

Tangram—天猫导购页面动态 化方案与实践

灰风@天猫无线



设计理念 概念模型 实现原理 拓展功能 落地实践

天猫导购页面动态化方案与实践



设计理念

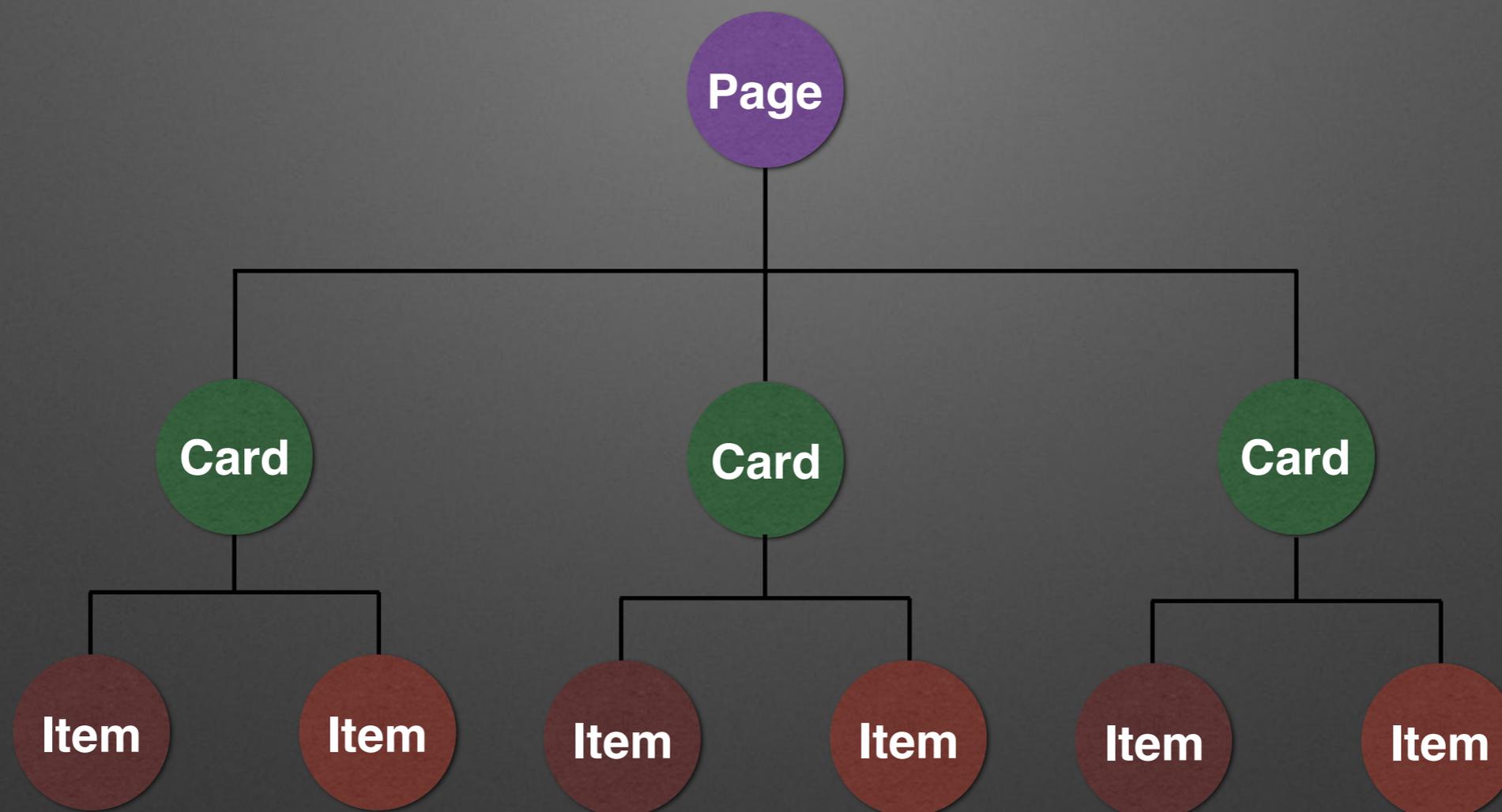
- 具备一定足够灵活性，满足运营日常需求，接近native体验与性能
 - 布局能力动态化
 - 组件业务化
 - 动态能力粗粒度化
 - 组件复用



概念模型



概念模型



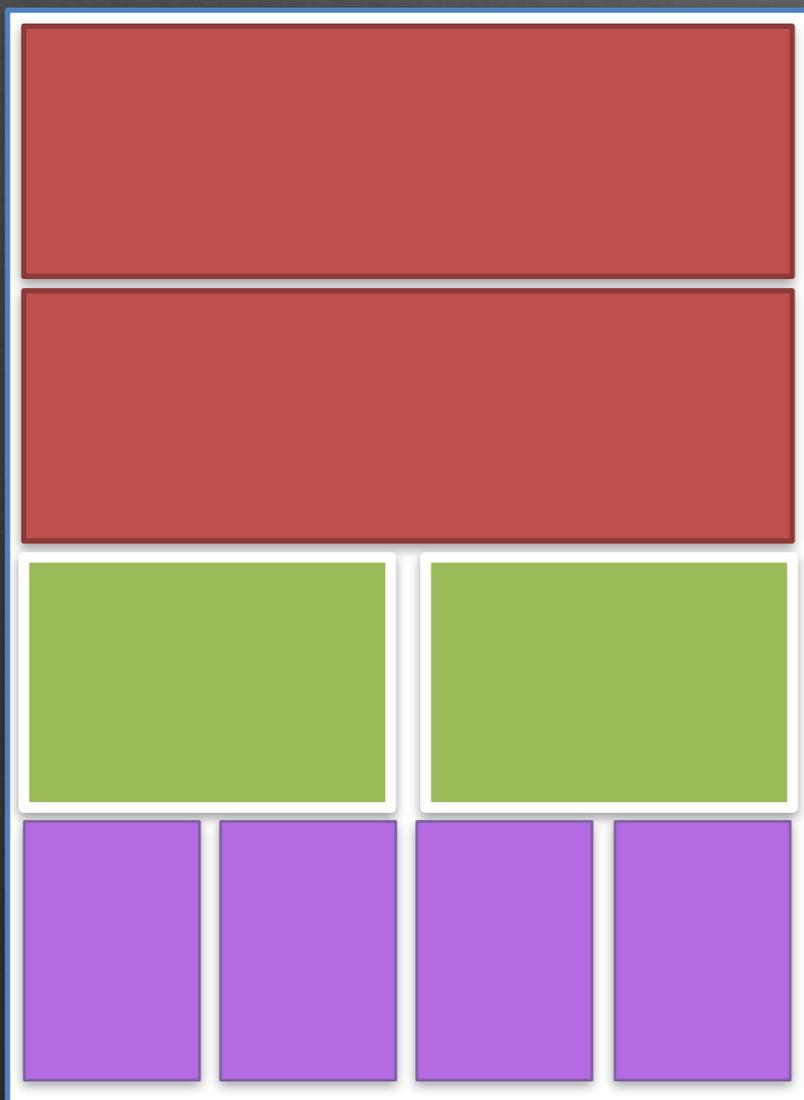
概念模型——卡片

- 负责布局逻辑，粗粒度化，不提供布局细节描述，只声明布局方式
 - header（可选）
 - footer（可选）
 - body（必选）
 - 流式布局
 - 瀑布流布局
 - 吸顶布局
 - 固定布局
 - 悬浮布局
 - 轮播布局
 - 样式（可选）

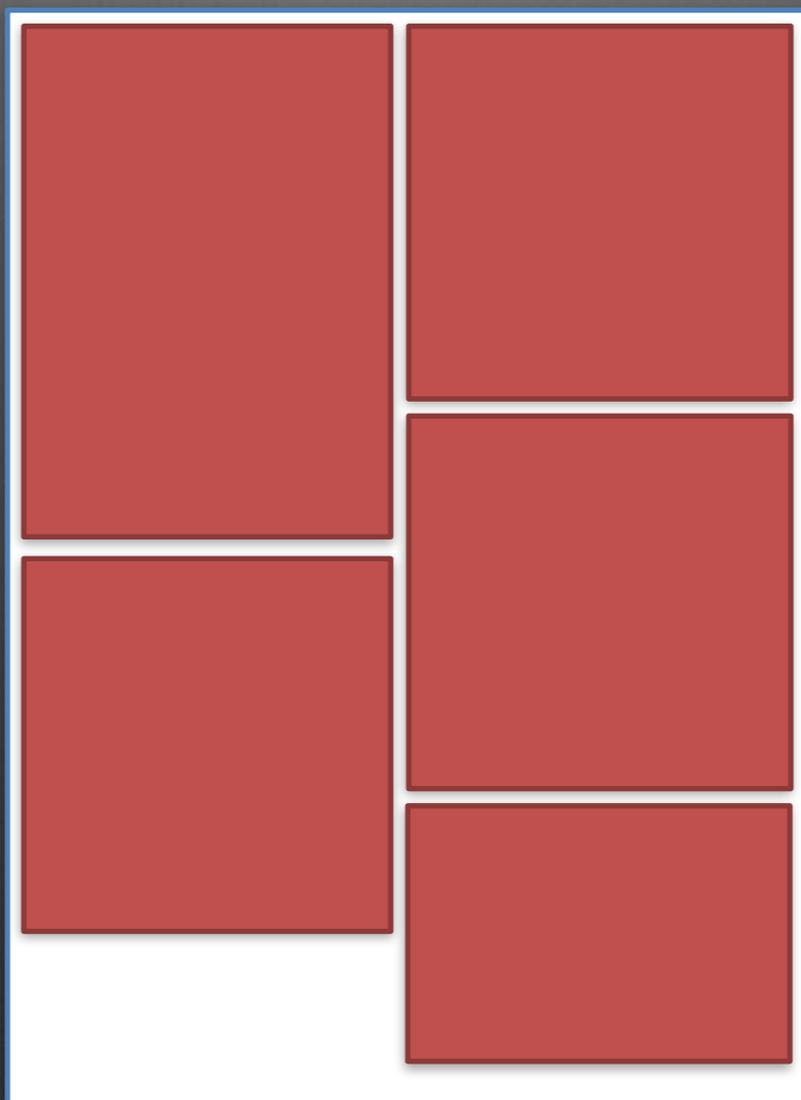


概念模型——卡片

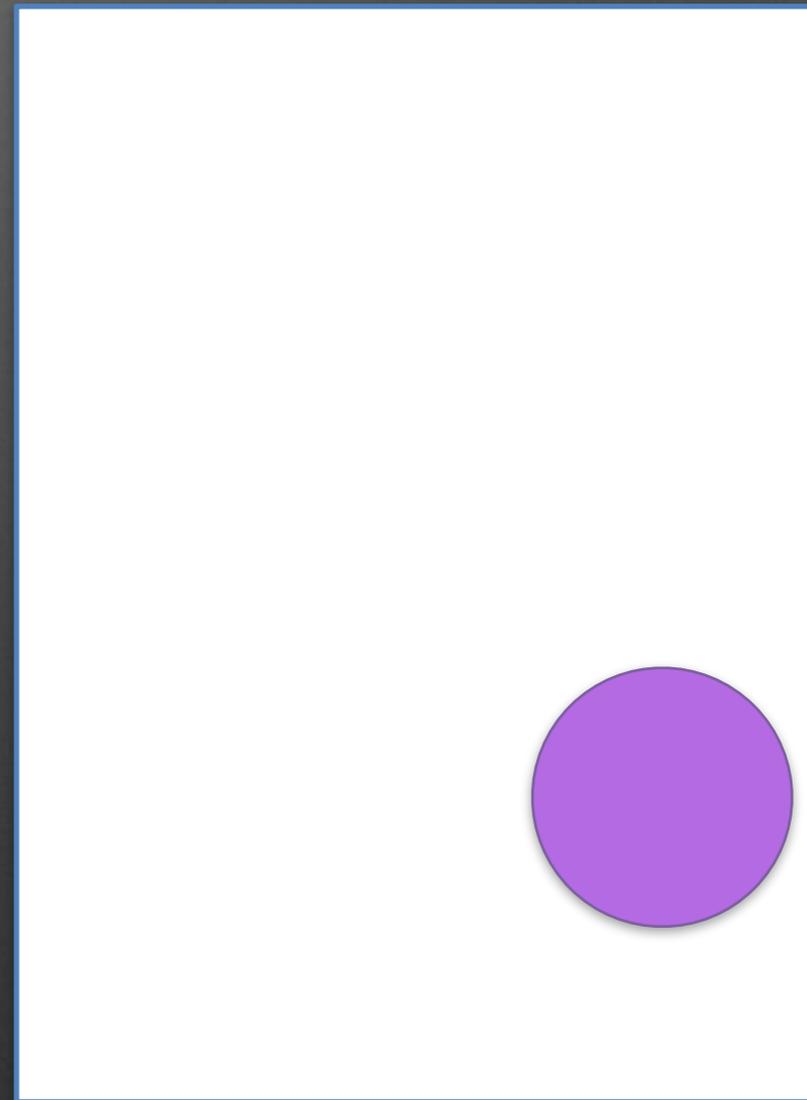
列布局



瀑布流



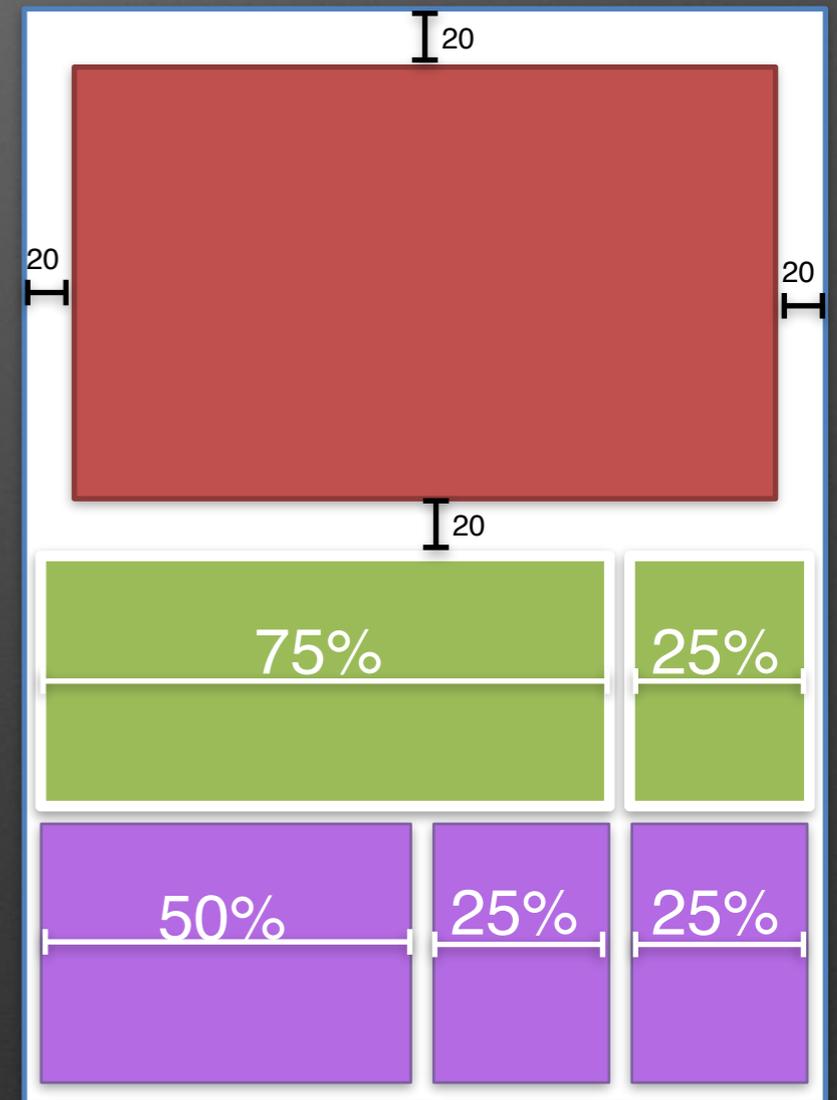
悬浮



概念模型——卡片

卡片的基本样式

- backgroundColor
- margin/padding
- gap
- cols



概念模型——组件

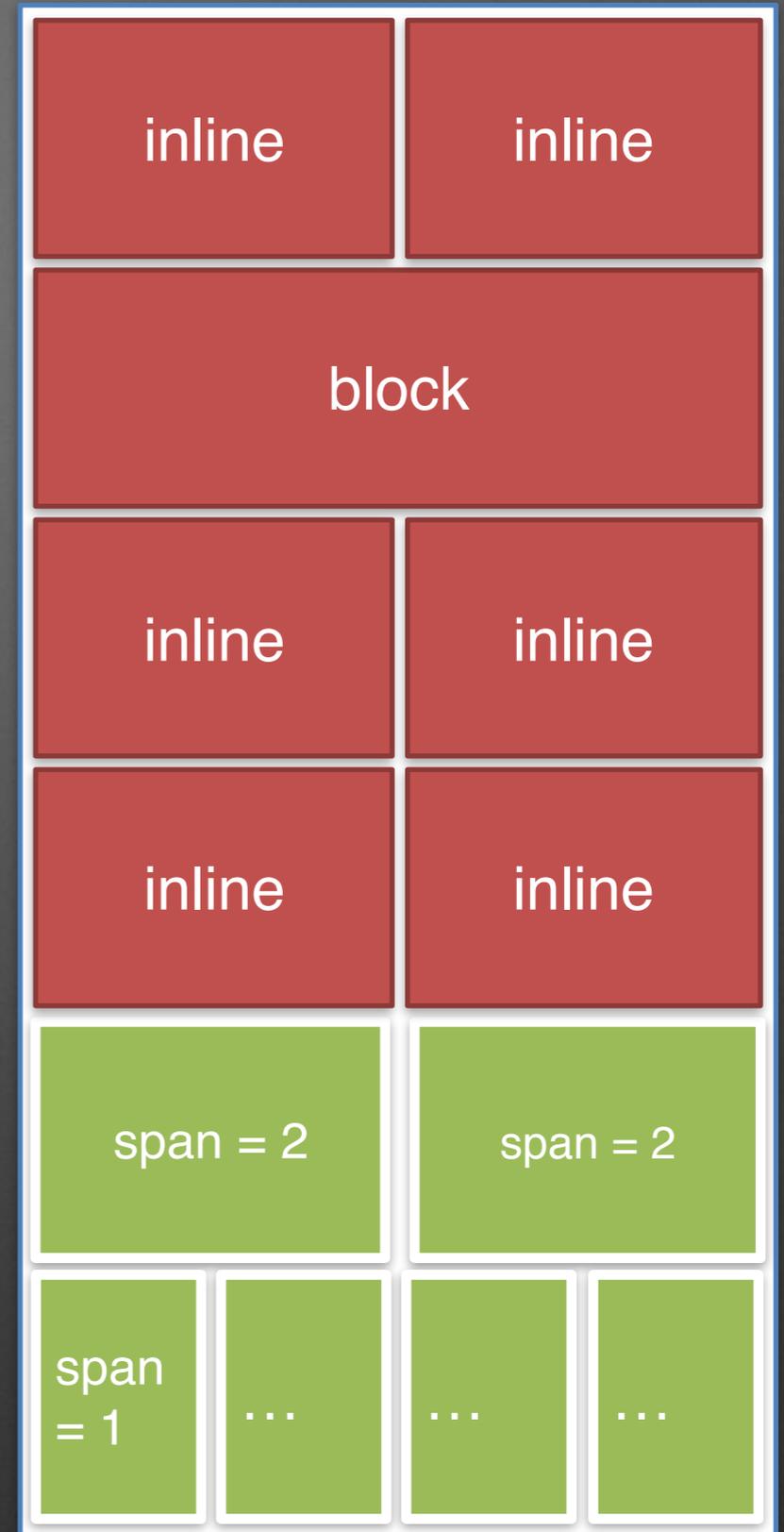
- 承载业务逻辑和UI元素展示，尽可能小的业务单元为粒度
 - 统一的ViewModel
 - 不同的View
 - 生命周期定义
 - init
 - bind
 - unbind
 - 样式



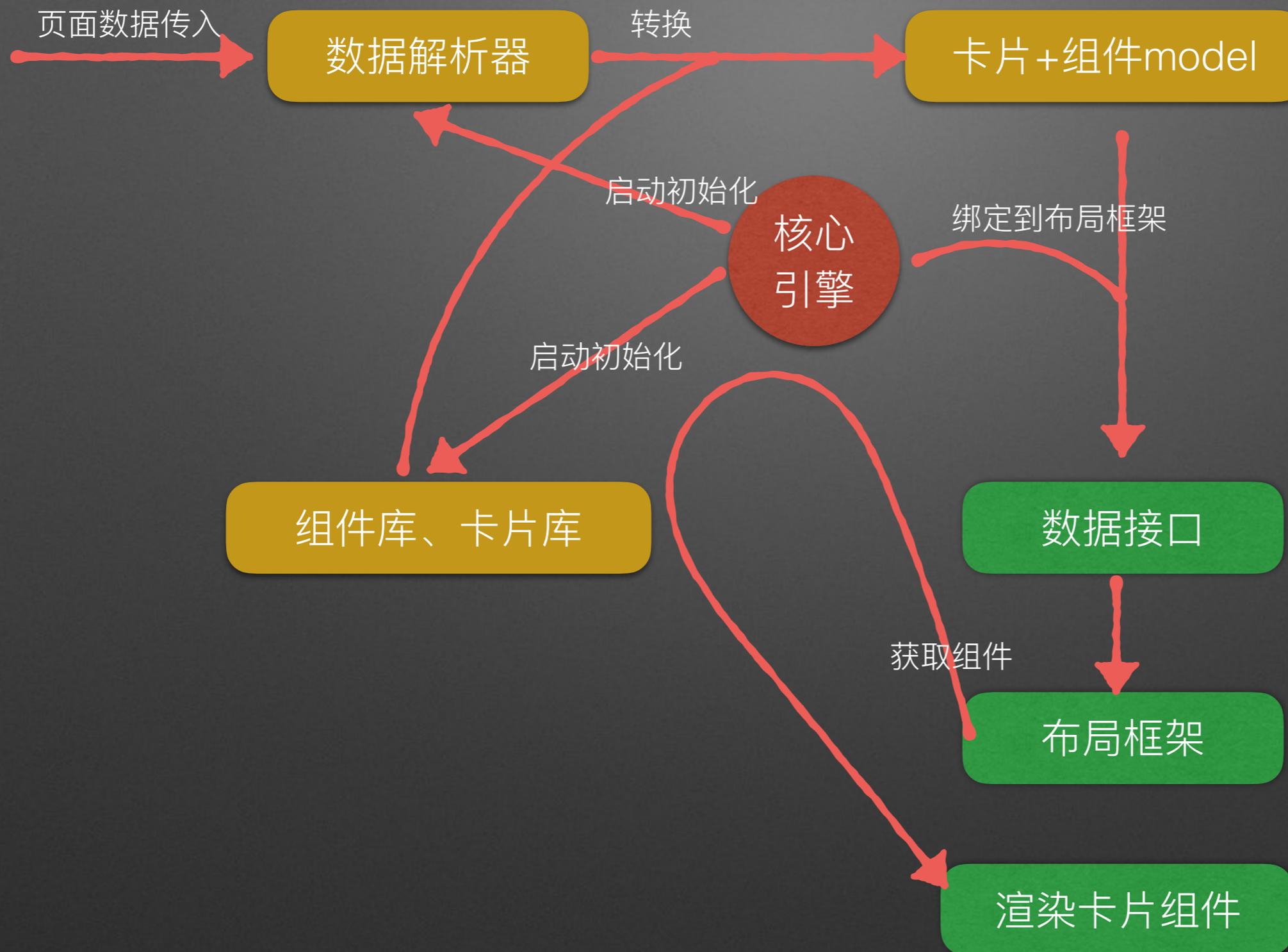
概念模型——组件

组件的基本样式

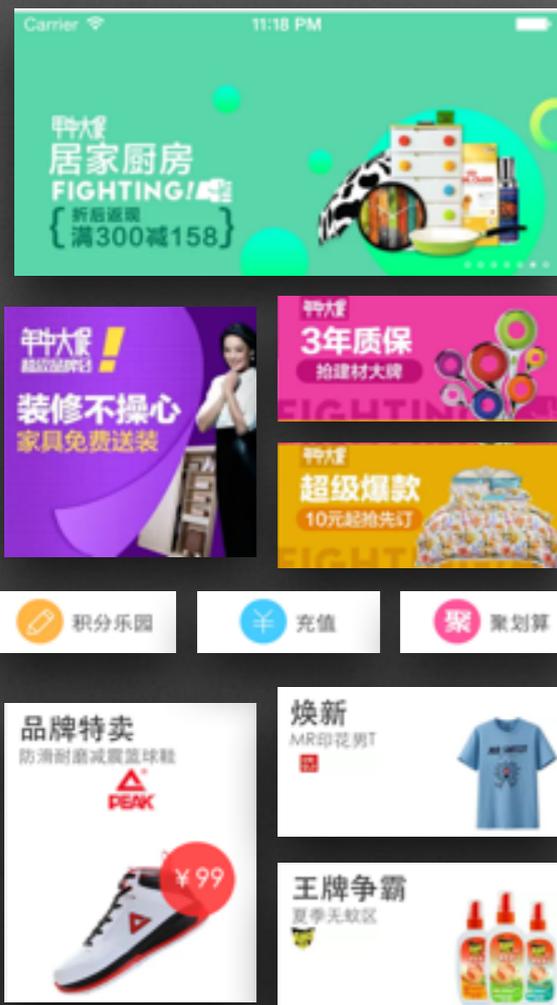
- backgroundColor
- margin/padding
- display
- colspan
- width/height



实现原理



实现原理



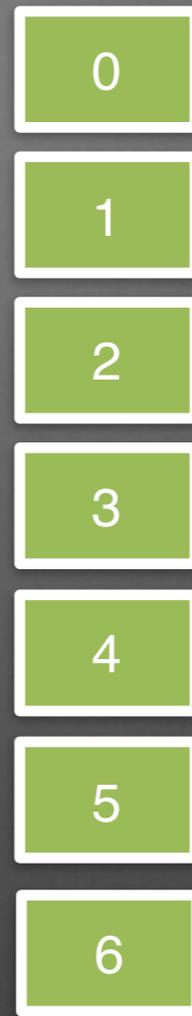
- 核心
 - Android基于 RecyclerView+自定义LayoutManager的方式
 - iOS基于自定义的LazyScrollView
 - 扁平化
 - 跨卡片的组件级别复用
 - 提供默认的系统实现
- 扩展
 - 外部注入



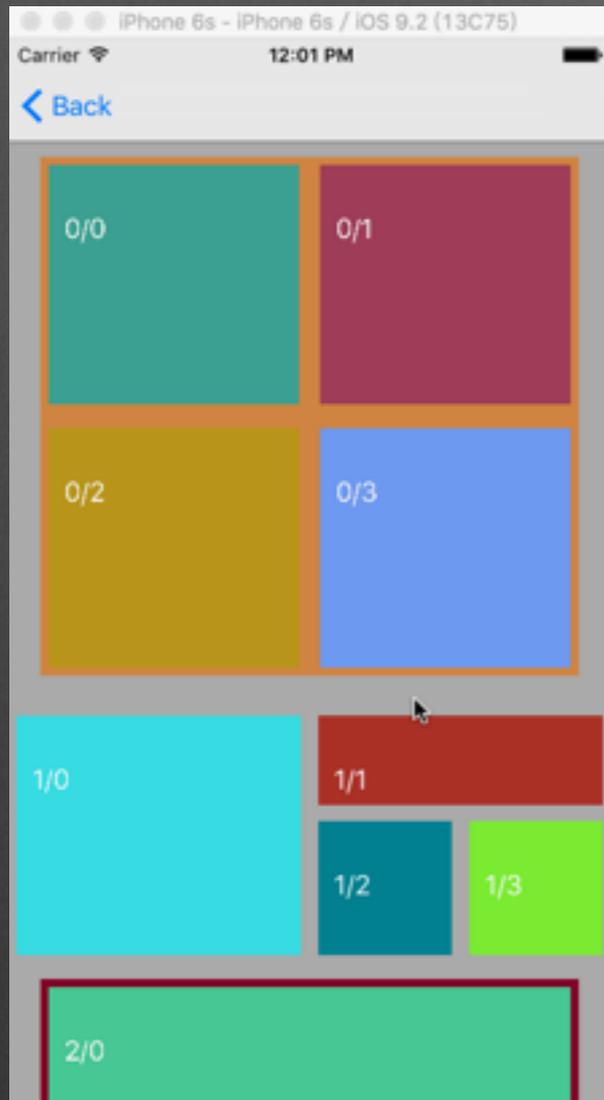
实现原理(Android) 数据

adapter

layoutManager



实现原理(iOS)



- 根据DataSource获取所有组件的位置
- 根据顶边(y)升序排序的索引和根据底边(y+height)降序排序的索引。
- 根据当前滚动距离取屏幕内的组件
- 回收不在可见区的组件，复用或生成新的组件



扩展

- 插件化扩展，外部可注入扩展模块、扩展卡片、扩展组件
 - 点击处理模块
 - 曝光处理模块
 - 通用定时器模块
 - 使用事件总线，做组件联动
 - 脚本动画
 - 纯动态组件
 - weex
 - weapp
 - 虚拟化view
 -
 - 通用请求



落地实践——客户端

- 首页等多块业务
 - 协议、规范统一：一份数据多端使用
 - 容错与保护：使用安全方法、数据解析保护、数组访问保护、空字段过滤
- 组件库
 - 多个业务共用组件：首页与猫超共用组件
- 容器化
 - 服务接入和客户端页面容器化，无开发搭建新页面：搜索专辑、推荐专辑
- 插件与解耦
 - 整合其他技术能力：动画库、weex



落地实践——日常运营

- 专门的管理后台
 - 应用、系统平台、版本、渠道多维度管理页面
- 稳定性
 - 预发布
 - 白名单预览
 - 时间机器预览
 - 发布自动对接流程平台
 - 权限管理
- 效率
 - 定时发布
 - 一键上下线页面、卡片
 - 批量复制创建卡片数据
 - 版本通配
- 快速试错
 - 页面、卡片级别的abtest能力



谢谢

团队博客：<http://www.pingguohe.net/>

