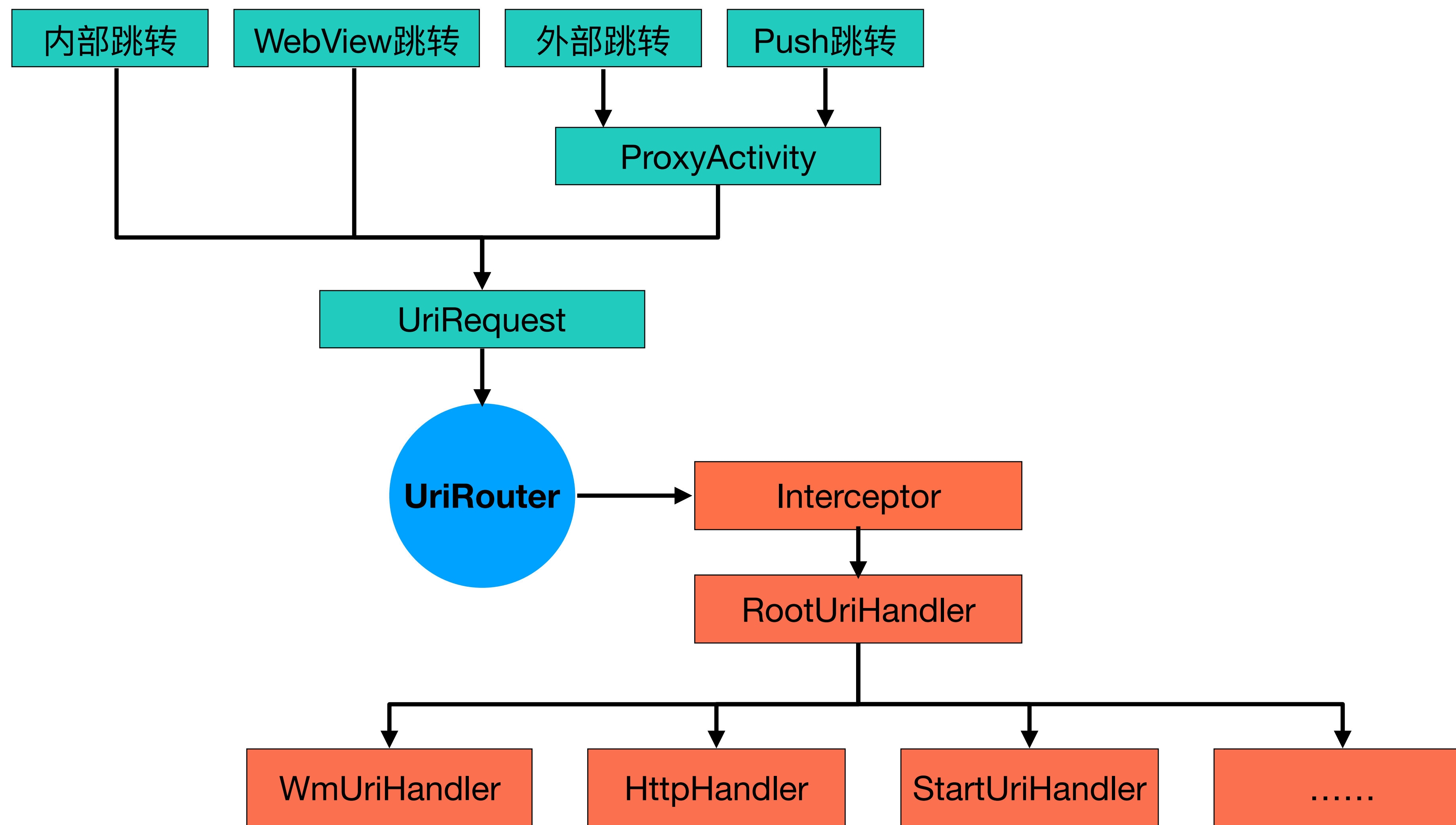
RD老人：修BUG用的Hack方案，不然外部Uri跳到我的红包，返回就直接到桌面了.....



RD新人：和我需求有冲突，我再加个标志位.....

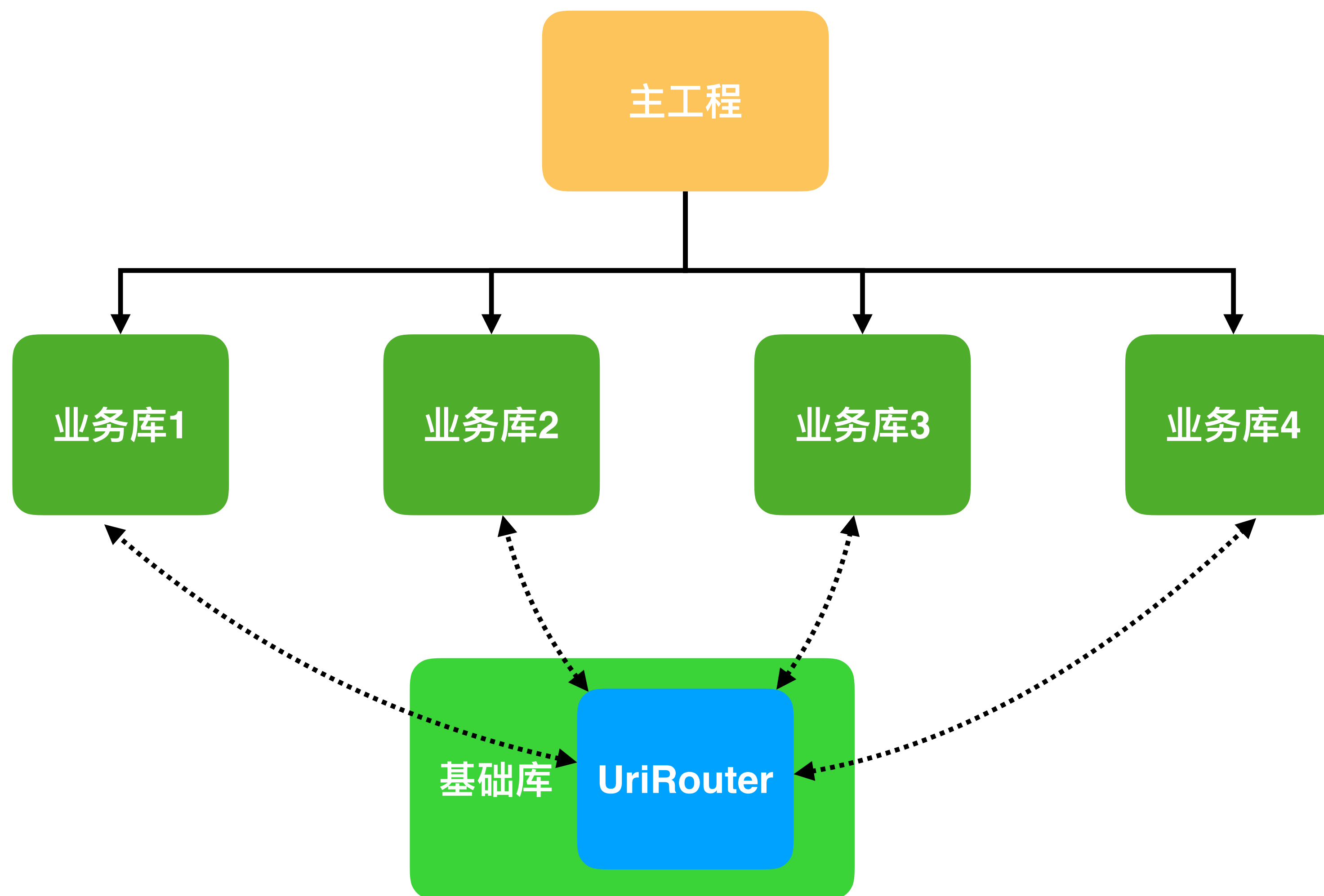
BUG多—>打补丁—>看不懂—>
给补丁打补丁—>更加看不懂

WMRouter实现 页面跳转

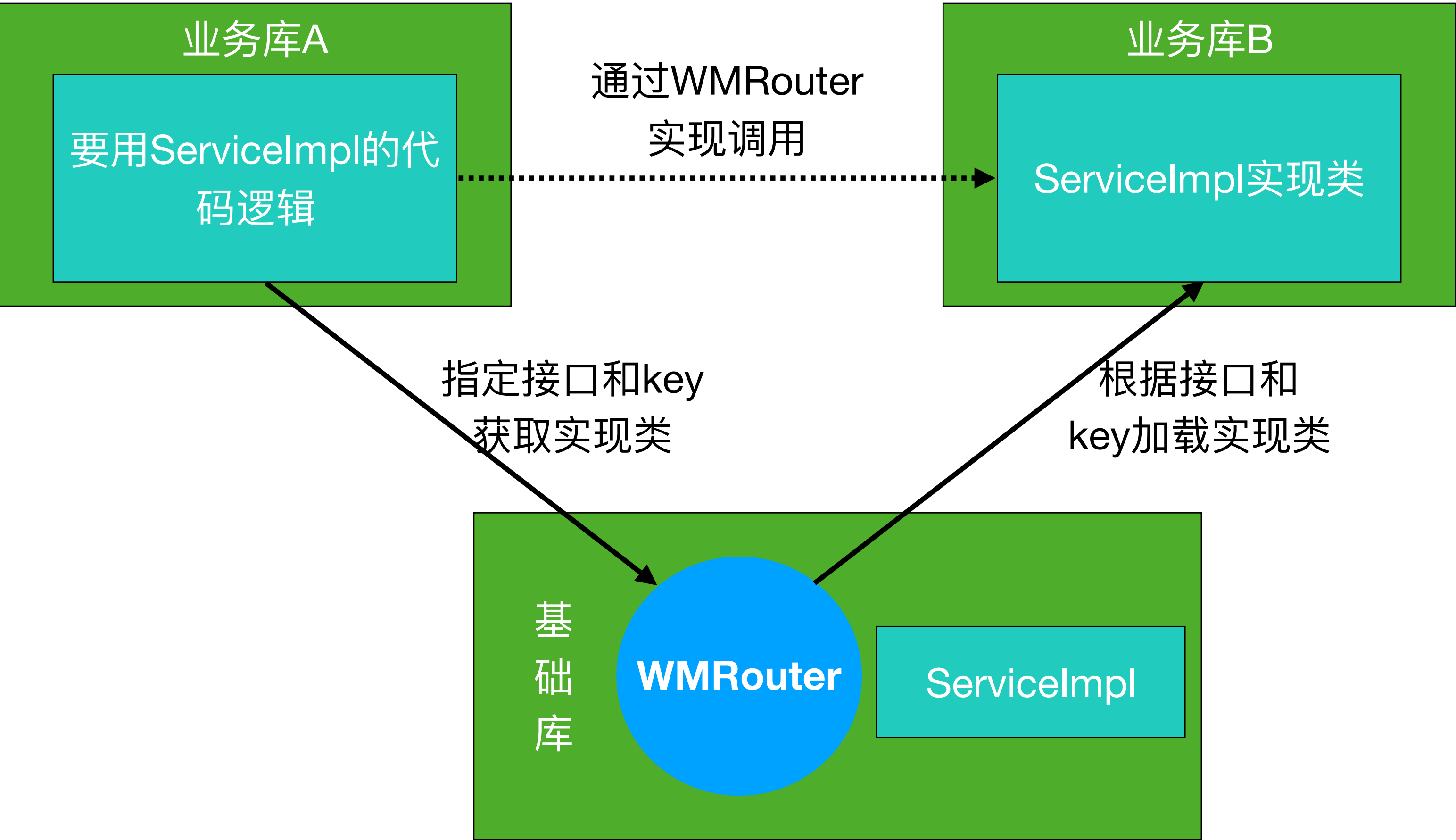


业务库通信

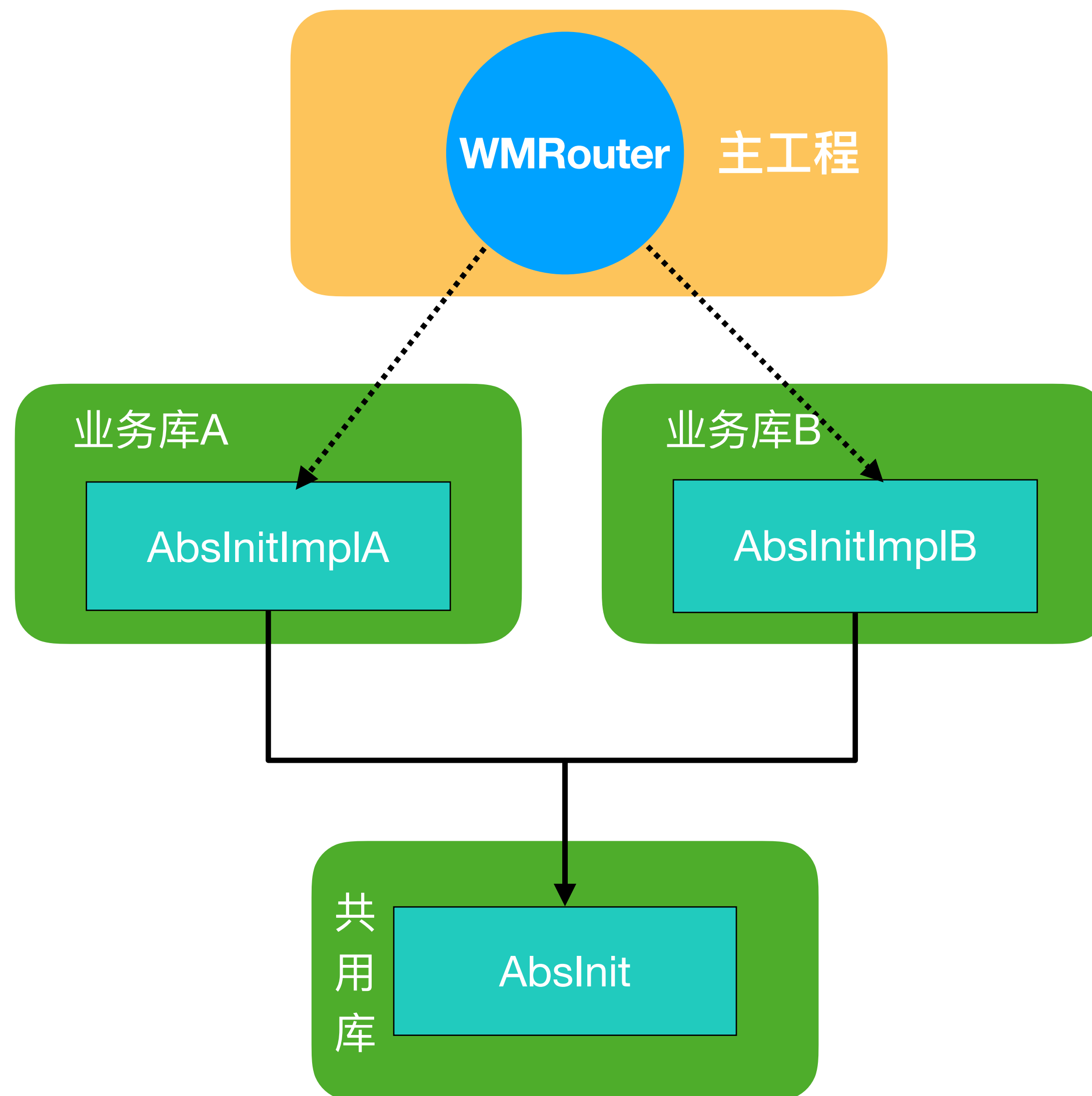
WMRouter



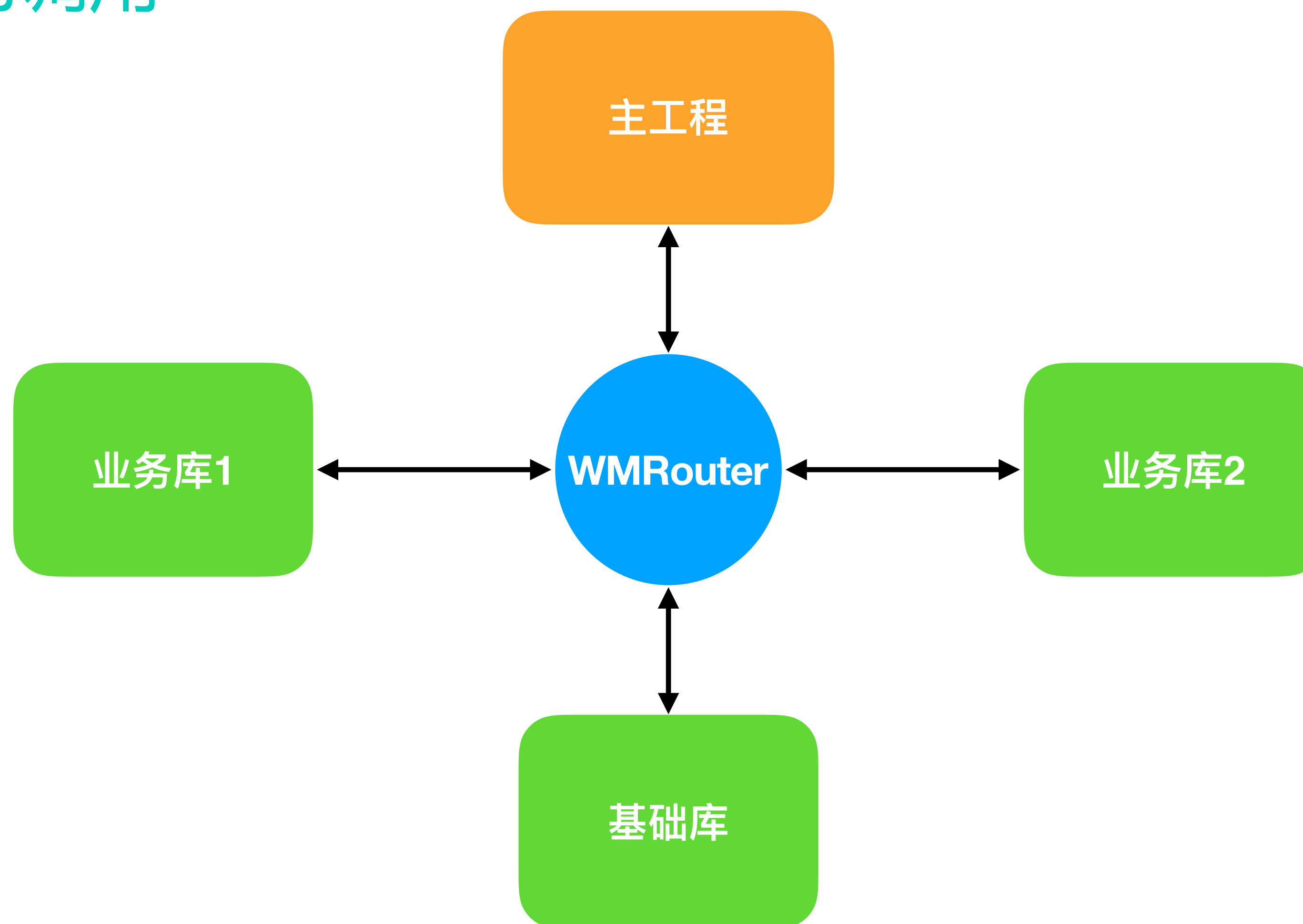
业务库代码调用



业务库初始化



各库之间的调用



总结

01

工程结构

复用工程结构

Pins工程划分及三级工程结构

02

代码复用

差异化管理：pins工程+Flavor

基础服务、组件复用

页面复用框架

03

棘手问题

WmUriRouter:

页面跳转

业务库代码复用

业务库初始化

Q&A

CODE A BETTER LIFE

一行代码 亿万生活



更多技术干货
欢迎关注“美团技术团队”

招聘：Android、iOS、FE 高级/资深工程师和技术专家，base 北京、上海、成都

邮箱：wukai05@meituan.com

