

美团点评移动端性能 监控方案Hertz

美团点评 吴凯

自我介绍

吴凯

西安电子科技大学

2012~2014 腾讯

2015~2016 创业

2016~今 美团点评

美团外卖Android负责人，公司性能监控、自动化埋点等基础设施负责人。

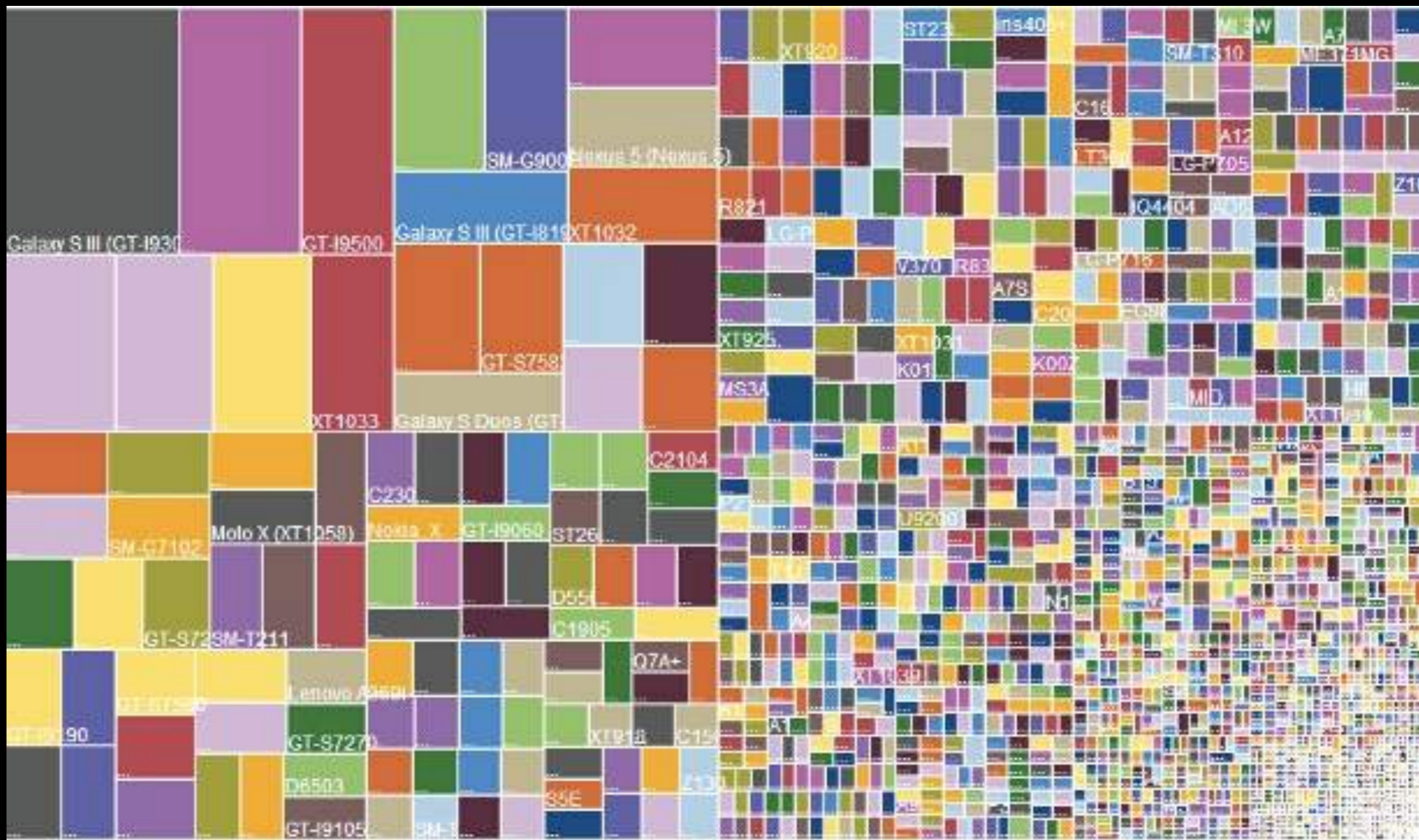


用户常常抱怨的性能问题

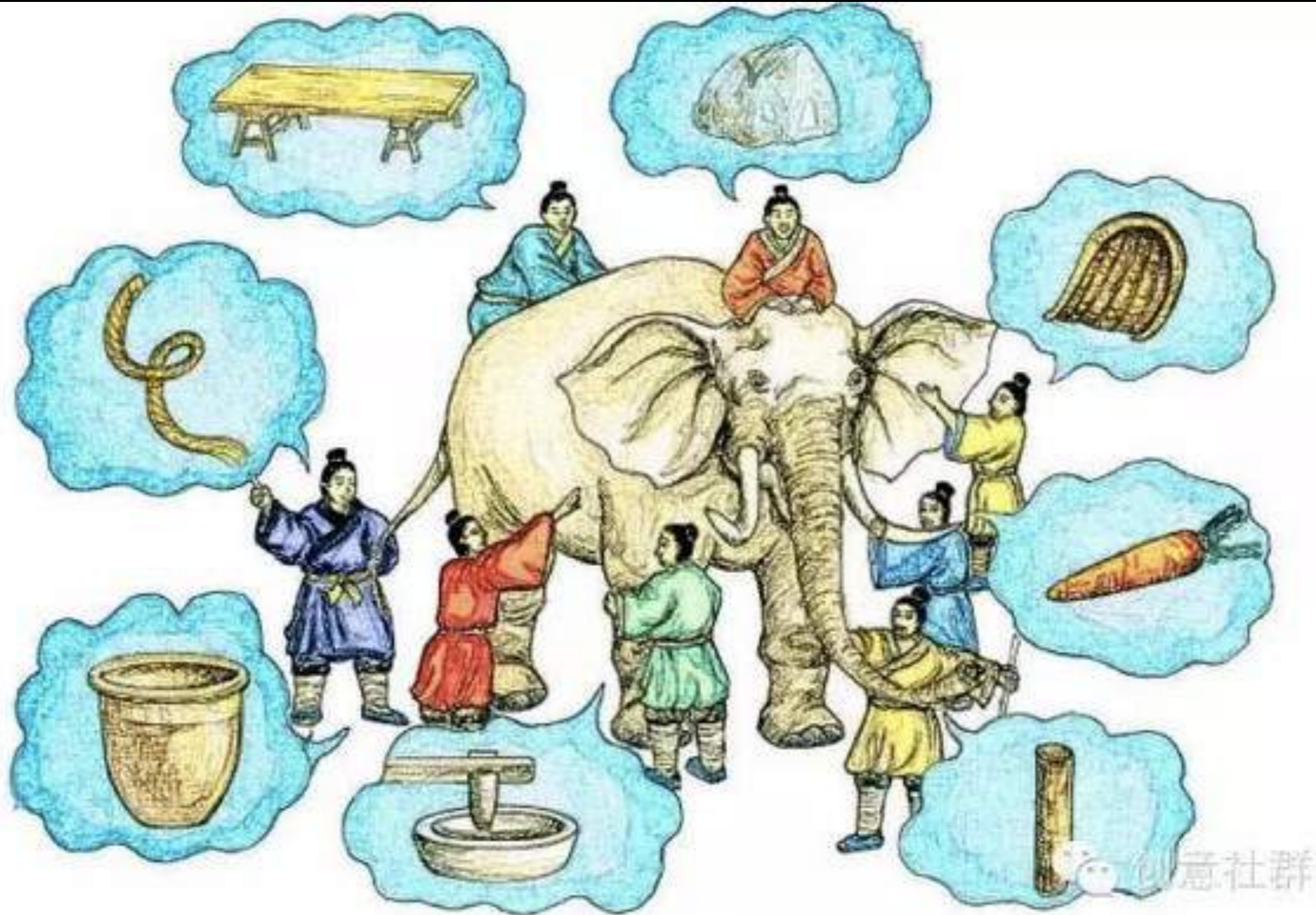
- 闪退
- 打开慢
- 滑动卡顿
- 网络加载慢
- 图片加载不出来
- 耗流量
- 耗电
- ...



移动端碎片化问题



性能测试的一致性问题的



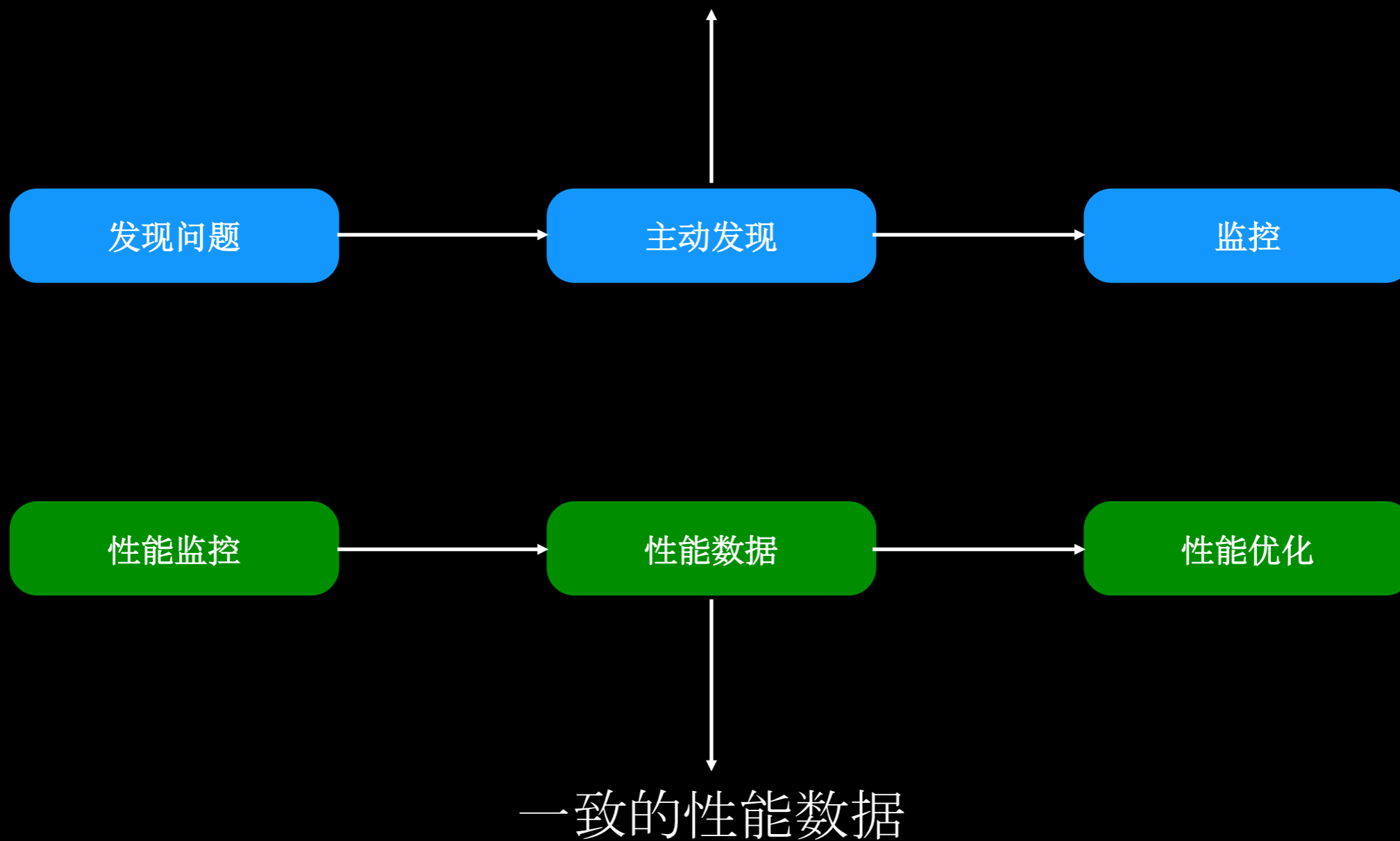


有没有银弹？



Try Hertz !

开发、测试、运维各阶段



Topics

- 采集什么样的数据
- 怎么采集
- 如何使用数据

采集什么样的数据

指标

CPU

内存

FPS

页面加载

冷启动

流量

电量



异常

卡顿

流量异常

Exception

内存泄漏

耗时方法

业务异常

FPS

```
int interval = 500;
Choreographer.getInstance().postFrameCallback(this);

@Override
public void doFrame(long frameTimeNanos) {
    frames++;
    currentTimeMills = TimeUnit.NANOSECONDS.toMillis(frameTimeNanos);
    timeSpan = currentTimeMills - startTimeMills;
    if (timeSpan >= interval) {
        fps = frames * 1000 / timeSpan;
        startTimeMills = currentTimeMills;
        frames = 0;
    }
}
```

Android

FPS

```
@property (nonatomic, strong) CADisplayLink *displayLink;
self.displayLink = [CADisplayLink displayLinkWithTarget:self
                                                         selector:@selector(tick:)];

- (void)tick:(CADisplayLink *)link
{
    self.count++;
    delta = link.timestamp - self.lastTime;
    if (delta >= 1.0) {
        self.fps = self.count / delta;
        self.lastTime = link.timestamp;
        self.count = 0;
    }
}
```

iOS

CPU

```
BufferedReader cpuReader = new BufferedReader(new InputStreamReader(  
    new FileInputStream("/proc/stat")), 1000);  
String cpuRate = cpuReader.readLine();  
  
BufferedReader pidReader = new BufferedReader(new InputStreamReader(  
    new FileInputStream("/proc/" + mPid + "/stat")), 1000);  
String pidCpuRate = pidReader.readLine();  
  
cpuRate = (appCpuTime - mAppCpuTimeLast) * 100L / totalTime
```

```
cpu 12249 3951 14287 17518 4117 0 139 0 0 0
```

```
cpu = user+nice+system+idle+iowait+irq+sofirq
```

Android

流量

- TrafficStats
- SocketFactory
- AOP

流量

```
@Pointcut("!within(retrofit.appengine.UrlFetchClient) " +
    "&& !within(okio.Okio) && !within(butterknife.internal.ButterKnifeProcessor) " +
    "&& !within(com.flurry.sdk.hb)" +
    "&& !within(rx.internal.util.unsafe.*)" +
    "&& !within(net.sf.cglib..*)" +
    "&& !within(com.huawei.android..*)" +
    "&& !within(com.sankuai.android.nettraffic..*)" +
    "&& !within(roboguice..*)" +
    "&& !within(com.alipay.sdk..*)")
protected void baseCondition() {
}

@Pointcut("call (java.io.OutputStream java.net.URLConnection.getOutputStream())" +
    "&& target(java.net.URLConnection)")
void getOutputStream() {
}
```

Android

流量

```
- (void)connection:(NSURLConnection *)connection didReceiveResponse:(NSURLResponse *)response
{
    [self.client URLProtocol:self
        didReceiveResponse:response
        cacheStoragePolicy:NSURLCacheStorageNotAllowed];
}

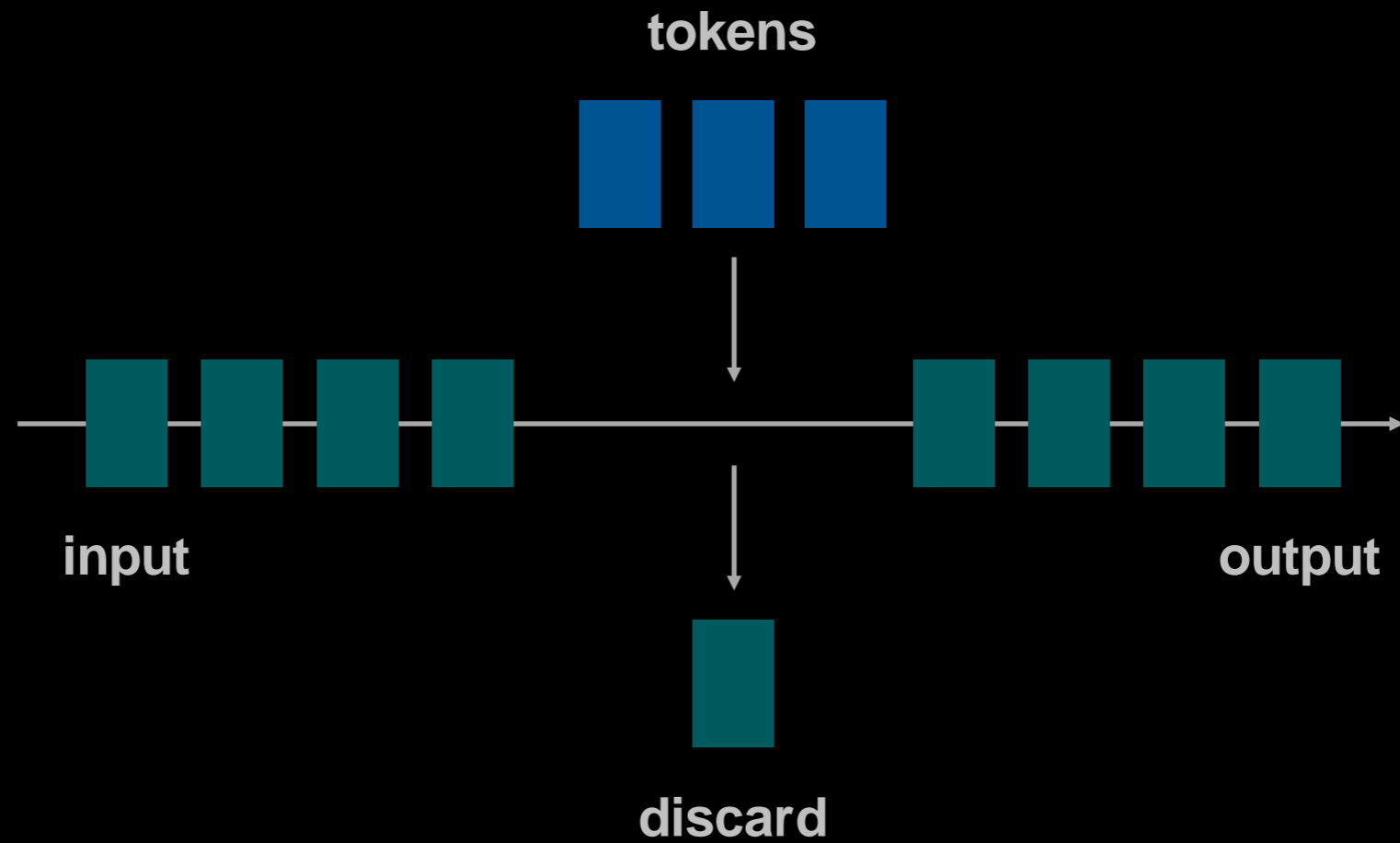
- (void)connection:(NSURLConnection *)connection didReceiveData:(NSData *)data
{
    [self.client URLProtocol:self didLoadData:data];
}

- (void)connectionDidFinishLoading:(NSURLConnection *)connection
{
    [self.client URLProtocolDidFinishLoading:self];
}
```

iOS

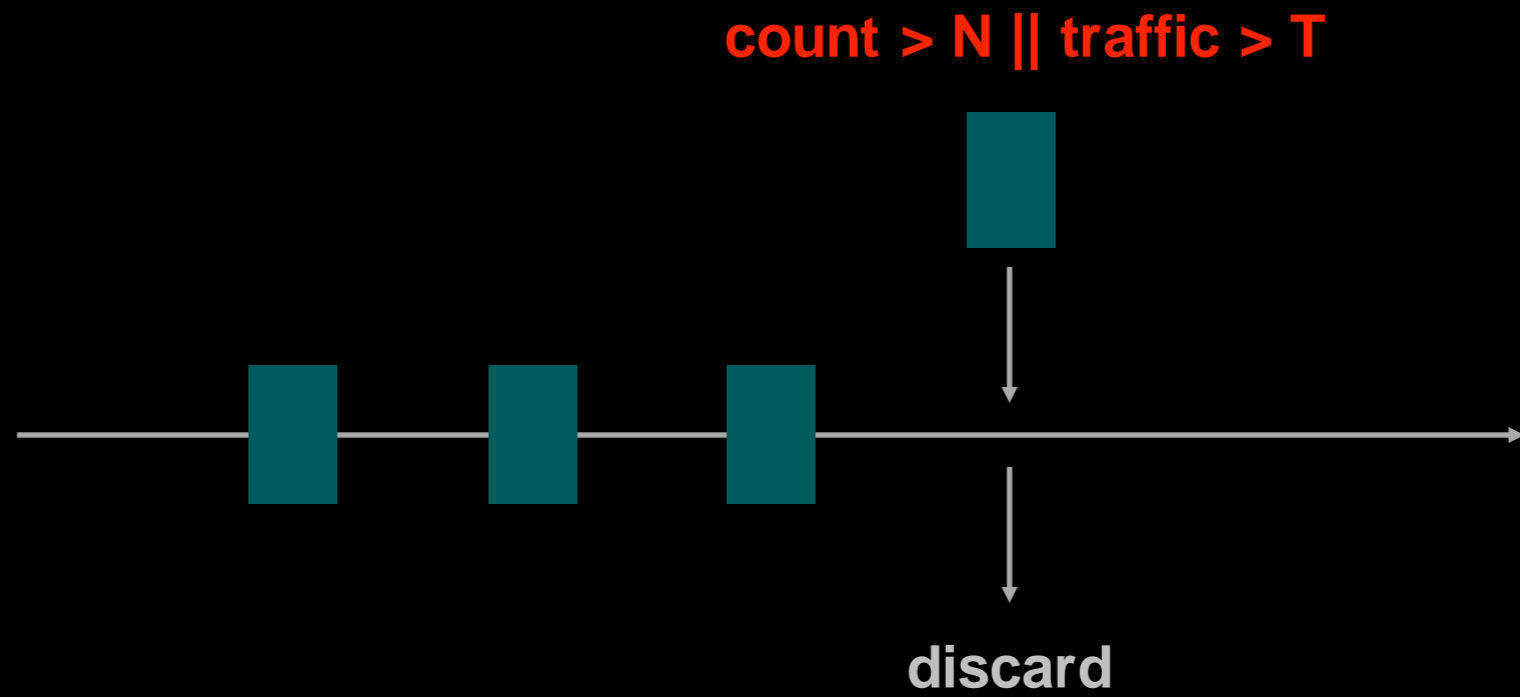
流量异常

- 限速
- 熔断



流量异常

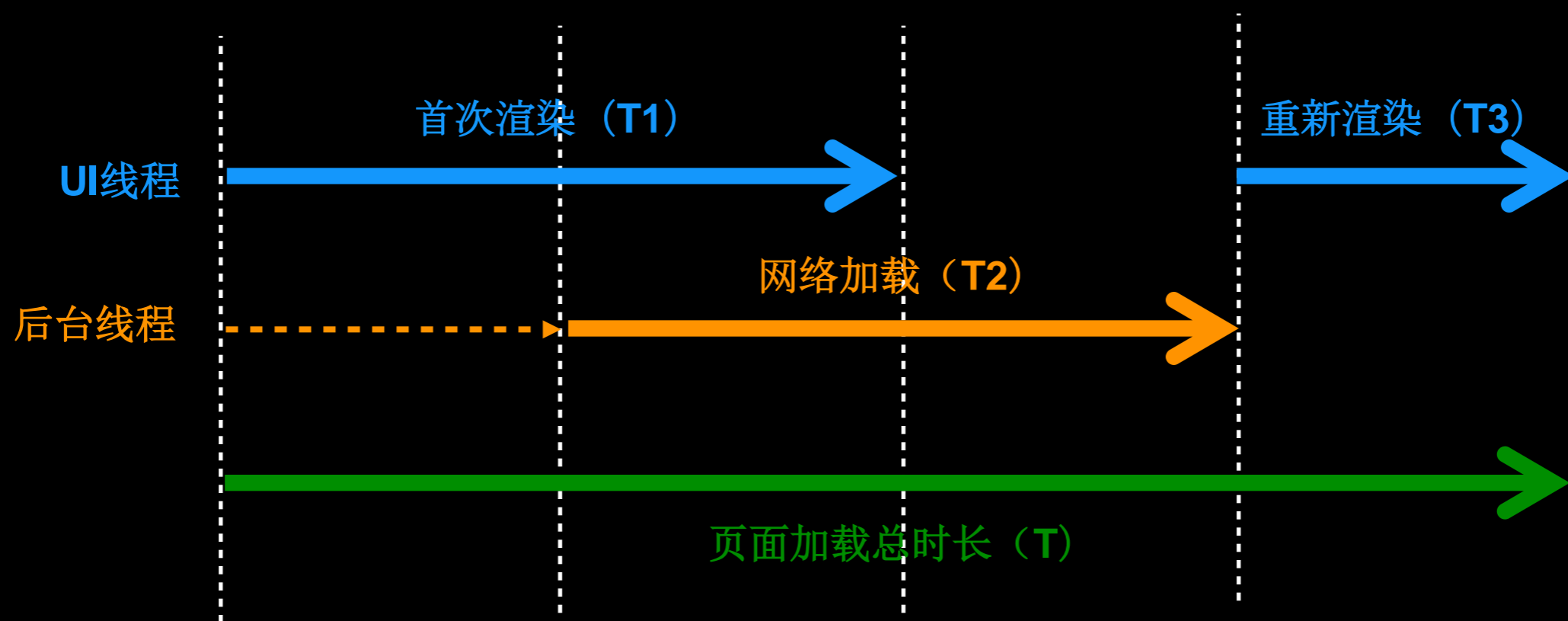
- 限速
- 熔断



页面加载

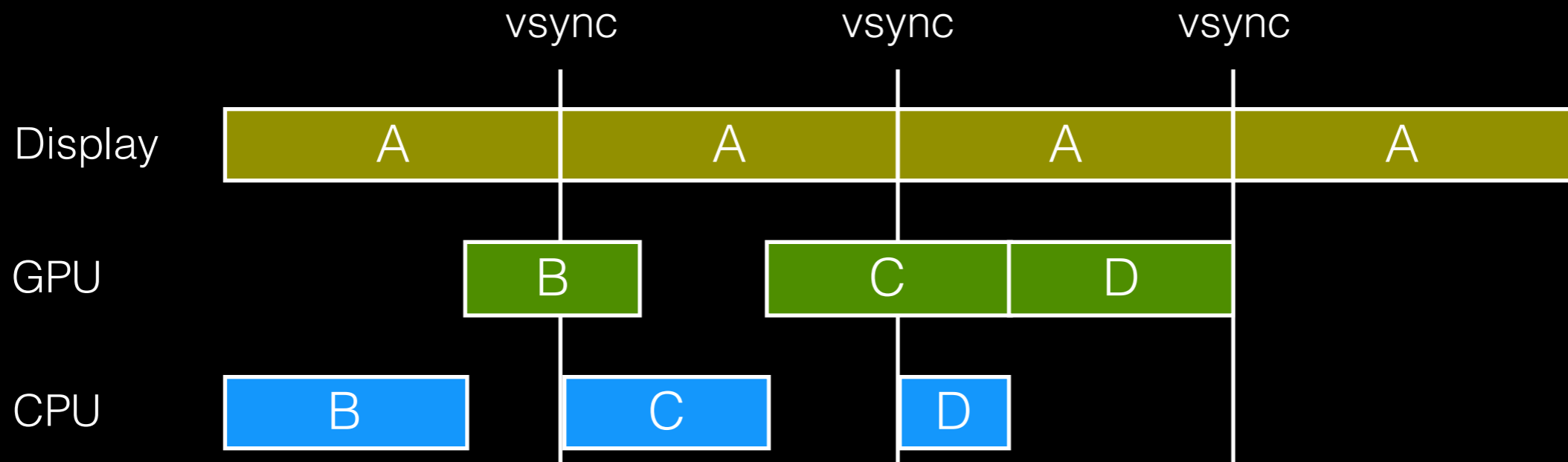
- 准确性
- 无埋点

典型的页面加载模型

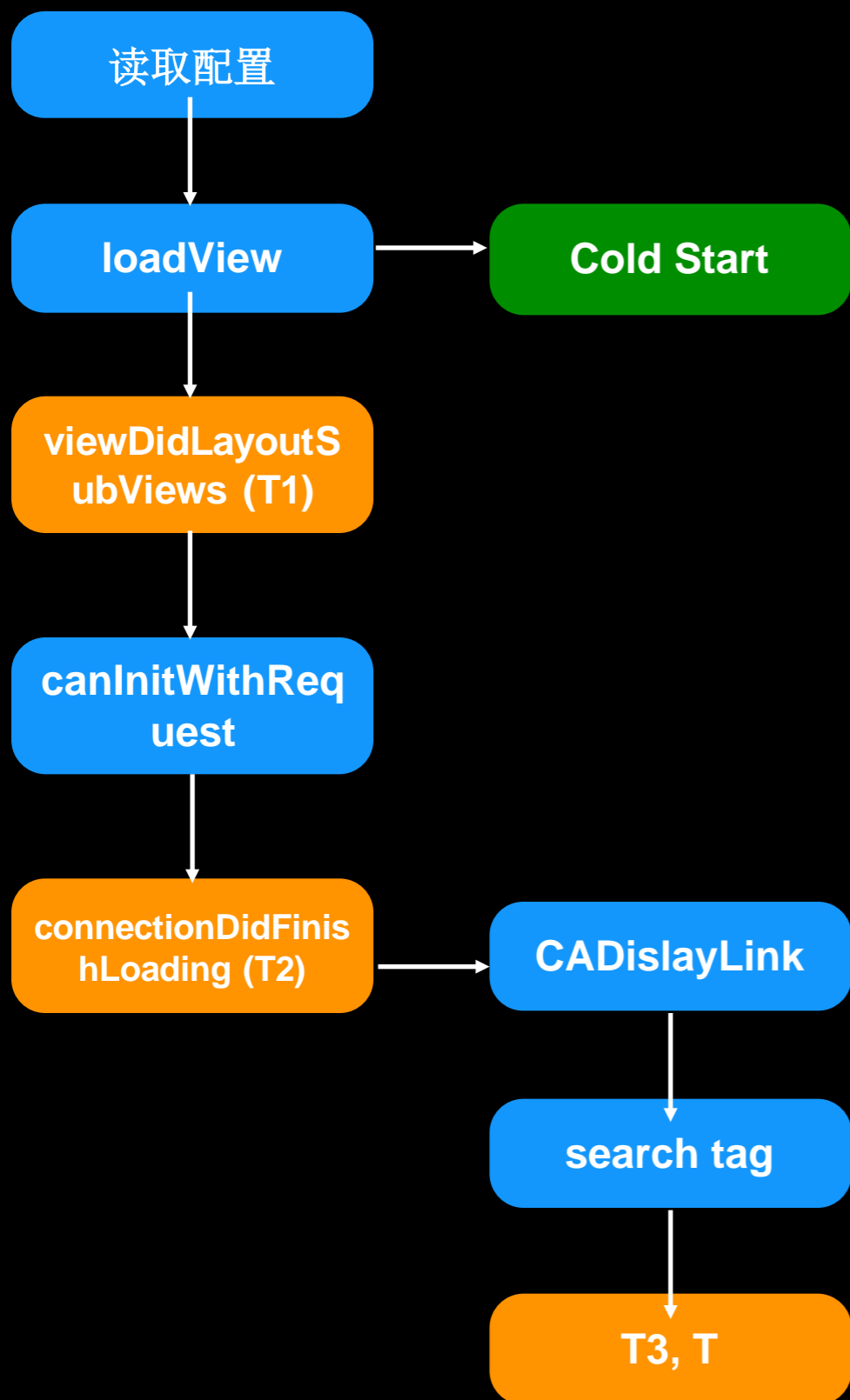


渲染时间的计算

- Runnable
- vsync信号
- dispatchDraw



页面加载配置

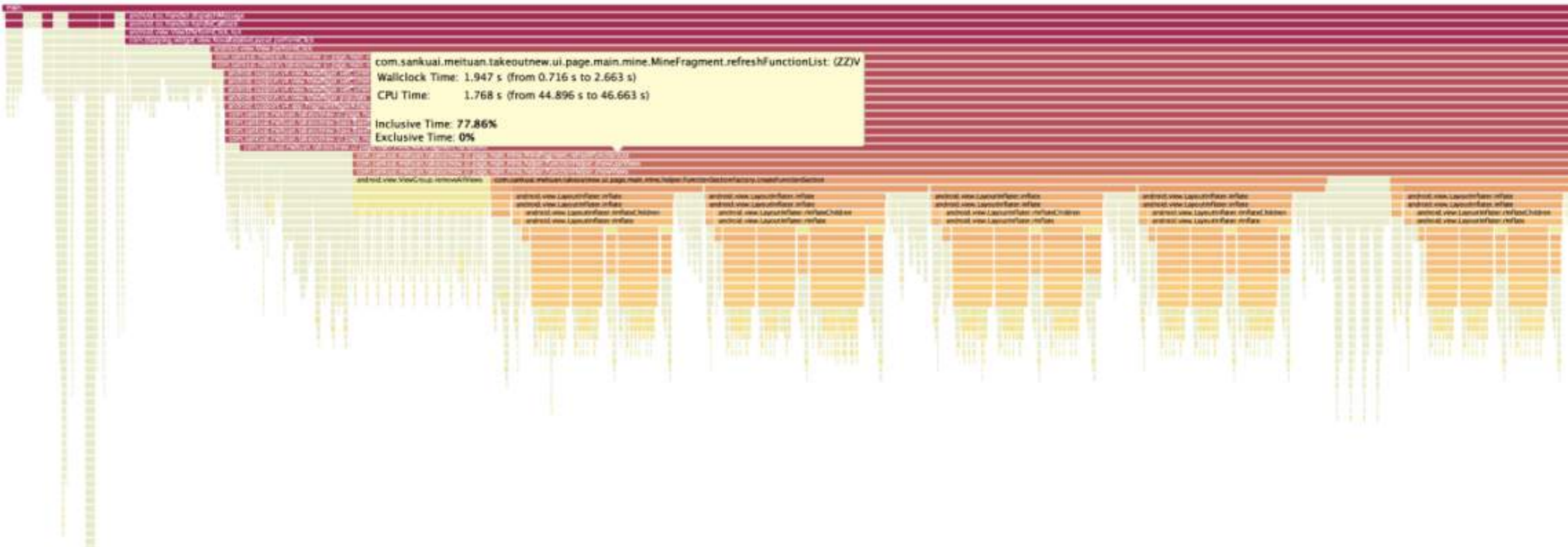


```
{
  "page": "WMHomeController",
  "tag": 1001,
  "api": ["/user/functions/list",
    "/home/head",
    "/poi/filter"],
  "is_main_page": true
},
{
  "page": "WMRestaurantViewController",
  "tag": 1002,
  "api": ["/poi/food"]
},
{
  "page": "WMGoodsDetailViewController",
  "tag": 1004,
  "api": ["/poi/product/info"]
},
{
  "page": "WMConfirmOrderViewController",
  "tag": 1005,
  "api": []
}
```

方法耗时

- AOP = Aspect Oriented Programming
- 分析指定包/类/方法中的耗时
- 跳过简单方法（根据class字节码的tableLength判断）
- 分析的线程：只分析主线程/分析所有线程
- 指定最大调用层级

方法耗时

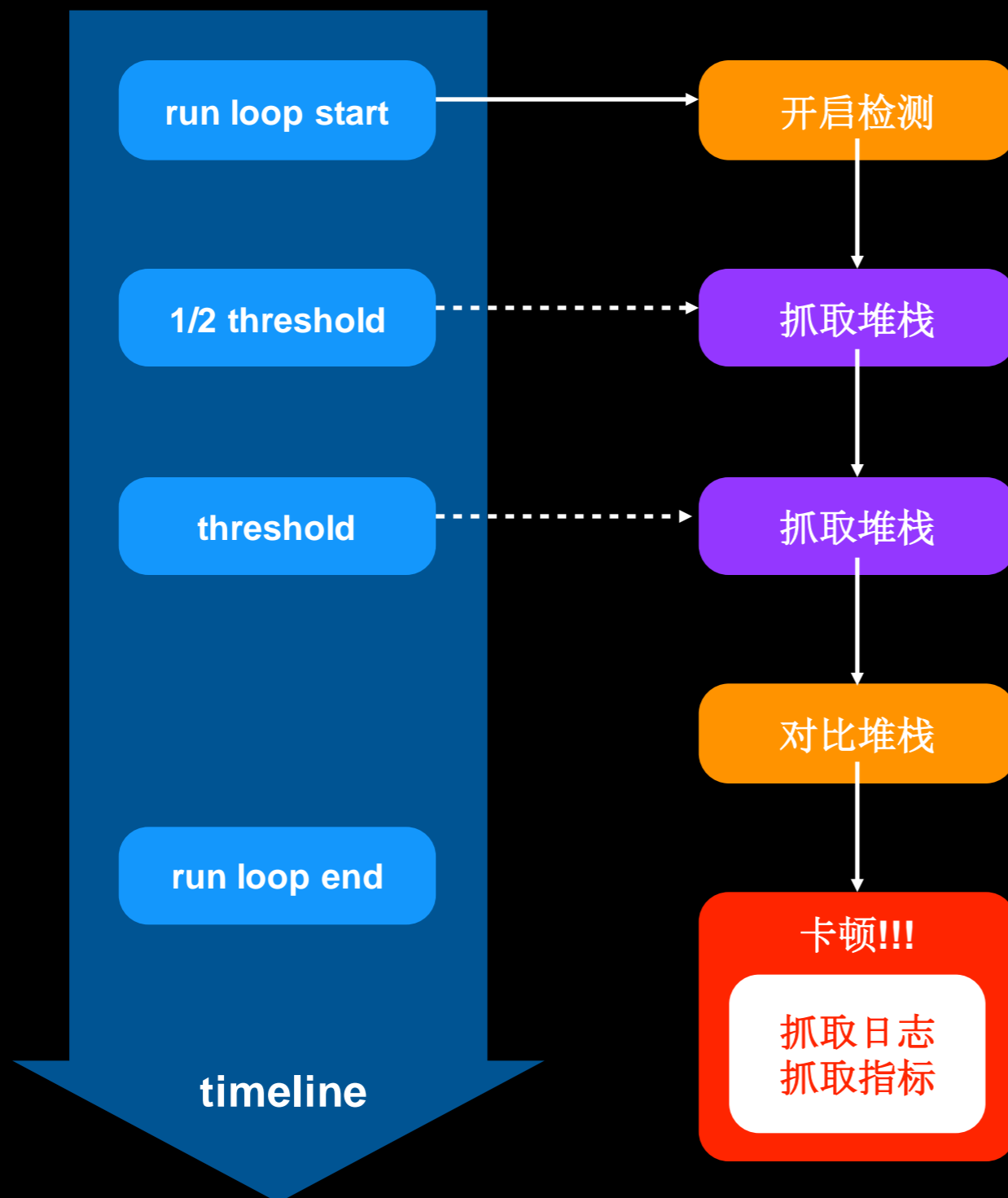


Name	Invocation Count	Inclusive Time (μs)	Exclusive Time (μs)
Thread main			
main	1	2,500,491 100.0%	691 0.0%
android.os.Handler.dispatchMessage	23	2,423,733 96.9%	281 0.0%
android.os.Handler.handleCallback	6	2,422,673 96.9%	125 0.0%
android.view.View\$PerformClick.run	1	2,306,530 92.2%	9 0.0%
com.dianping.widget.view.NovaRelativeLayout.performClick	1	2,306,521 92.2%	29 0.0%
android.view.View.performClick	1	2,172,292 86.9%	22 0.0%
com.sankuai.meituan.takeoutnew.ui.page.main.view.TitleIndicator.onClick	1	2,169,800 86.8%	18 0.0%
com.sankuai.meituan.takeoutnew.ui.page.main.view.TitleIndicator.setCurrentTab	1	2,169,775 86.8%	43 0.0%

卡顿

- FPS
- run loop

卡顿检测



- 防止误判
- 有效堆栈

卡顿检测

```
class LooperPrinter implements Printer {
    private static final String START_TAG=">>>> Dispatching to";
    private static final String FINISH_TAG="<<<<< Finished to";

    @Override
    public void println(String x) {
        switch (isStart(x)){
            case START:
                mStartedPrinting = true;
                detect();
                break;
            case FINISH:
                mStartedPrinting = false;
                if (isBlock()) {
                    ...
                }
                break;
            default:
        }
    }
}
```

Android

卡顿检测

```
- (void)main
{
    CFRunLoopAddObserver(CFRunLoopGetMain(), self.runloopObserver, kCFRunLoopCommonModes);
    self.semaphore = dispatch_semaphore_create(0);
    while (!self.cancelled) {
        long status = dispatch_semaphore_wait(self.semaphore, dispatch_time(DISPATCH_TIME_NOW, self.t
        if (status != 0 && !self.suspended) {
            if (self.cancelled || self.semaphore == NULL) {
                return;
            }
            if (self.lastActivity == kCFRunLoopBeforeSources || self.lastActivity == kCFRunLoopAfter
                if (++self.counter < self.maxCount) {
                    continue;
                }
                self.handlerBlock();
                dispatch_semaphore_wait(self.semaphore, DISPATCH_TIME_FOREVER);
            }
        }
        self.counter = 0;
    }
}
```

iOS

堆栈归类

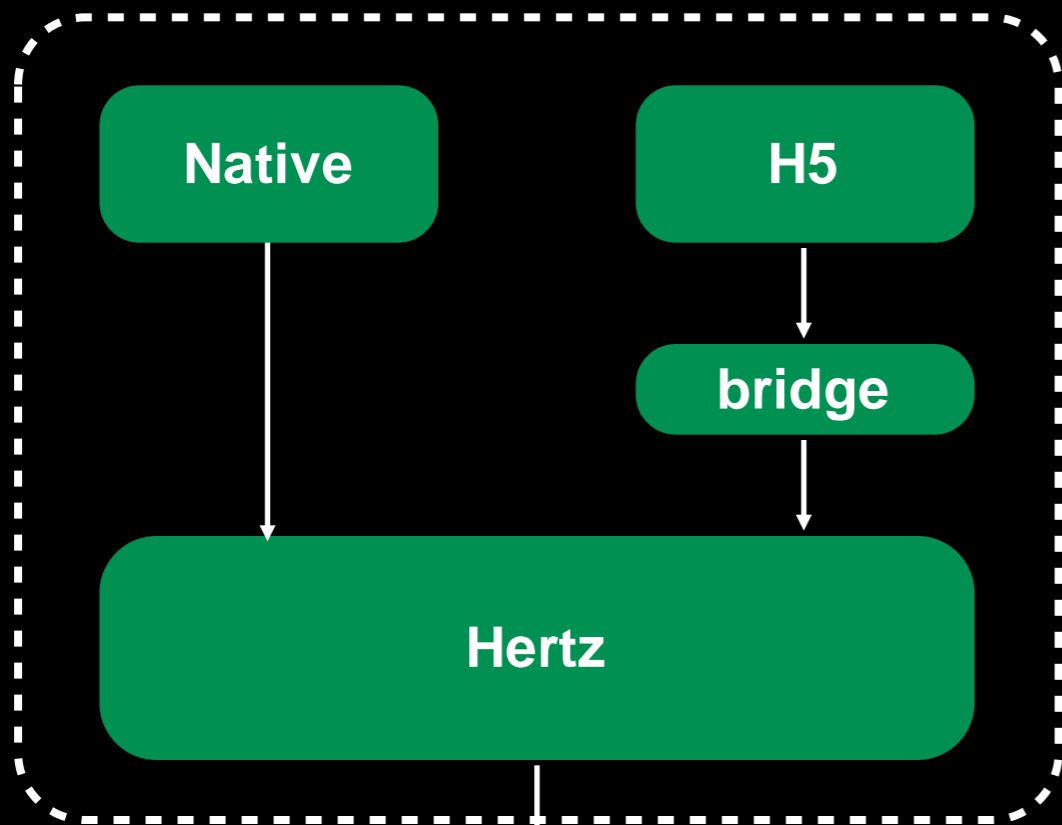
- 最内层归类
- 最外层归类
- 自定义规则归类
- 相似度归类

堆栈归类

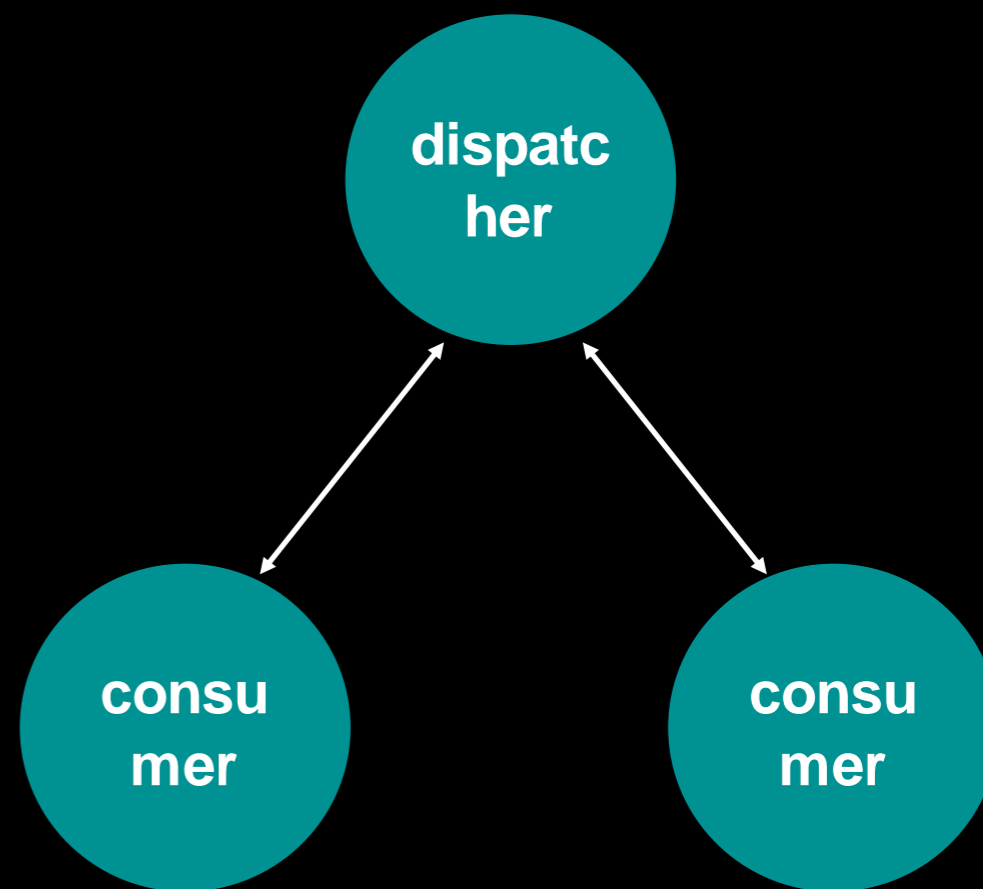
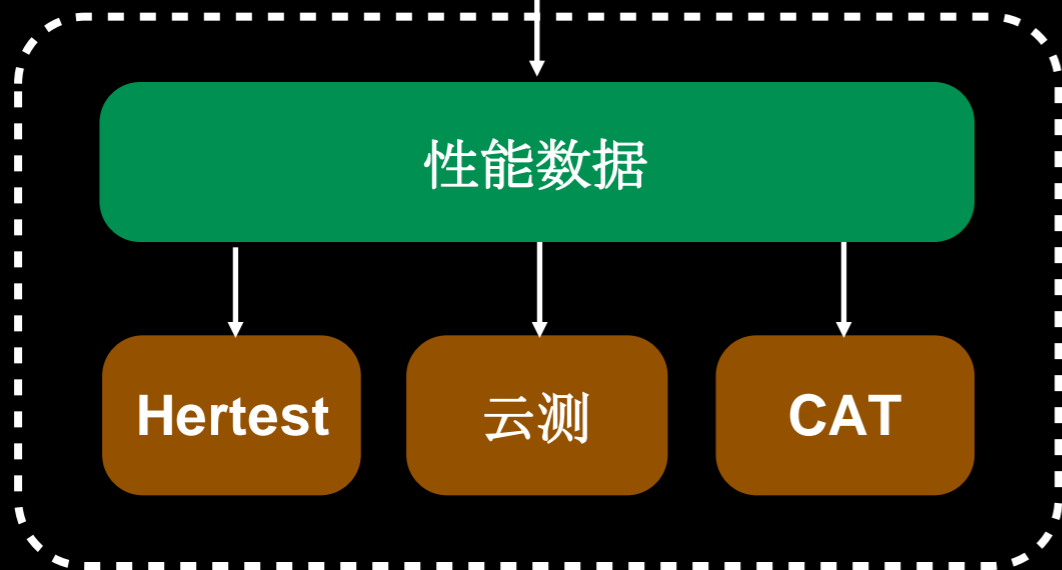
```
android.database.sqlite.SQLiteConnection.nativeExecuteForLastInsertedRowId(Native Method)
android.database.sqlite.SQLiteConnection.executeForLastInsertedRowId(SQLiteConnection.java:784)
android.database.sqlite.SQLiteSession.executeForLastInsertedRowId(SQLiteSession.java:788)
android.database.sqlite.SQLiteStatement.executeInsert(SQLiteStatement.java:86)
de.greenrobot.dao.AbstractDao.executeInsert(ProGuard:306)
de.greenrobot.dao.AbstractDao.insert(ProGuard:276)
com.sankuai.meituan.takeoutnew.db.dao.BaseAbstractDao.insert(ProGuard:25) ←
com.sankuai.meituan.takeoutnew.log.LogDataUtil.a(ProGuard:1244)
com.sankuai.meituan.takeoutnew.log.LogDataUtil.b(ProGuard:134)
com.sankuai.meituan.takeoutnew.log.LogDataUtil.a(ProGuard:130)
cuj$1.a(ProGuard:62)
cuq$1.onLocationChanged(ProGuard:40)
aqj.handleMessage(ProGuard:0)
android.os.Handler.dispatchMessage(Handler.java:99)
android.os.Looper.loop(Looper.java:137)
android.app.ActivityThread.main(ActivityThread.java:5473)
java.lang.reflect.Method.invokeNative(Native Method)
java.lang.reflect.Method.invoke(Method.java:525)
com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:804)
com.android.internal.os.ZygoteInit.main(ZygoteInit.java:620)
dalvik.system.NativeStart.main(Native Method)
```

如何使用数据

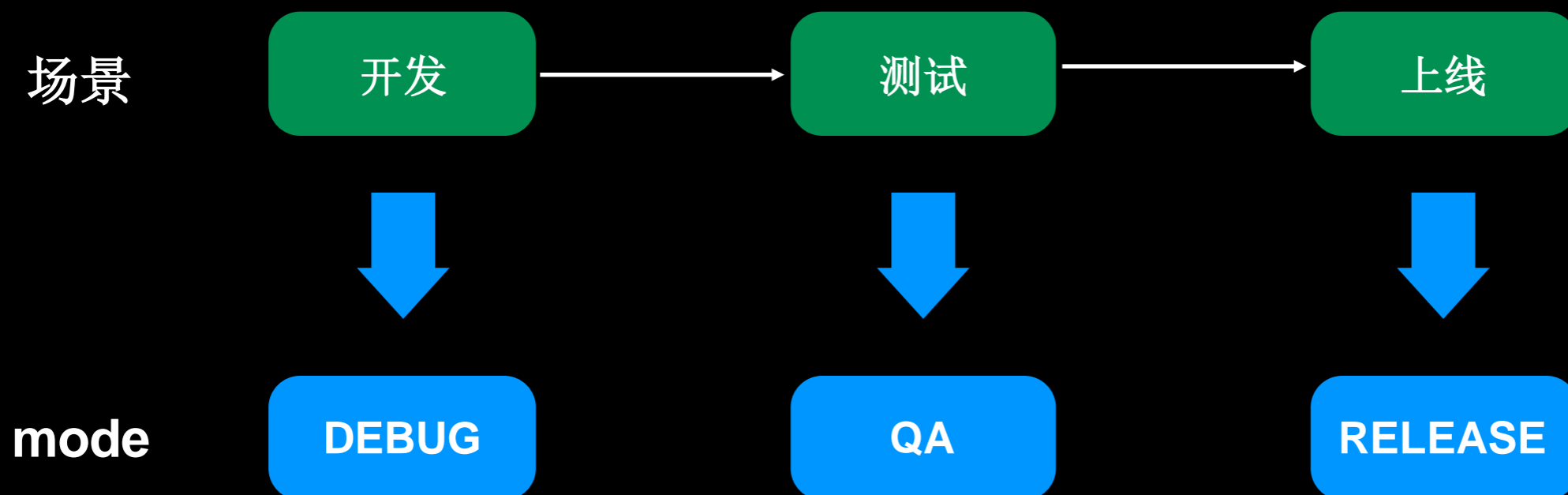
采集



上报



如何使用数据

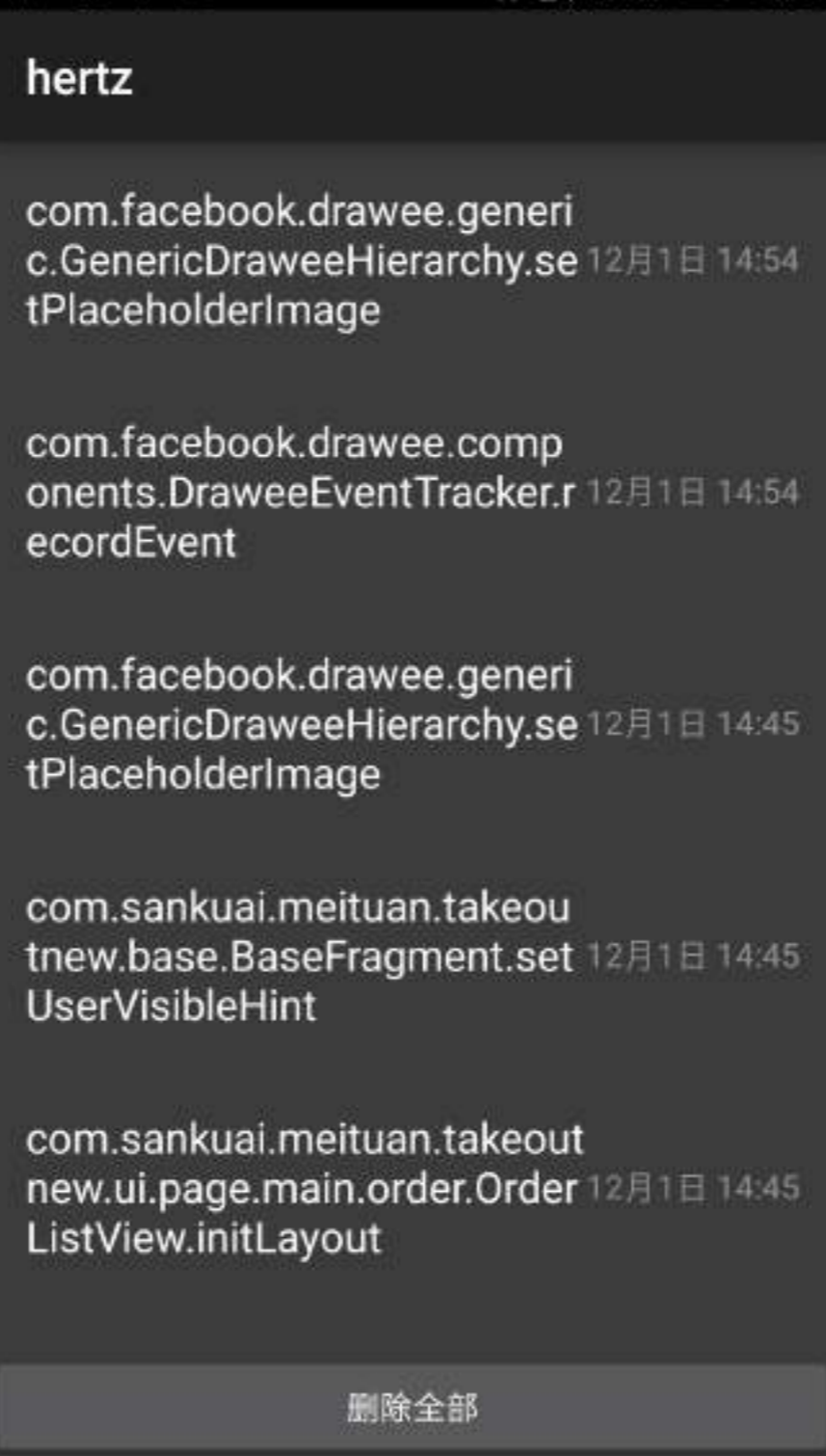


Native

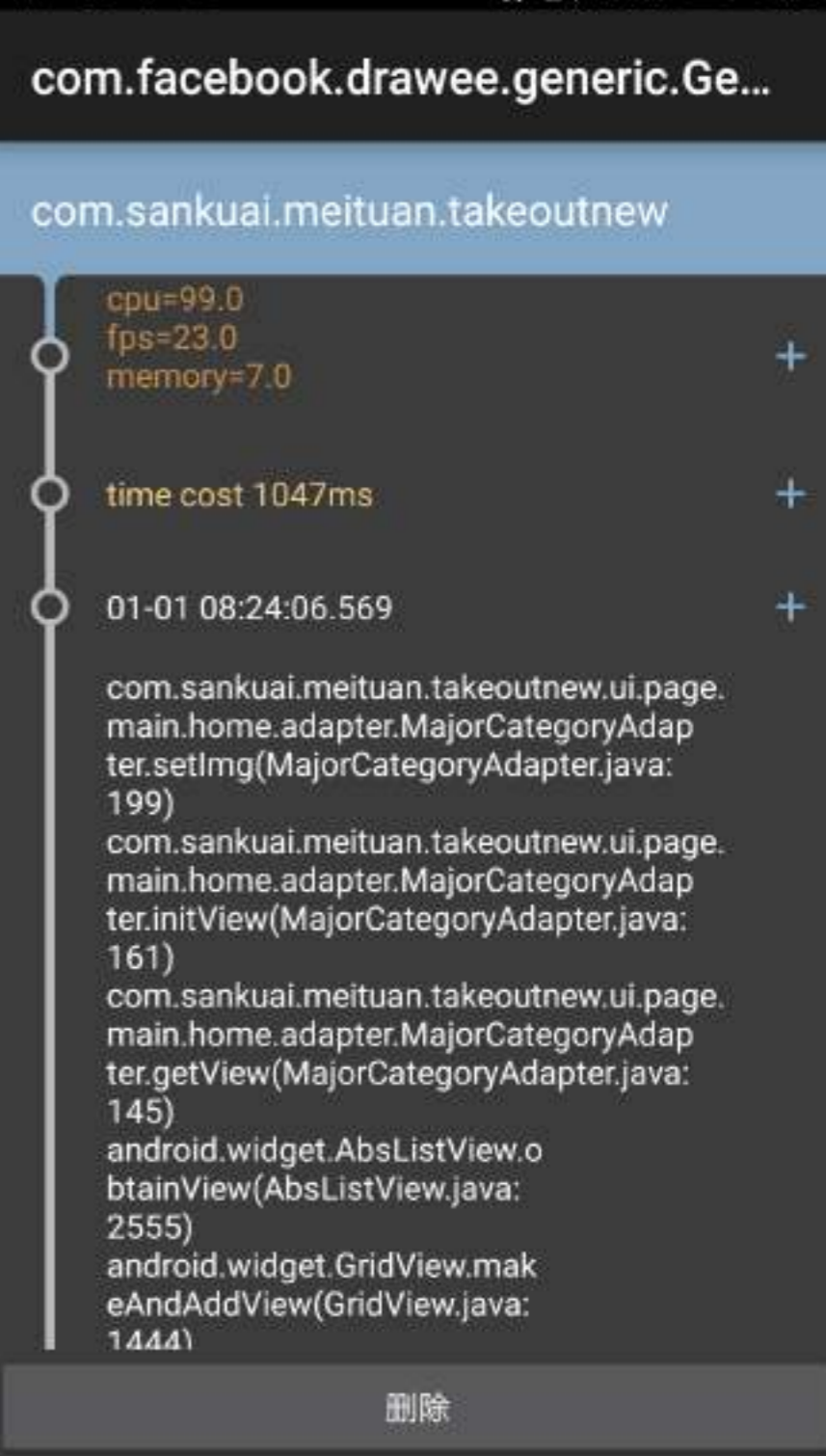
100% 14:55



100% 14:57



100% 14:57



E/Hertz: =====

|12-01 14:42:19.538

当前卡顿堆栈信息:

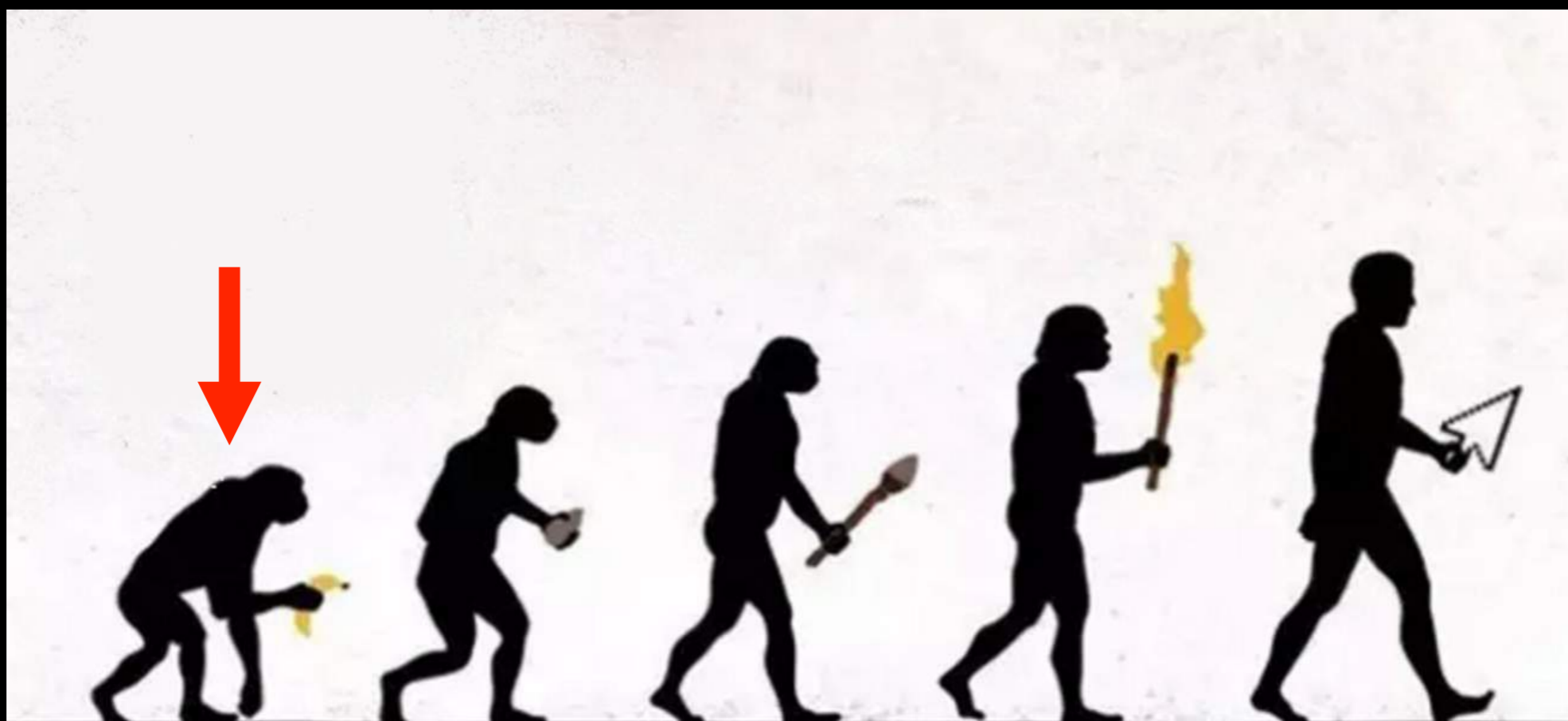
```
java.lang.reflect.Constructor.newInstance(Native Method)
android.view.LayoutInflater.createView(LayoutInflater.java:619)
android.view.LayoutInflater.createViewFromTag(LayoutInflater.java:764)
android.view.LayoutInflater.createViewFromTag(LayoutInflater.java:704)
android.view.LayoutInflater.rInflate(LayoutInflater.java:835)
android.view.LayoutInflater.rInflateChildren(LayoutInflater.java:798)
android.view.LayoutInflater.rInflate(LayoutInflater.java:838)
android.view.LayoutInflater.rInflateChildren(LayoutInflater.java:798)
android.view.LayoutInflater.rInflate(LayoutInflater.java:838)
android.view.LayoutInflater.rInflateChildren(LayoutInflater.java:798)
android.view.LayoutInflater.rInflate(LayoutInflater.java:838)
android.view.LayoutInflater.rInflateChildren(LayoutInflater.java:798)
android.view.LayoutInflater.inflate(LayoutInflater.java:515)
android.view.LayoutInflater.inflate(LayoutInflater.java:423)
com.sankuai.meituan.takeoutnew.ui.page.main.mine.MineFragment.onCreateView(MineFragment.java:221)
android.support.v4.app.Fragment.performCreateView(Fragment.java:1974)
android.support.v4.app.FragmentManagerImpl.moveToState(FragmentManager.java:1067)
android.support.v4.app.FragmentManagerImpl.moveToState(FragmentManager.java:1252)
android.support.v4.app.BackStackRecord.run(BackStackRecord.java:742)
android.support.v4.app.FragmentManagerImpl.execPendingActions(FragmentManager.java:1617)
android.support.v4.app.FragmentManagerImpl.executePendingTransactions(FragmentManager.java:570)
android.support.v4.app.FragmentPagerAdapter.finishUpdate(FragmentPagerAdapter.java:141)
android.support.v4.view.ViewPager.populate(ViewPager.java:1177)
android.support.v4.view.ViewPager.populate(ViewPager.java:1025)
android.support.v4.view.ViewPager.onMeasure(ViewPager.java:1545)
android.view.View.measure(View.java:19260)
```

测试报告?

Hertz → 导出日志 → 执行脚本 →



测试报告?



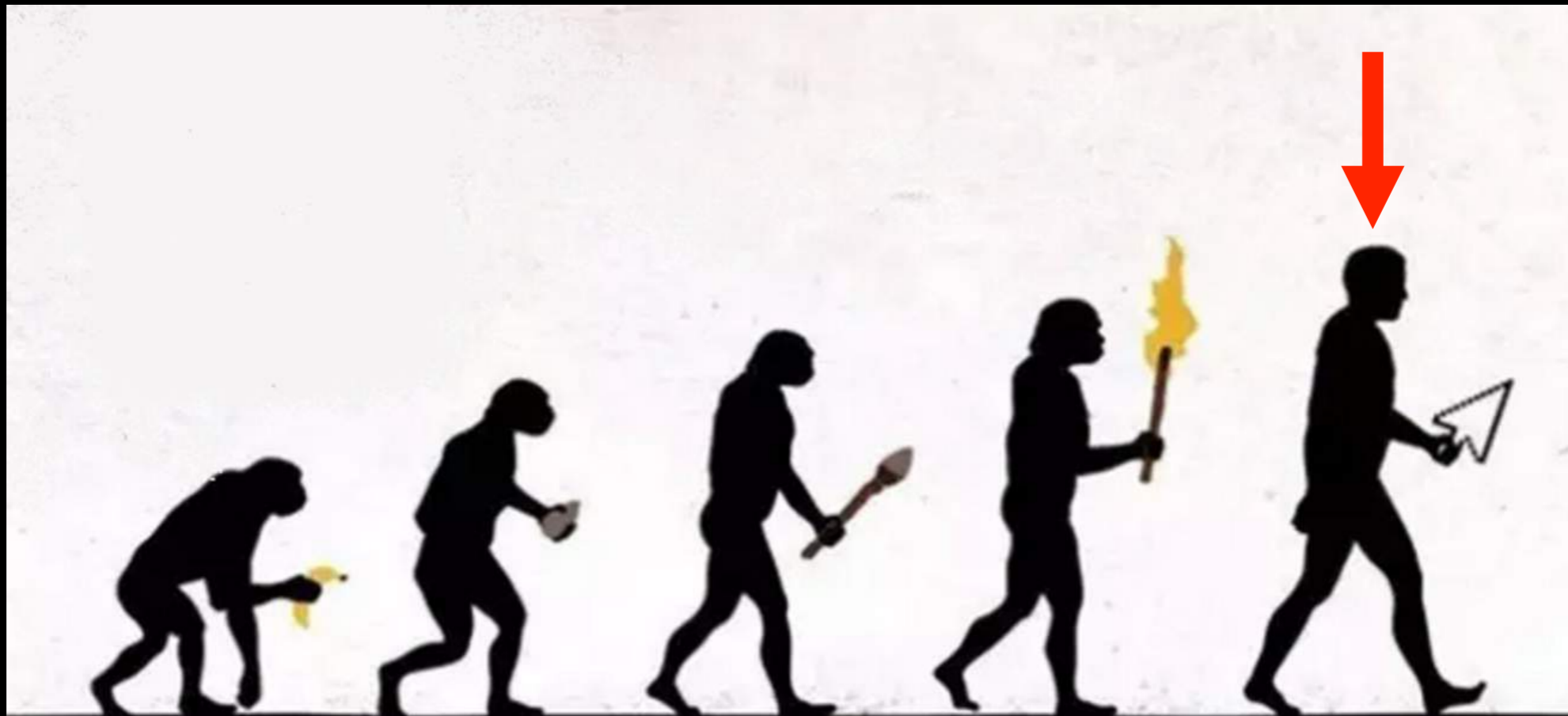
测试报告?

Hertz周期性采集日志报告 (第二轮第二天-1107)

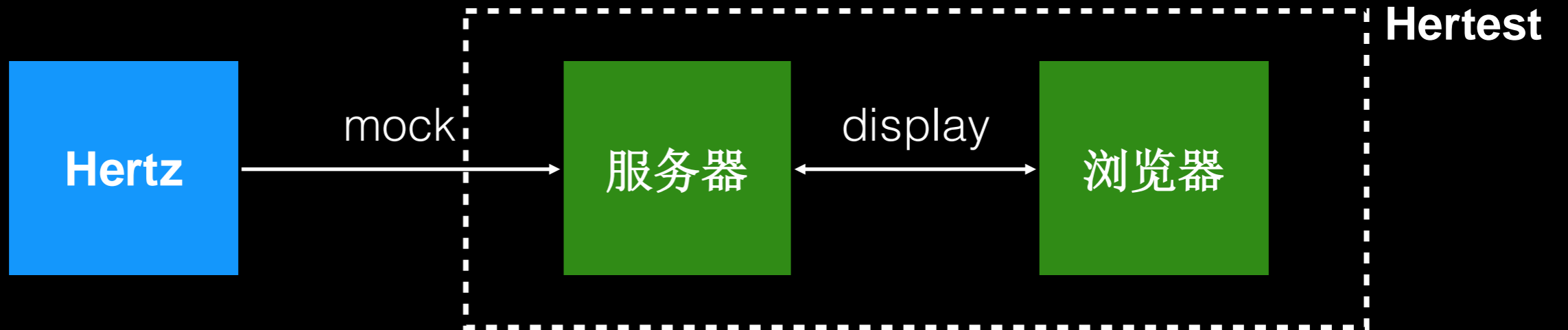
测试前置条件	机型	OPPO R7Plust			
	系统	Android 5.0			
	网络	公司WIFI			
	UUID	EA498B26BB78BE1FB57EDFEB2ABCDA0F510EEA3B516AE0D0E361968E0CFA825D			
	版本	外卖App V5.2.0			
测试执行时间	2016年11月7日				
页面采集	页面	CPU	MEM	FPS	-
	BindPhoneActivity	34	4	55	-
	BindWxFriendsActivity	18	12	54	-
	WXEntryActivity	35	10	51	-
	LoginActivity	33	5	59	-
	MyAccountActivity	27	9	55	-
	PoiListFilterActivity	28	8	41	-
页面测速	页面	页面加载总时间	初次渲染时间	加载数据时间	重新渲染时间
	MainActivity	2510ms	946ms	1274ms	176ms
	PoiListFilterActivity	1242ms	198ms	913ms	212ms
卡顿次数	52				
CAT系统数据汇总	CAT系统-外卖APP Android V5.2.0第二轮第二天Hertz SDK性能测试数据日志 (logcat日志见附件)				

Aha, Hertest !

可追溯, 可视化



Hertest



```
{  
  "topic": "performance.cpu:16228771",  
  "payload": {  
    "value": 10,  
    "tm":1479795870288  
  },  
  "common": {  
    "app": "美团外卖",  
    "platform": "iOS",  
    "device": "iPhone6",  
    "system": "10.0.2",  
    "uuid": "325ED1844489FF68B6***"  
  }  
}
```

Sessions

UnionId: 日期:

UnionID	App	Platform	Device	Start Time	Duration	
be24760c-b561-11e6-9a79-a0999b0f89bf	美团	iOS	iPhone7	2016-11-28 19:56:48	90 sec	详情
bde8cd30-b561-11e6-8b80-a0999b0f89bf	美团外卖	iOS	iPhone7	2016-11-28 19:56:48	90 sec	详情
bdad8be6-b561-11e6-8e40-a0999b0f89bf	大众点评	iOS	iPhone7	2016-11-28 19:56:48	90 sec	详情
bd716640-b561-11e6-b475-a0999b0f89bf	美团	iOS	iPhone6	2016-11-28 19:56:47	90 sec	详情
bd359a4f-b561-11e6-9b17-a0999b0f89bf	大众点评	iOS	iPhone7	2016-11-28 19:56:47	90 sec	详情
bcf691b5-b561-11e6-9bc2-a0999b0f89bf	美团	iOS	iPhone6	2016-11-28 19:56:46	90 sec	详情
bcba3c47-b561-11e6-886d-a0999b0f89bf	大众点评	iOS	iPhone6s	2016-11-28 19:56:46	90 sec	详情
bc7dc694-b561-11e6-bd2e-a0999b0f89bf	美团外卖	iOS	iPhone6	2016-11-28 19:56:46	90 sec	详情
bc3afa45-b561-11e6-8630-a0999b0f89bf	美团	iOS	iPhone6	2016-11-28 19:56:45	90 sec	详情
bbf85d26-b561-11e6-b7b8-a0999b0f89bf	美团	iOS	iPhone7	2016-11-28 19:56:45	90 sec	详情
de7a8b78-b55f-11e6-8768-a0999b0f89bf	美团	iOS	iPhone6	2016-11-28 19:43:24	90 sec	详情
de439078-b55f-11e6-b8ab-a0999b0f89bf	美团	iOS	iPhone6	2016-11-28 19:43:23	90 sec	详情

Overview

CPU

内存

FPS

测速

冷启动

流量

电量

卡顿

异常

日志

Overview

App: 11

Device: Nexus 6P

Platform: android

StartTime: 2017-01-06 17:03:54

System: 7.0

Duration: 3m46s

UnionID: 315D634A137AF6E6FA46DE54F5E053A18E1BD79030AB07F9AA2336909B44BB64

指标	数值	基线值
CPU	24%	8%
内存	79M	50M
FPS	56	55
冷启动	3899 msec	3000 msec
卡顿	0	0
流量	NA	NA
电量	NA	NA
异常	NA	NA

Overview

CPU

内存

FPS

测速

冷启动

流量

电量

卡顿

异常

日志

内存



均值	方差	基线值
79M	568	50M

Page	Value	Count
RestaurantActivity	96M	9
MainActivity	57M	7

Overview

CPU

内存

FPS

测速

冷启动

流量

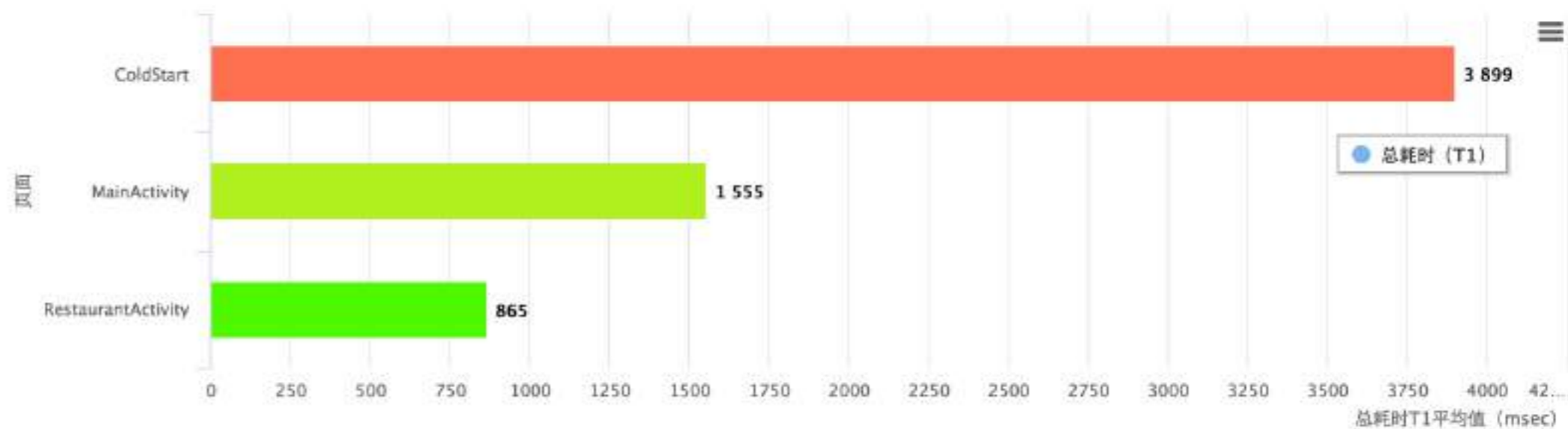
电量

卡顿

异常

日志

测速



MainActivity

Page	Step	Count	Cost(msec)
MainActivity	T1	2	1555
MainActivity	T3	2	777
MainActivity	T2	2	541
MainActivity	T4	2	39

RestaurantActivity

Page	Step	Count	Cost(msec)
RestaurantActivity	T1	2	865

Overview

CPU

内存

FPS

测速

冷启动

流量

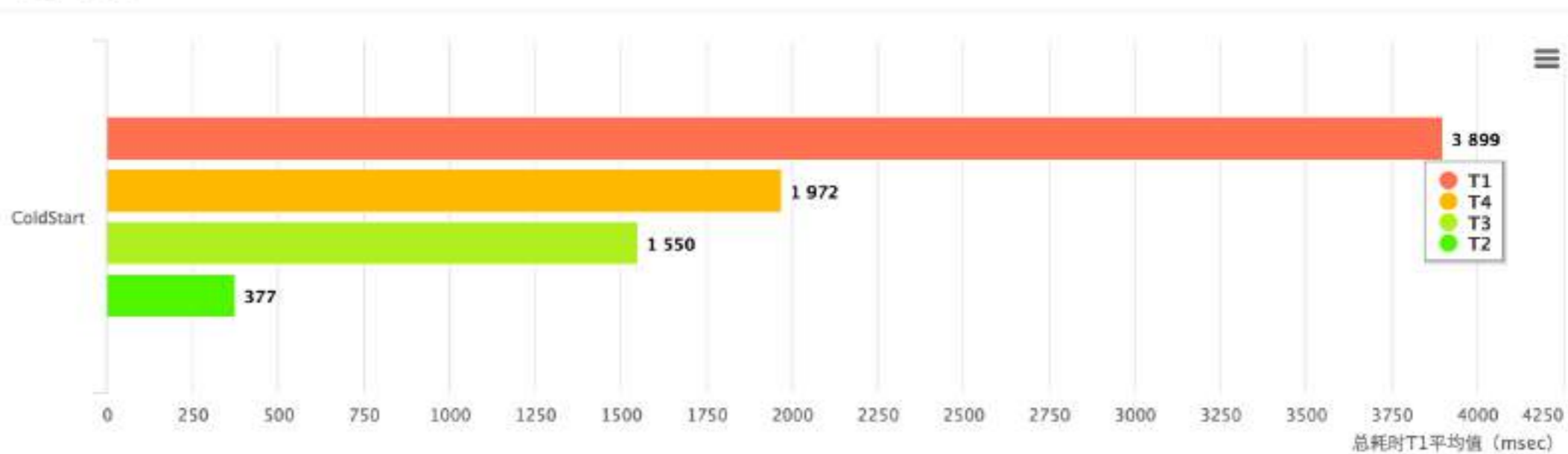
电量

卡顿

异常

日志

冷启动



Page	Step	StartTime	Tm	Cost(msec)
ColdStart	T1	1970-01-01 08:00:00	2017-01-06 17:03:54	3899
ColdStart	T2	1970-01-01 08:00:00	2017-01-06 17:03:54	377
ColdStart	T3	1970-01-01 08:00:00	2017-01-06 17:03:54	1550
ColdStart	T4	1970-01-01 08:00:01	2017-01-06 17:03:54	1972

Overview

CPU

内存

FPS

测速

冷启动

流量

电量

卡顿

卡顿

卡顿次数: 5次

result	time	cost(msec)
com.sankuai.android.hertz.demo.performance.LaggyCreateActivity.onCreate	2017-01-06 16:50:13	534
com.sankuai.android.hertz.demo.performance.LaggyListActivity\$DemoAdapter.getView	2017-01-06 16:50:10	337
com.sankuai.android.hertz.demo.performance.LaggyListActivity\$DemoAdapter.getView	2017-01-06 16:50:08	337
com.sankuai.android.hertz.demo.performance.LaggyListActivity\$DemoAdapter.getView	2017-01-06 16:49:56	304
com.sankuai.android.hertz.demo.performance.LaggyListActivity\$DemoAdapter.getView	2017-01-06 16:49:55	304

FPS

测速

冷启动

流量

电量

卡顿

异常

日志

time	2017-01-06 16:50:13
result	com.sankuai.android.hertz.demo.performance.LaggyCreateActivity.onCreate
stack	<pre> java.lang.Object.wait(Native Method) java.lang.Object.wait(Object.java:407) java.lang.Thread.join(Thread.java:1290) java.lang.Thread.join(Thread.java:1366) com.sankuai.android.hertz.demo.performance.LaggyCreateActivity.onCreate(LaggyCreateActivity.java:34) android.app.Activity.performCreate(Activity.java:6664) android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1118) android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2599) android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:2707) android.app.ActivityThread.-wrap12(ActivityThread.java) android.app.ActivityThread\$H.handleMessage(ActivityThread.java:1460) android.os.Handler.dispatchMessage(Handler.java:102) android.os.Looper.loop(Looper.java:154) android.app.ActivityThread.main(ActivityThread.java:6077) java.lang.reflect.Method.invoke(Native Method) com.android.internal.os.ZygoteInit\$MethodAndArgsCaller.run(ZygoteInit.java:865) com.android.internal.os.ZygoteInit.main(ZygoteInit.java:755) </pre>

Hertest还能做什么

- 同一设备多维度的可视化监控
- 收集运行日志
- 收集崩溃
- 配置管理
- 定位QA妹纸的行踪？
- ...

APP的另一个界面

CAT Performance

- 线上统计能做什么？
 - 配置
 - 统计
 - 报警
 - 个案追查

- Dashboard
- Mobile
- Performance
- Config

开始 2016-11-09 19:00 结束 20:28 平台 android App 美团外卖 Type ALL Processor ALL UnionId

查询

APP版本 All 5.2.0 5.1.7 1.0

平台版本 All 6.0.1 6.0 5.1.1 5.1 5.0.2 5.0.1 5.0 4.4.4 4.4.3 4.4.2 4.3 4.2.1

模块 All Unknown

级别 All ERROR

设备

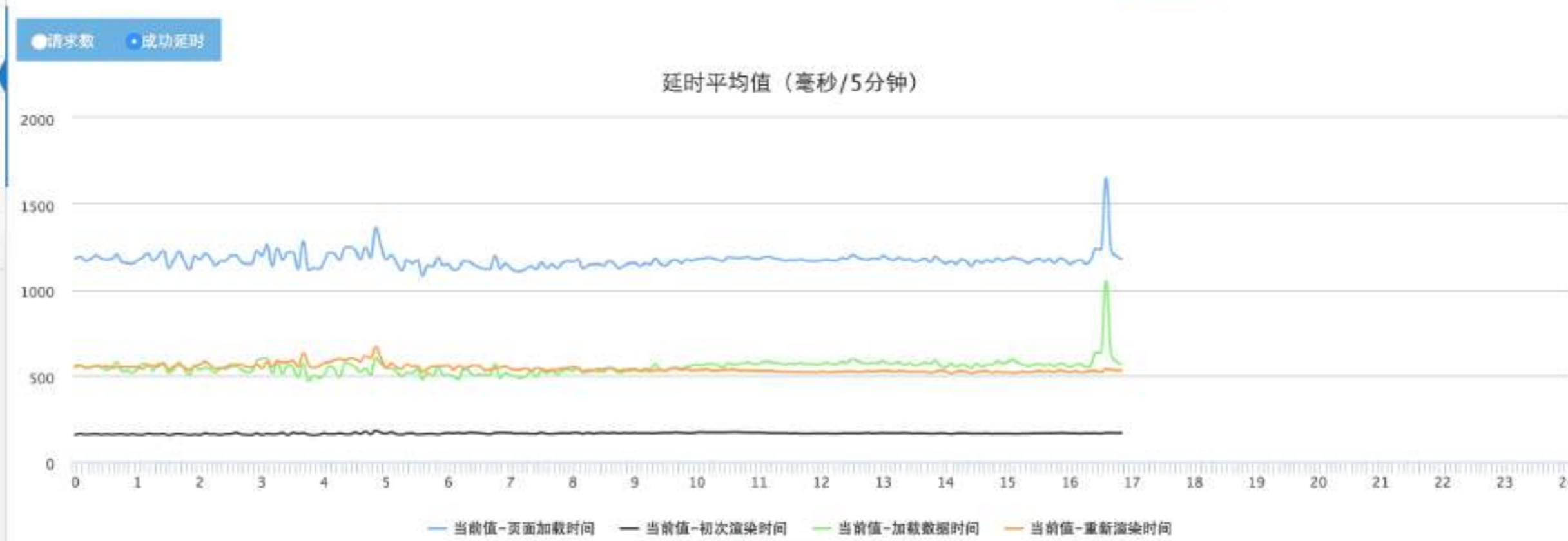
详细日志 统计图表

Msg	Count	SampleLinks
Total	33	
[:: show ::] com.meituan.android.common.candy.CandyPreprocessor	5	Log
[:: show ::] pnf.this.object.does.not.Exist	4	Log
[:: show ::] com.meituan.android.common.fingerprint.FingerprintManager	4	Log
[:: show ::] com.sankuai.meituan.takeoutnew.adapter.FoodCategoryAdapter	3	Log
[:: show ::] com.sankuai.meituan.takeoutnew.adapter.FreshFruitListAdapter	2	Lg
[:: show ::] com.meituan.waimai.pbi.library.TraceTagManager	1	L
[:: show ::] com.sankuai.android.hertz.utils.L	1	L
[:: show ::] libcore.icu.NativeDecimalFormat	1	L

- Dashboard
- Mobile
- Performance
 - APP页面测速
 - 性能监控报表
 - 自定义报表
- Config

日期: 2016-11-25 App: 美团外卖 页面: RestaurantActivity 网络类型: All APP版本: All

平台: All 平台版本: All 地区: All 运营商: All 设备: 输入机型 查询 选择对比



汇总信息

计算规则: 所选天中288个5分钟点数据求和再平均的结果

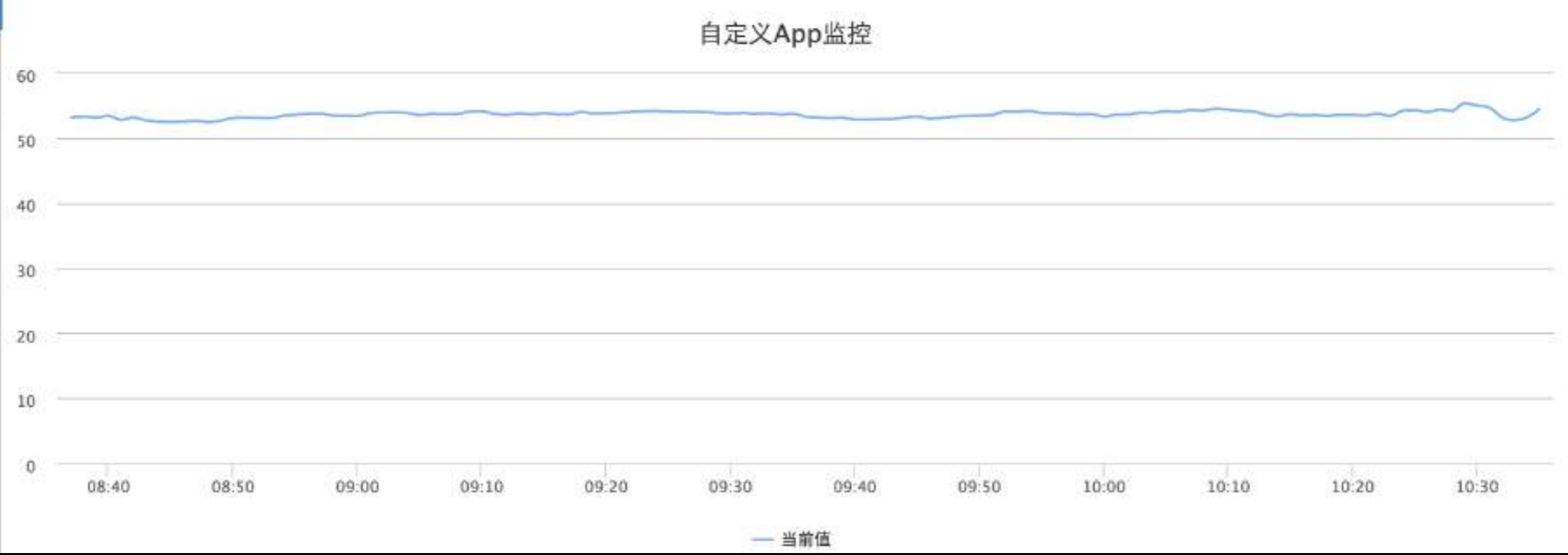
测速点	时间	访问总次数	平均延时(ms)
当前值-页面加载时间	2016-11-25	3,239,697	1,177
当前值-初次渲染时间	2016-11-25	3,267,258	165
当前值-加载数据时间	2016-11-25	3,240,862	568
当前值-重新渲染时间	2016-11-25	3,236,674	527

- Dashboard
- Mobile
 - APP页面测速
 - 性能监控报表
 - 自定义报表
- Config

开始 2016-11-30 08:37 结束 2016-11-30 10:37 App 美团外卖 页面 RestaurantActivity 指标 fps
 APP版本 All 平台 All 平台版本 All 设备 Q 输入机型 聚合 求平均 查询 选择对比

当天分位值统计

类别	tp90	tp95	tp99	tp999
当前值	57.126	57.139	57.145	57.145



CAT Performance

- 统计值怎么计算？
 - 平均
 - 低分位
- 哪些数据是有用的？
 - 最差机型
 - 最差页面

Dashboard

Mobile

Performance

Config

Appid注册

App命令字

Api分组

Api规则

App返回码

App测速

App告警模板

App告警

App常量

App版本

自定义配置

机型配置

测速映射

Crash配置

Crash告警

日志告警

选择App

美团外卖

操作 +



页面

加载阶段

DemoRenderSimpleActivity

测速点ID

测速点说明

1

页面加载总时间

2

初次渲染时间

3

加载数据时间

4

重新渲染时间

PoiListFilterActivity

测速点ID

测速点说明

1

页面加载总时间

2

初次渲染时间

3

加载数据时间

4

重新渲染时间

WMRestaurantViewController

测速点ID

测速点说明

1

页面加载总时间

2

初次渲染时间

3

加载数据时间

4

重新渲染时间

WMGoodsDetailViewController

测速点ID

测速点说明

1

页面加载总时间

2

初次渲染时间

Dashboard

Mobile

Performance

Config

Appid注册

App命令字

Api分组

Api规则

App返回码

App测速

App告警模板

App告警

App常量

App版本

自定义配置

机型配置

测速映射

移动SDK配置

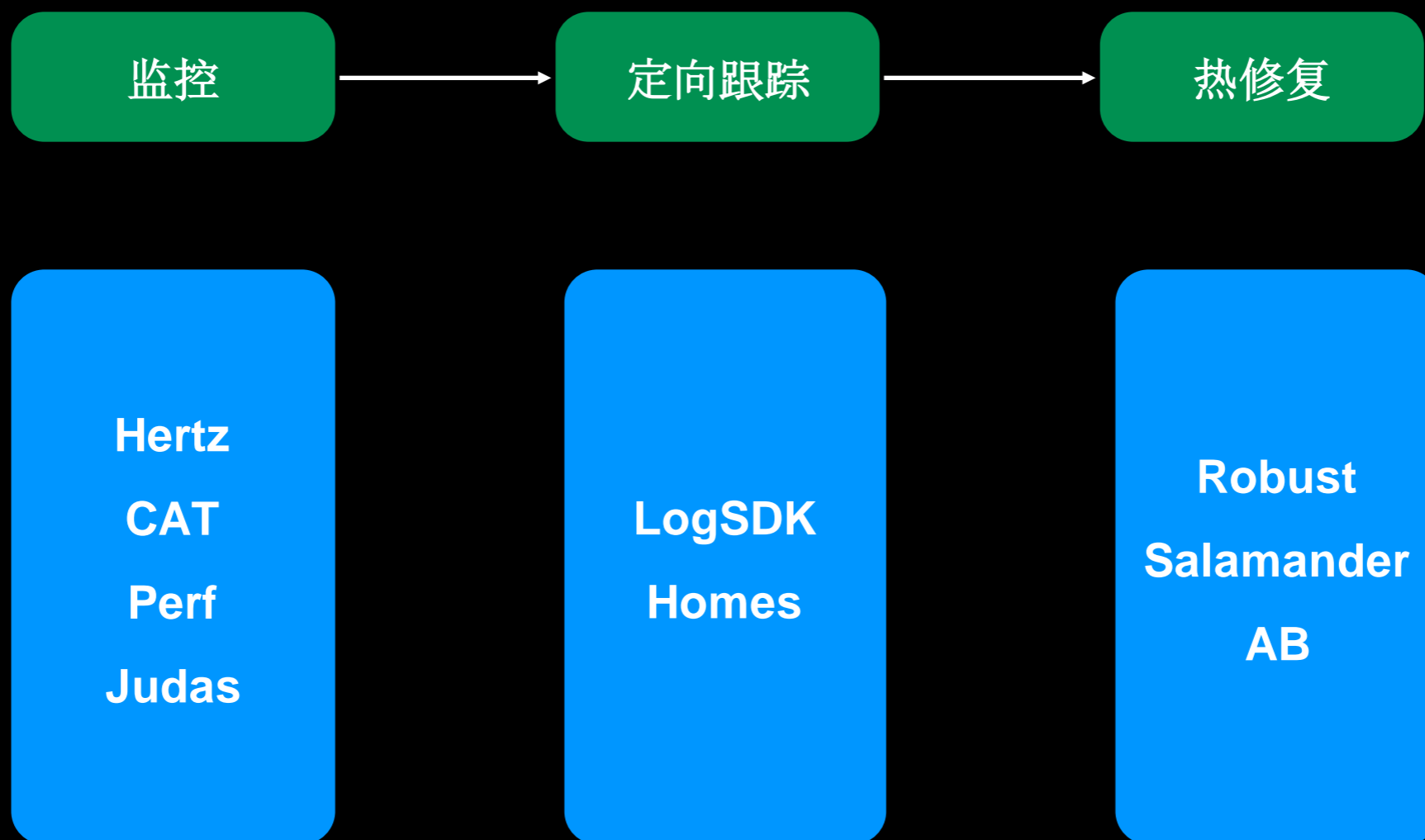
```
17 </app>
18 <app id="11">
19   <type id="base" enable="true"/>
20   <type id="crash" enable="true"/>
21   <type id="codelog" enable="true"/>
22   <type id="anr" enable="true"/>
23   <hertz>
24     <hertz-config type="andriod">
25       <property key="enable" value="false"/>
26       <property key="user-sample-rate" value="0.02"/>
27       <property key="sample-rate" value="0.1"/>
28       <property key="looper-block-threshold" value="2000"/>
29       <property key="looper-block-count" value="20"/>
30     <speed-config>
31       <page id="MainActivity1" tag="1001">
32         <api id="/home/head"/>
33         <api id="/poi/filter"/>
34         <api id="/getbanner"/>
35       </page>
36       <page id="MainActivity2" tag="1001">
37         <api id="/home/head"/>
38         <api id="/poi/filter"/>
39         <api id="/getbanner"/>
40       </page>
41     </speed-config>
42   </hertz-config>
43   <hertz-config type="ios">
44     <property key="enable" value="true"/>
45     <property key="user-sample-rate" value="0.1"/>
46     <property key="sample-rate" value="0.02"/>
```

提交

线上统计还能做什么

- 异常设备分析与筛选
- 与App诊断结合
- 定向跟踪
- 可视化，可视化，可视化

美团点评移动端线上运维能力



总结

- 采集什么样的数据
 - 指标、异常、有用的数据
- 怎么采集
 - 定义模型、反复实践
- 怎么使用数据
 - 设定场景、指标化、可视化



欢迎关注美团点评技术团队

Q&A

Thanks