DevOpsDays Beijing

DevOpsDays

2017 DevOpsDays Beijing

时间：2017.03.18 地点：北京朝阳区万达索菲特大酒店

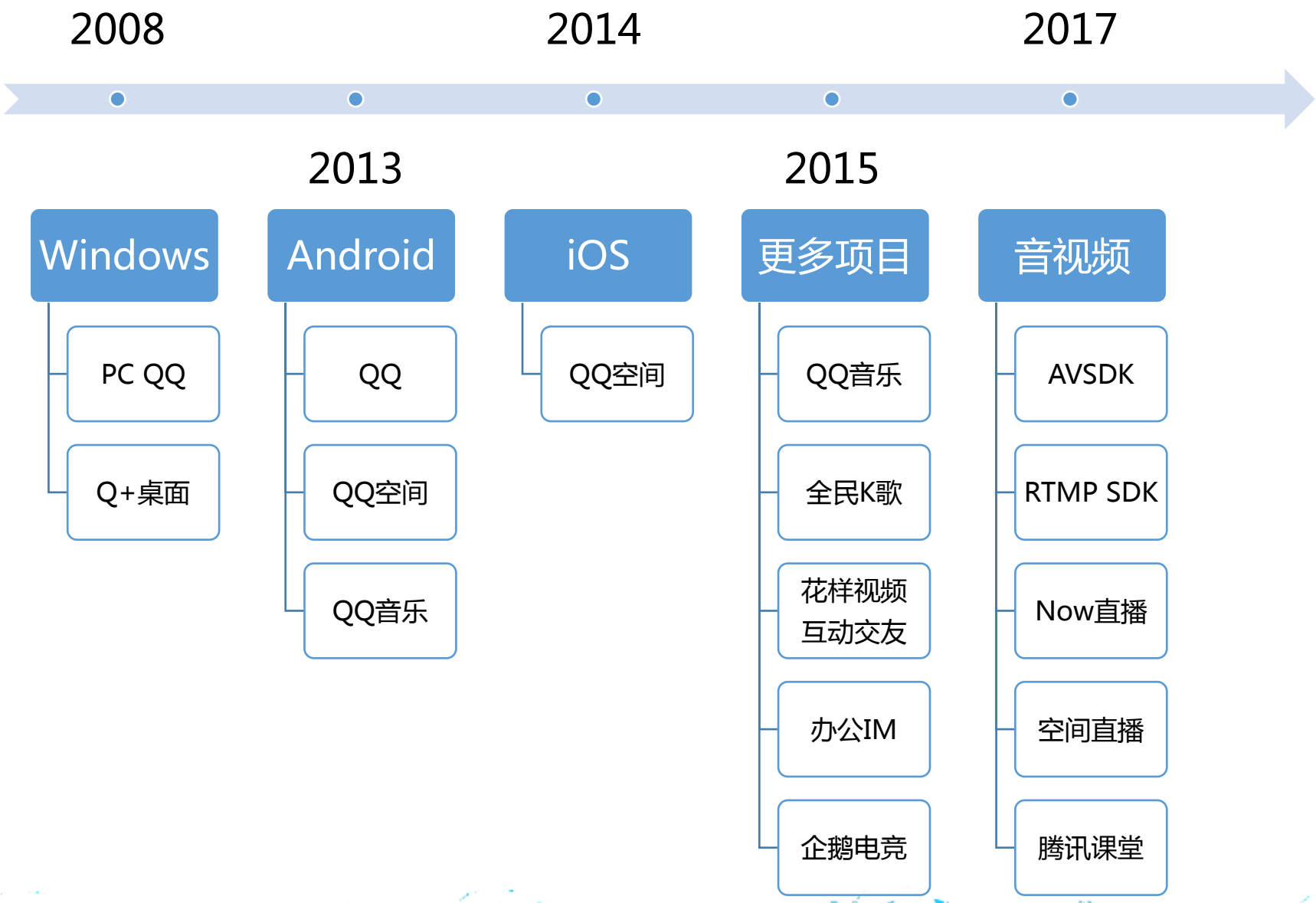
主办单位：高效运维社区 GreatOPS Community　Best Practice 最佳实践

腾讯的专项测试之道

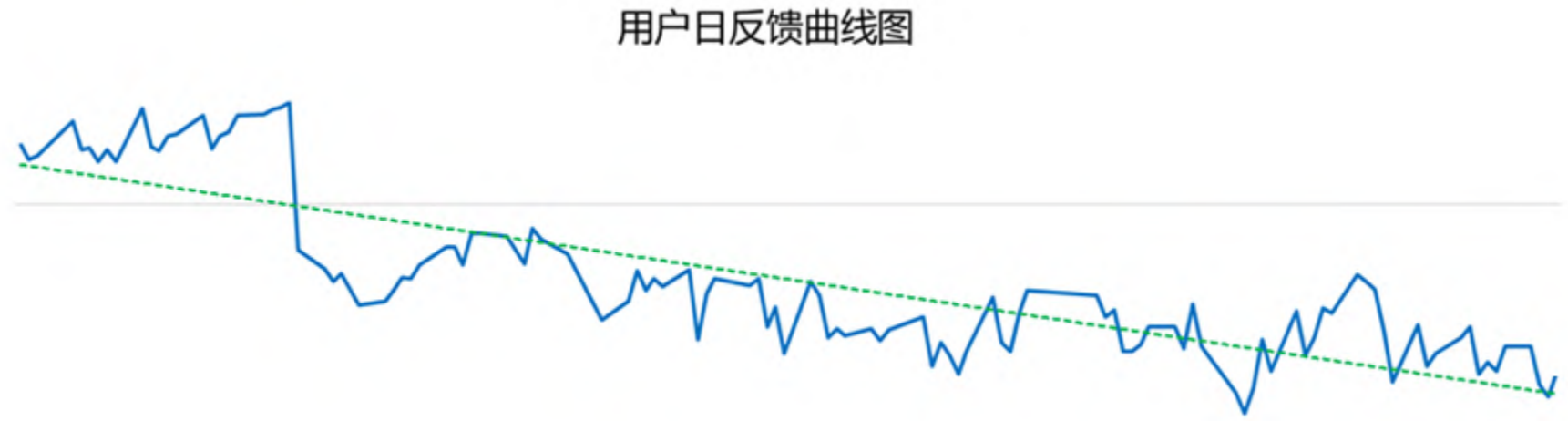
benjaminli（李昶博）
腾讯公司—SNG社交网络质量部

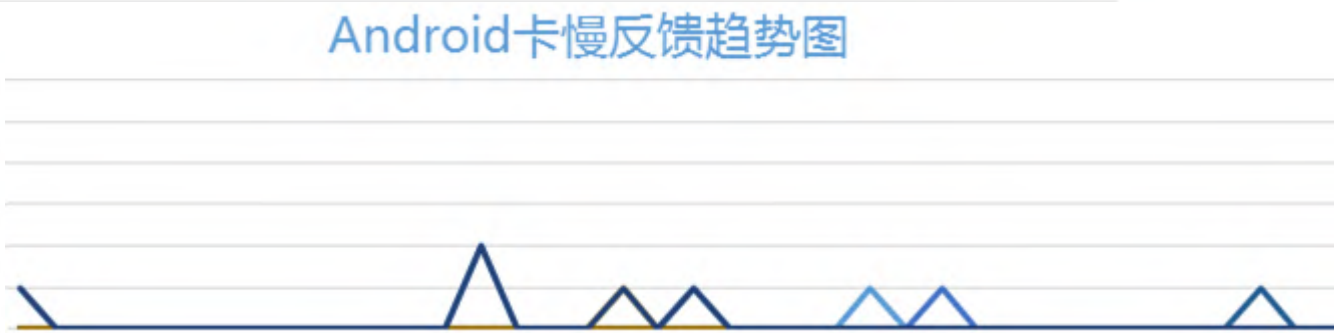
目录

我有哪些项目经验？



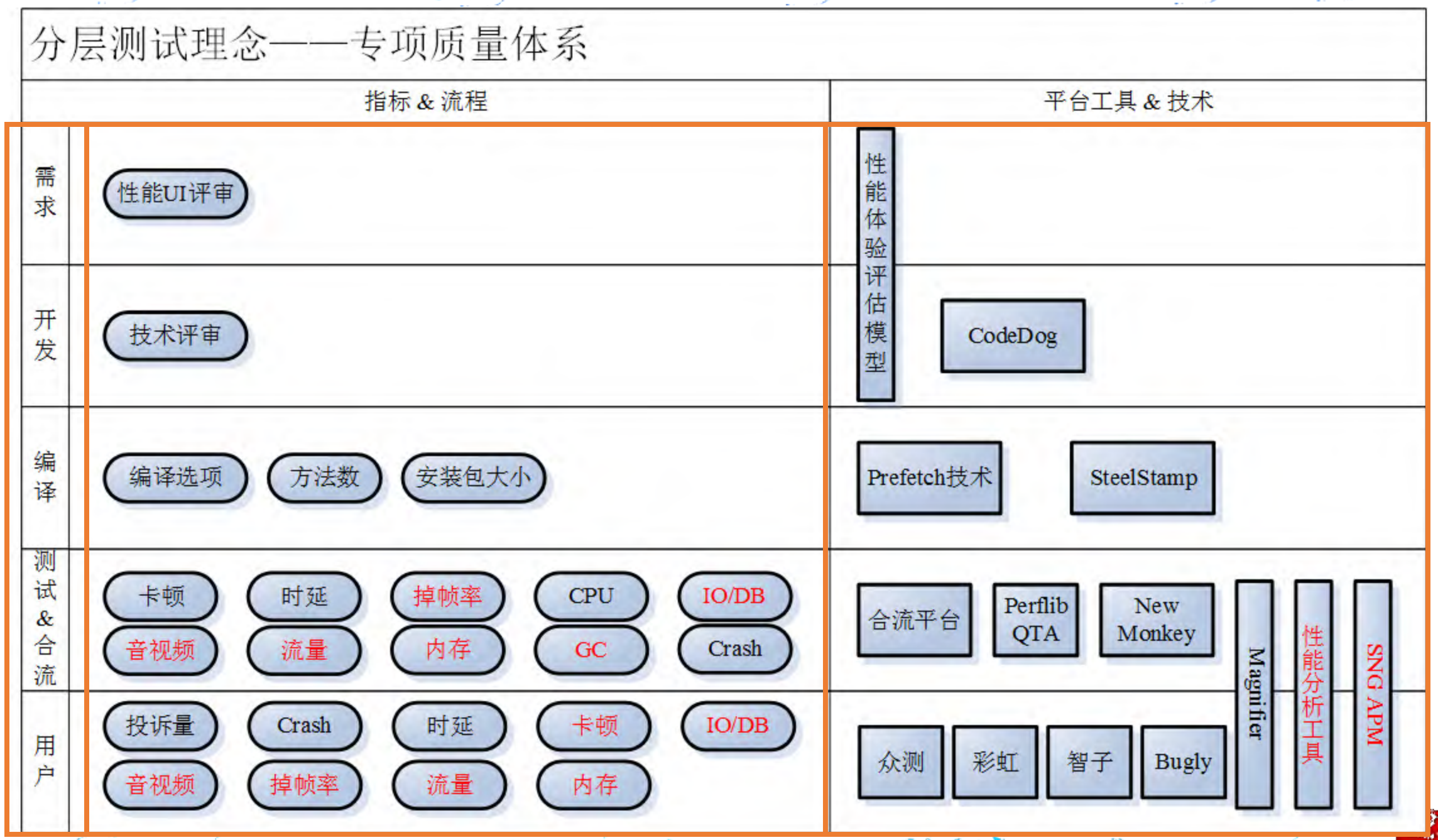
我给腾讯公司带来的价值





目录

分层测试理念——专项质量体系



腾讯专项技术测试员工能力模型

	探索性测试	自动化	分析定位	OS源码	提解决方案
T3	√	√	√	√	√
T2	√	√	√	√/×	√/×
T1	√	√	√	×	×
外包	√	√/×	√/×	×	×
实习	√	√/×	×	√/×	×

纵轴：级别更高

横轴：难度更高

速度体验评测模型

交互模式	举例	用户预期	成效
用户操作立即回馈		< 400ms	1.指导测试用例的编写 2. 性能测试执行可以交给外包 3.无标准时代，每个性能Bug的重要性依赖leader判断。现在，开发不再PK，聚焦于性能优化。
忙模式		< 4秒	
进度条		最多等待4秒，看到进度条前进	
实时动画	游戏，视频等	掉帧率0%	
卡时间	窗口展示后不卡	< 50ms	
竞品法则	任何情况下，始终要快于竞争对手		

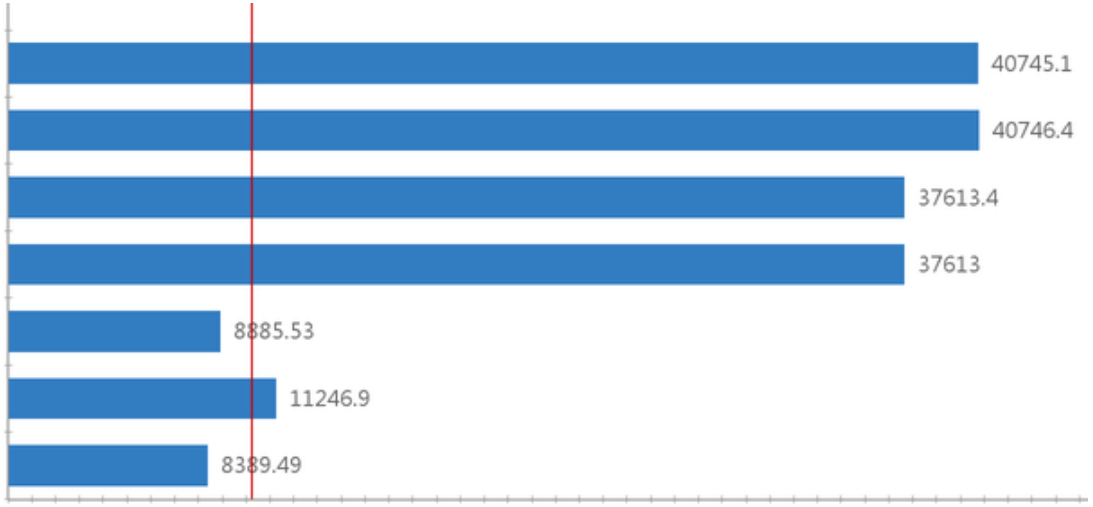
技术评审模型

分类	细项	产品需求PK	技术方案PK	性能测试点
C/S	处理大批量数据的协议	不需要	增量拉取/大数据通道	窄带环境下测试速度
	两个或以上的C/S协议来回	不需要	考虑合并为一个来回	高丢包环境下测试速度
	下载/上传数据量 > 100K	不需要	不需要	I/O测试、流量测试
	下载/上传数据量 > 500K	进度条	不需要	下载速度测试
代码结构	独立的vcproj工程	不需要	尽量使用静态LIB	有共享价值才可用DLL
	常驻内存的数据段（static数据、一旦申请就不再释放的逻辑）	不需要	按需申请、按场景释放	验证内存增量/释放
	Static类/全局变量	不需要	非必要情况不要使用	验证启动程序时/DLL加载时，不加载此模块的代码
B/S	是否有内嵌网页	不需要	跨进程WEB展现	页面测速/内存泄漏测试
	是否跳转到浏览器	不需要	不需要	页面测速
磁盘I/O	为界面层提供数据查询/存储接口（封装DB、ini、注册表等）	不需要	接口查询效率	I/O测试
	多次磁盘写操作（移动、复制、删除等）	不需要	不需要	I/O测试
	数据遍历（SQL、复合文档、文件系统）	需求合理？	考虑无遍历的替代方案	I/O测试
界面效果	有多个同类界面元素（举例：多个桌面图标）	不需要	异步界面展示/按需加载设计	异步展示 按需加载 I/O测试
	动画效果	不需要	Timer实现/阻塞式动画	测试FPS
关键路径	逻辑是否和关键路径挂钩：启动时、登录面板、登录时/登录后、退出	不需要	按需加载设计	按需加载 关键路径性能测试

目录

研发支撑平台：CI/合流/发布

安装包质量监控：体积、方法数



部门	安装包增量配额（KB）	方法数增量配额
SNG-增值产品部		
SNG-云平台部		
SNG-应用部-群		
SNG-应用部-非群		
SNG-数字音乐部		
SNG-社交网络运营部		
SNG-社交平台部		
SNG-开放平台部		
SNG-即通产品部-基础		
SNG-即通产品部-PC		
MIG-浏览器产品部		
IEG-内容与版权业务部		

需要处理的问题

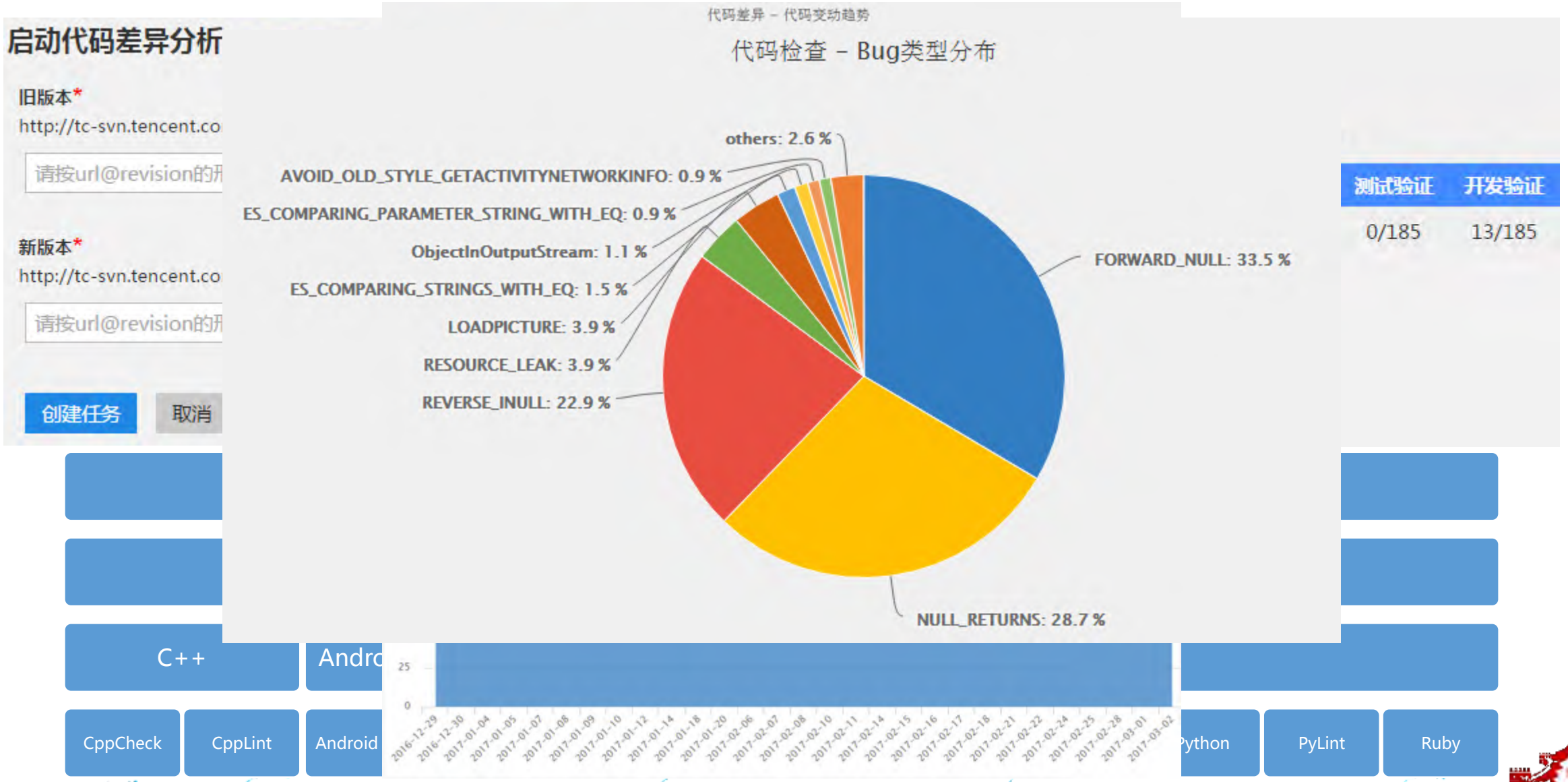
- [QTA] exe的CSC&Log检查结果
- [安装包大小]安装包大小60.345MB -

查看全部问题

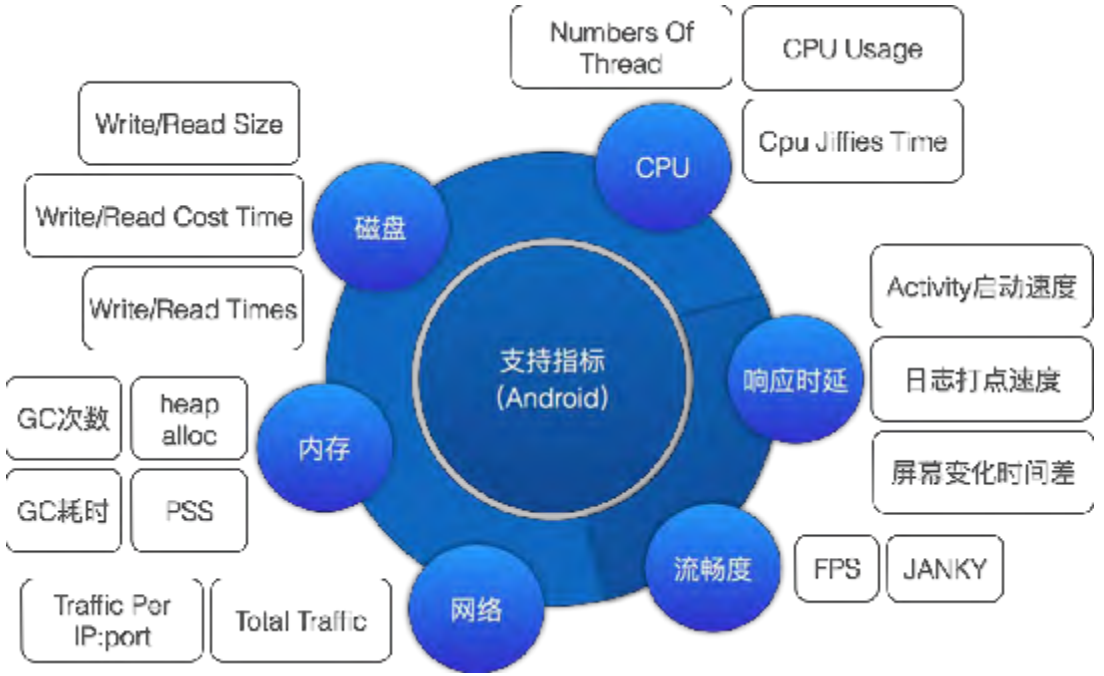
运行的检查项

检查项目	检查结果	负责人	详情
安装包大小检查	Fail		Detail
安装包属性检查	Pass		Detail
文件列表检查(解压)	Pass		Detail
文件列表检查(安装)	Pass		Detail
资源大小变化检查	Fail		Detail
版本信息规范	Pass		Detail
PDB匹配检查(解压)	Pass		Detail
PDB匹配检查(安装)	Pass		Detail
数字签名检查(解压)	Pass		Detail
数字签名检查(安装)	Pass		Detail
签名一致性校验	Pass		Detail
Pubno校验	Pass		Detail
是否发布版本	Pass		Detail
license检查	Pass		Detail
插件信息检查	Pass		Detail
prf文件检查	Pass		Detail
protobuf检查	Pass		Detail
Q盾驱动版本号检查	Pass		Detail
Q盾签名检查	Pass		Detail
QTA辅助检查	Fail		Detail
文件版本号递增检查	Pass		Detail

CodeDog：静态代码扫描



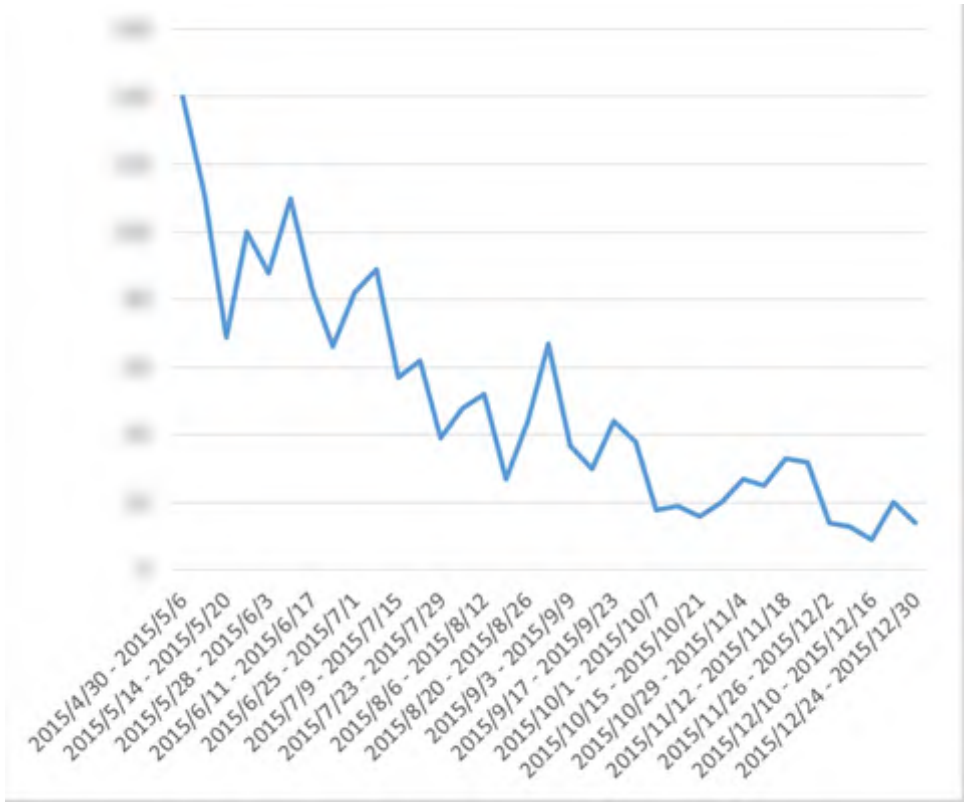
Perflib & QTA：性能自动化测试



New Monkey：稳定性测试

序号	项目名称	部门
1	手Q	即通产品部
2	空间独立版	社交平台部
3	空间结合版	社交平台部
4	QQ Music	数字音乐部
5	全民K歌	数字音乐部
6	应用宝	MIG
7	腾讯新闻	OMG
8	天天快报	OMG
9	腾讯视频	OMG
10	企鹅办公	即通产品部
11	腾讯课堂	即通应用部
12	天天P图	社交平台部
13	Now直播	即通应用部
14	企鹅FM	社交平台部
15	腾讯视频OEM	OMG
16	腾讯视频少儿版	OMG
17	花样直播	即通应用部
18	企鹅辅导	即通应用部
19	微云	云平台部
20	企鹅电竞	增值产品部
21	闪咖	社交平台部
22	腾讯直播	OMG
23	音视频开放SDK	音视频实验室

Magnifier：卡顿监控



目录

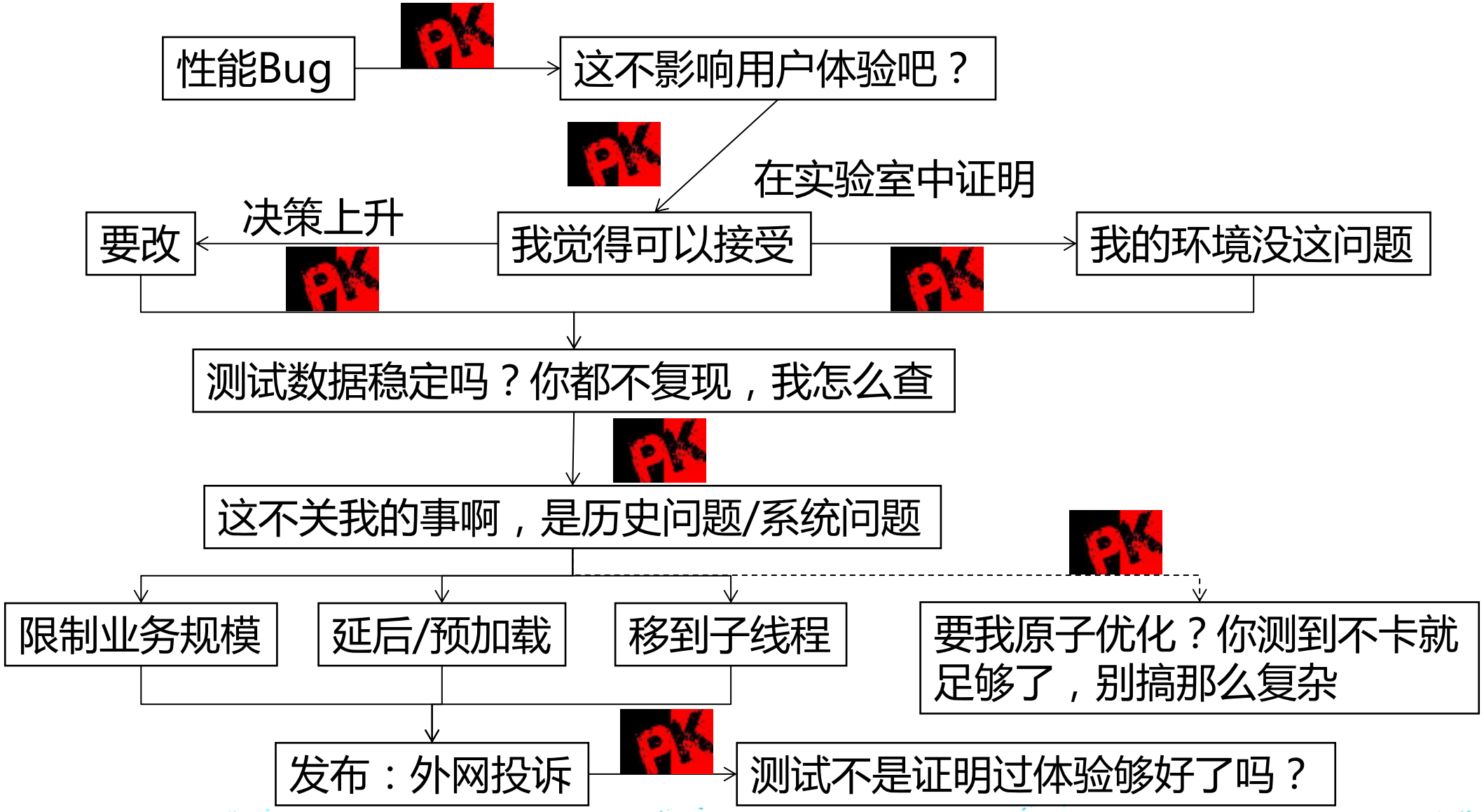
专项质量体系

Windows	卡顿	时延	CPU	句柄	页错误	IO	线程	内存	稳定性
采集	perflib.win								全网Crash上报
分析	Magnifier + Windows Performance Analyzer							WinDbg	
外网	卡上报	全网性能上报 哈勃							

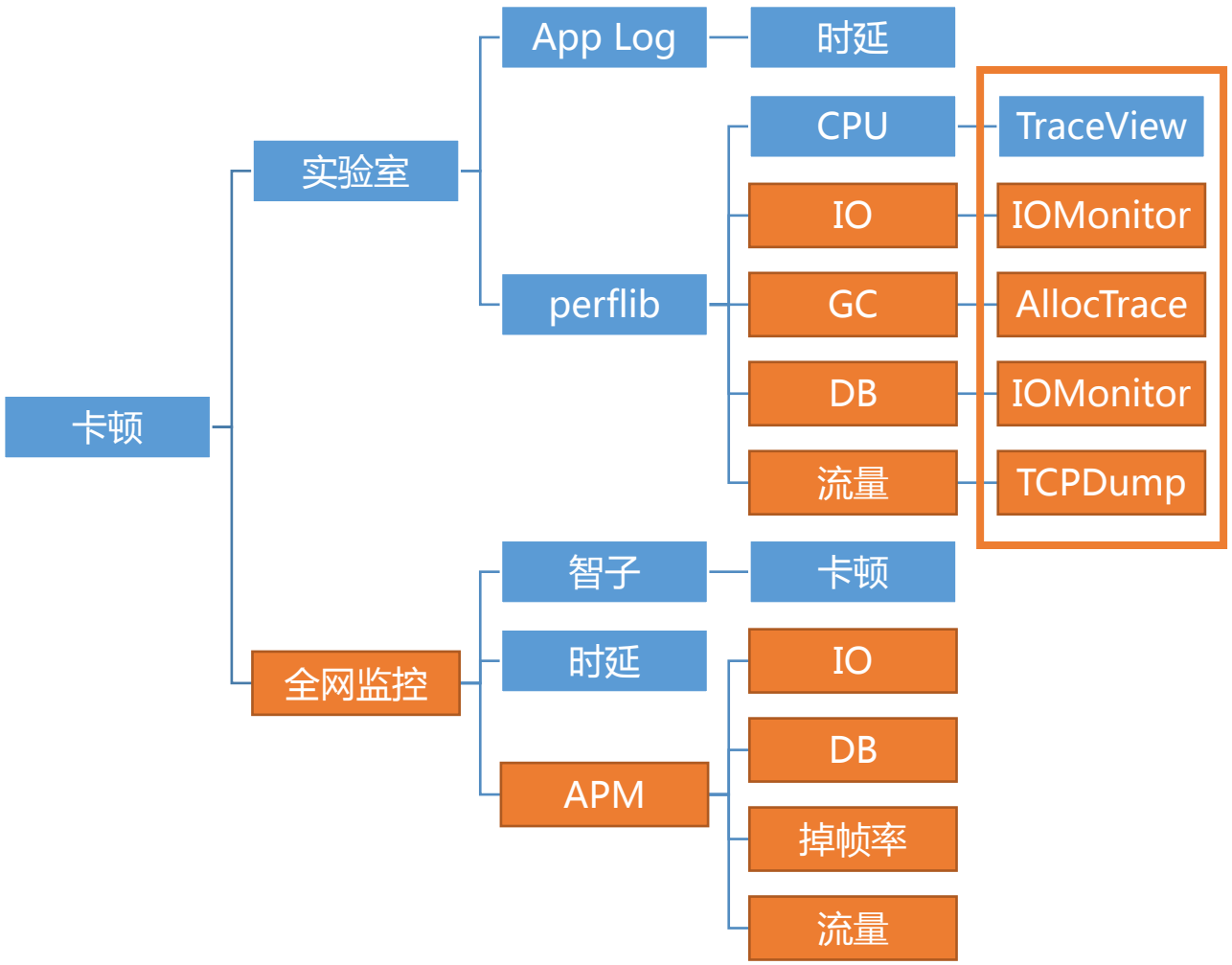
Android	卡顿	电量	稳定性	时延	线程	流量	流畅度	CPU	内存	IO	DB	GC	音视频
采集	×	电流仪	New Monkey	perflib.android									×
分析	×	×		Log	Log	Wire Shark	Trace View		分析云			Alloc Trace	SNG APM
外网	智子	×	RQD	哈勃	×	×	SNG APM					×	

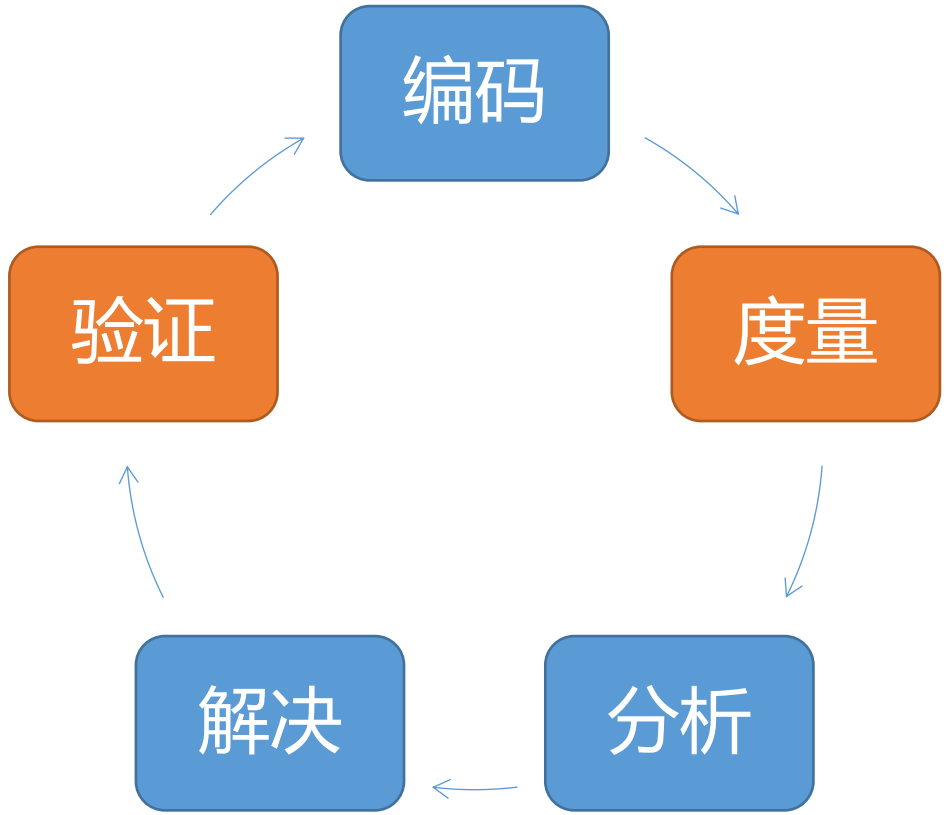
iOS	卡顿	时延	流畅度	CPU	内存	线程	IO	流量	电量	稳定性
采集	×	perflib.ios						TCPDump	体温枪	New Monkey
分析	×	Log	Apple Instruments					WireShark	×	
外网	智子	哈勃	SNG APM			×		×	×	RQD

框架效应：这样PK，效率多低？



无需重现规律定位随机性能Bug



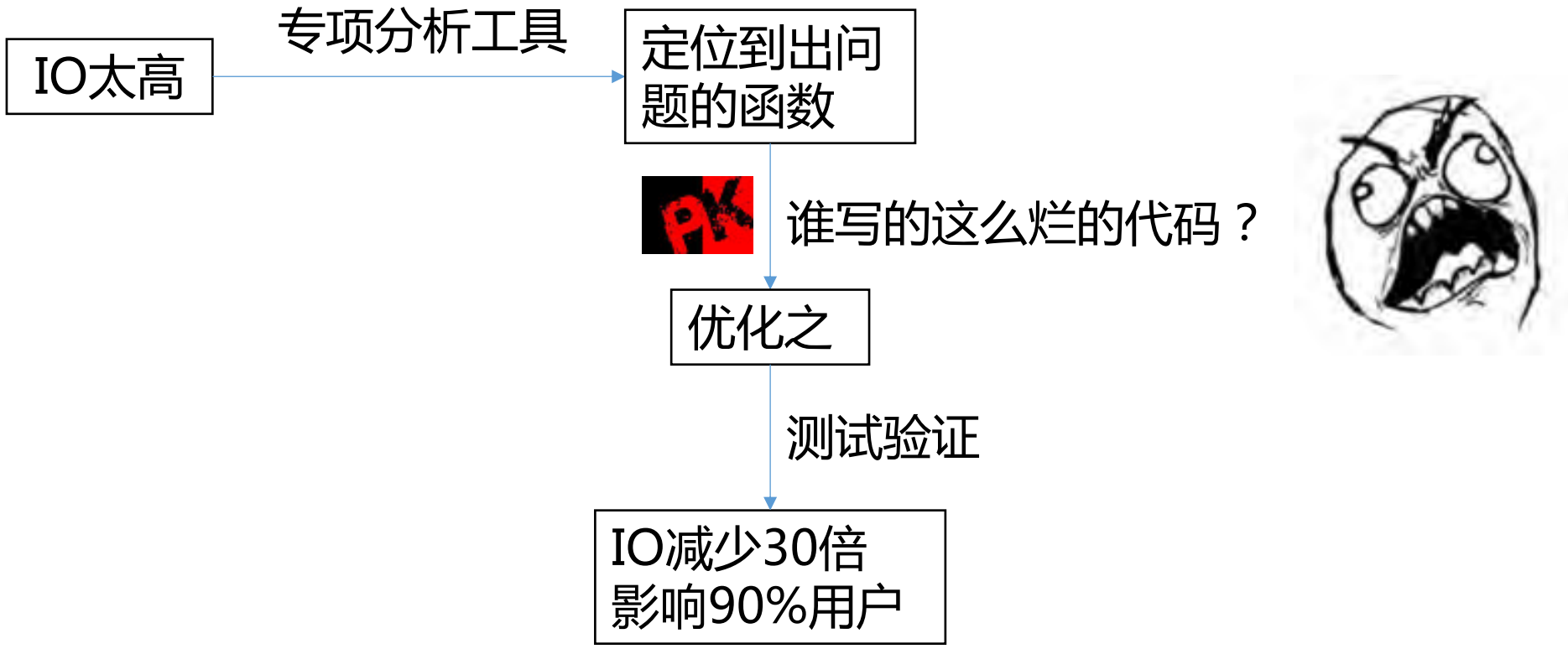


带分析能力的性能自动化

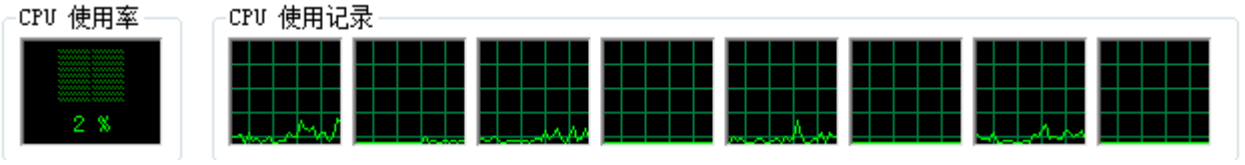
首次启动（时延、内存、CPU、IO）：

指标	被测版本	对比版本	最大值	变化	备注	说明
耗时					增量小于10%	通过
CPU时间(Linux)					增量小于30%	通过
线程数						通过
PSS						通过
流量						通过
网络包						通过
HeapAlloc						通过
文件IO次数						通过
文件IO大小						通过
Activity泄露个数					当前值小于1	通过
GC次数					当前版本小于对比版本或波动在10%以内	通过
GC阻塞时间						通过
GC总时间						通过
Hprof AllocSize					相对参照值偏差小于500K	通过

结果：不吵架了，直接解Bug



CPU测试




```
top | grep User
User 0%, System 0%, IOW 2%, IRQ 0%
User 105 + Nice 11 + Sys 49 + Idle 1601632 + IOW 45048 + IRQ 0 + SIRQ 10 = 1646855
User 0%, System 0%, IOW 0%, IRQ 0%
User 90 + Nice 9 + Sys 88 + Idle -981328 + IOW -25780 + IRQ 0 + SIRQ 13 = -1006908
```

CPU变频特性

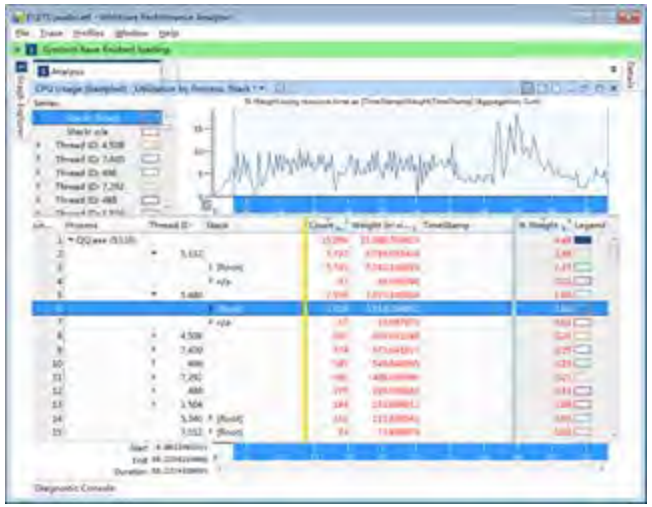
其他APP串扰

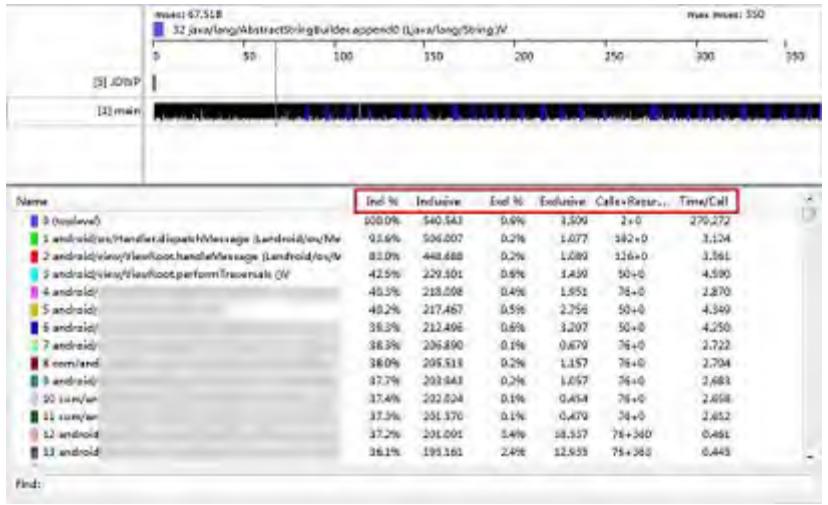
如何判定性能问题

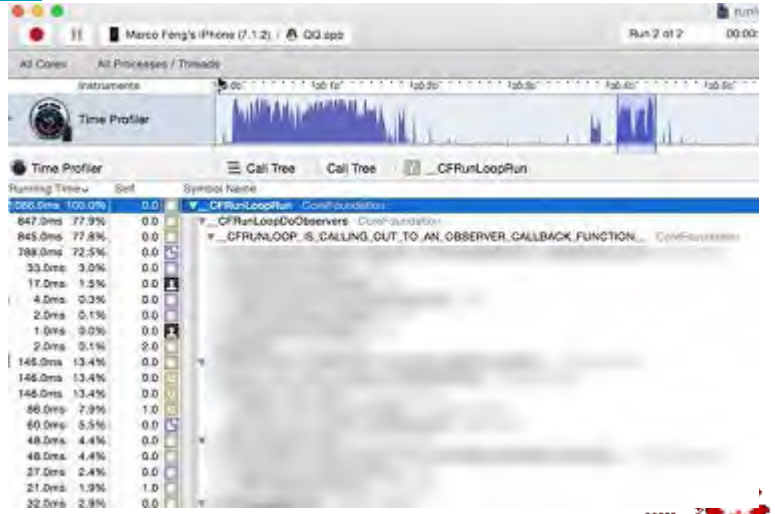
Android 4.x有Bug

指标	被测版本	对比版本	最大值	变化	备注	说明
CPU时间(Linux)	2207.0 jiffies	2773.5 jiffies	2268.0 jiffies	↓ 566.5 jiffies(20.0%)	增量小于30%	通过

指标	被测版本	对比版本	最大值	变化	备注	说明
CPU时间	1977.1 ms	1934.6 ms	2064.7 ms	↑ 42.5 ms(2.0%)	增量< 500ms	通过







IO测试

效率提升60倍

Bug：启动慢至12秒 → 连续3周没有分析出原因 → IO分析 → 2小时内找到某组件最大

filepath	process	thread	writecount	writebytes	writetime	stacktrace
			1403	11185056	10020	java.io.FileOutput
		Normal_HandlerThread	481	5586108	8296	java.io.FileOutput
		Normal_HandlerThread	510	4752744	3067	java.io.FileOutput
		Normal_HandlerThread	1161	4752744	417	java.io.FileOutput
		Normal_HandlerThread	1354	2018138	721	java.io.FileOutput
		Normal_HandlerThread	205	1910900	165	java.io.FileOutput
		Normal_HandlerThread	188	1538921	111	java.io.FileOutput
			142	1131884	380	java.io.FileOutput

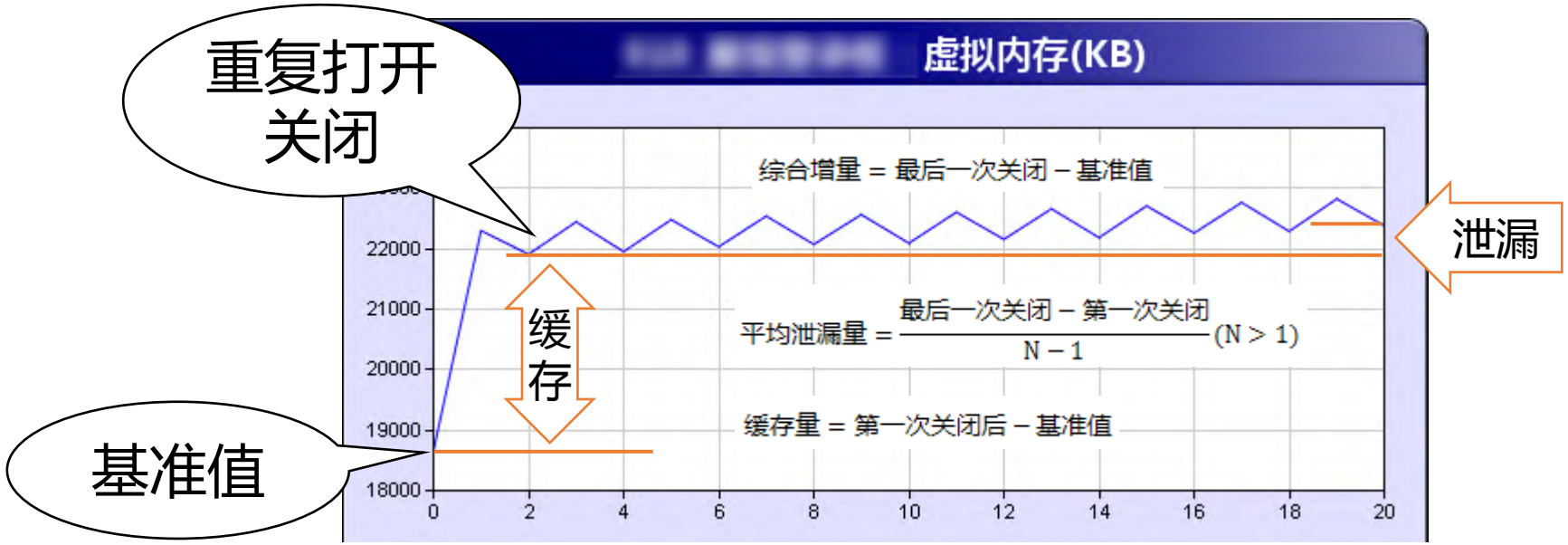
指标	被测版本	对比版本	变化
文件IO次数	936.5 次	10389.5 次	性能提升30倍
文件IO大小	1384.5 KB	41894.8 KB	性能提升30倍
主线程进行IO的文件数	0.5 个	8.0 个	几乎清扫到零
重复IO的文件数	0.0 个	24.5 个	彻底清扫到零

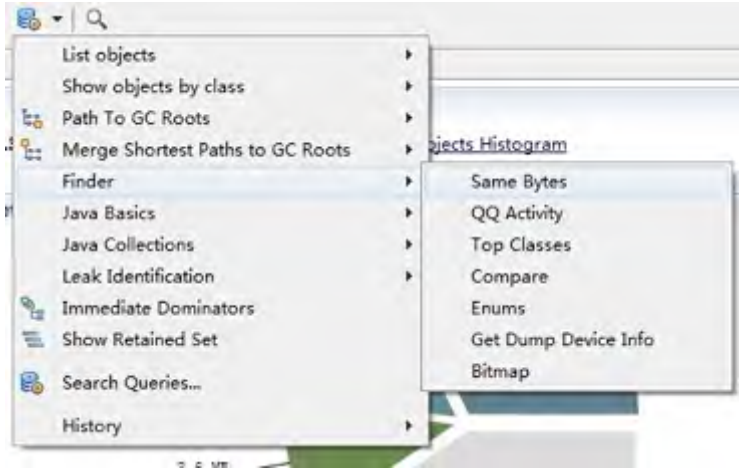
指标	被测版本	对比版本	变化
文件IO次数	6790.0 次	10658.5 次	新引入TBS的情况下，性能提升0.6倍
文件IO大小	28881.6 KB	42760.9 KB	新引入TBS的情况下，性能提升0.5倍
主线程进行IO的文件数	4.5 个	8.5 个	解决了一半的问题
重复IO的文件数	0.0 个	36.0 个	彻底清扫到零
IO Buffer太小的文件数	0.0 个	4.5 个	彻底清扫到零

指标	被测版本	对比版本	变化
文件IO次数	811.5 次	17148.5 次	性能提升30倍
文件IO大小	2694.7 KB	17269.2 KB	性能提升5.4倍
主线程进行IO的文件数	2.0 个	6.5 个	几乎清扫到零
重复IO的文件数	1.0 个	11.0 个	几乎清扫到零
IO Buffer太小的文件数	0.0 个	5.0 个	彻底清扫到零

指标	被测版本	对比版本	变化
文件IO次数	122.5 次	316.5 次	性能提升1.6倍
文件IO大小	212.2 KB	1505.4 KB	性能提升6倍
主线程进行IO的文件数	0.0 个	—	没有问题
重复IO的文件数	0.0 个	4.0 个	彻底清扫到零
IO Buffer太小的文件数	0.0 个	2.0 个	彻底清扫到零

内存测试



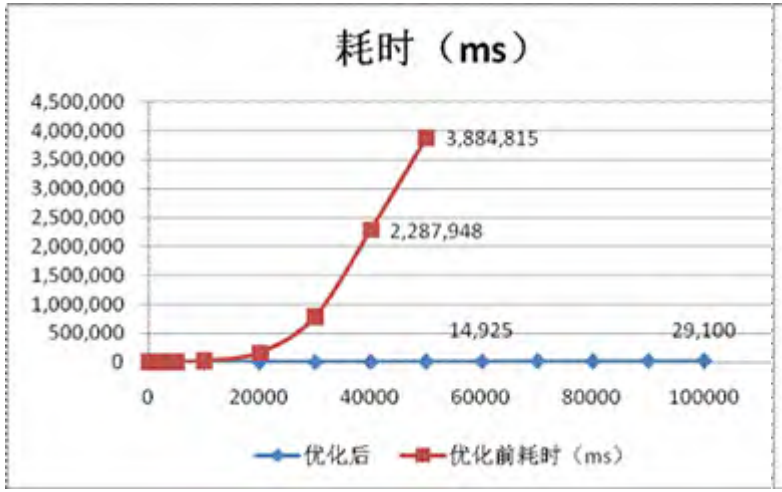


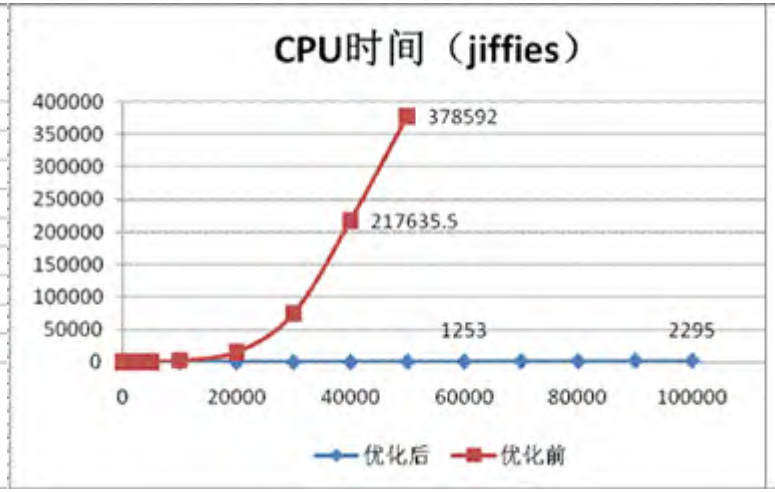

```
self.snapshot() #取资源值
self.memdump.snapshot() #取内存Dump
for _ in range(10):
    mp.Controls['加号'].click()
    plusNum = PlusNumPanel(mzapp)
    self.sleep(5)
    plusNum.ensure_back()
    self.sleep(5)
    self.snapshot() #取资源值
self.logcatmon.wait_for_idle(15)
self.snapshot() #取资源值
self.memdump.snapshot() #取内存Dump
```

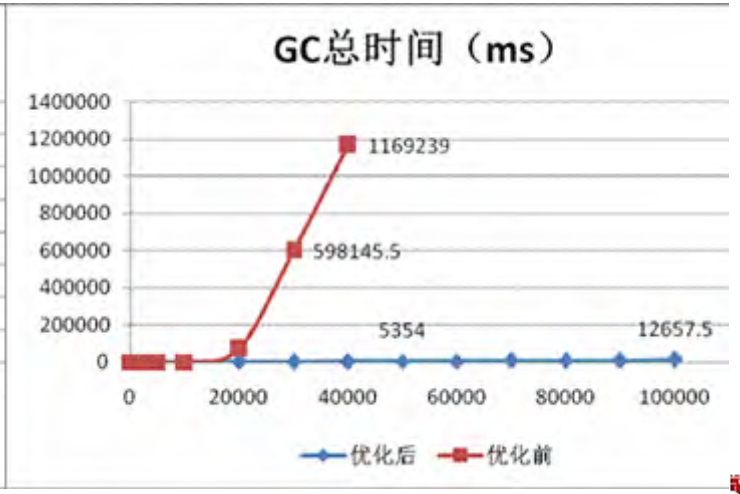
GC测试

GC问题	CPU问题
• 内存复用	• 函数耗时
• 可解！！	• 无解？？

数据规模	优化前(ms)	优化后(ms)	性能提升
10000	27,635	2,853	9倍
20000	164,235	4,380	36倍
30000	789,600	6,945	113倍
40000	2,287,948	9,910	230倍
50000	3,884,815	16,038	241倍
100000	远超1小时，无法测出	29,100	-







GC分析工具原理

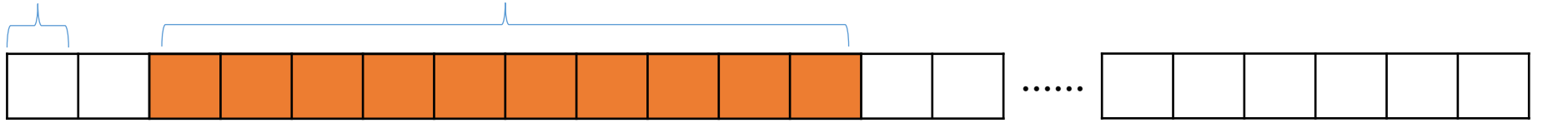
- Google Allocation Tracker最多记录64K
- 自研的AllocTrace能记录无限多信息

ClassName	Size	times	sum	ThreadId	StackTraces
byte[]	215312	3764	810434368	1	at android.graphics.BitmapFactory.nativeDecodeStream(BitmapFactory.java:-2)-> at android.graphics.BitmapFactory.decodeStreamInternal(BitmapFactory.java:627)-> at android.graphics.BitmapFactory.decodeStream(BitmapFactory.java:603)-> at android.view.View.performClick(View.java:4444)-> at android.view.View$PerformClick.run(View.java:18440)->
byte[]	16400	3917	64238800	1	at android.graphics.BitmapFactory.decodeStreamInternal(BitmapFactory.java:626)-> at android.graphics.BitmapFactory.decodeStream(BitmapFactory.java:603)-> at android.view.View.performClick(View.java:4444)-> at android.view.View$PerformClick.run(View.java:18440)->

掉帧率（流畅度）测试

更敏感

16.67ms **掉10帧，167ms**



FPS = 50	掉帧率 = 16.7%

FPS = 50	掉帧率 = 0%

避免平均化

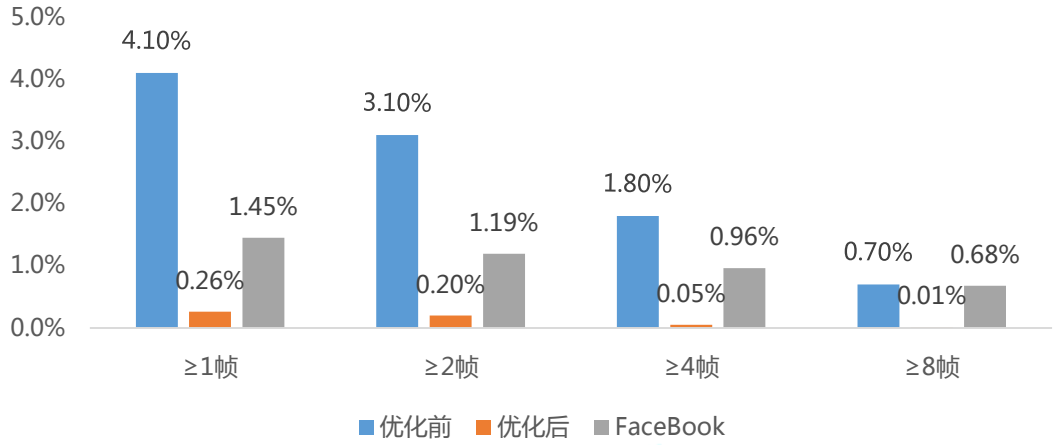
每天运行10轮：

60	50	60	60	60	60	60	60	60	60

→ 59

每天运行1轮：

0%
16.7%



流量测试

首次启动（时延、内存、CPU、IO）：

指标	被测版本	对比版本	最大值	变化	说明
流量	2236.39 KB	2044.94 KB	2254.8 KB	↑ 191.45 KB(9.0%)	通过
网络包	3500.25 个	3272.67 个	3565.0 个	↑ 227.58 个(7.0%)	通过

自己解析 /proc/net /xt_qtaguid/ stats

TrafficStats 含有本地回环的流量

Added in API level 8

TrafficStats

Summary: Constants | Ctors | Methods | Inherited Methods | [Expand All]

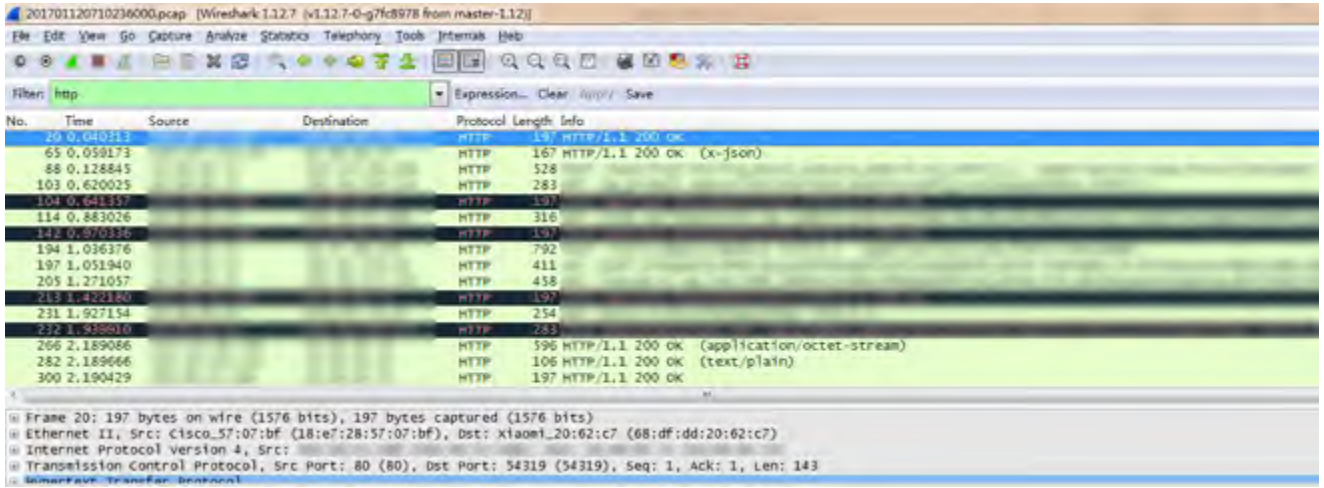
Class that provides network traffic statistics. These statistics include bytes transmitted and received and network packets transmitted and received, over all interfaces, over the mobile interface, and on a per-UID basis.

Name Last modified Size Description

Parent Directory -

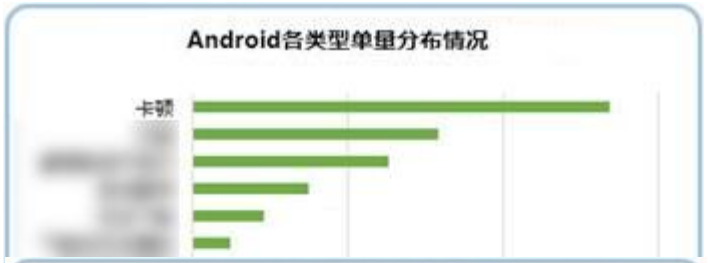
20170112071023600O.pcap 12-Jan-2017 05:50 17M

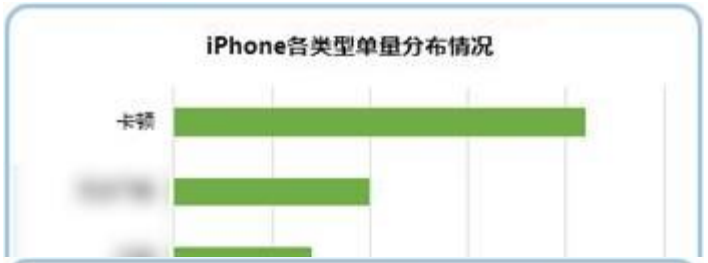
20170112071023600O.txt 12-Jan-2017 05:50 6.3K

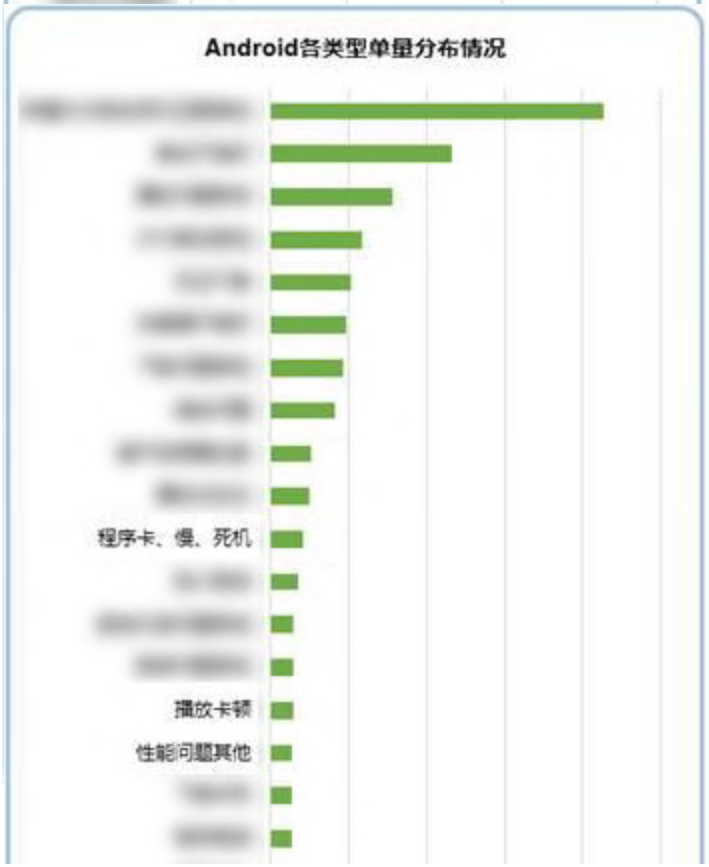


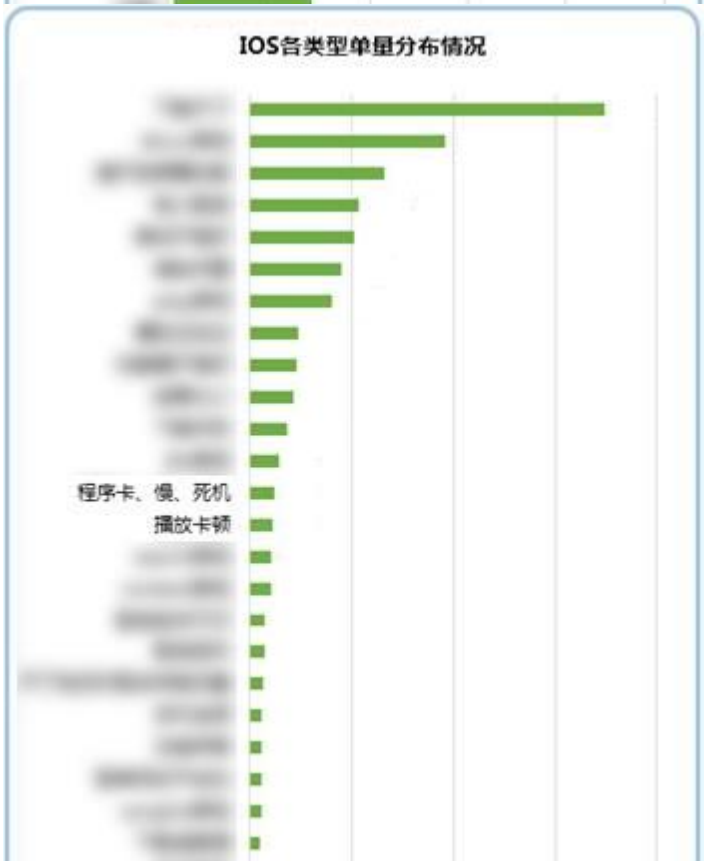
目录

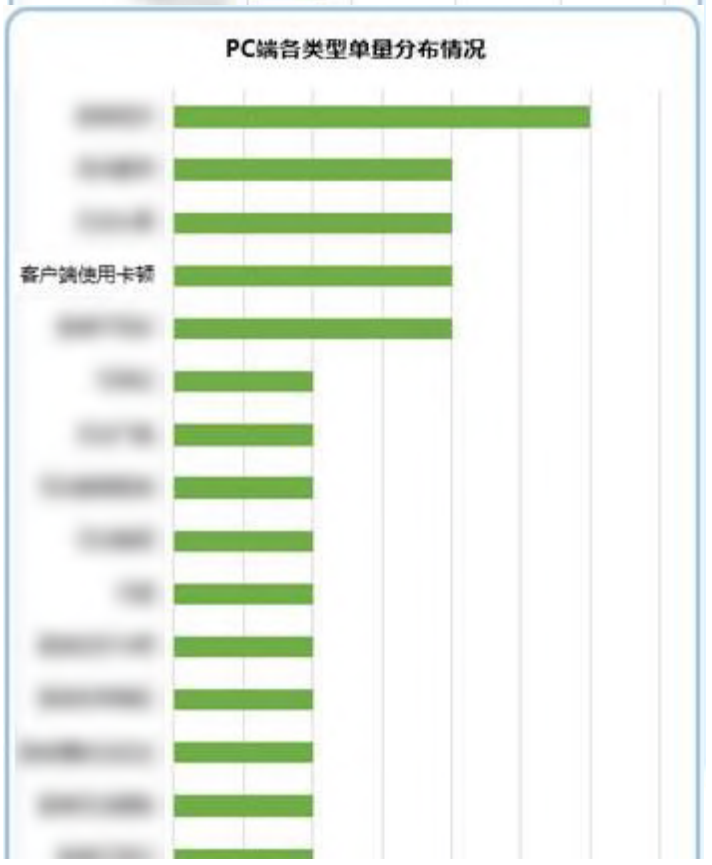
用户投诉跟进



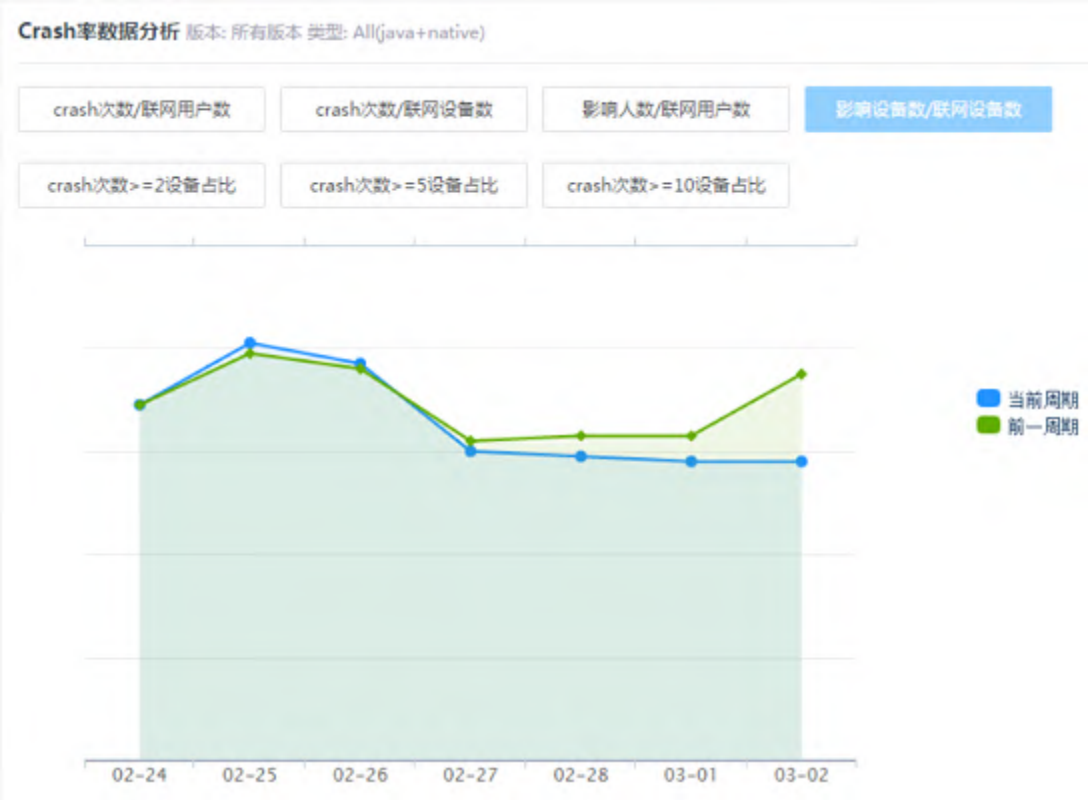








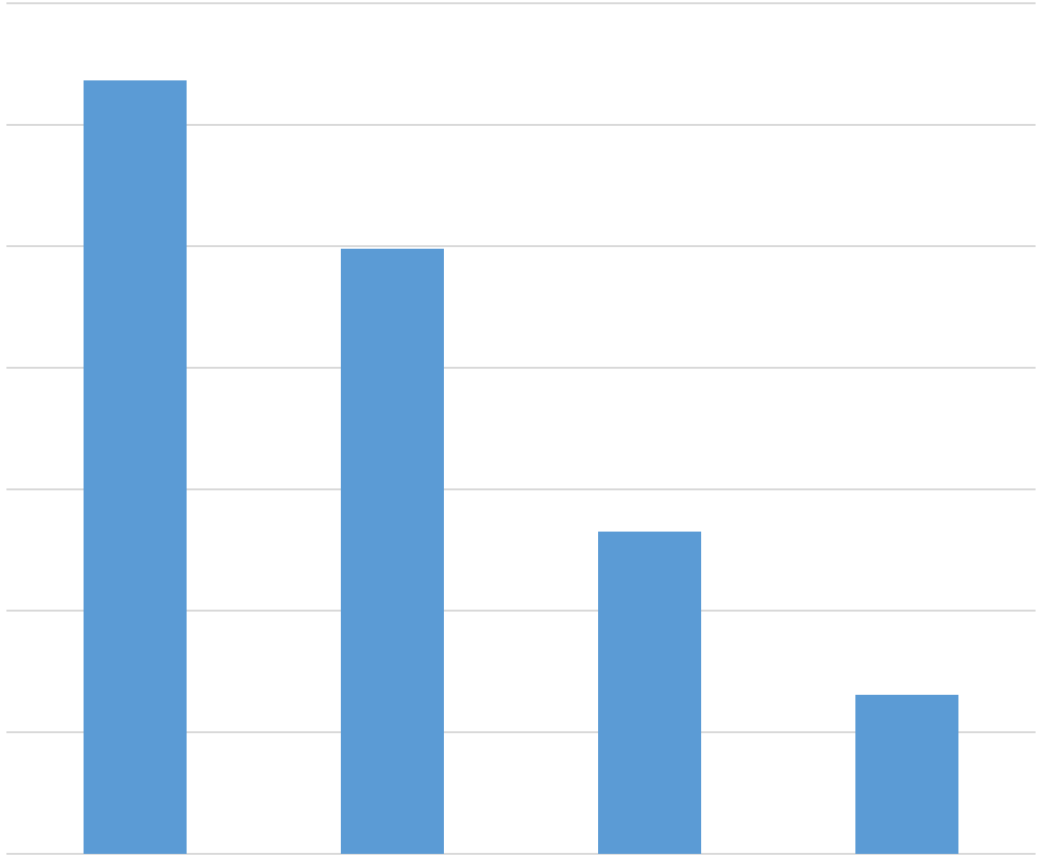
Crash全网上报



指标集	外网质量指标 ↑	精品标准	iPhone	iPhone	iPhone	iPhone	iPhone
1.crash率	crash率						
2.速率类	图片下载成功率2G						
	图片下载成功率3G						
	图片下载成功率4G						
	图片下载成功率WIFI						
	图片上传成功率2G						
	图片上传成功率3G						
	图片上传成功率4G						
	图片上传成功率WIFI						
	视频播放成功率2G						
	视频播放成功率3G						
	视频播放成功率4G						

卡顿全网上报

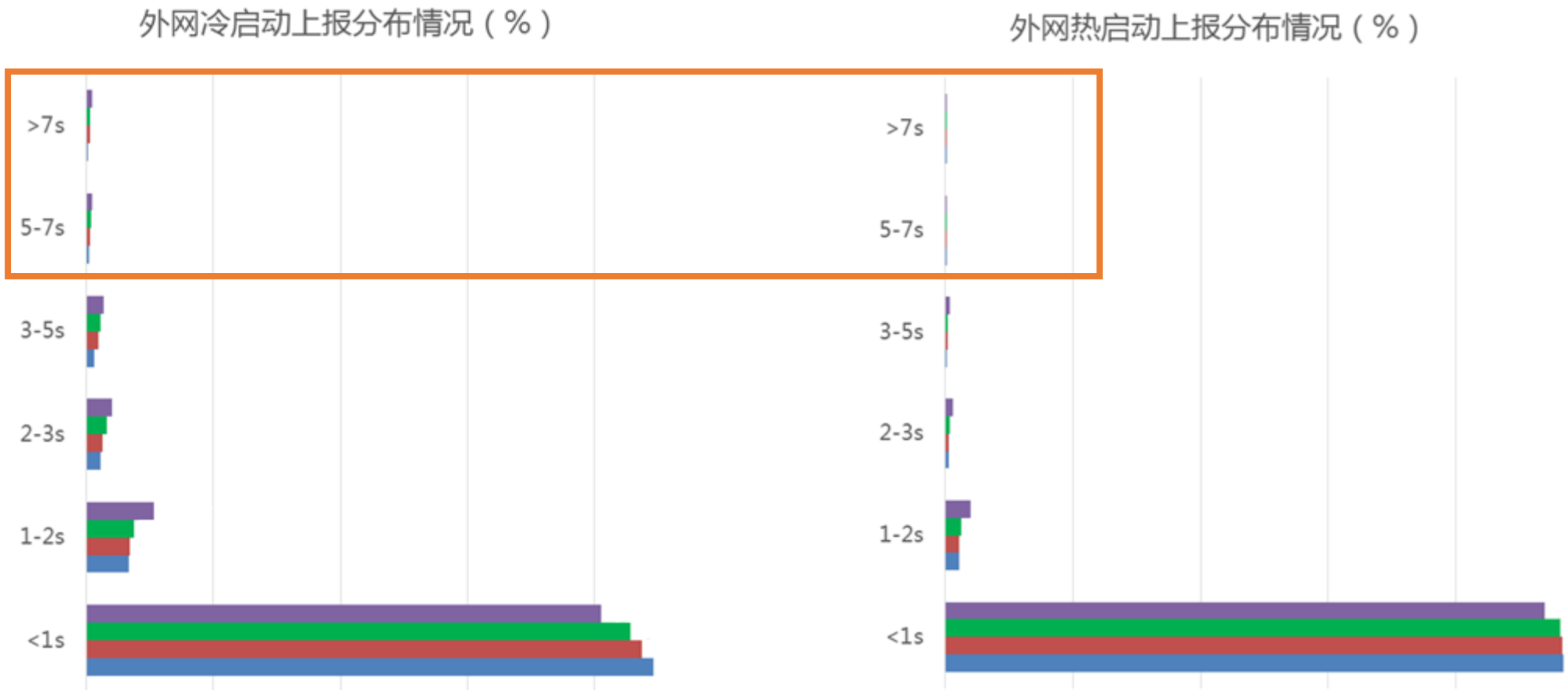
外网用户卡顿率



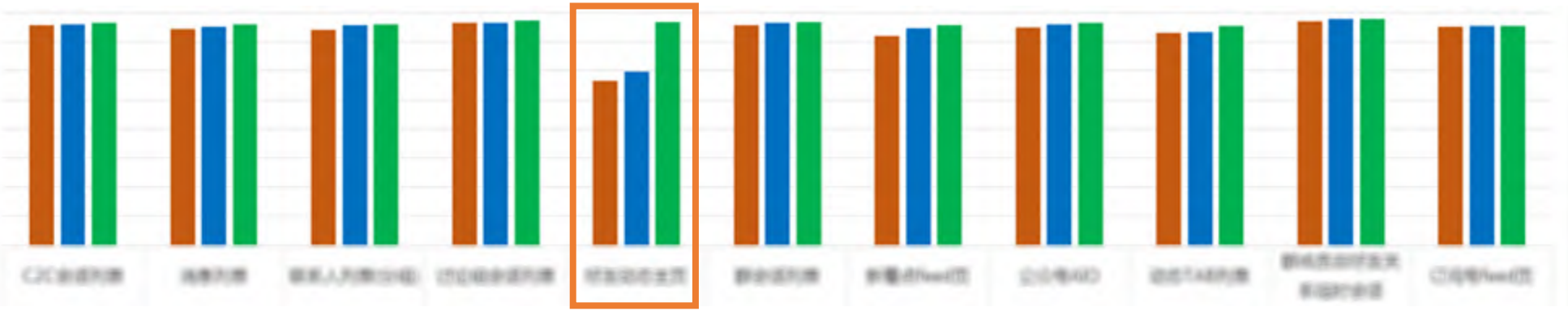
堆栈: Enter text 版本号: Search

版本号	堆栈	更新时间
[redacted]	[redacted]	2015-07-07 17:41:57
[redacted]	[redacted]	2015-07-07 17:43:09
[redacted]	[redacted]	2015-07-07 17:30:54
[redacted]	[redacted]	2015-07-07 17:46:41
[redacted]	[redacted]	2015-07-06 17:51:56
[redacted]	[redacted]	2015-07-07 17:16:05
[redacted]	[redacted]	2015-07-07 17:30:21
[redacted]	[redacted]	2015-07-07 17:37:53
[redacted]	[redacted]	2015-07-07 17:09:59
[redacted]	[redacted]	2015-07-07 17:46:02

时延全网上报



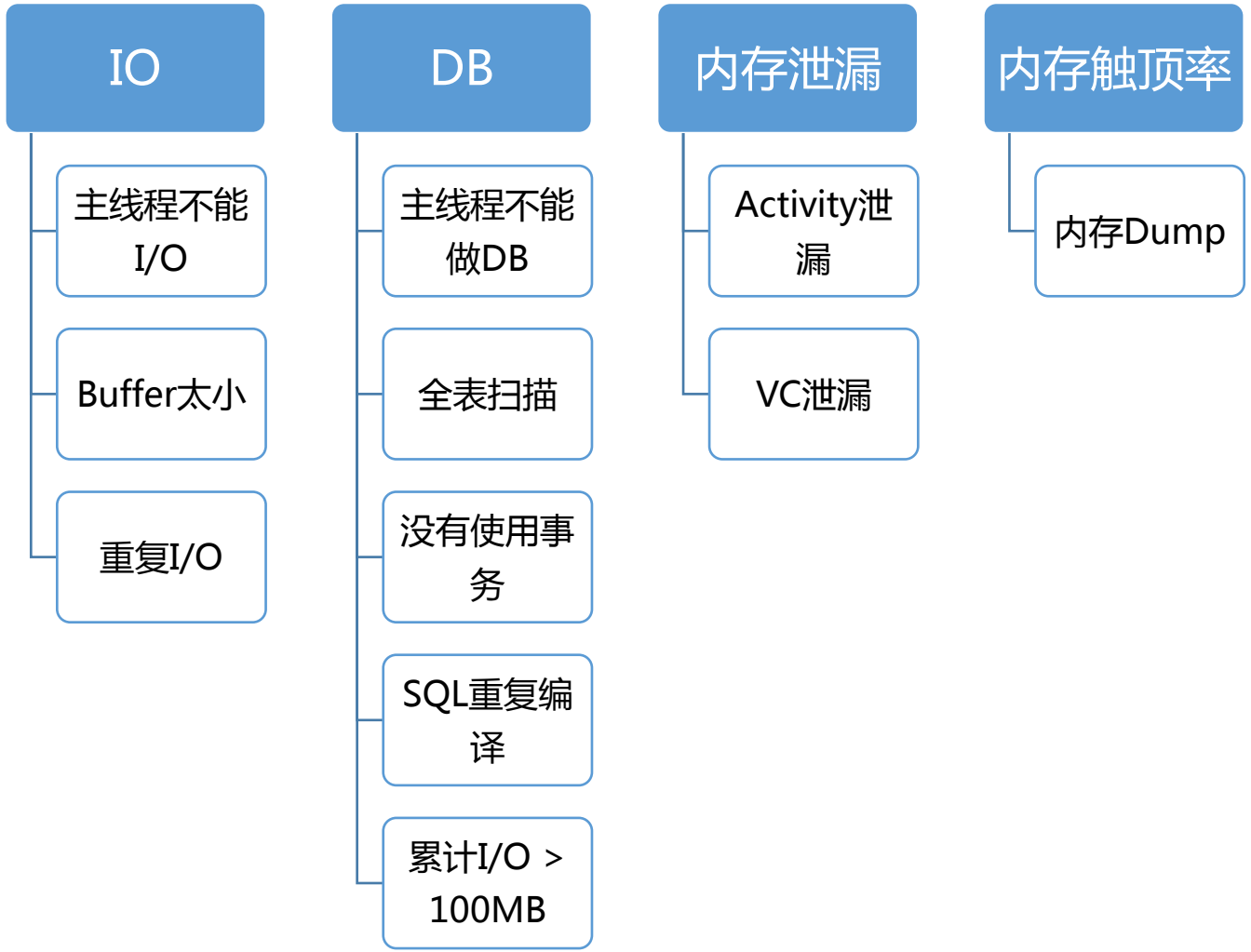
SNG APM：掉帧率



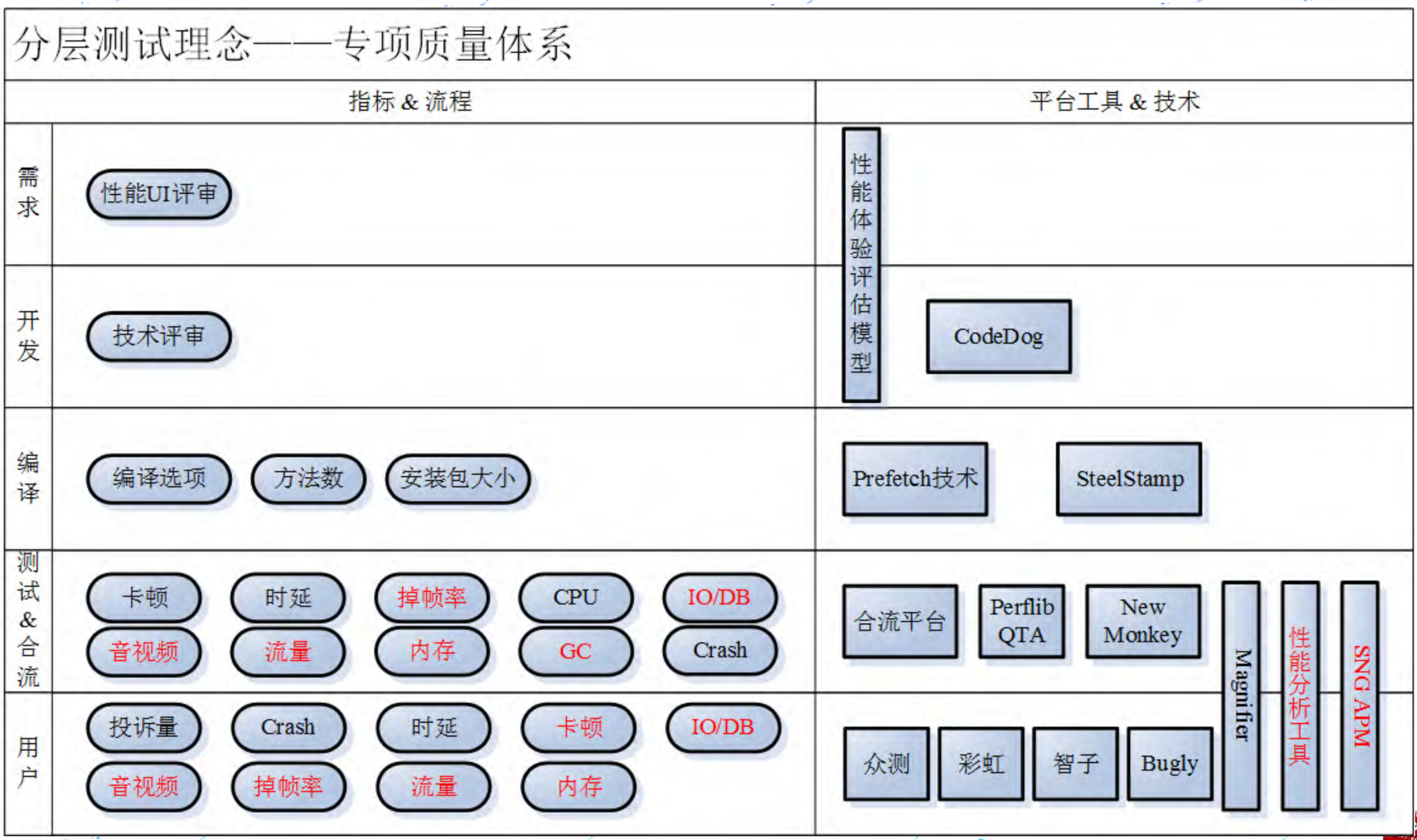
掉帧耗时树枝分析

	节点名	总体耗时	平均耗时	上报次数	总体耗时占比
1	[redacted]	78840.7	305.29	262	49.51%
2	[redacted]	40113.4	297.77	134	50.88%
3	[redacted]	40096.7	297.62	134	99.96%
4	[redacted]	14545.7	196.26	77	18.
5	[redacted]	7181	252.14	29	9

SNG APM：IO、DB、内存



分层测试理念——专项质量体系



高效运维社区

GreatOPS Community

GOPS 4月深圳 / 7月北京 / 11月上海

EXIN DevOps Master 认证研修

DevOpsDays 3月北京 / 8月上海

DevOps 企业内训 / 咨询服务

DevOps China 全国巡回技术沙龙

其他量身定制服务项目

商务经理：刘静女士
电话 / 微信：13021082989
邮箱：liujing@greatops.co

DevOpsDays Beijing

Thanks

荣誉出品

高效运维社区

国际最佳实践管理联盟

DevOpsDays Beijing