



WEEX Conf

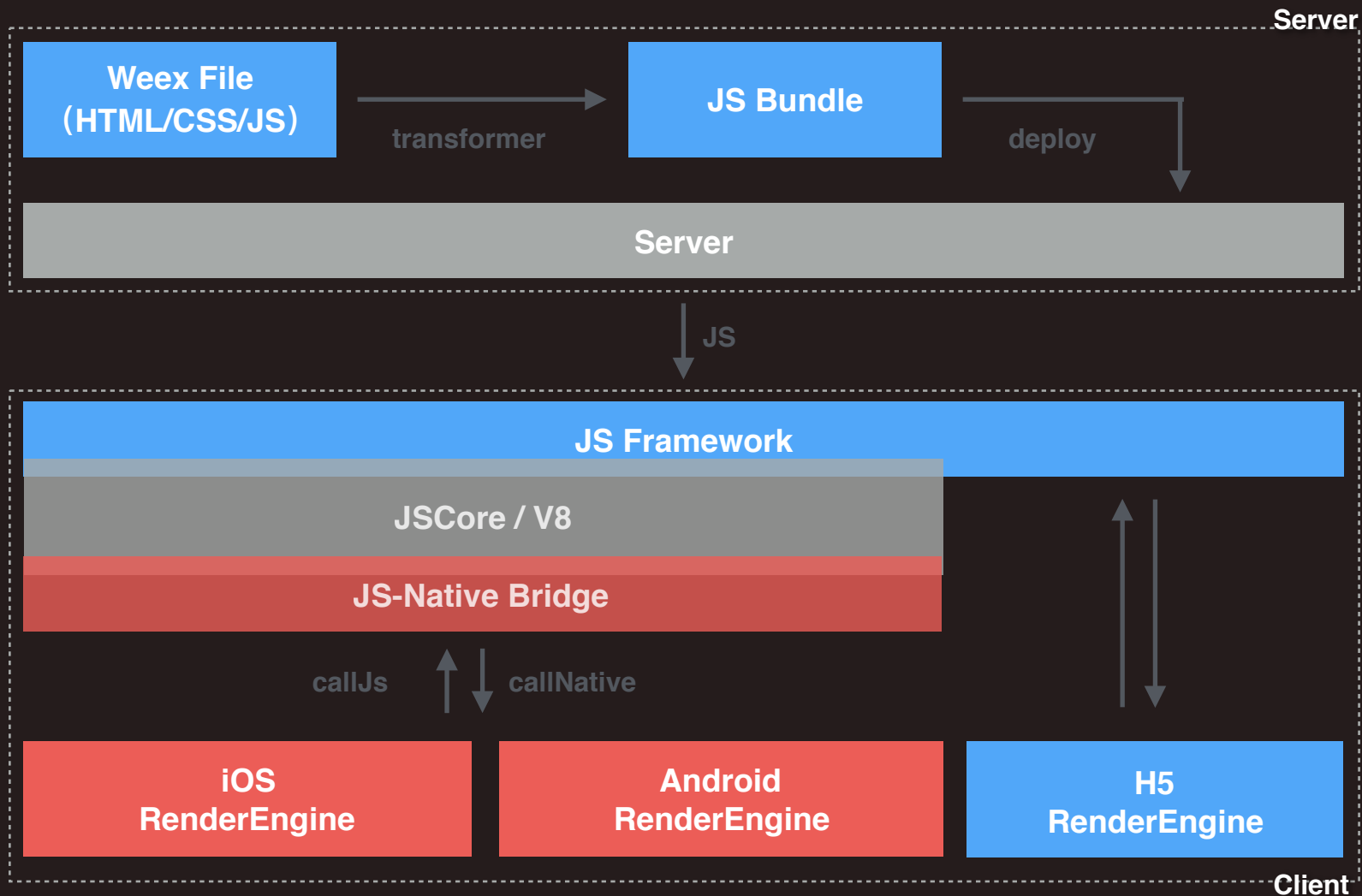
Native 性能稳定性极致优化

郑士汉（花名：饮源）

目录

- Weex简明架构
- 性能优化
- 最佳实践

简明架构图



性能优化

提升用户体验

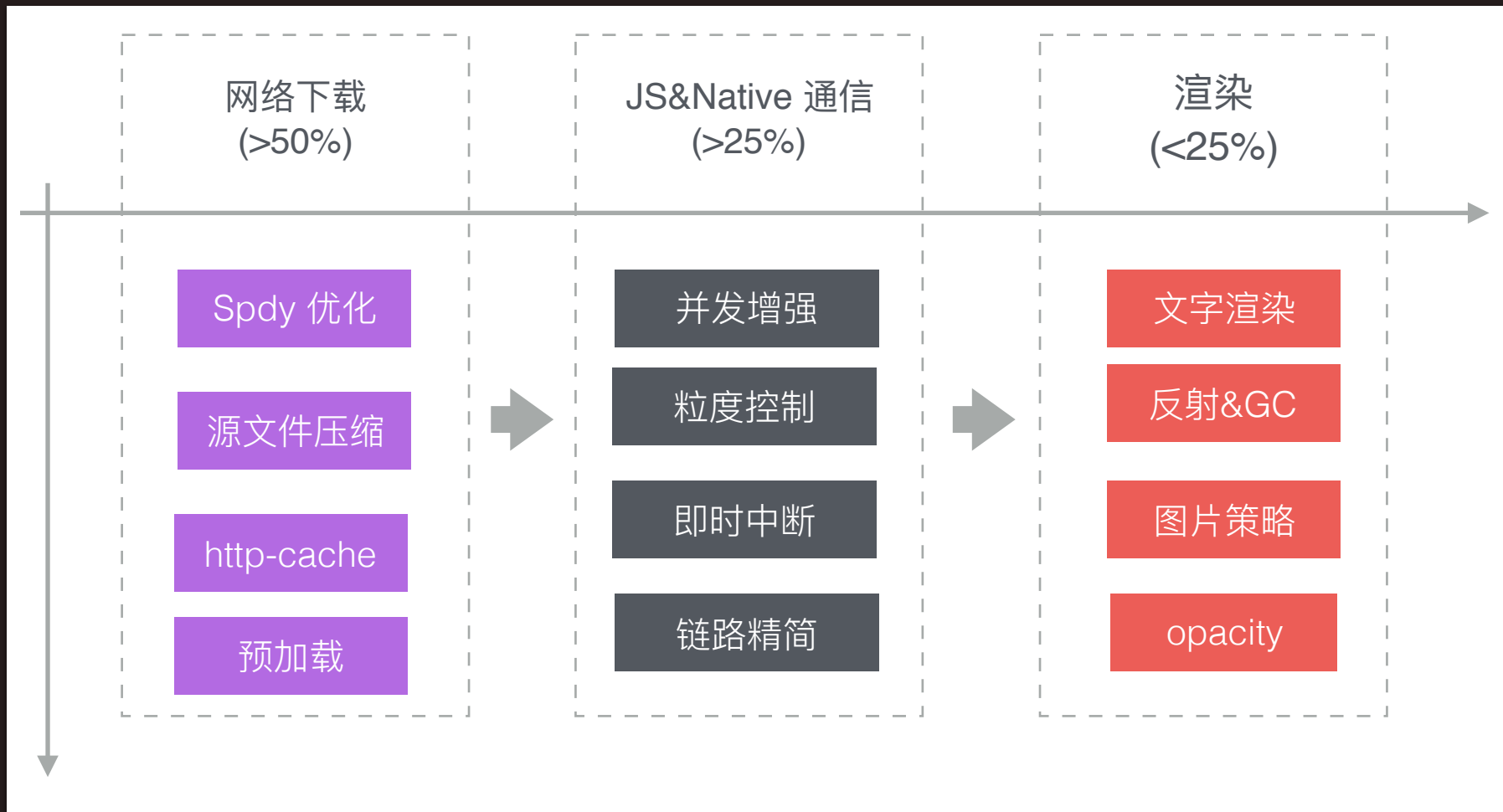
性能优化

网络 + 首屏渲染 < 1S

问题与挑战

- JS bundle下载慢（压缩后60k, NetworkTime>800ms）
- JS和Native通信效率低，拖慢加载时间
- 长页面VDom渲染时间慢，占首屏时间40%左右
- JSThread过于繁忙（Json解析&反射调用）
- JS Task无法抢占执行，导致新页面打开慢
- 复杂页面滚动帧率低的问题

Weex TimeLine



网络下载

网络优化

问题:

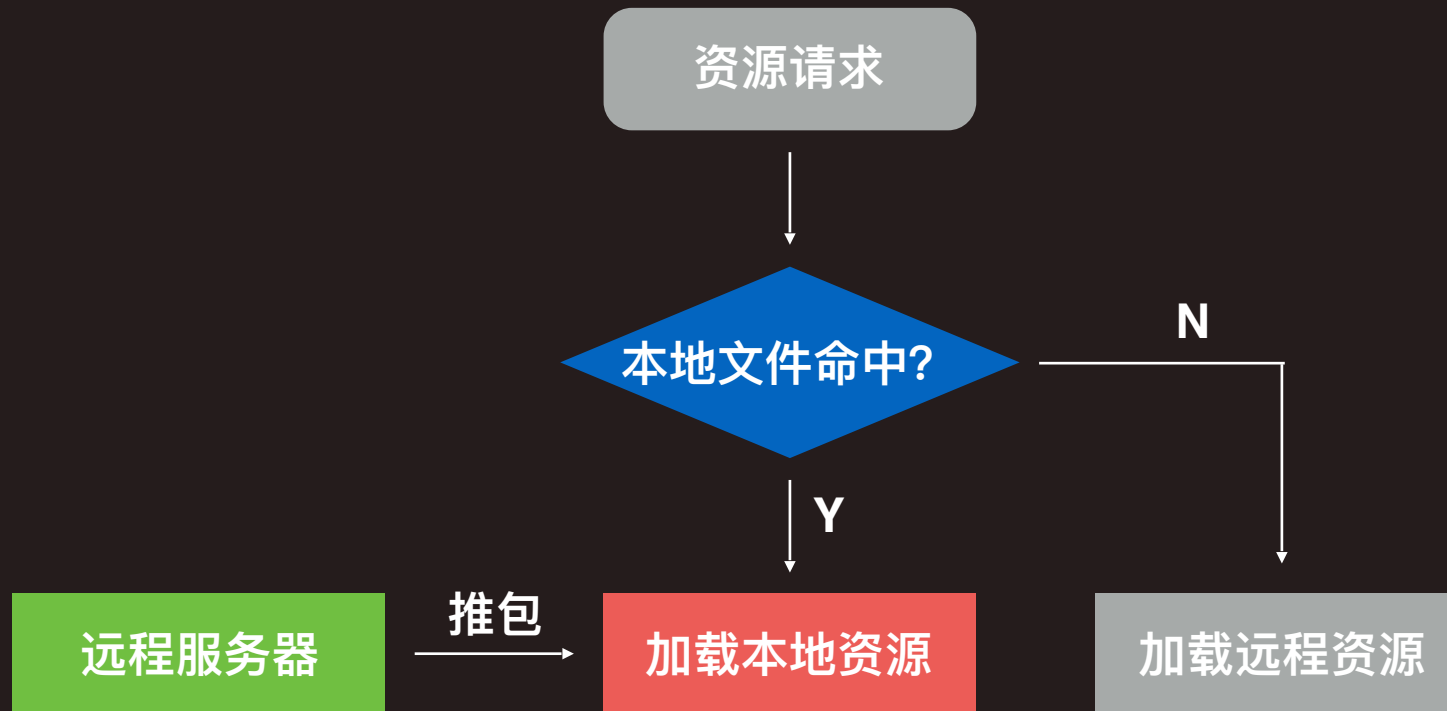
- 优化前状况: iOS 280~450ms Android 700~900ms
- 业务JS文件过大, gzip压缩后 (60~80k)
- 域名没有收敛, 配置Spdy协议
- 不支持httpCache, 200请求

线上性能数据对比:

| 日期 | App版本 | 页面名称 | 连接类型 | 样本数 | networkTime | 样本数占比 (本页) |
|------------|-------|--------------------|---------------|-----|-------------|------------|
| 2016-10-09 | 6.0.0 | DetailWeexFragment | cache | 1 | 106.69 | 0.64% |
| 2016-10-09 | 6.0.0 | DetailWeexFragment | http | 1 | 13159.59 | 0.01% |
| 2016-10-09 | 6.0.0 | DetailWeexFragment | https | 1 | 10397.83 | 1.38% |
| 2016-10-09 | 6.0.0 | DetailWeexFragment | packageApp | 1 | 40.19 | 23.55% |
| 2016-10-09 | 6.0.0 | DetailWeexFragment | spdy | 1 | 248.30 | 1.28% |
| 2016-10-09 | 6.0.0 | DetailWeexFragment | spdy_0rtt_cdn | 1 | 266.83 | 73.14% |

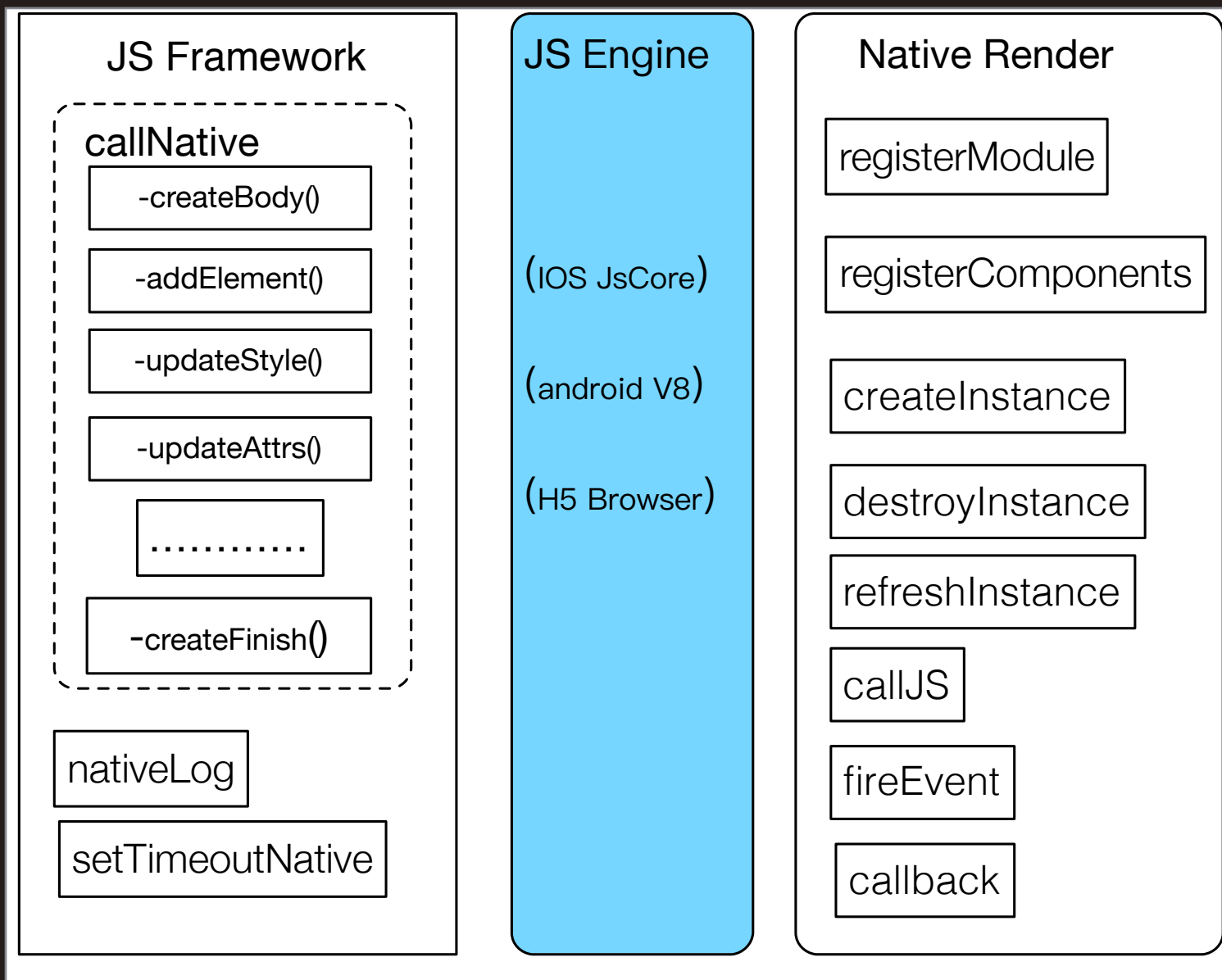
JS bundle预加载

- App启动预先下载JS
- 通过长连通道push
- 全量/增量、被动/强制更新结合
- 400ms->10ms(iOS)
- 850ms->35ms(Android)

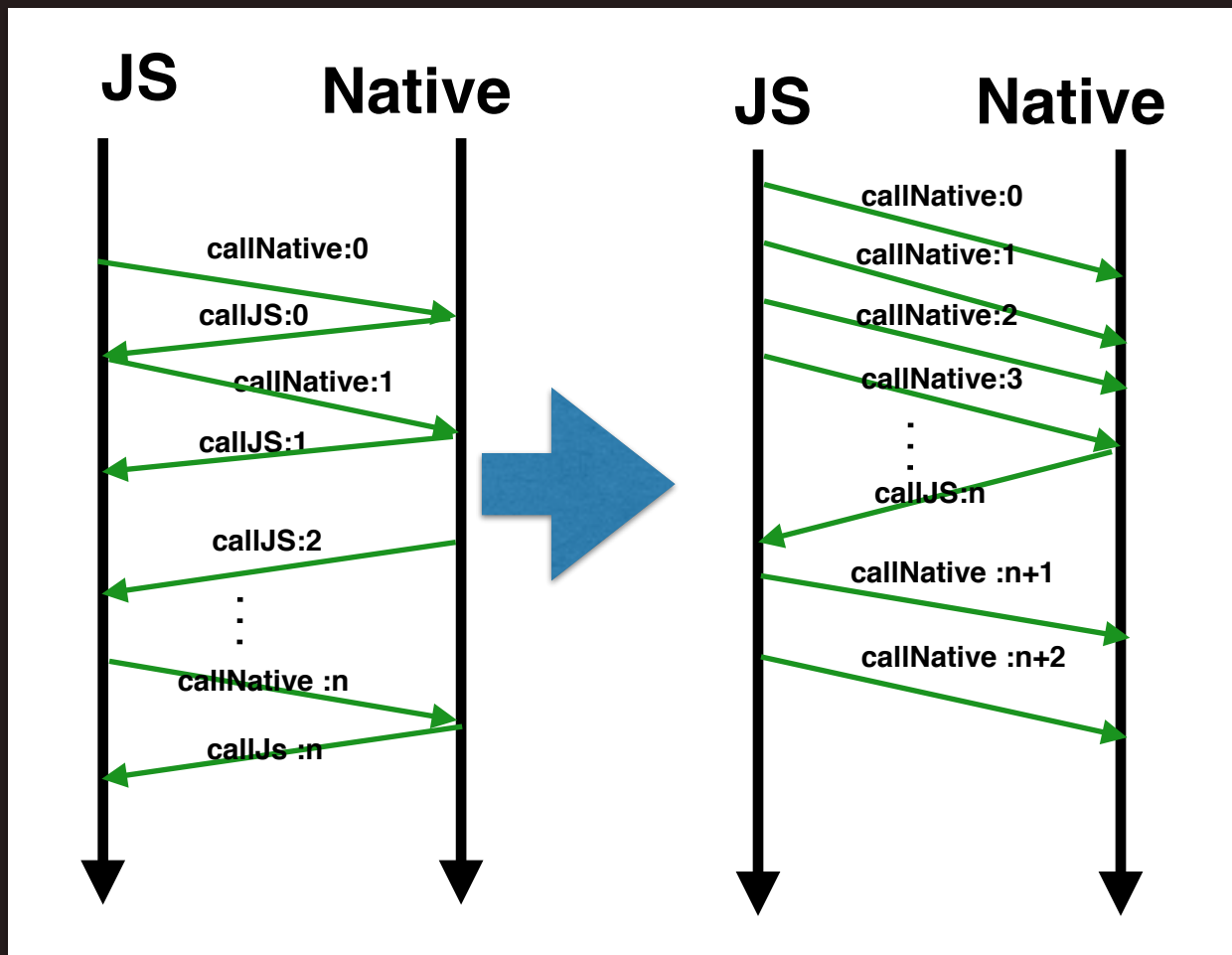


JS&Native 通信机制

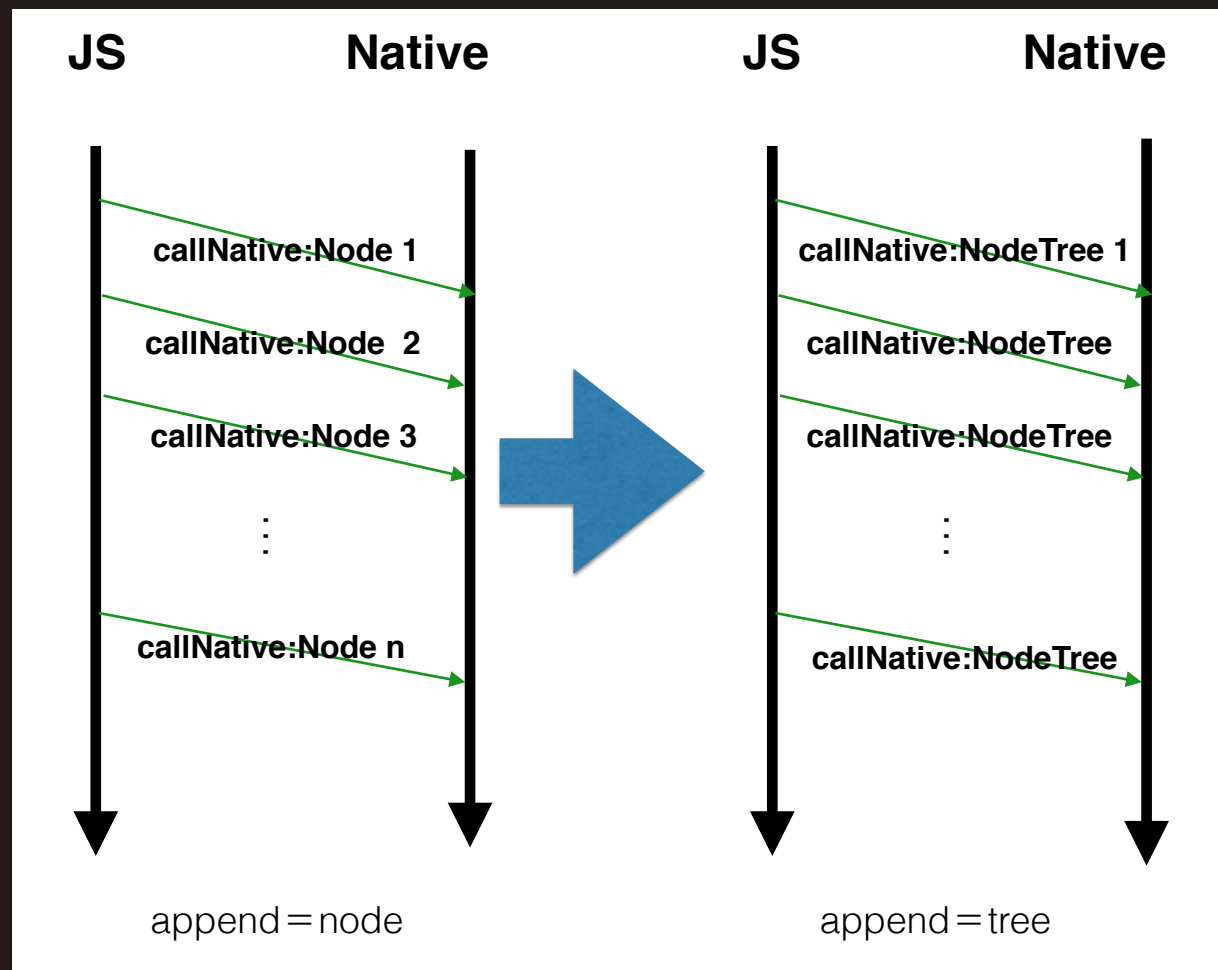
JS&Native 通信机制



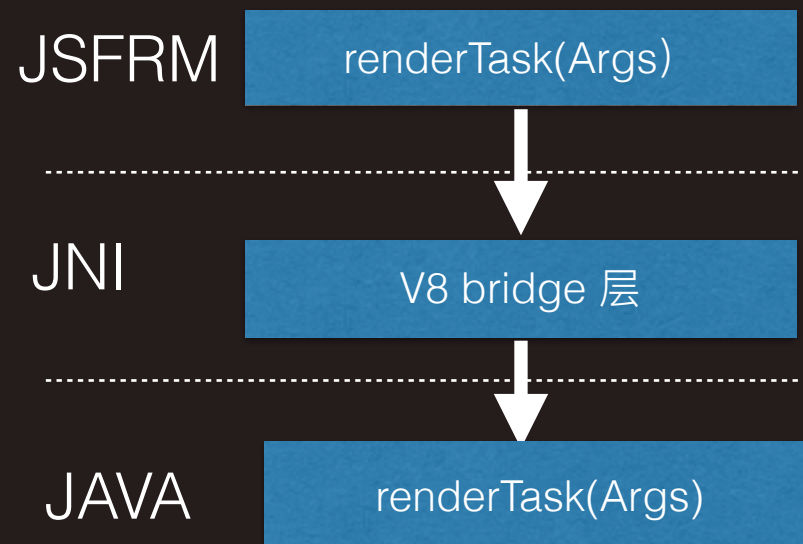
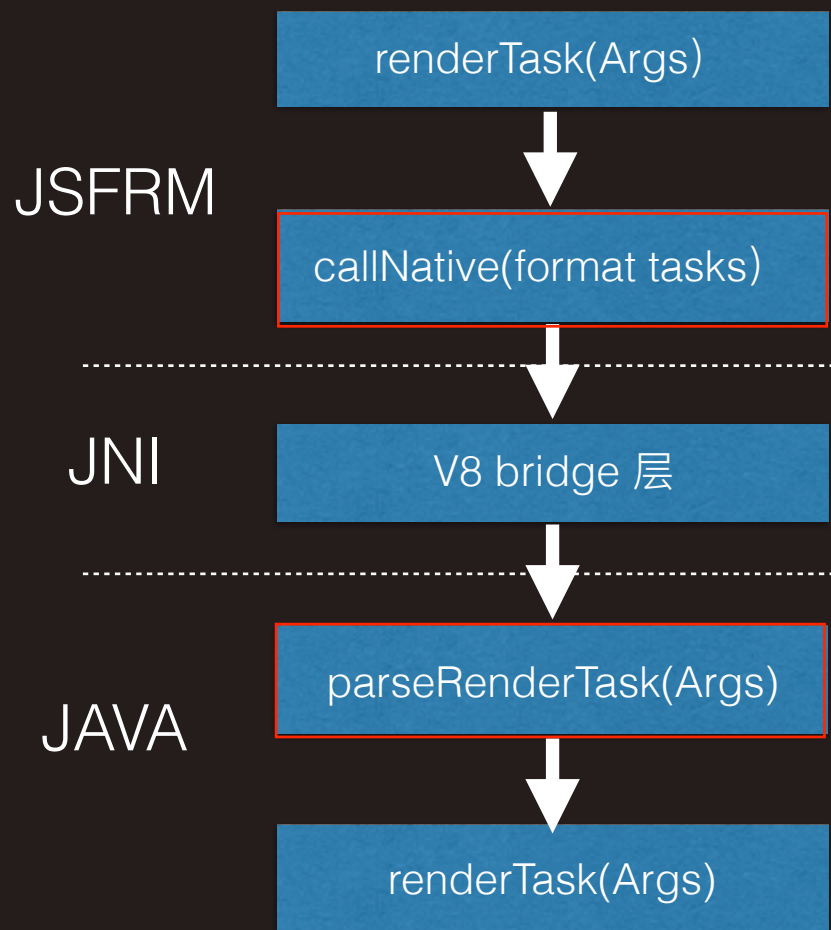
增强并发



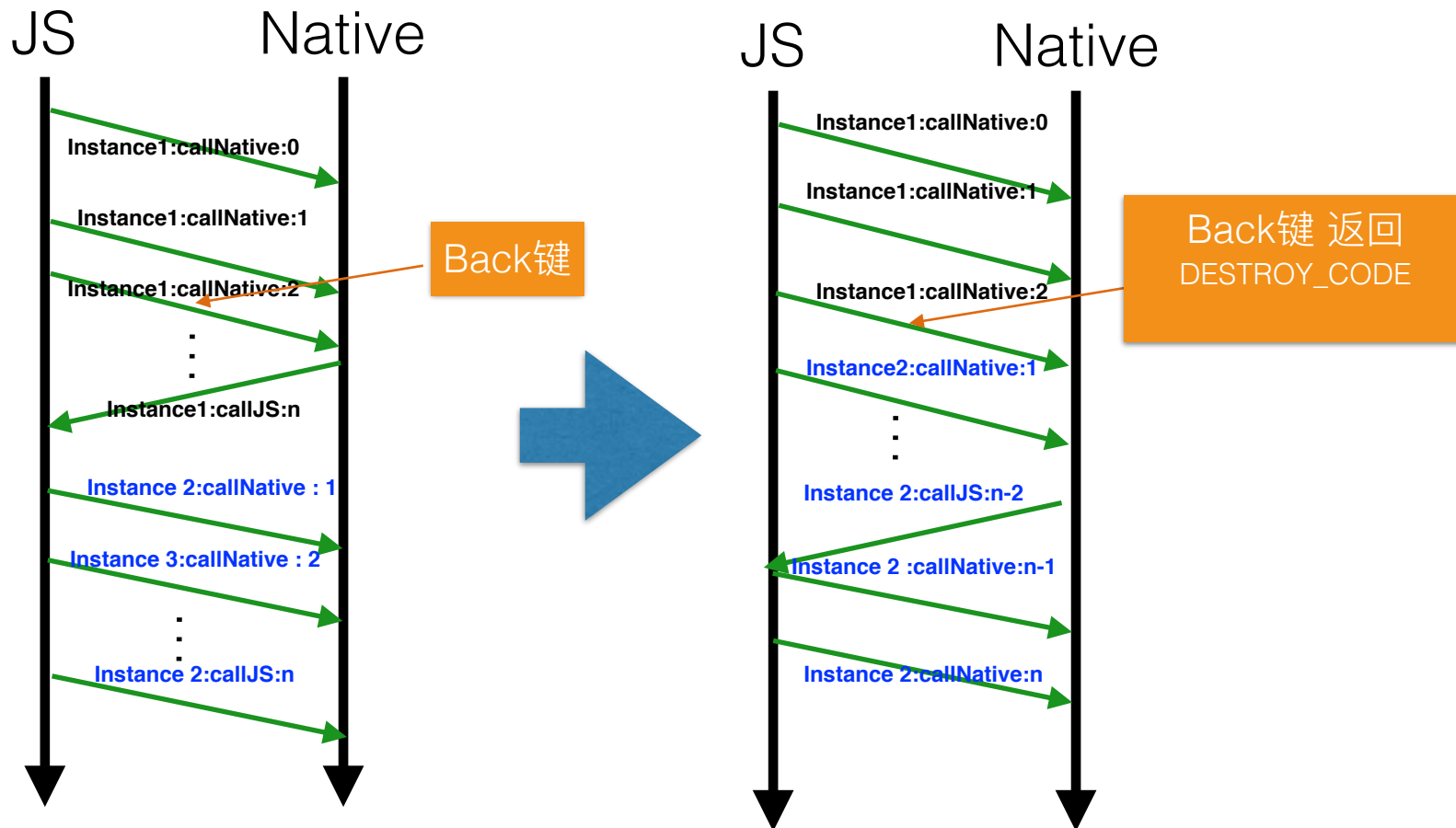
Node & Tree



精简链路



中断控制



解决长页面快速进出，无法中断的问题

渲染优化

文字渲染

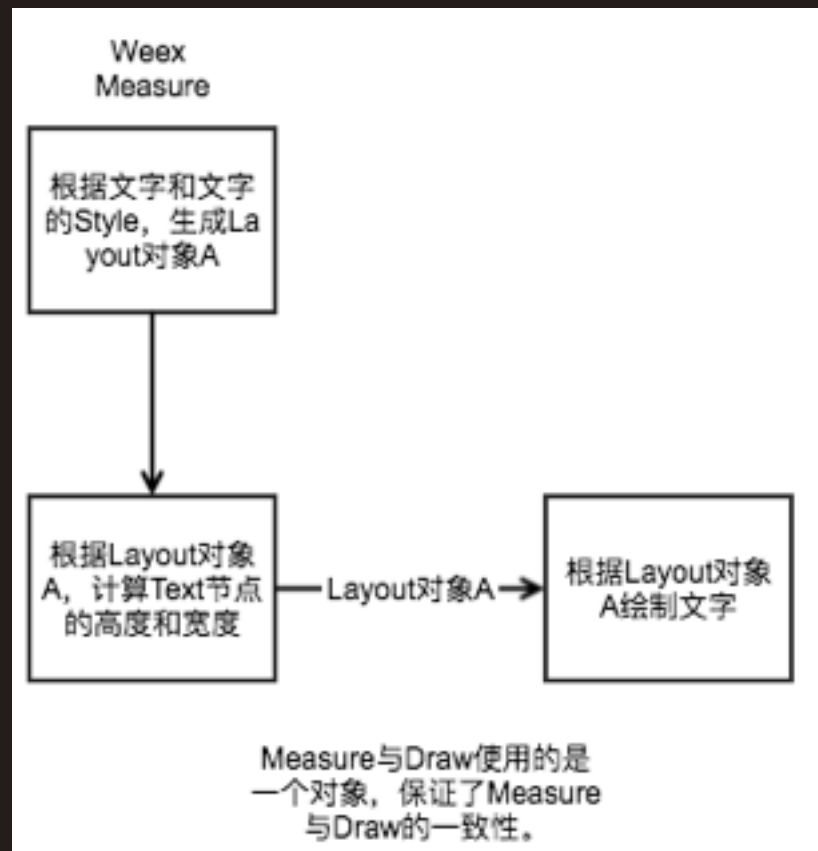
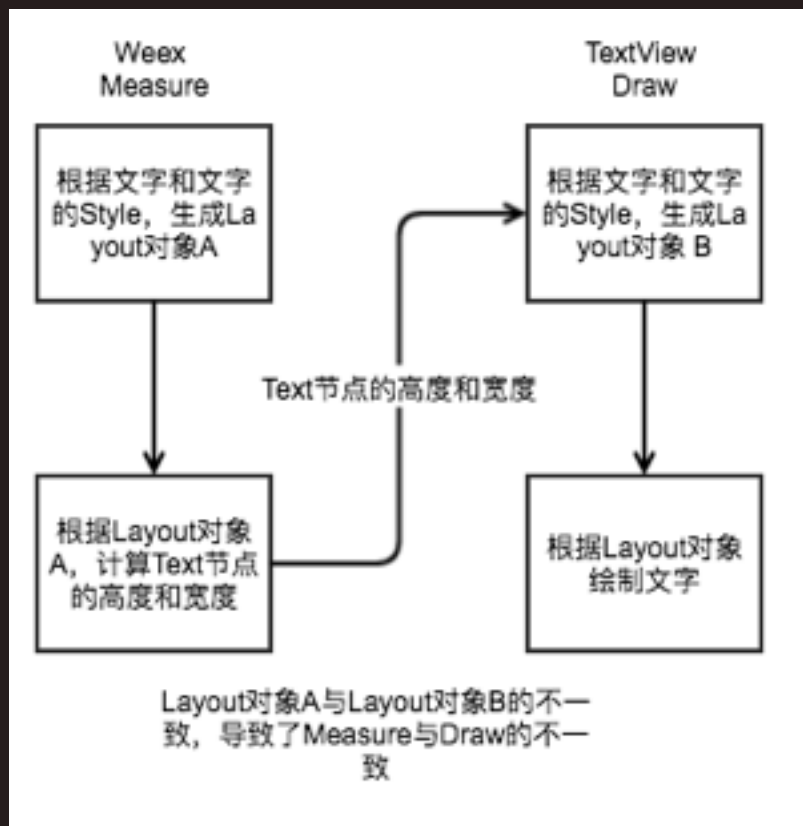
优化前(文字被截断)



优化后(正常渲染)

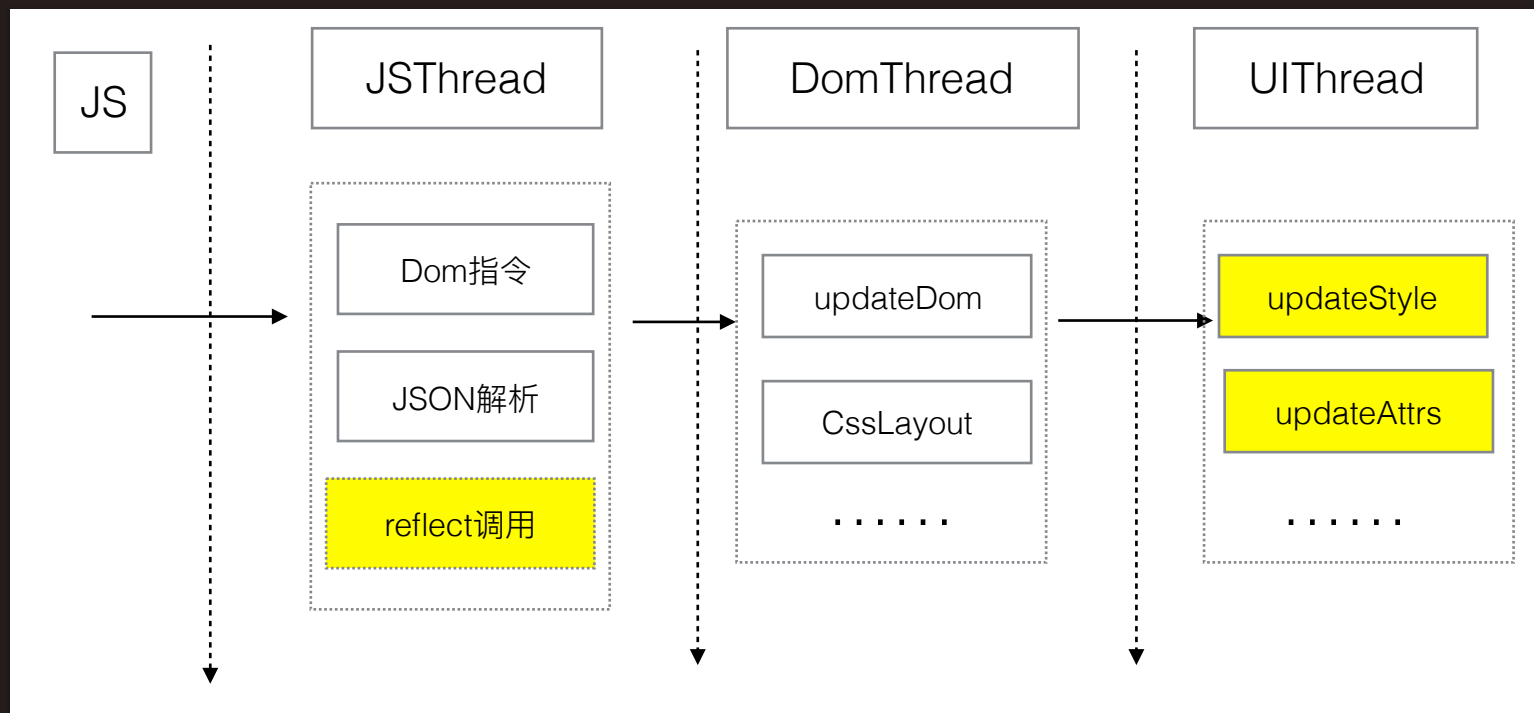


文字渲染



Measure和Draw使用了同一个对象, 避免了额外的对象创建。

反射&GC优化



- 高频的反射接口
- 固化的属性和样式设置

Opacity 属性问题



解决方案: `setLayerType(View.LAYER_TYPE_HARDWARE, null);`
优化效果: 38帧 -> 55帧

最佳实践

List 最佳实践

- 尽量使用list，而不是scroller
- 尽可能细粒度拆分cell
- 尽可能减少首次加载的内容，多用懒加载
- 尽可能复用List组件（多tab切换场景）
- 尽可能使用 a 链接跳转，而不是通过响应 click

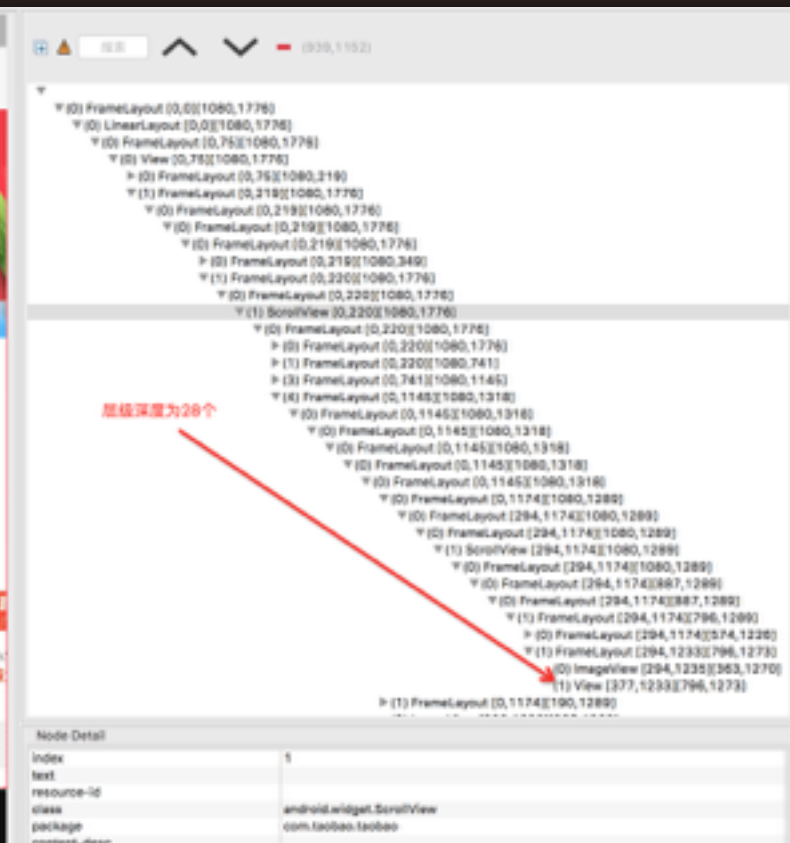
层级过深导致StackOverflowError

原因：

- 页面切套层级过深
- 递归调用层次过深, 触发系统保护

java.lang.StackOverflowError

```
at android.text.Layout.draw(Layout.java:186)
at com.taobao.weex.ui.view.WXTextView.onDraw(WXTextView
at android.view.View.draw(View.java:11281)
at android.view.View.getDisplayList(View.java:10720)
at android.view.ViewGroup.drawChild(ViewGroup.java:3240)
at android.view.ViewGroup.dispatchDraw(ViewGroup.java:2
at android.view.View.getDisplayList(View.java:10718)
at android.view.ViewGroup.drawChild(ViewGroup.java:3240)
at android.view.ViewGroup.dispatchDraw(ViewGroup.java:2
at android.view.View.getDisplayList(View.java:10718)
at android.view.ViewGroup.drawChild(ViewGroup.java:3240)
at android.view.ViewGroup.dispatchDraw(ViewGroup.java:2
at android.view.View.getDisplayList(View.java:10718)
at android.view.ViewGroup.drawChild(ViewGroup.java:3240)
at android.view.ViewGroup.dispatchDraw(ViewGroup.java:2
at android.view.View.getDisplayList(View.java:10718)
at android.view.ViewGroup.drawChild(ViewGroup.java:3240)
at android.view.ViewGroup.dispatchDraw(ViewGroup.java:2
at com.taobao.weex.ui.view.WXScrollView.dispatchDraw(WX
at android.view.View.draw(View.java:11284)
at android.widget.FrameLayout.draw(FrameLayout.java:456)
at android.widget.ScrollView.draw(ScrollView.java:1791)
at android.view.View.getDisplayList(View.java:10720)
at android.view.ViewGroup.drawChild(ViewGroup.java:3240)
at android.view.ViewGroup.dispatchDraw(ViewGroup.java:2
at android.view.View.getDisplayList(View.java:10718)
at android.view.ViewGroup.drawChild(ViewGroup.java:3240)
at android.view.ViewGroup.dispatchDraw(ViewGroup.java:2
at android.view.View.getDisplayList(View.java:10718)
at android.view.ViewGroup.drawChild(ViewGroup.java:3240)
```



层级过深导致StackOverflowError

https://pages.tmall.com/wow/portal/act/fpmw6 引起的crash次数分布



解决方案:

- 删除冗余的嵌套
- 删除冗余的嵌套
- 删除冗余的嵌套

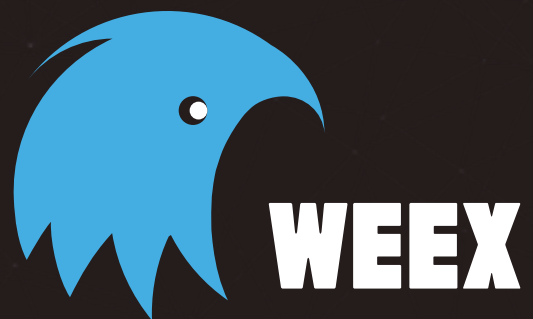
https://huodong.taobao.com/wow/tb-20161212/act/index 引起的crash次数分布



```
<div deep="20">
  <div deep="20">
    <div deep="20">
      <scroller id="scrollView" scrollDirection="vertical" loadmoreoffset="500"
        showScrollbar="true" deep="20">
        <div deep="20">
          <div deep="11">_</div>
          <div deep="20">
            <div deep="16">_</div>
            <div data-spm="decisions" deep="20">
              <div deep="10">_</div>
              <div deep="20">
                <div deep="20">
                  <tbimage resize="cover"
                    src="https://img.alicdn.com/tps/TB1JRal0pXXXXaGXVXXXXXXXXXX-666-272.png"
                    deep="11"></tbimage>
                  <div deep="20">_</div>
                </div>
              <div deep="15">_</div>
              <div deep="13">_</div>
            </div>
          <div data-spm="shop" deep="15">_</div>
          <div data-spm="live" deep="20">_</div>
        </div>
      </div>
    </div>
  </div>
</div>
```

动画最佳实践

- 尽量使用Transform动画（GPU加速）
 - `translate/translateX/translateY(string)`
 - `rotate(string)`
 - `scale(string)`
- 离开页面停止定时器
- 回到页面重新开启定时器



谢谢

<https://github.com/alibaba/weex>