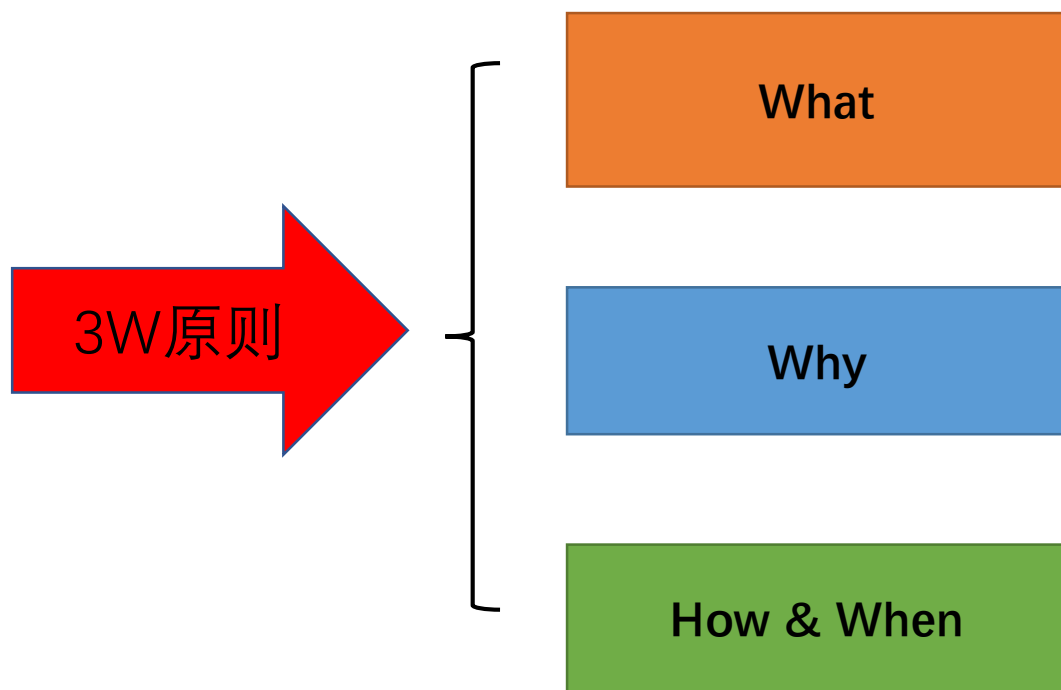


大爆炸重构分享

By 王武军

- Part 1 重构理论
- Part 2 大爆炸重构
- Part 3 心得分享

Part 1 重构理论



what

- 重构 – refactoring

“在不改变代码外在行为的前提下，对代码做出修改，改进程序的内部结构”

– Martin Flower



- 不改变代码外在行为 – 操作逻辑、视图效果
- 修改、改进 – 修改代码、调整逻辑；改进项目结构

- 重构的作用

1. 重构改进软件设计

调控代码、优化项目结构、易于扩展

2. 重构使软件更容易理解

代码易读性、逻辑清晰

3. 重构提高代码质量

减少bug、提升代码运行性能

理论支撑

```
graph TD; A[理论支撑] --- B[6大设计原则]; A --- C[23种设计模式]; A --- D[重构方法];
```

6大设计原则

23种设计模式

重构方法

6大设计原则

```
graph LR; A[6大设计原则] --- B[单一职责原则]; A --- C[里式替换原则]; A --- D[依赖倒置原则]; A --- E[接口隔离原则]; A --- F[迪米特法则]; A --- G[开闭原则];
```

单一职责原则

里式替换原则

依赖倒置原则

接口隔离原则

迪米特法则

开闭原则

23种设计模式

创建型：

工厂方法模式、抽象工厂模式、单例模式、建造者模式、原型模式

结构型：

适配器模式、装饰器模式、代理模式、外观模式、桥接模式、组合模式、享元模式

行为型：

策略模式、模板方法模式、观察者模式、迭代子模式、责任链模式、命令模式、备忘录模式、状态模式、访问者模式、中介者模式、解释器模式

重构方法

- 1) 重新组织函数（提炼函数、替换算法）
- 2) 在对象间搬移特性（提炼类、搬移函数）
- 3) 重新组织数据（以对象取代数据、封装字段、以对象取代数组）
- 4) 简化条件表达式（分解表达式、以多态取代条件表达式）
- 5) 简化函数调用（函数改名、添加参数）
- 6) 处理概括关系（字段上移、提炼接口、提炼子类）

Part 2 大爆炸重构

- 大爆炸简介
- 大爆炸重构的过程
- 大爆炸重构结果

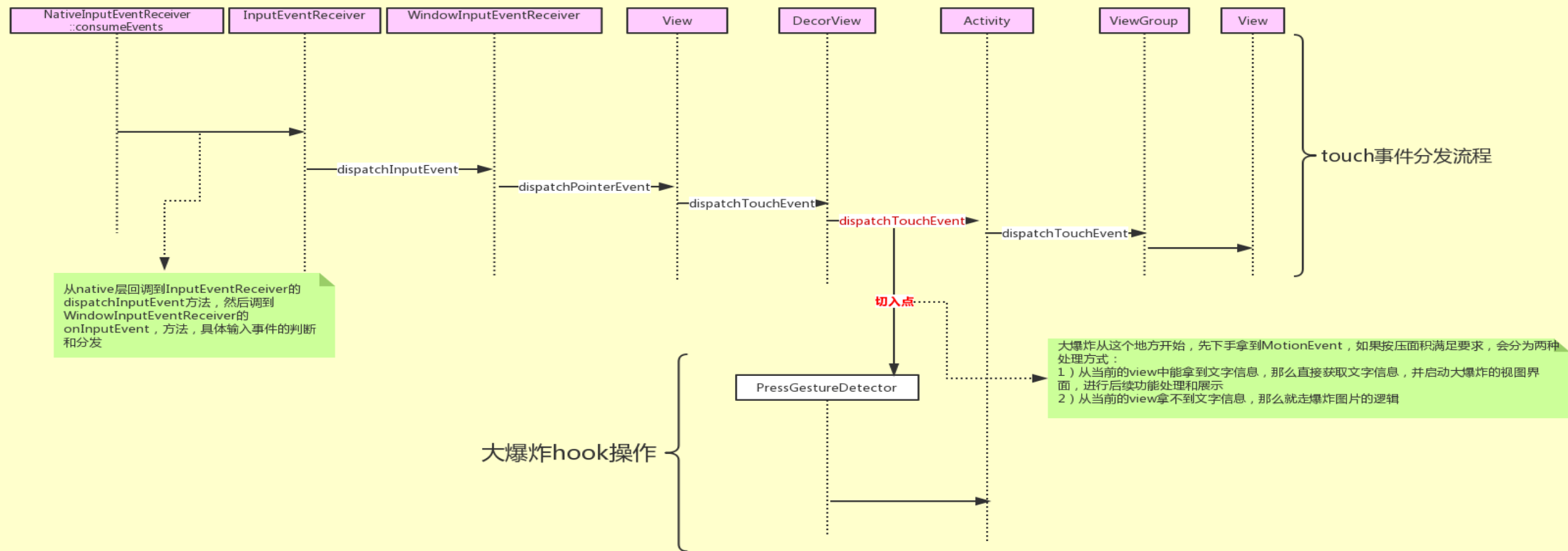
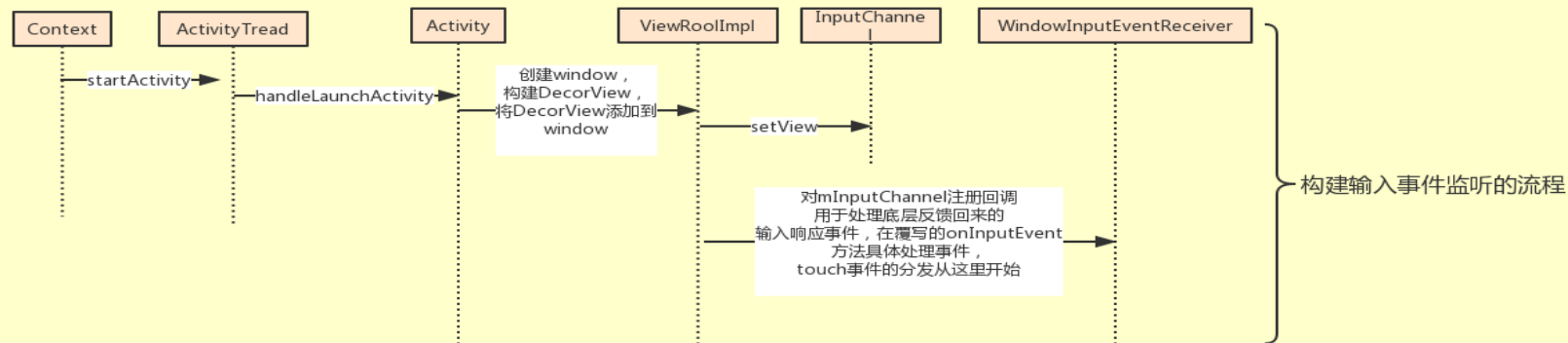
大爆炸简介

• 官方介绍

大爆炸瞄准了智能手机在编辑信息环节的一大痛点，只要用拇指大面积按压屏幕中的文字，“大爆炸”会将按住的那段文字全部“炸”开，按照语义智能拆分成独立的字和词，选择文字即可搜索、分享和复制，开创性地解决了在手机屏幕上难于处理文本信息的用户痛点。

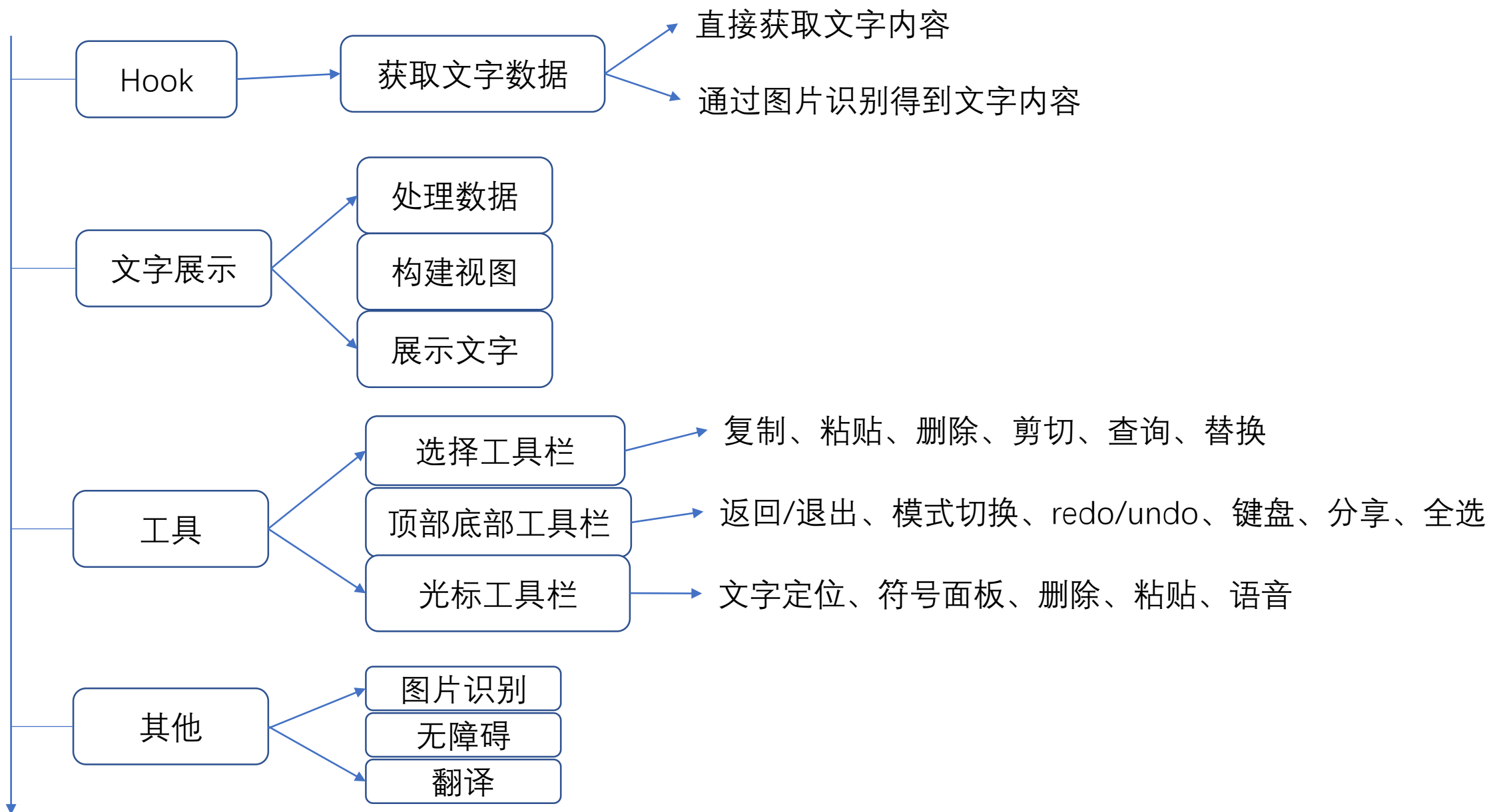
• 理论简述

用类似hook的方式，截取触摸事件分发，进而获取文字数据，最后对获取的文字数据进行各种处理



理论流程分析图

• 功能模块介绍



大爆炸重构的过程

重构之前存在的问题

- UI控制层职责不明确，未能实现调控各个view模块的作用
- View模块之间调用关系混乱
- View模块的数据处理未做隔离
- Activity职责太复杂，引用泛滥
- 功能类存在太多强关联

总结就是

类功能不单一，接口使用目的不明确，模块隔离度不够

重构方案和策略

1. 分包

按功能模块进行分包，将关联密切的类划分到同一包下

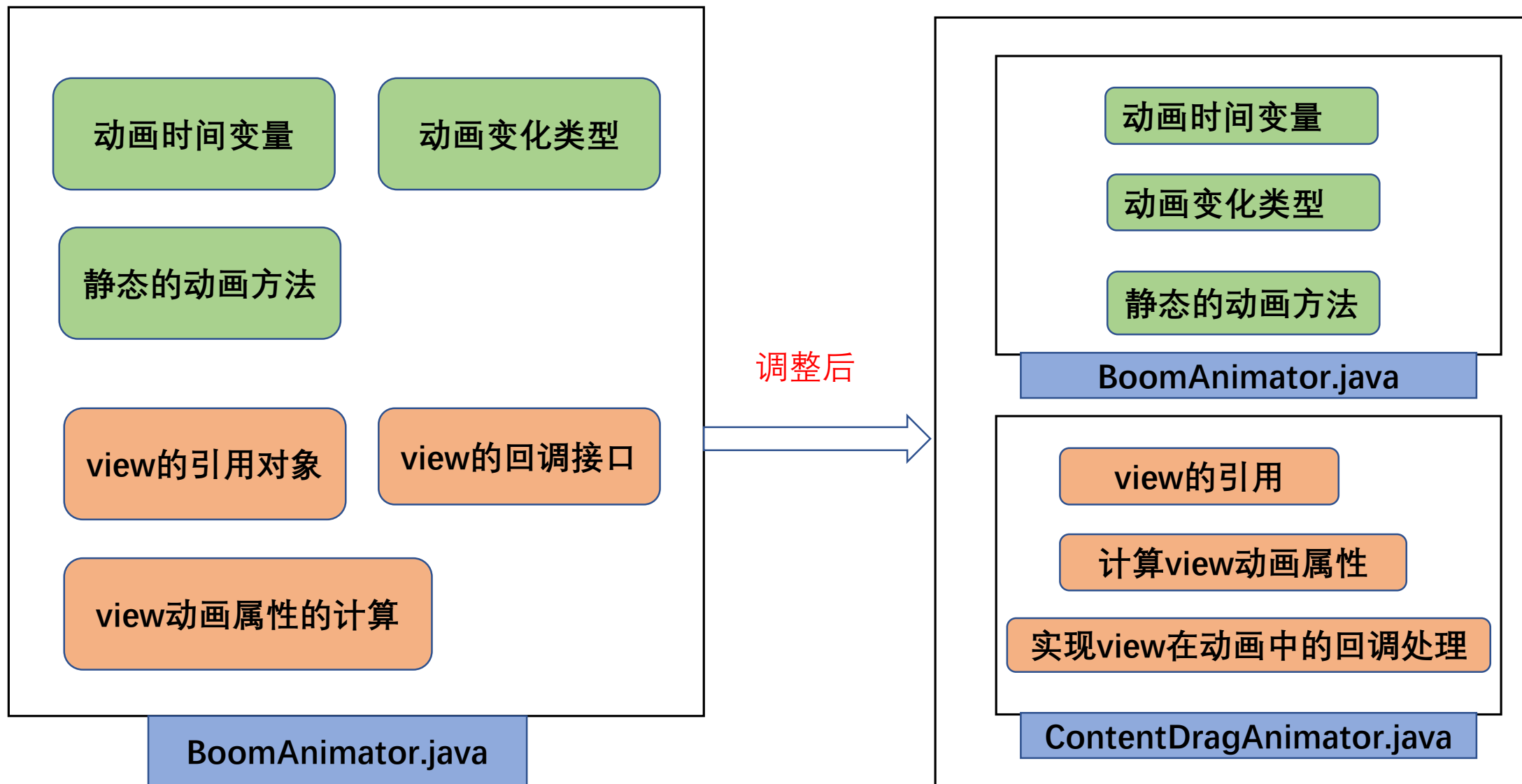


调整后



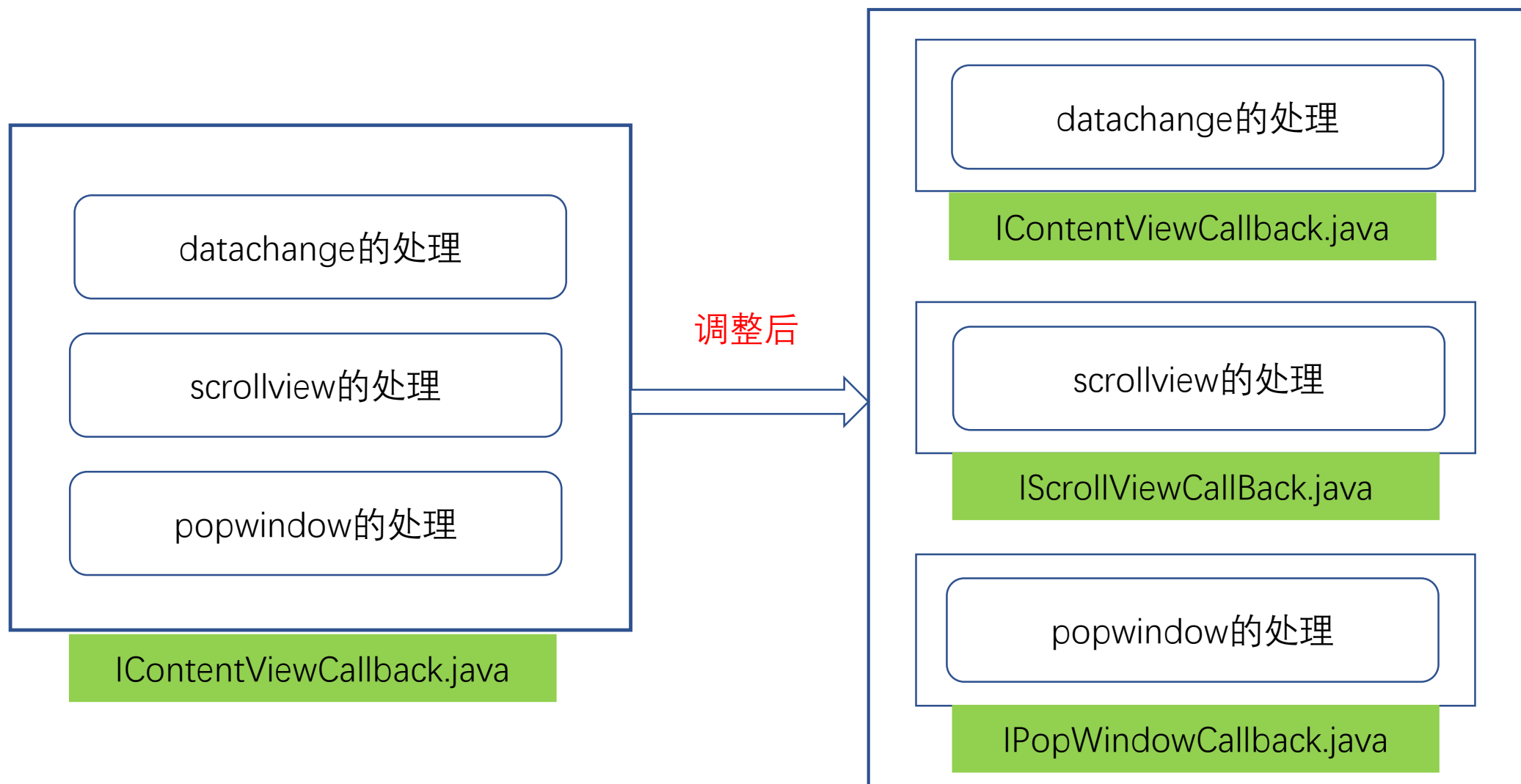
2.按单一职责进行类提取和抽离

BoomAnimator.java



3.接口隔离

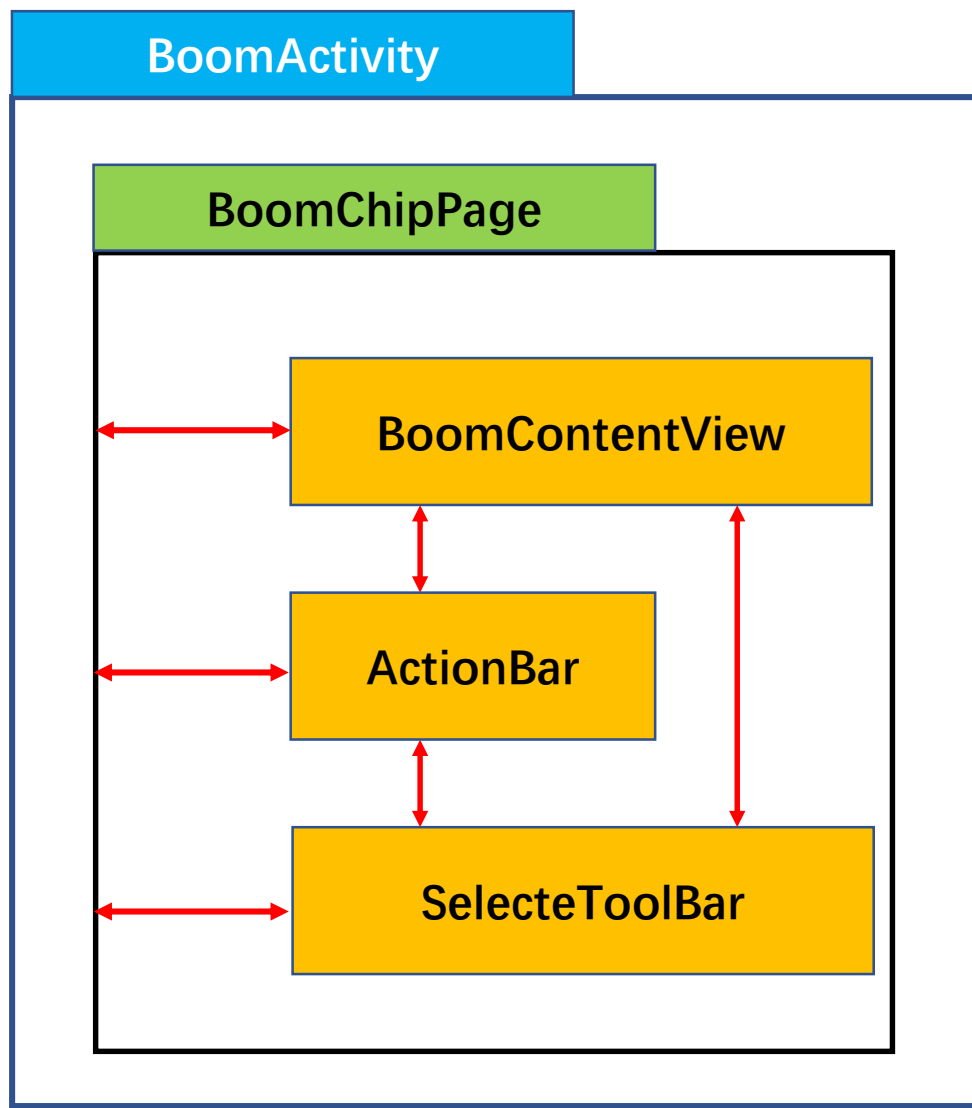
细化接口的功能，将完成单一功能模块的方法聚合在一个接口里面



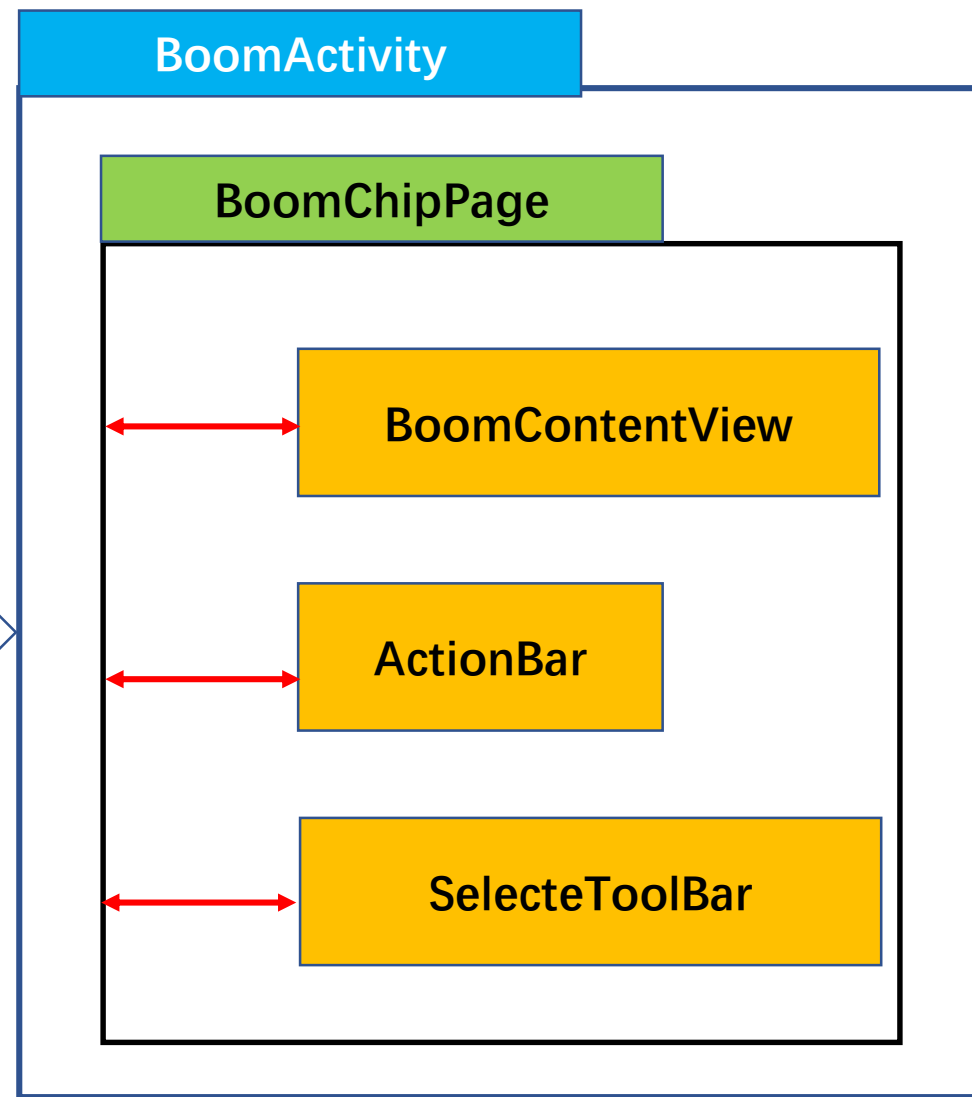
4.迪米特原则

一个对象应该对其他对象保持最少的了解

- a) view模块彼此间应该隔离，去掉相互引用;
- b) 属于view内部自己相关的功能就内聚在自己的类里;
- c) 与其他模块的view进行交互，都统一交给UI管理者进行调度;
- d) 与UI管理者的交互通过接口回调来完成



调整后



大爆炸重构结果

- 完成重构之初指定的计划（改进项目结构、修改优化代码）
- 用开闭原则、单一职责等原则进行项目的构建
- 测试验证重构代码，确保重构质量

Part 3 心得分享

1、统一思想

开发规则的统一

明确类、方法、接口粒度大小

2、节奏控制

划分任务、确立目标

合理分配bug、需求、重构的时间

组员之间有效的配合

3、方法策略

分清任务主次

善于借鉴优秀的解决方法

自测验证不可缺少

4、重构时机

解决bug时

做需求时

发现更加合理和优秀的实现方法时

重构一直在路上！ 大家共勉！ 谢谢！