

# AliExpress APP质量保障体系

阿里巴巴 程召

## MTSC2018

第四届中国移动互联网测试开发大会

- 全球速卖通，海外用户的淘宝乐园
- 全球买家数超1亿
- 中国第2个获得Google编辑精选的APP

AliExpress™  
Smarter Shopping, Better Living!



# APP质量保障体系

## MTSC2018

第四届中国移动互联网测试开发大会



角色

开发

测试

交互

视觉

PD

运营

...

业务场景

业务质量

开发自测

功能预演

功能测试

APP专项

验收测试

回归测试

发布质量

灰度

A/B

线上质量

业务监控

性能监控

稳定性监控

舆情监控

过程质量

项目管理

流程控制

技术风险

资损防控

工具平台

真机适配

Android Monkey

iOS Monkey

H5适配

快速适配

稳定性

深度遍历

远程真机

Android远程租用

iOS远程租用

多机同步

自动化

Android UI

iOS UI

Android 分层

iOS 分层

多语言

多币种

APP专项

秒开

帧率

图片

overdraw

电量

流量

弱网络

打点

安全

覆盖安装

SDK

代码扫描

辅助功能

包管理

设备托管

包大小检查

权限检查

SDK测试

度量

后台

无线机房

国内机房

海外机房

无线后台

设备管理

任务管理

适配策略中心

基础服务

打包

扫描

监控

舆情



## 研发模式

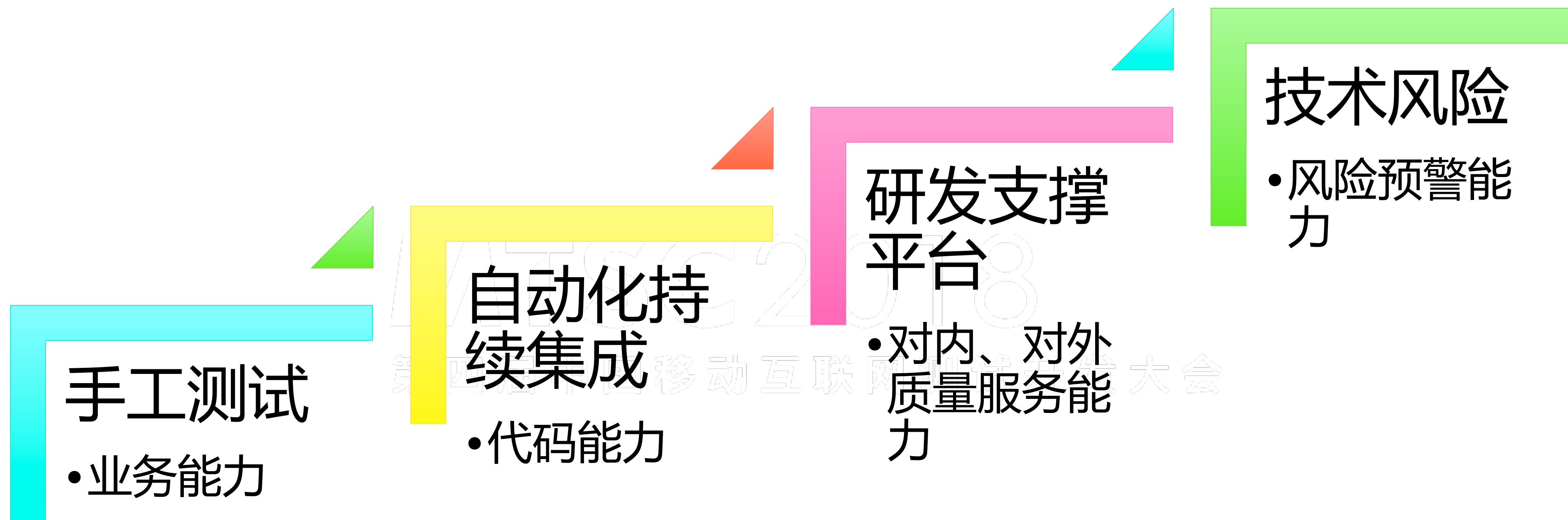
- 定版会上车
- 项目流程
- 研发迭代模式

## 专项测试

- 自动化
- 适配
- 性能
- 安全

## 无线平台

- Native适配
- H5适配
- 远程真机
- 多机同步



研发流程 + 测试环境 + 测试数据

BRD不详细

需求变更更多

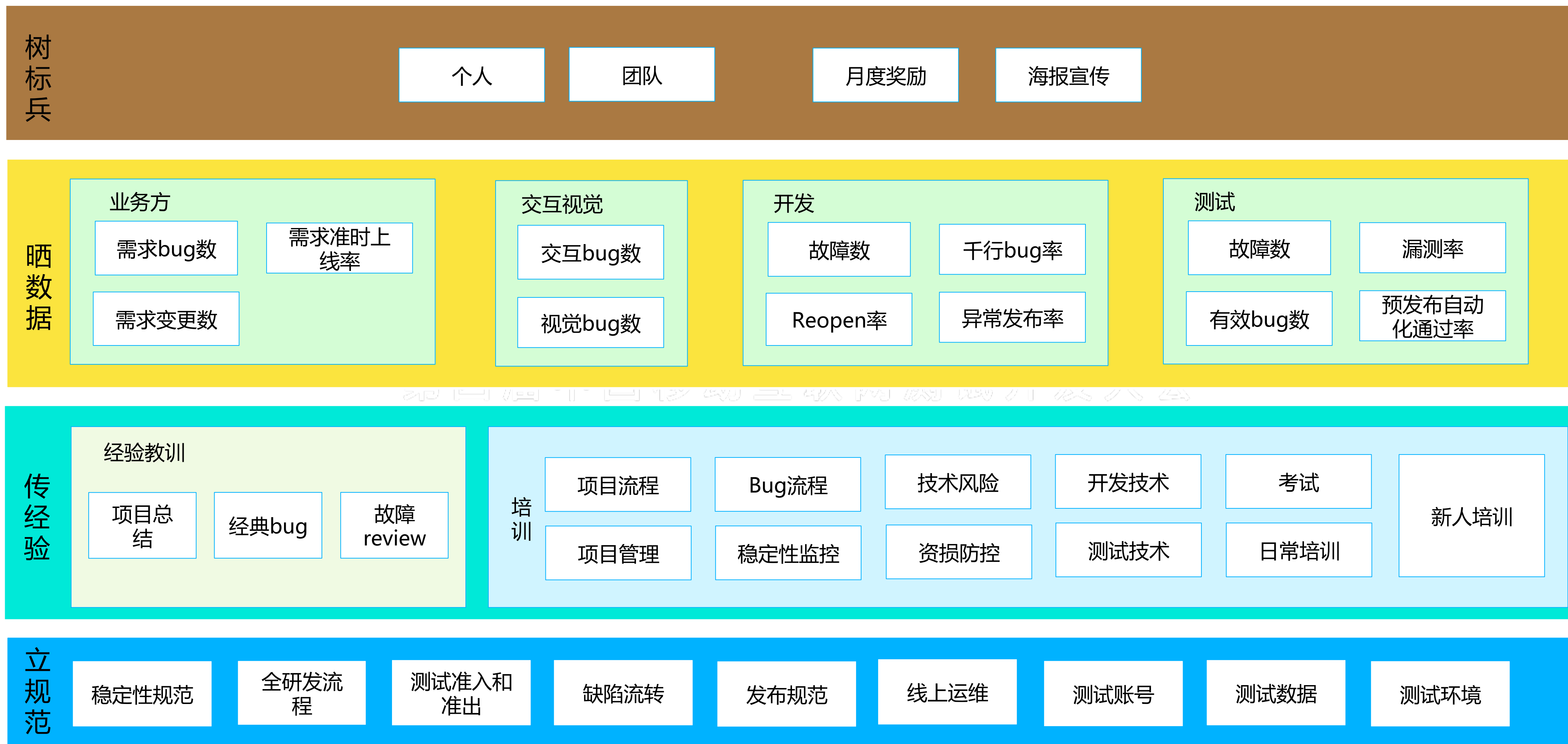
测试时间被压缩

?

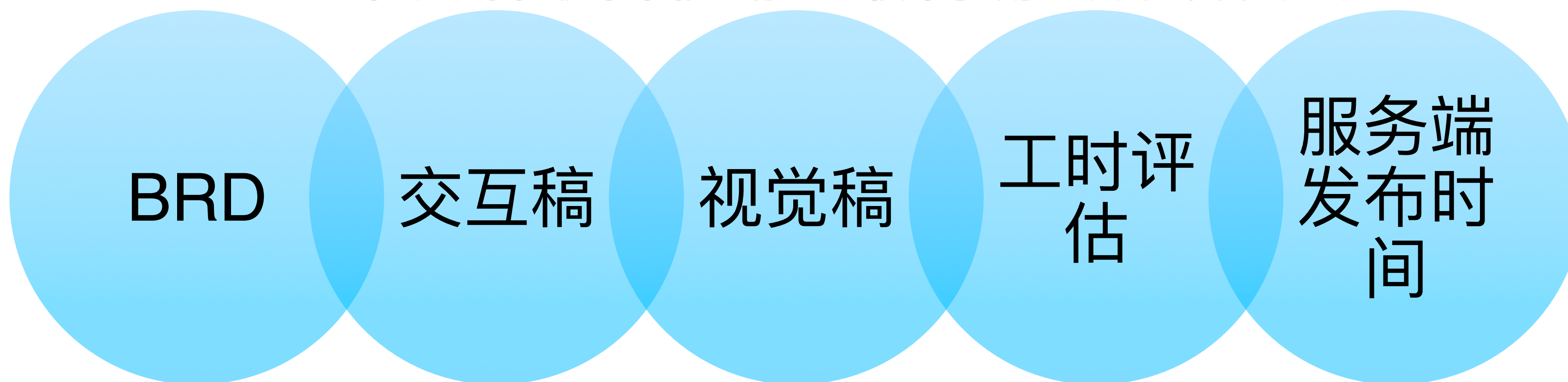
质量文化

合作方质量意识不强

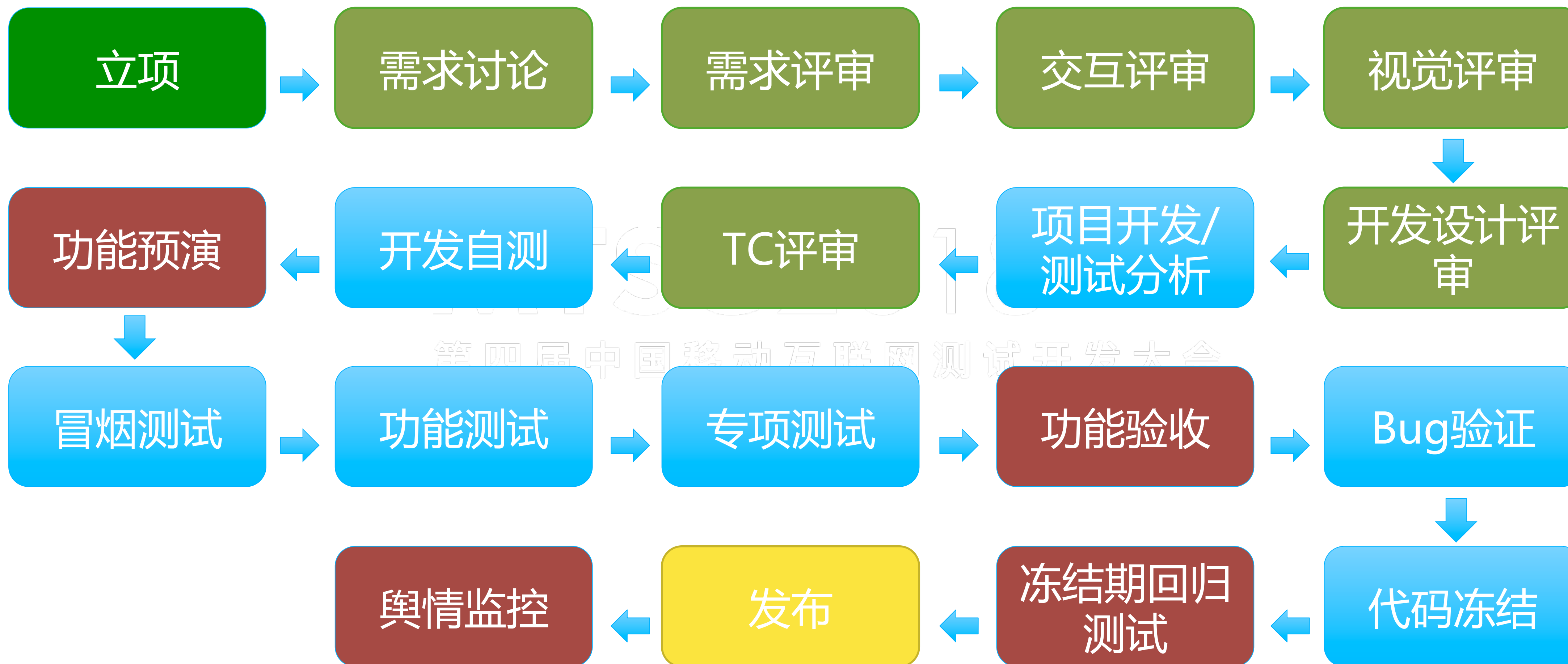
兜底式保姆式测试

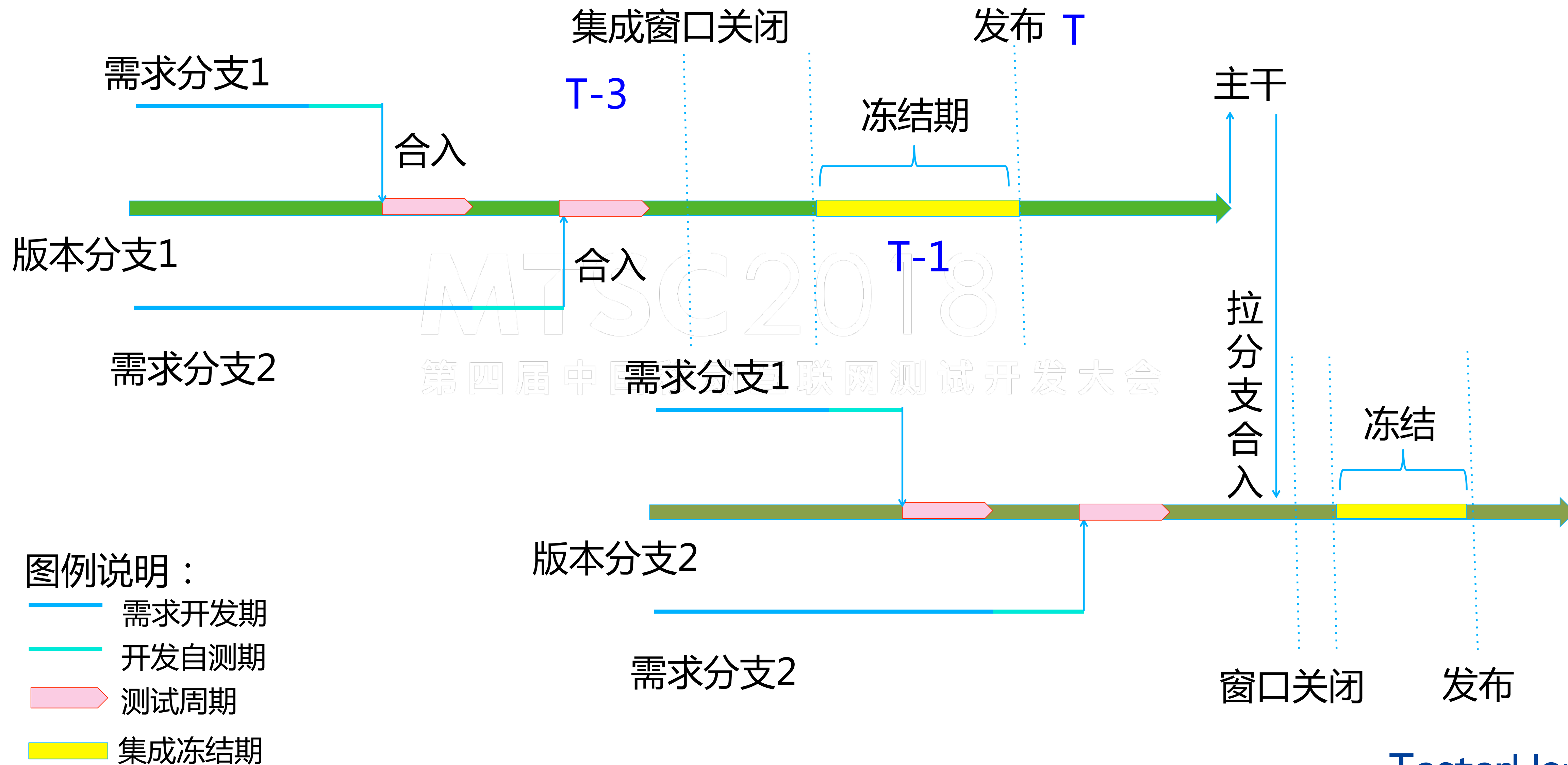


- 目标：一个月APP发版两次，促进业务快速发展
- 方式：定版会上车模式，确定下一个月两个版本的需求
  - 一个月一次
  - 各方参与
  - 五件套：缺一不可，否则需求顺延











## 研发模式

- 定版会上车
- 项目流程
- 研发迭代模式

## 专项测试

- 自动化
- 适配
- 性能
- 安全

## 无线平台

- Native适配
- H5适配
- 远程真机
- 多机同步



### 自动化

- UI
- 分层/模块化

### 适配

- OS
- 分辨率
- 横竖屏
- 第三方组件
- 多语言
- 多币种

### 性能

- 秒开
- 卡顿
- 过度绘制
- 电量
- 图片

### 其他

- 覆盖安装
- 打点
- 安全
- 包大小
- 权限检查
- SDK测试



秒开

卡顿

过度绘制

电量

图片

- 线上：性能预警
- 线下：质量卡口

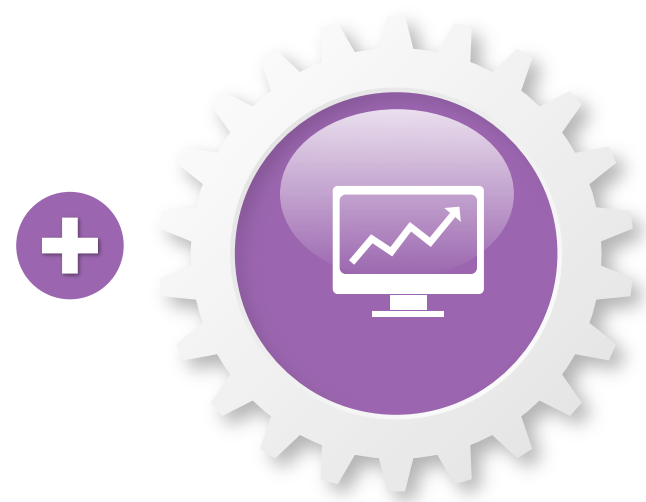


- 数据采集：代码打点、SDK采集
- 质量卡口：核心页面秒开、固定机型/固定网络性能指标不高于基线值
- 性能报表：秒开场景报告、基线对比报告



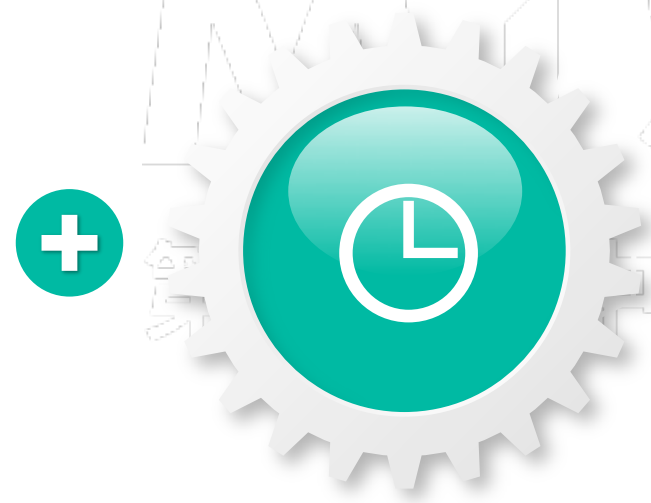


1、分析APP大小，  
减少启动内存开销



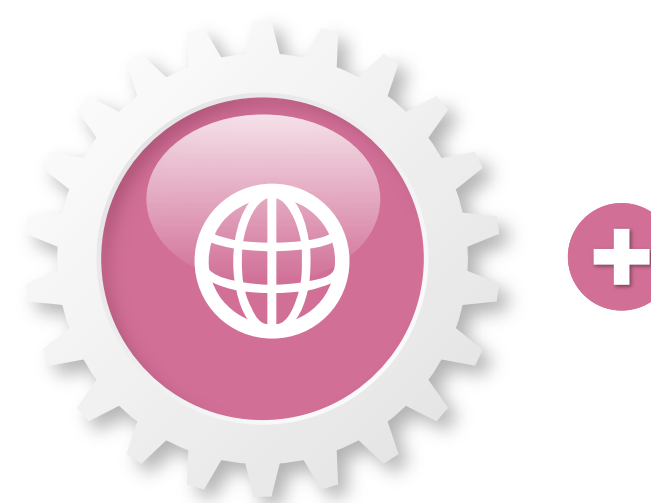
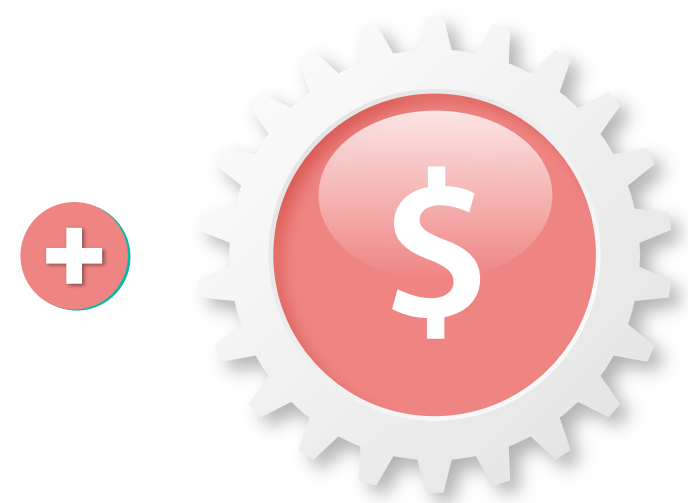
2、启动方法梳理，  
优化耗时方法

3、非核心场景加载  
延迟



4、统一预加载，首  
页图片缓存

5、检查首页View复  
杂度，尽快显示UI



6、高中低不同机  
型，确立不同目  
标，持续优化

## 测试方法

- 固定机型，固定网络
- 帧率自动化检测

## 质量卡口

- 平均帧率：不低于48帧
- 丢帧率：帧率 < 30，机率不超过10%

## 报 表

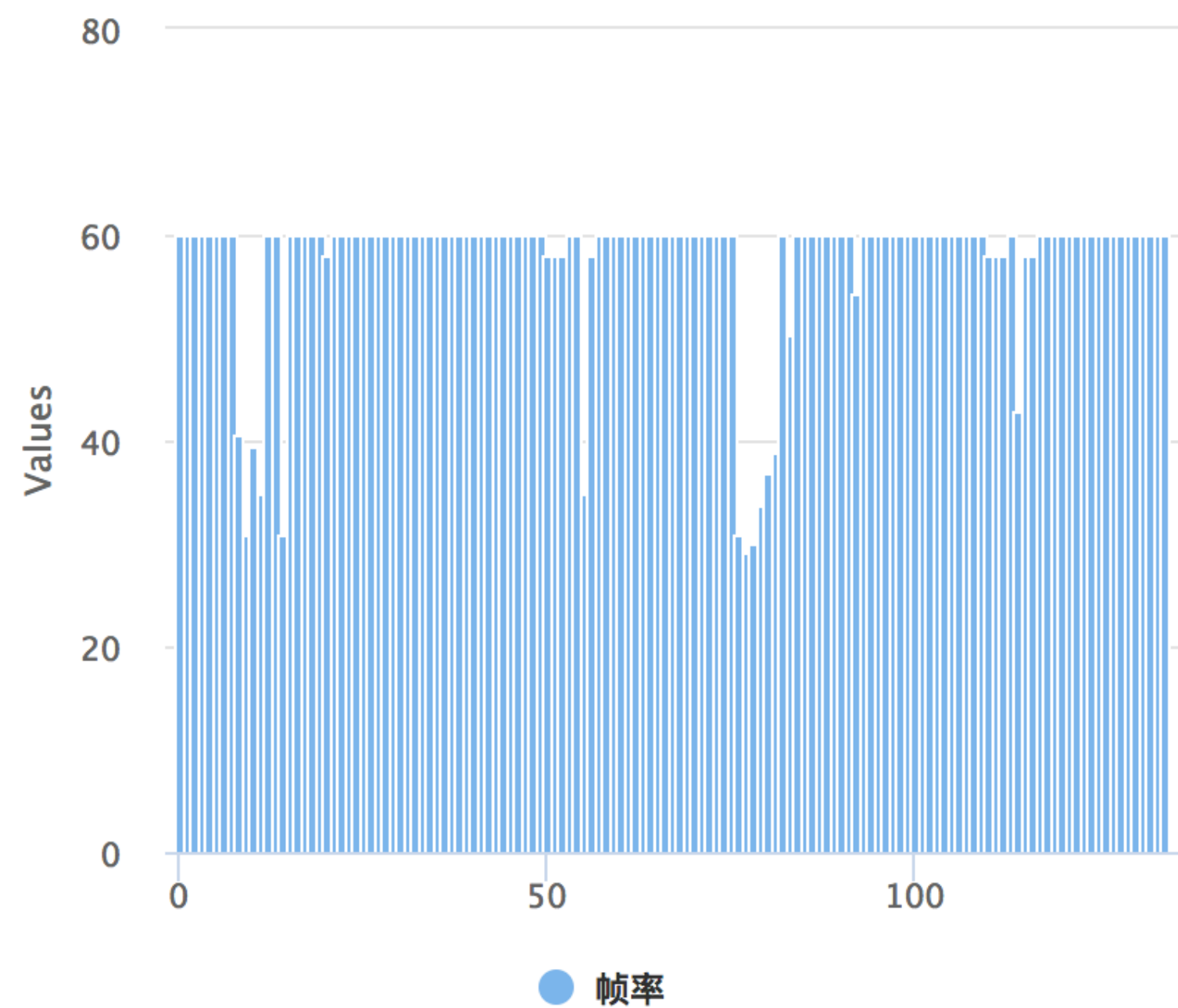
- 自动化报告



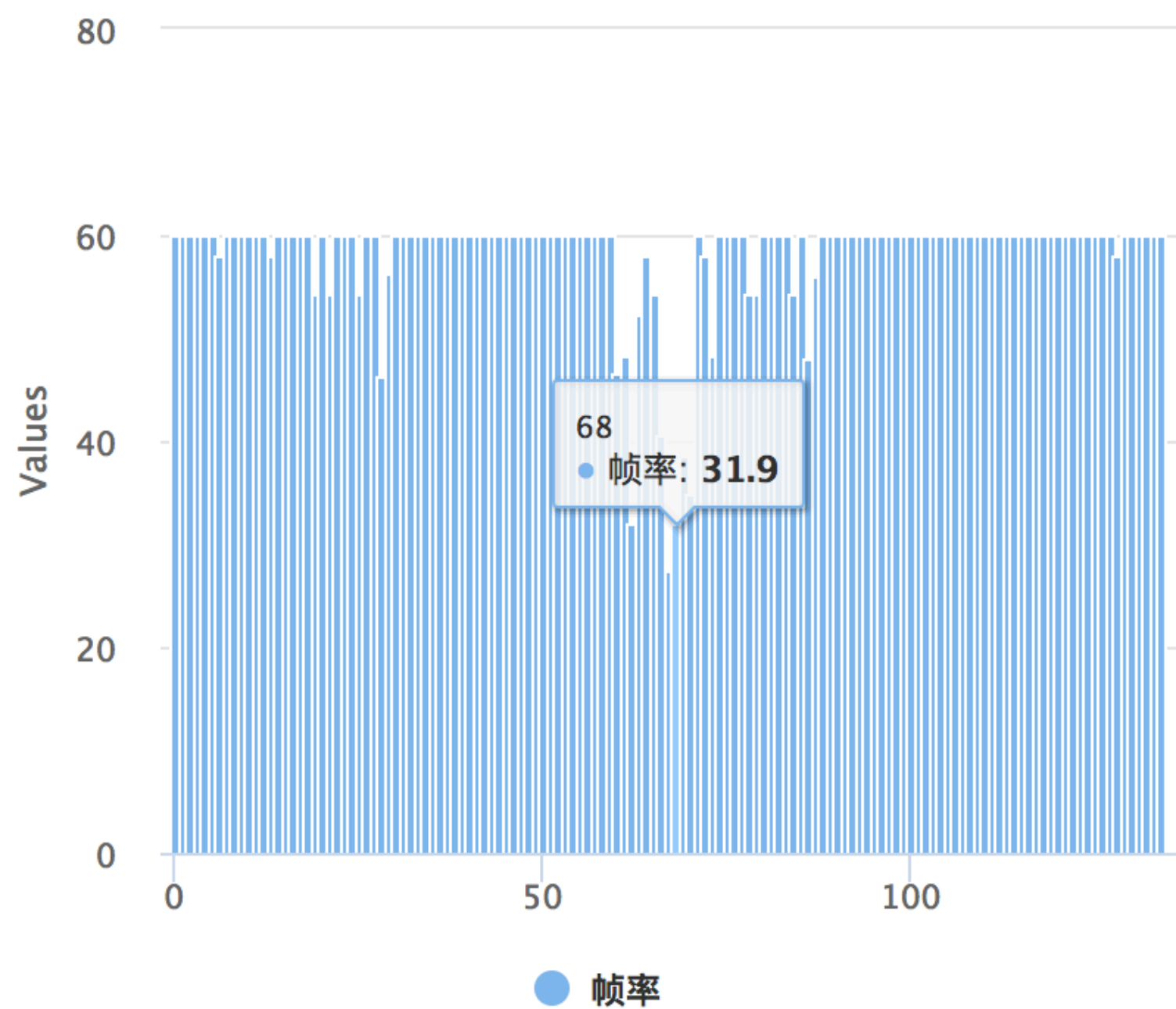
## Android帧率测试报告

用例:

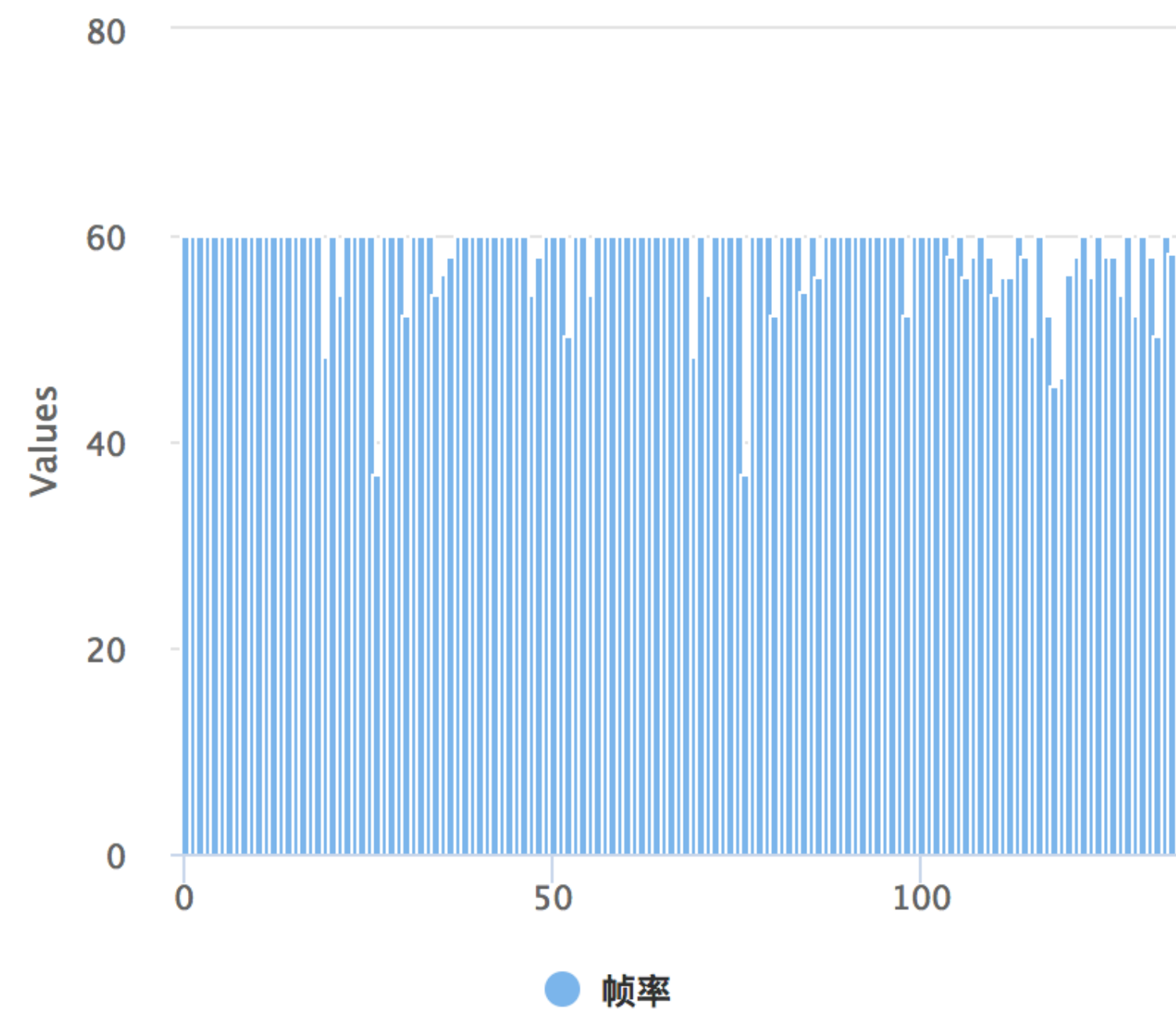
第一次执行



第二次执行



第三次执行



启动detail页面, 连续滑动20s

第1次测试

57.78

38.7

0.00%

启动Detail页面, 静置30s

第1次测试

57.37

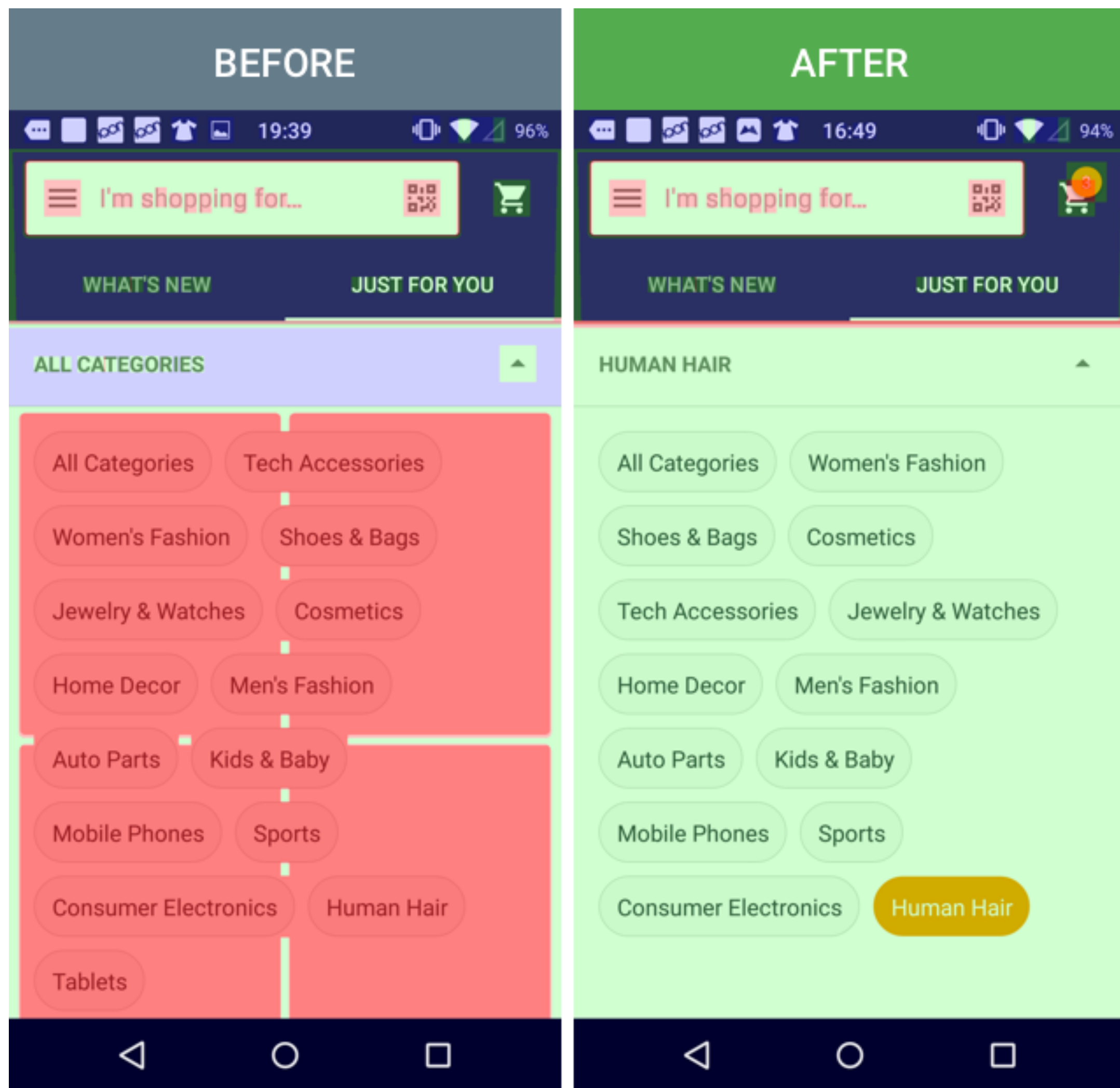
29.1

1.47%

# 卡顿：如何获取帧率？

通过Choreographer.FrameCallback 接口来实现，每当VSYNC到来时，doFrame函数将被调用，调度一帧开始，如果发现调度这一帧的时间和当前的时间已经超过间隔时间，那就认为丢帧了，从这里反推出当前的帧率。

```
46     @Override
47     public void doFrame(long frameTimeNanos) {
48         long currentTimeMillis = TimeUnit.NANOSECONDS.toMillis(frameTimeNanos);
49
50         if (frameStartTime > 0) {
51             // take the span in milliseconds
52             final long timeSpan = currentTimeMillis - frameStartTime;
53             framesRendered++;
54
55             if (timeSpan > interval) {
56                 final double fps = framesRendered * 1000 / (double) timeSpan;
57
58                 frameStartTime = currentTimeMillis;
59                 framesRendered = 0;
60                 if(listener!=null){
61                     listener.heartbeat(fps);
62                 }
63             }
64         } else {
65             frameStartTime = currentTimeMillis;
66         }
67
68         choreographer.postFrameCallback(this);
69     }
```



### 质量卡口

- 红色（4次以上过度绘制）不能出现，且极力避免
- 粉色（3次过度绘制）不应超过屏幕的四分之一
- 绿色和蓝色为比较理想状态

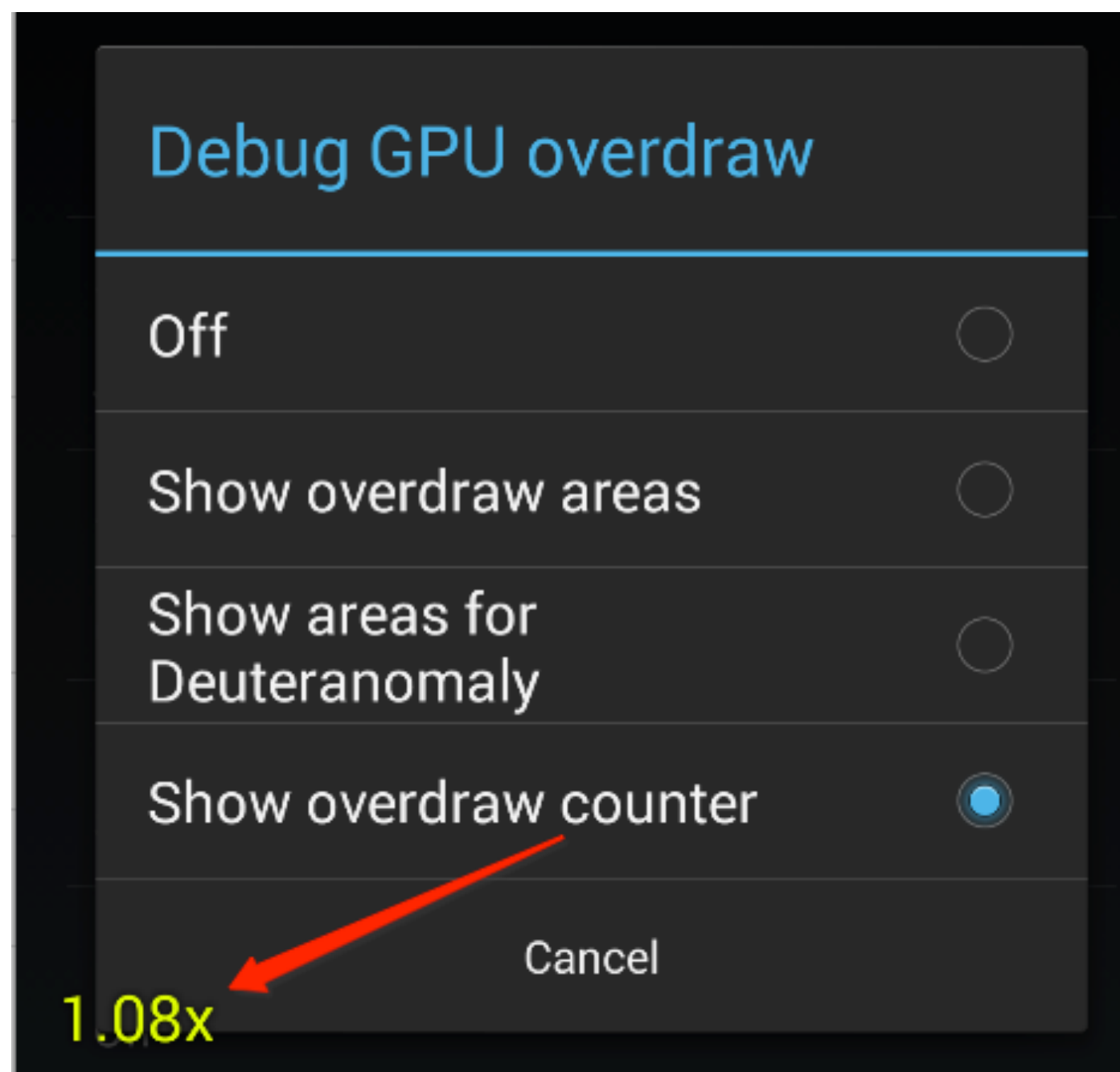
### 测试方法

- 过度绘制自动化检测

# 过度绘制自动化方案

## Counter

- 什么是counter？



```
4 //frameworks/base/libs/hwui/OpenGLRenderer.cpp
5
6 void OpenGLRenderer::renderOverdraw() {
7     .....
8
9     // 1x overflow
10    mCaches.stencil.enableDebugTest(2);
11    drawColor(mCaches.getOverdrawColor(1), SkXfermode::kSrcOver_Mode);
12
13    // 2x overflow
14    mCaches.stencil.enableDebugTest(3);
15    drawColor(mCaches.getOverdrawColor(2), SkXfermode::kSrcOver_Mode);
16
17    // 3x overflow
18    mCaches.stencil.enableDebugTest(4);
19    drawColor(mCaches.getOverdrawColor(3), SkXfermode::kSrcOver_Mode);
20
21    // 4x overflow and higher
22    mCaches.stencil.enableDebugTest(4, true);
23    drawColor(mCaches.getOverdrawColor(4), SkXfermode::kSrcOver_Mode);
24
25    mCaches.stencil.disable();
26 }
27 }
28
29 void OpenGLRenderer::countOverdraw() {
30     size_t count = mWidth * mHeight;
31     uint32_t* buffer = new uint32_t[count];
32     glReadPixels(0, 0, mWidth, mHeight, GL_RGBA, GL_UNSIGNED_BYTE, &buffer[0]);
33
34     size_t total = 0;
35     for (size_t i = 0; i < count; i++) {
36         total += buffer[i] & 0xff;
37     }
38
39     mOverdraw = total / float(count);
40
41     delete[] buffer;
42 }
```

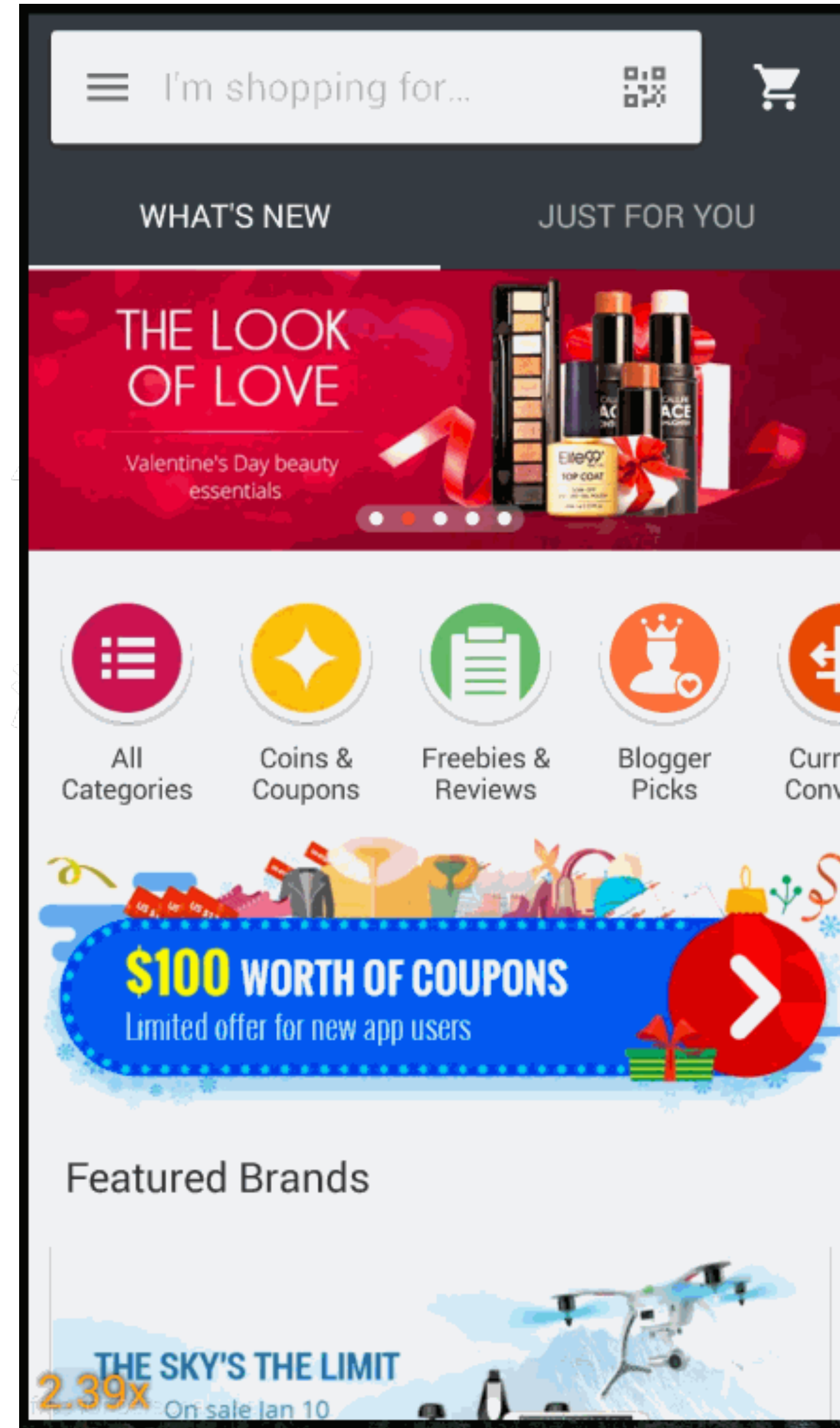


## Counter

- 如何获取？

```
5 //Framework/base/core/Java/android/view/HardwareRender.java
6
7 private void drawOverdrawCounter(HardwareCanvas canvas, float overdraw, float density) {
8     final String text = String.format("%.2fx", overdraw);
9     final Paint paint = setupPaint(density);
10    // HSBtoColor will clamp the values in the 0..1 range
11    paint.setColor(Color.HSBtoColor(0.28f - 0.28f * overdraw / 3.5f, 0.8f, 1.0f));
12    canvas.drawText(text, density * 4.0f, mHeight - paint.getFontMetrics().bottom, paint);
13 }
```

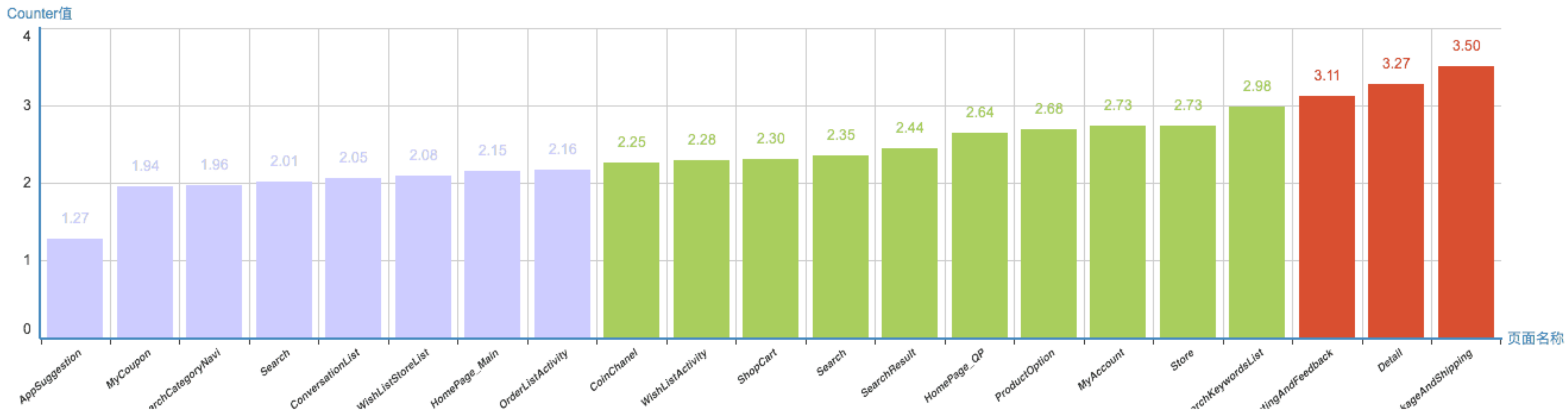
- 通过Xposed hook系统函数drawOverdrawCounter方法来获取过度绘制的Counter值





### AE Android 页面过度绘制统计图

Aliexpress overdraw chart





- 测试工具：Power Monitor、组装手机
- 质量卡口：单版本电流趋势平稳无毛刺，基线对比未超前一版本





### 静默测试

- APP启动后，操作Power、Home、Back后的息屏电流

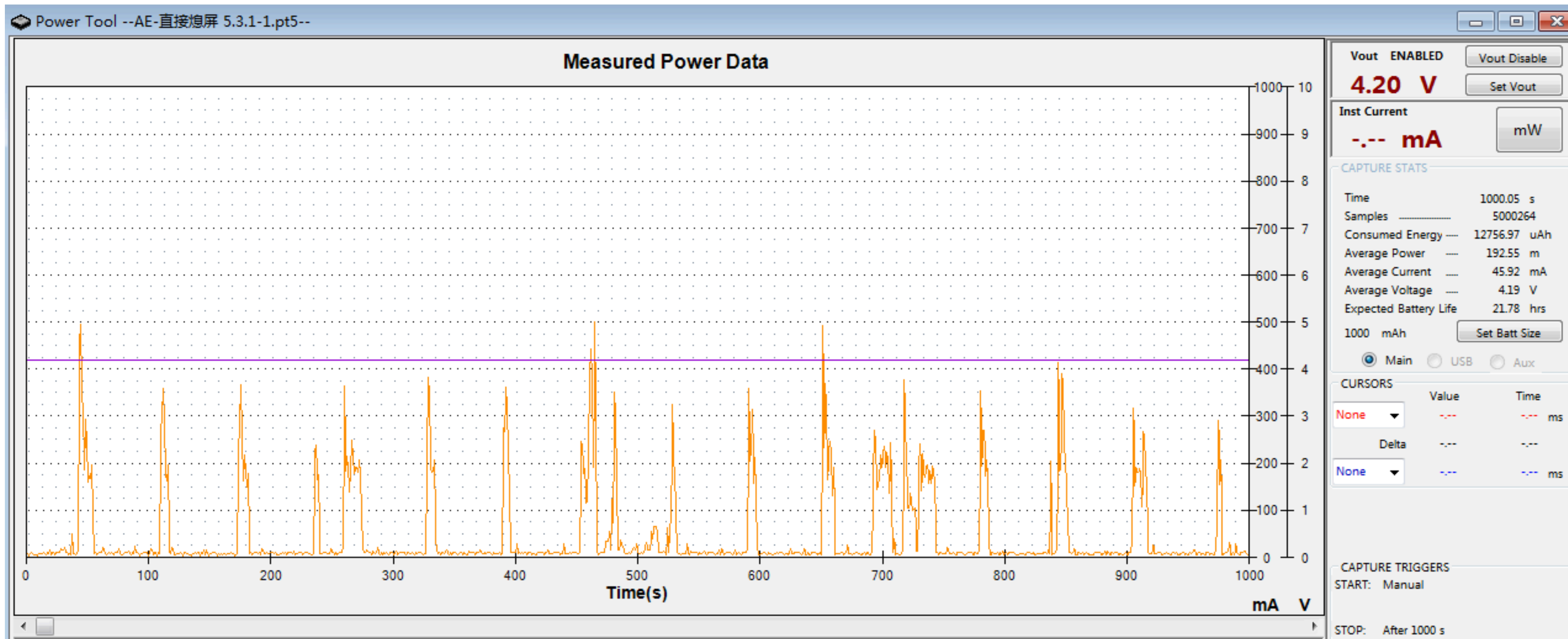
### 场景电流

- 对单一场景进行重复操作

### 链路电流

- 对链路场景进行操作

# 耗电：问题举例1



唤醒过多，每次耗电较多



唤醒后长时间不睡



## 研发模式

- 定版会上车
- 项目流程
- 研发迭代模式

## 专项测试

- 自动化
- 适配
- 性能
- 安全

## 无线平台

- Native适配
- H5适配
- 远程真机
- 多机同步



- AliExpress无线测试中心
- 首页
- 适配测试
- 远程真机
- 多机同步
- H5适配
- SDK测试
- 包管理
- 设备托管
- 全球化测试



### Android适配

支持Android适配测试  
重点发现crash、anr、leaks问题



### iOS适配

支持iOS适配测试  
重点发现crash问题



### 远程真机租用

支持Andorid真机远程调试  
Web化真机测试方便快捷

## 稳定性测试

1.应用上传

2.设备选择

3.提交测试

### 设备推荐:

- 自定义设备
- 空闲设备
- 覆盖所有品牌
- 覆盖所有版本
- 覆盖所有分辨率
- 覆盖安装
- 私有设备

### 海外设备推荐: beta

- 海外空闲设备
- 俄罗斯
- 美国
- 西班牙
- 法国
- 巴西
- 荷兰
- 以色列
- 英国
- 智利
- 意大利
- 东南亚

### 系统版本:

- 4.3
- 4.4.4
- 4.4.2
- 6.0.1
- 6.0.8-RS-20170821.1313
- 6.0
- 7.0
- 5.1
- 5.0.1
- 7.0
- 5.1
- 4.1.2
- 4.2.2
- 7.1.1
- 5.0.2
- 8.0.0
- 7.1.2
- 8.1.0

### 品牌:

- OPPO
- HONOR
- HUAWEI
- YOTAPHONE
- SAMSUNG
- XIAOMI
- MBX
- MOTOROLA
- LENOVO
- HTC
- MEIZU
- LGE
- ONEPLUS
- ZTE
- SONY
- NOKIA
- BQ
- GOOGLE
- ASUS
- LETV

### 已选择 26 款

- SAMSUNG(6款)
  - GALAXY GRAND 2 (4.3)
  - GALAXY NOTE 2 (4.3)
  - GALAXY NOTE 2 (4.3)
  - GALAXY J5 (7.1.1)
  - SM-N950U (7.1.1)
  - GALAXY NOTE 8 (7.1.1)
- OPPO(3款)
  - FIND 7 (4.3)
  - R6007 (4.3)
  - OPPO R831S (4.3)
- ONEPLUS(3款)
  - ONE (4.3)
  - ONE (4.3)
  - ONEPLUS 5 (7.1.1)
- HUAWEI(3款)
  - HUAWEI G750-T01 (4.2.2)
  - 荣耀 3C (4.2.2)
  - ASCEND P6 (4.2.2)
- SONY(2款)
  - XPERIA Z (4.2.2)
  - XPERIA Z5 (7.1.1)
- BQ(2款)
  - AQUARIS X (7.1.1)
  - AQUARIS U PLUS (7.1.1)

推荐策略

## 场景覆盖率



包名 : com.alibaba.aliexpresshd

测试状态 : 已完成

结果汇总 : Crash 20 个, ANR 45 个, Leak 17 个



### 未覆盖Activity(34/99) :

- com.alibaba.aliexpresshd.auth.test.Tok
- com.alibaba.aliexpresshd.web.TestWel
- com.alibaba.aliexpresshd.module.pron
- com.alibaba.aliexpresshd.module.proc
- com.alibaba.aliexpresshd.module.mes: y

### 错误堆栈信息

### 设备执行情况

## 问题去重、堆栈详情、提交bug

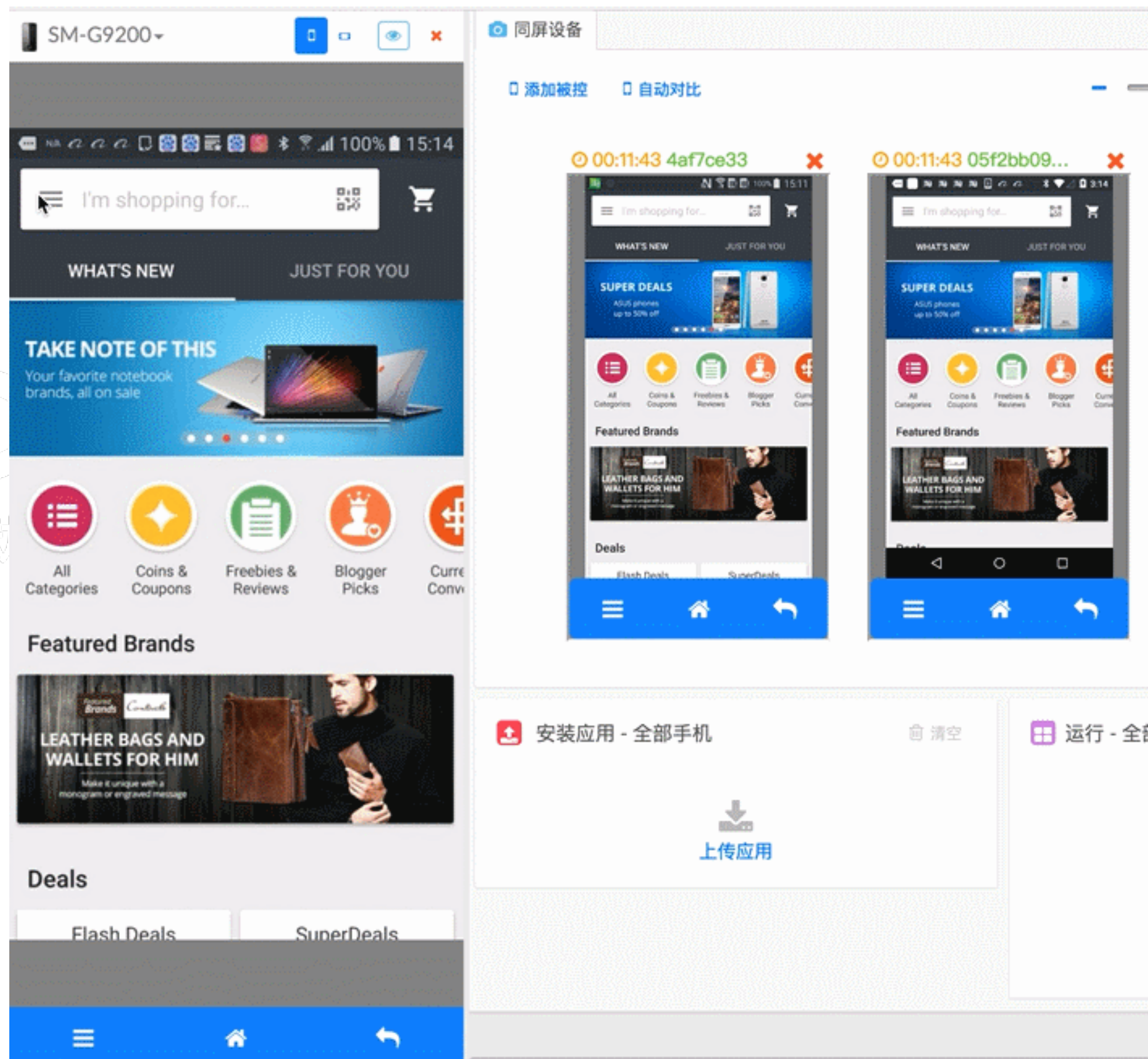
堆栈信息	设备	错误类型	出现次数	操作
java.lang.IllegalStateException: Fragment already added: RefinePagerFragment{8b354...	<a href="#">Huawei ALE-UL00 5.0.2</a>	Crash	1	<a href="#">查看详情</a>   <a href="#">提交bug</a>
android.view.WindowManager\$BadTokenException: Unable to add window -- token a...	<a href="#">samsung SM-N900 5.0</a>	Crash	1	<a href="#">查看详情</a>   <a href="#">提交bug</a>
java.lang.IllegalStateException: Fragment already added: RefinePagerFragment{39bba...	<a href="#">HUAWEI HUAWEI GRA-UL00 6.0</a>	Crash	1	<a href="#">查看详情</a>   <a href="#">提交bug</a>
Native crash: Aborted	<a href="#">htc HTC 919d 4.4.2</a> <a href="#">null null null</a>	Crash	7	<a href="#">查看详情</a>   <a href="#">提交bug</a>
Native crash: Bus error	<a href="#">htc HTC 919d 4.4.2</a>	Crash	1	<a href="#">查看详情</a>   <a href="#">提交bug</a>
Native crash: Segmentation fault	<a href="#">null null null</a>	Crash	2	<a href="#">查看详情</a>   <a href="#">提交bug</a>

## 背景：

- 安卓机型碎片化严重，适配成本高

## 多机同步：

- 测试主控设备，同步测试所有被控设备
- 所有设备界面一致性自动对比





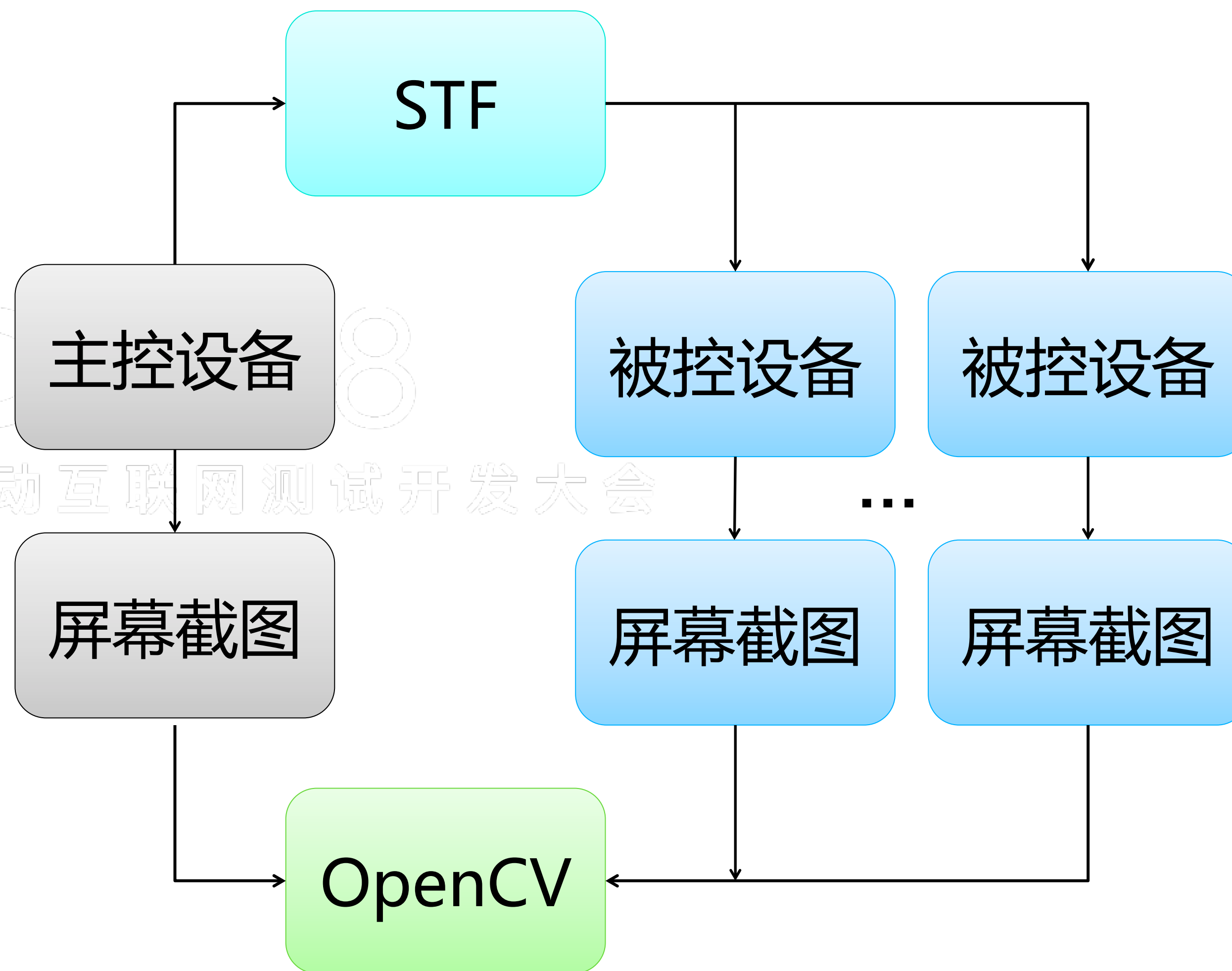


### 底层技术：

- 基于STF二次开发
- 基于OpenCV进行图片对比

### 实现：

- `stf/control/control-service.js`
- `stf/control/index.js`
- `stf/install/install-service.js`



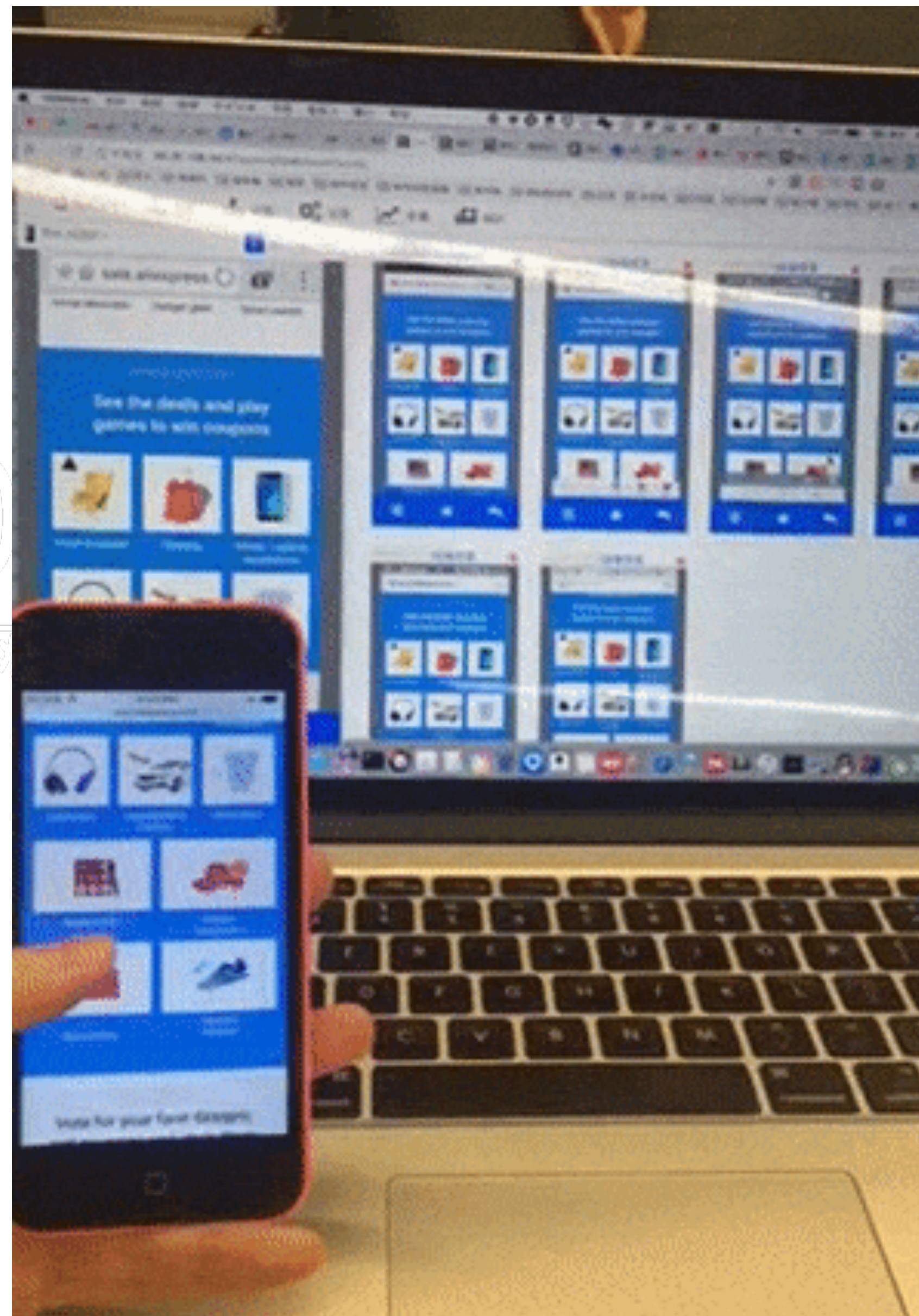


背景：

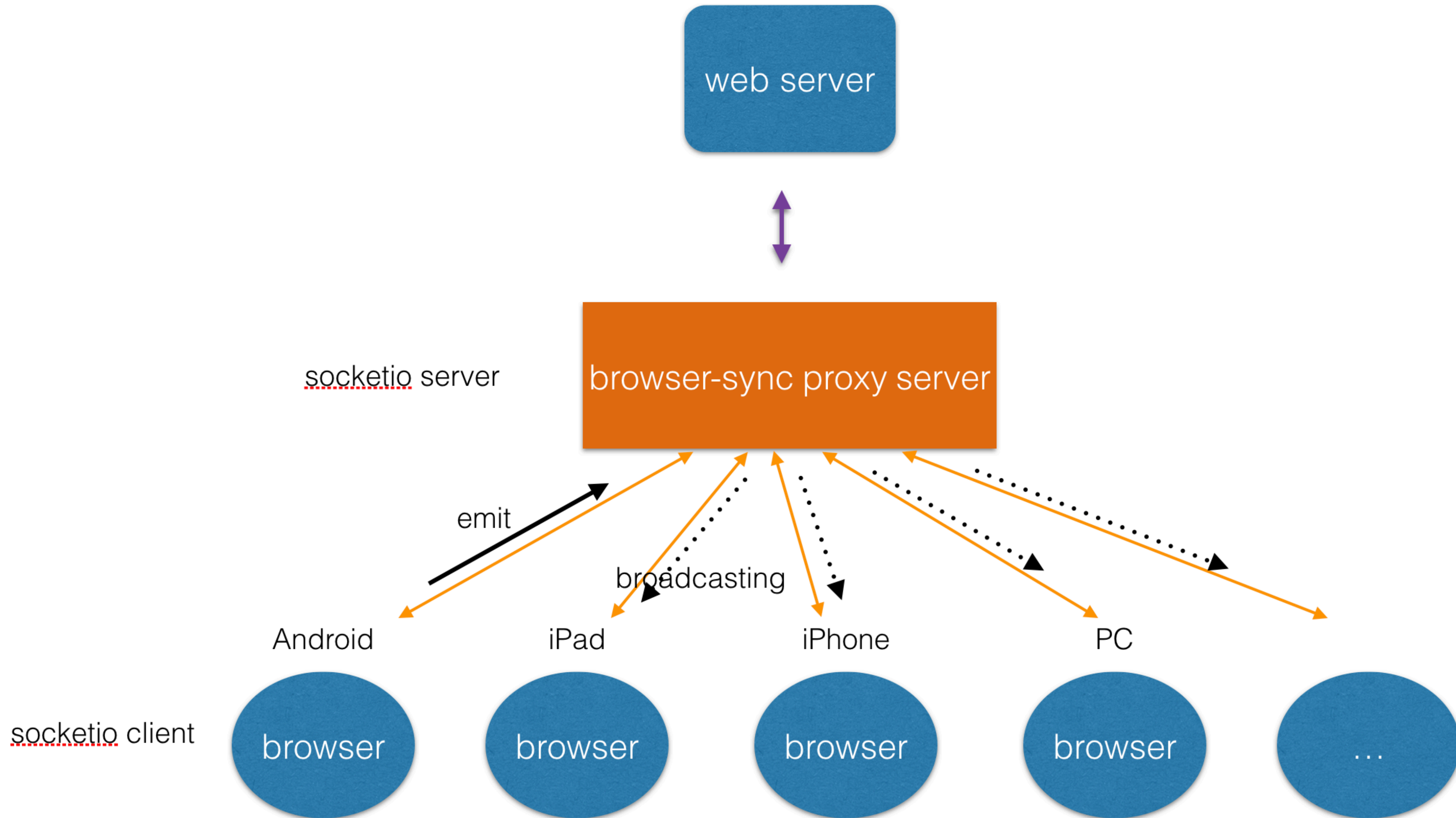
- PC端、移动端 H5页面适配需求量大、成本高

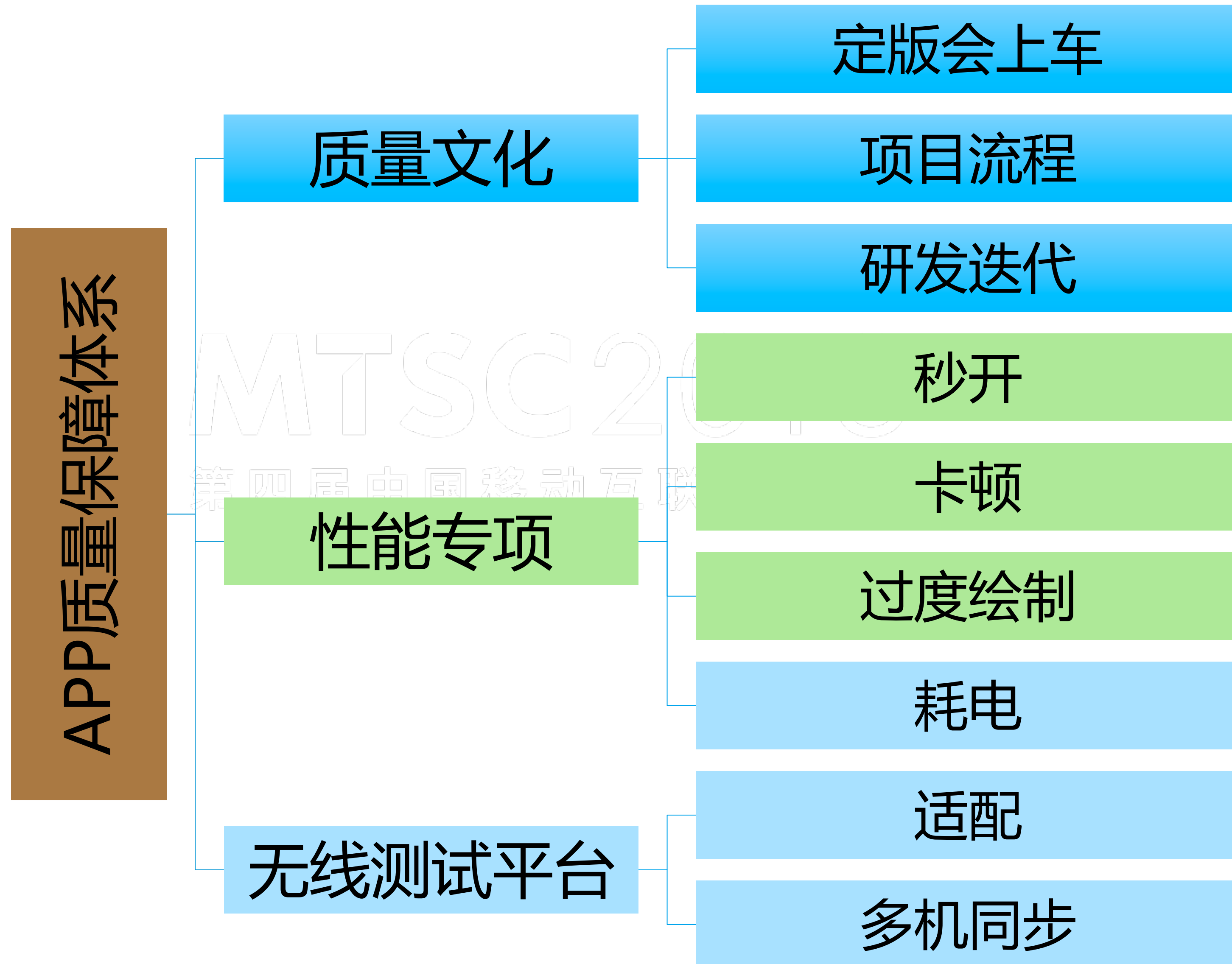
browser-sync实现H5多机同步

- 支持PC/平板/手机下的浏览器和APP同步测试
- 任意操作行为，实时同步到其他设备



# H5多机同步：原理







Q&A

第四届中国移动互



欢迎关注交流：<https://testerhome.com/topics/8176>



# MTSC2018

第四届中国移动互联网测试开发大会