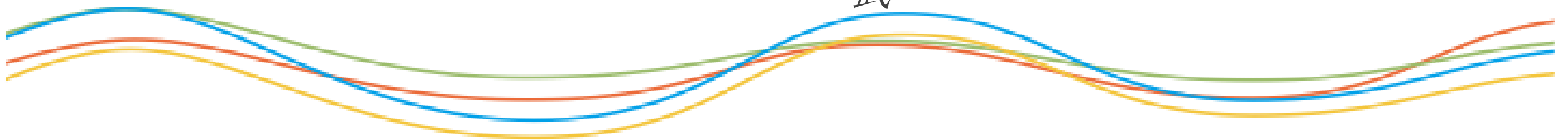


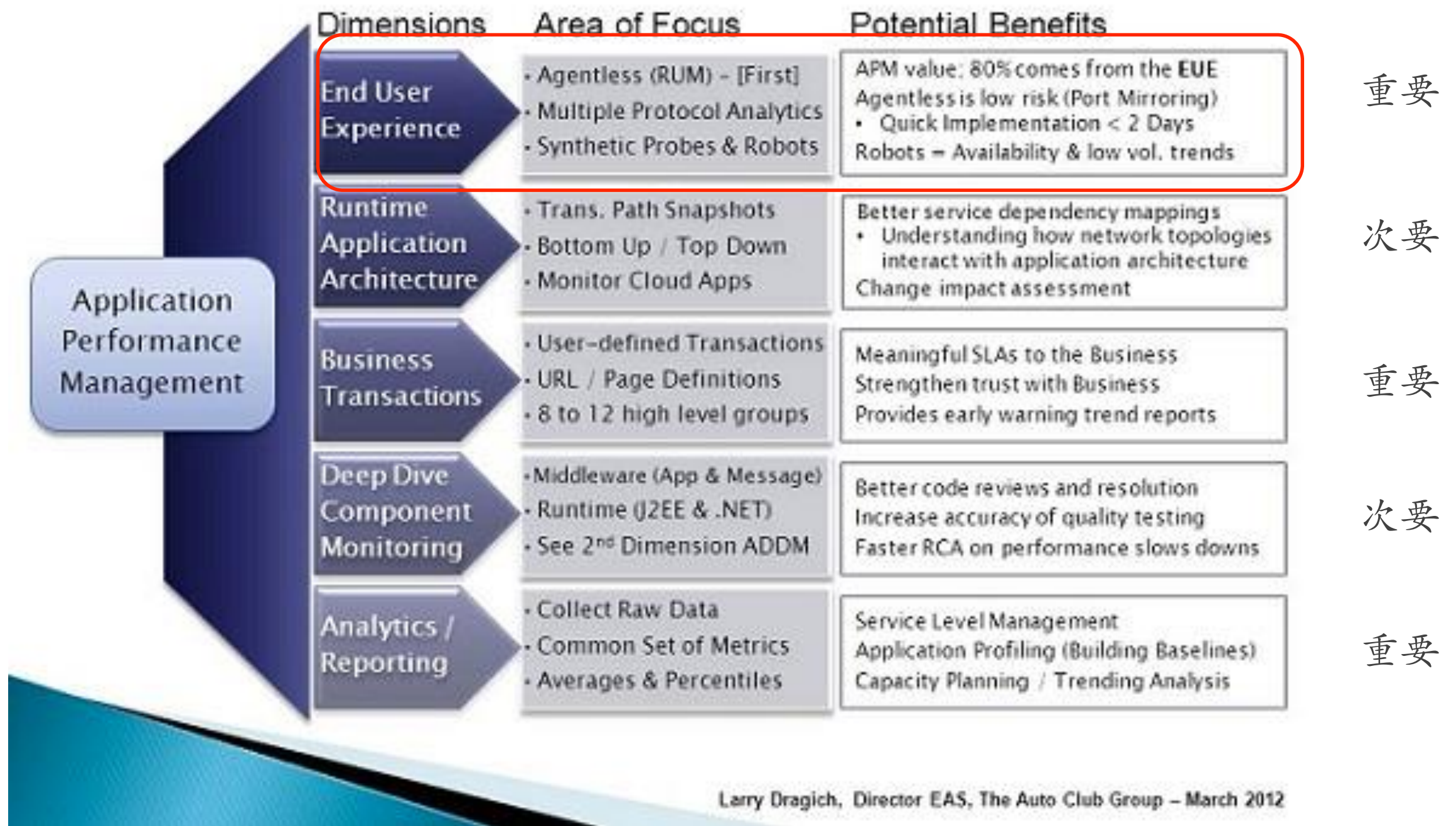
性能可视化实践之路

讲师： 阿里百川高级无线技术专家- 陈武



APM Conceptual Framework

Prioritizing Gartner's APM Model




- ❖ 稳定的运行环境
- ❖ 稳定的网络连接
- ❖ 流畅的操作界面
- ❖ 尽量少的资源消耗
- ❖ 漂亮的视觉交互

<http://www.apmdigest.com/apm-evolution-end-user-experience>

稳定的运行环境

每天有大量用户会遇到多次应用崩溃

	日均发生崩溃的设备比例	单个应用在发生崩溃的设备上 日均崩溃次数
	3.0%	2.3次
	3.5%	1.9次

Crash分析与处理方法

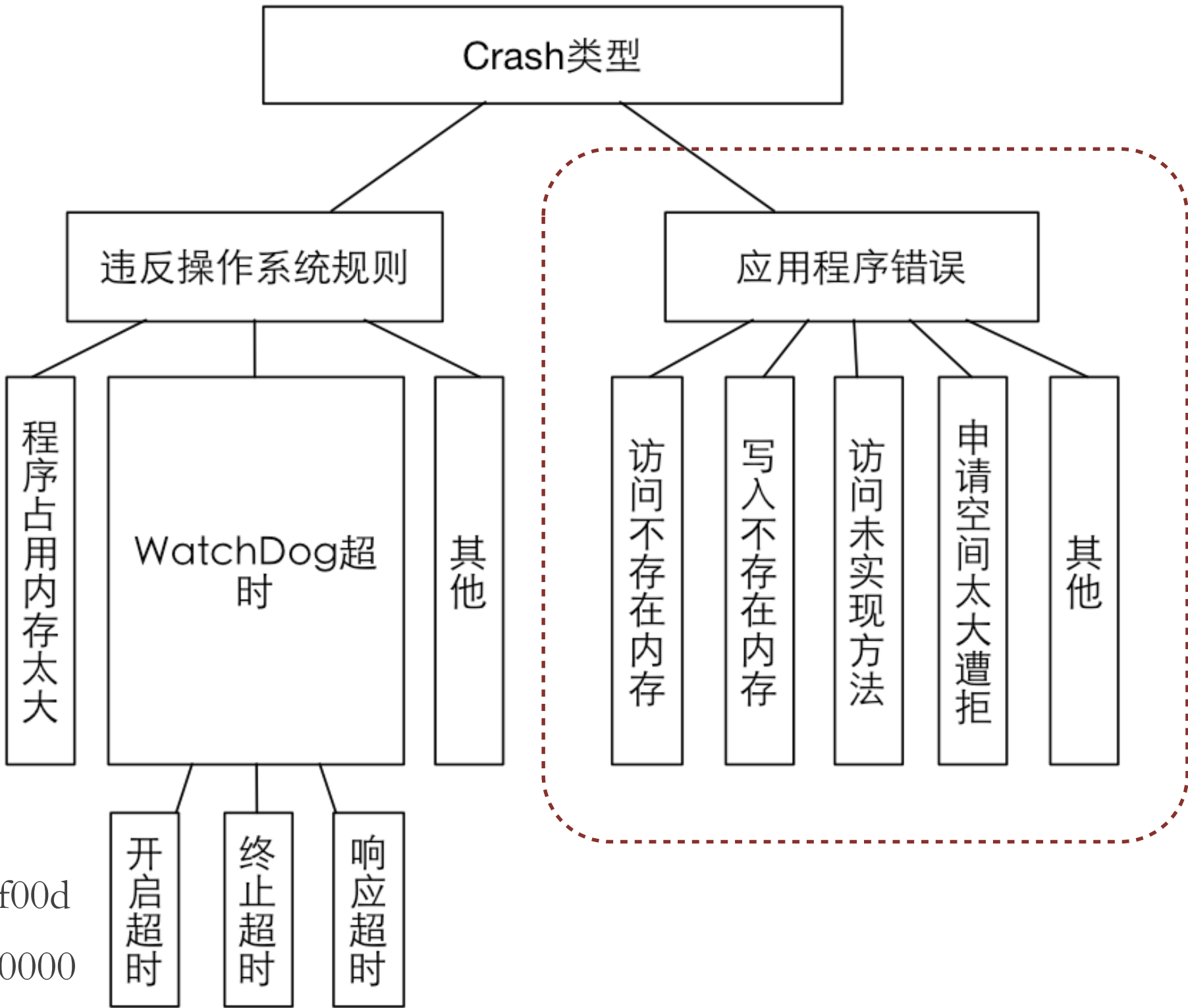
传统方式

- 开发阶段
 - 开发人员Debug调试验证
 - 测试人员测试验证，提供设备奔溃日志给开发人员分析
- 发布阶段
 - 用户反馈，开发人员debug复现
 - 用户反馈，导出crashlog给开发分析
 - 通过iTunes connect服务提供的Crash Report日志

APM服务

- 集成SDK
- 开发&发布阶段
 - ✓ Crash日志云端查询
 - ✓ Crash率异常短信告警

Crash 类型分类

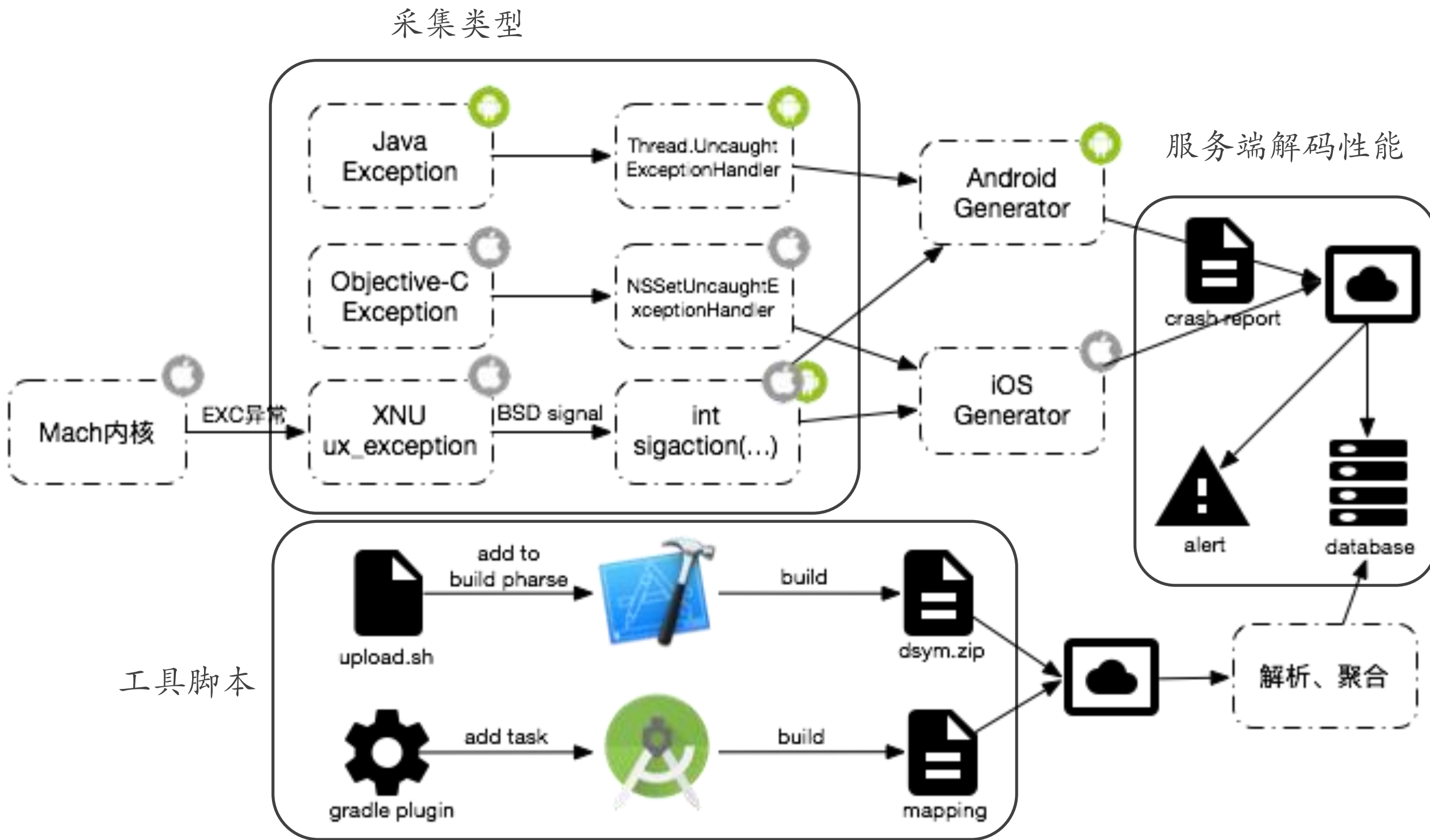


0x000000008badf00d

0x0000000000000000

...

Crash日志流程图



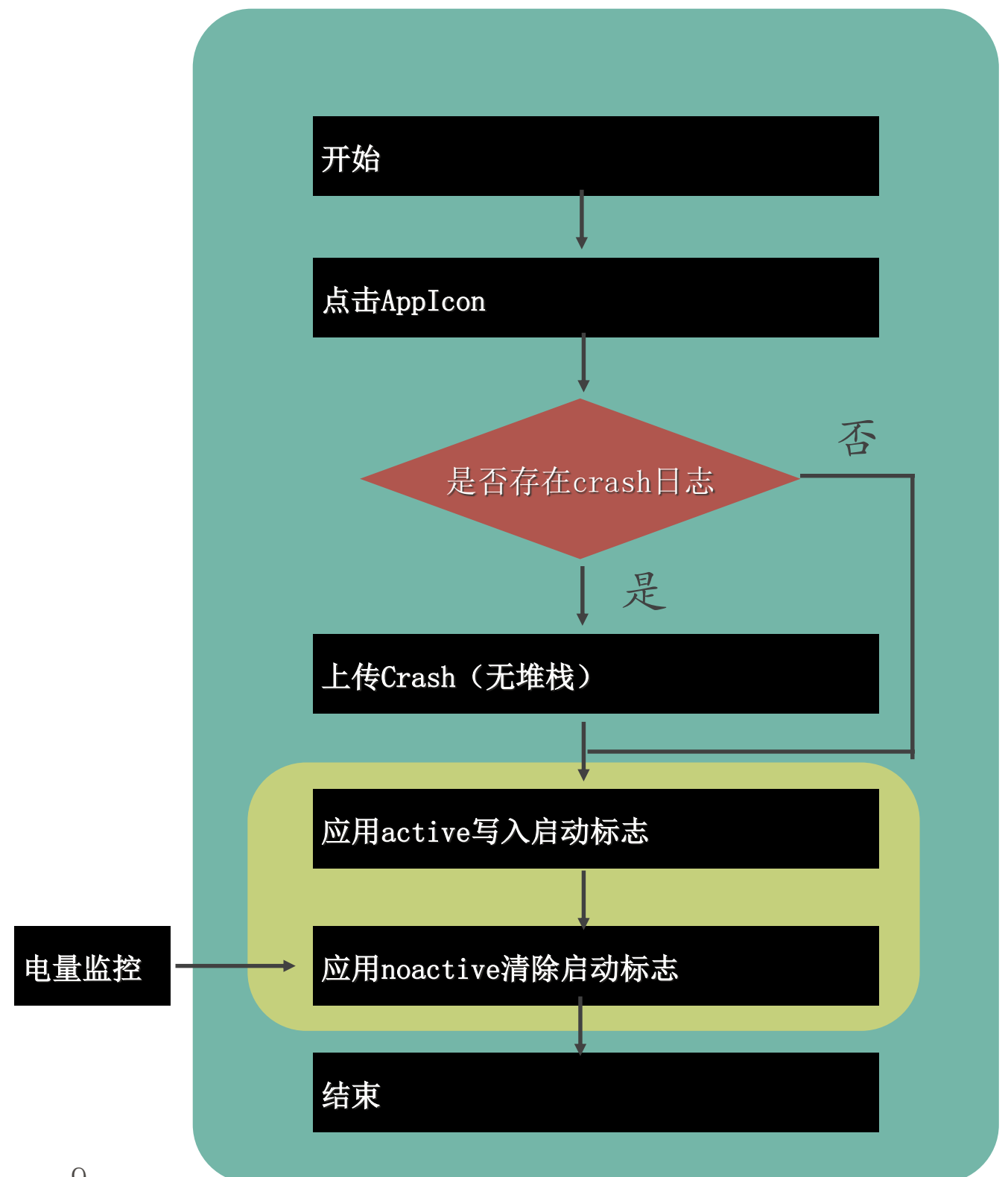
Crash

Low Memory Crash

Watchdog timeout (0x8badf00d)

原理

应用Active时，写入标记文件
应用非active,清除标记文件
应用启动时是否有标记文件
，上传Crash埋点



首页 / 崩溃 / 实时Crash

当前: V2.1 2016-08-03 对比: V2.1 2016-07-21

Java Crash次数

13 昨日同期
↑ 333%

Java Crash率

21% 昨日同期
↑ 250%

Native Crash次数

1 昨日同期
--

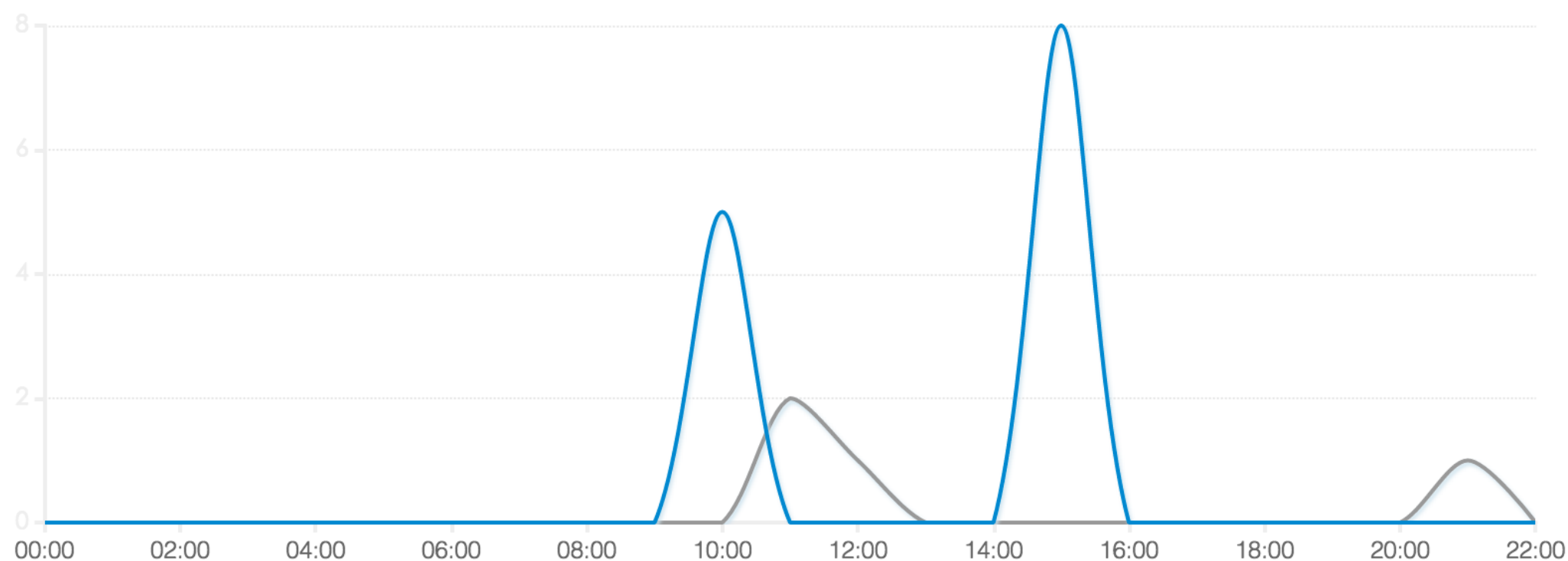
Native Crash率

1% 昨日同期
--

总体Crash率

22% 昨日同期
↑ 266%

影响设备数: 4



—○— 当前版本: 2.1 日期:2016-08-03

—○— 对比版本: 2.1 日期:2016-07-21

Exception Type: SIGABRT
Exception Codes: #0 at 0x195523270
Triggered by Thread: 0

Application Specific Information:

*** Terminating app due to uncaught exception 'NSRangeException', reason: '*** -[__NSArrayI objectAtIndex:]: index 1 beyond bounds [0 .. 0]'

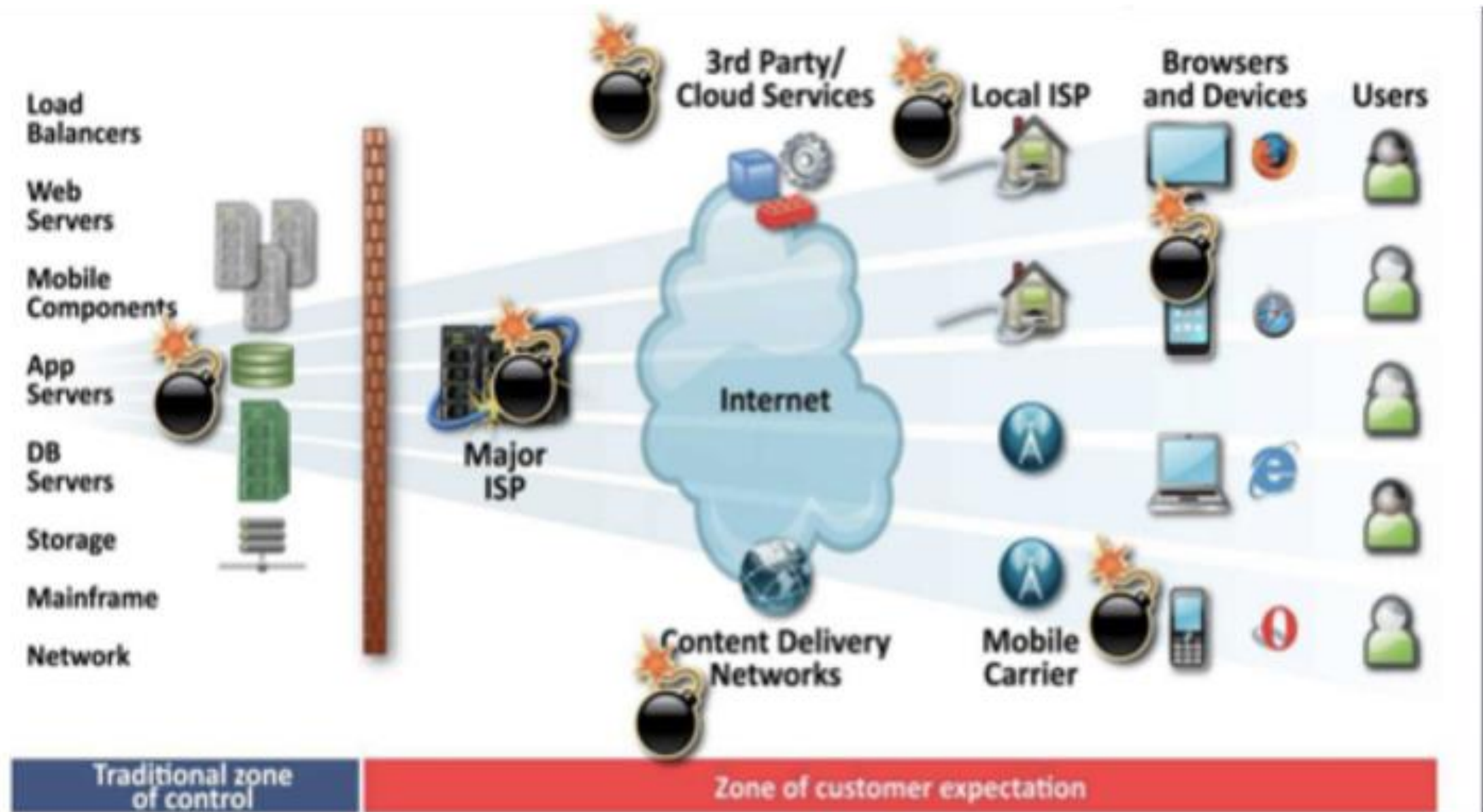
Last Exception Backtrace:

```
0  CoreFoundation      0x00000001835602d8 __exceptionPreprocess :132 (in CoreFoundation)
1  libobjc.A.dylib     0x0000000194d8c0e4 _objc_exception_throw :60 (in libobjc.A.dylib)
2  CoreFoundation      0x0000000183443b34 -[__NSArrayI objectAtIndex:] :224 (in CoreFoundation)
3  iOSTest              0x00000001000fcf88 +[Skeleton outOfRange] Skeleton.m:33 (in iOSTest)
4  iOSTest              0x0000000100100fbc _TTSf4g_g_d___TFC7iOSTest14ViewController9tableViewFTCSo11UITableView23didSelectRowAtI
5  iOSTest              0x00000001000ffaa4 _TToFC7iOSTest14ViewController9tableViewFTCSo11UITableView23didSelectRowAtIndexPathCSc
6  UIKit                0x00000001880c9474 -[UITableView _selectRowAtIndexPath:animated:scrollPosition:notifyDelegate:] :1280 (ir
7  UIKit                0x0000000188183790 -[UITableView _userSelectRowAtIndexPath:] :276 (in UIKit)
8  UIKit                0x0000000188024240 __applyBlockToCFArrayCopiedToStack :356 (in UIKit)
9  UIKit                0x0000000187f946ec __afterCACommitHandler :536 (in UIKit)
10 CoreFoundation      0x00000001835182a4 __CFRunLoop_IS_CALLING_OUT_TO_AN_OBSERVER_CALLBACK_FUNCTION__ :32 (in CoreFoundation)
11 CoreFoundation      0x0000000183515230 __CFRunLoopDoObservers :360 (in CoreFoundation)
12 CoreFoundation      0x0000000183515610 __CFRunLoopRun :836 (in CoreFoundation)
13 CoreFoundation      0x00000001834412d4 _CFRunLoopRunSpecific :396 (in CoreFoundation)
14 GraphicsServices    0x000000018cc576fc _GSEventRunModal :168 (in GraphicsServices)
15 UIKit                0x0000000188006fac _UIApplicationMain :1488 (in UIKit)
16 iOSTest              0x00000001000fd158 main main.m:15 (in iOSTest)
17 libdyld.dylib        0x000000019540aa08 _start :4 (in libdyld.dylib)
```

Thread 0 Crashed:

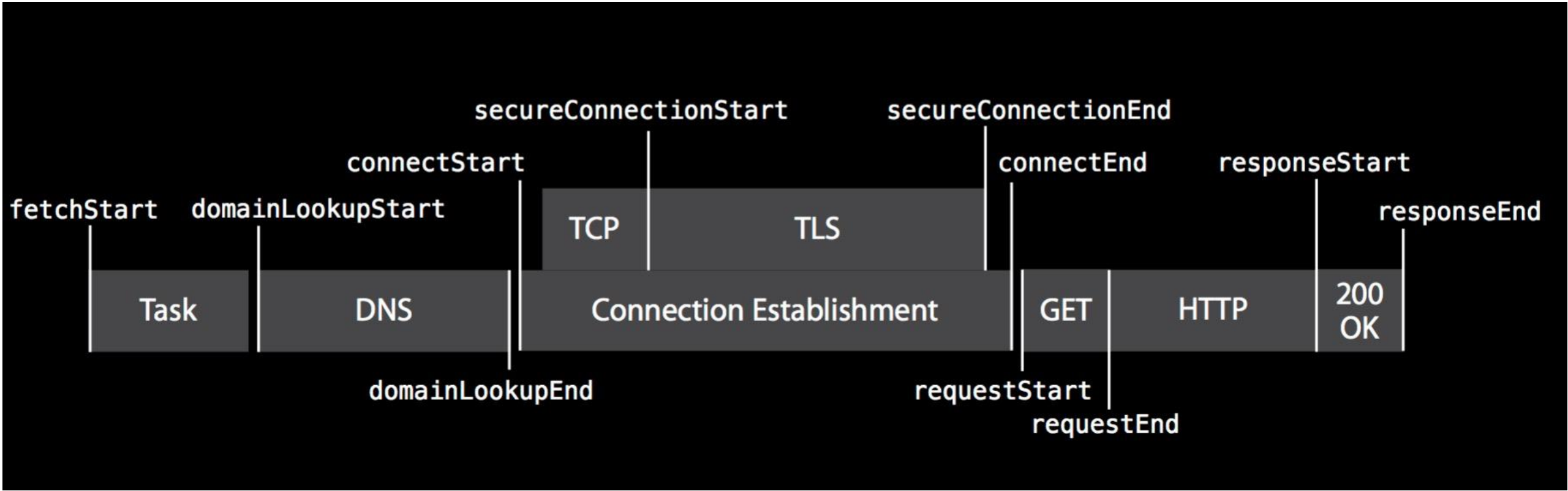
```
0  libsystem_kernel.dylib 0x0000000195523270 pthread_kill :8 (in libsystem_kernel.dylib)
```

稳定的网络连接



NSURLSessionTaskTransactionMetrics

指标
错误率
每分钟请求量
平均响应时间



NSURLConnection

- + sendAsynchronousRequest:queue:completionHandler:
- + sendSynchronousRequest:returningResponse:error:
- initWithRequest:delegate:startImmediately:
- initWithRequest:delegate:
- start
- cancel

NSURLSession

- delegate
- + sessionWithConfiguration:
- + sessionWithConfiguration:delegate:delegateQueue:
- dataTaskWithURL:
- dataTaskWithURL:completionHandler:
- dataTaskWithRequest:completionHandler:
- dataTaskWithRequest:
- downloadTaskWithURL:
- downloadTaskWithResumeData:
- downloadTaskWithRequest:
- downloadTaskWithURL:completionHandler:
- downloadTaskWithResumeData:completionHandler:
- downloadTaskWithRequest:completionHandler:
- uploadTaskWithRequest:fromFile:
- uploadTaskWithRequest:fromData:
- uploadTaskWithRequest:fromFile:completionHandler:
- uploadTaskWithRequest:fromData:completionHandler:
- uploadTaskWithStreamedRequest:

NSURLSessionTask

- resume

CFNetwork

- CFReadStreamCreateForHTTPRequest(...)
- CFReadStreamCreateForStreamedHTTPRequest(...)
- CFReadStreamSetClient(...)
- CFReadStreamOpen(...)
- CFReadStreamRead(...)

NSInputStream

- setDelegate:
- delegate
- open
- read:maxLength:

Connection
Delegate
Proxy

起始日期
结束日期
请求
响应
错误
错误数据
发送数据量
接收数据量

Session
Delegate
Proxy

起始日期
结束日期
请求
响应
错误
错误数据
发送数据量
接收数据量

Surveyor

起始日期
结束日期
请求
响应
错误
错误数据
发送数据量
接收数据量

InputStream
Delegate
Proxy

Mirror
Reporter

起始日期
结束日期
请求
响应
错误
错误数据
发送数据量
接收数据量

Reporter
DI

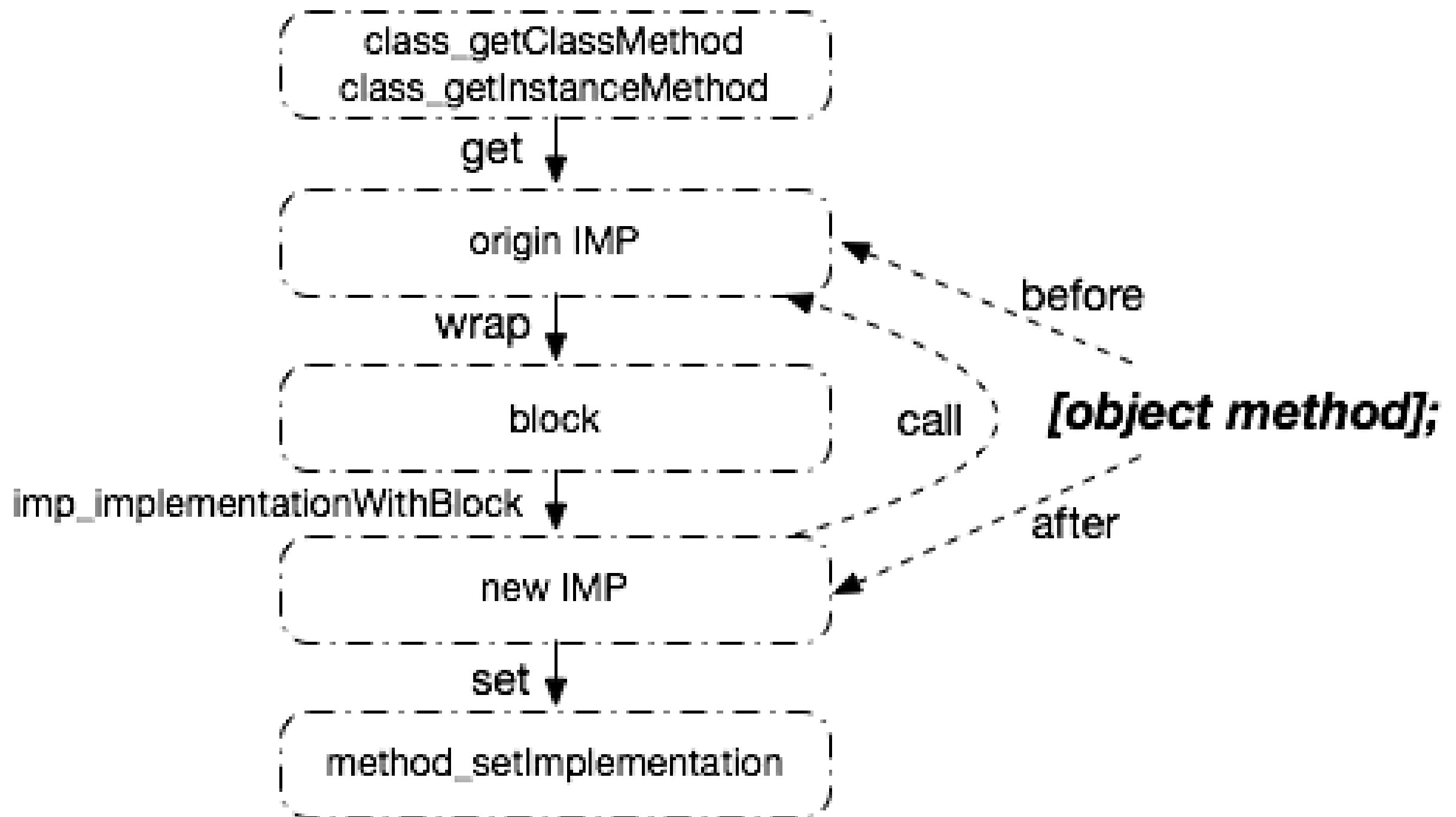
AppMonitor
Reporter

起始日期
结束日期
请求
响应
错误
错误数据
发送数据量
接收数据量

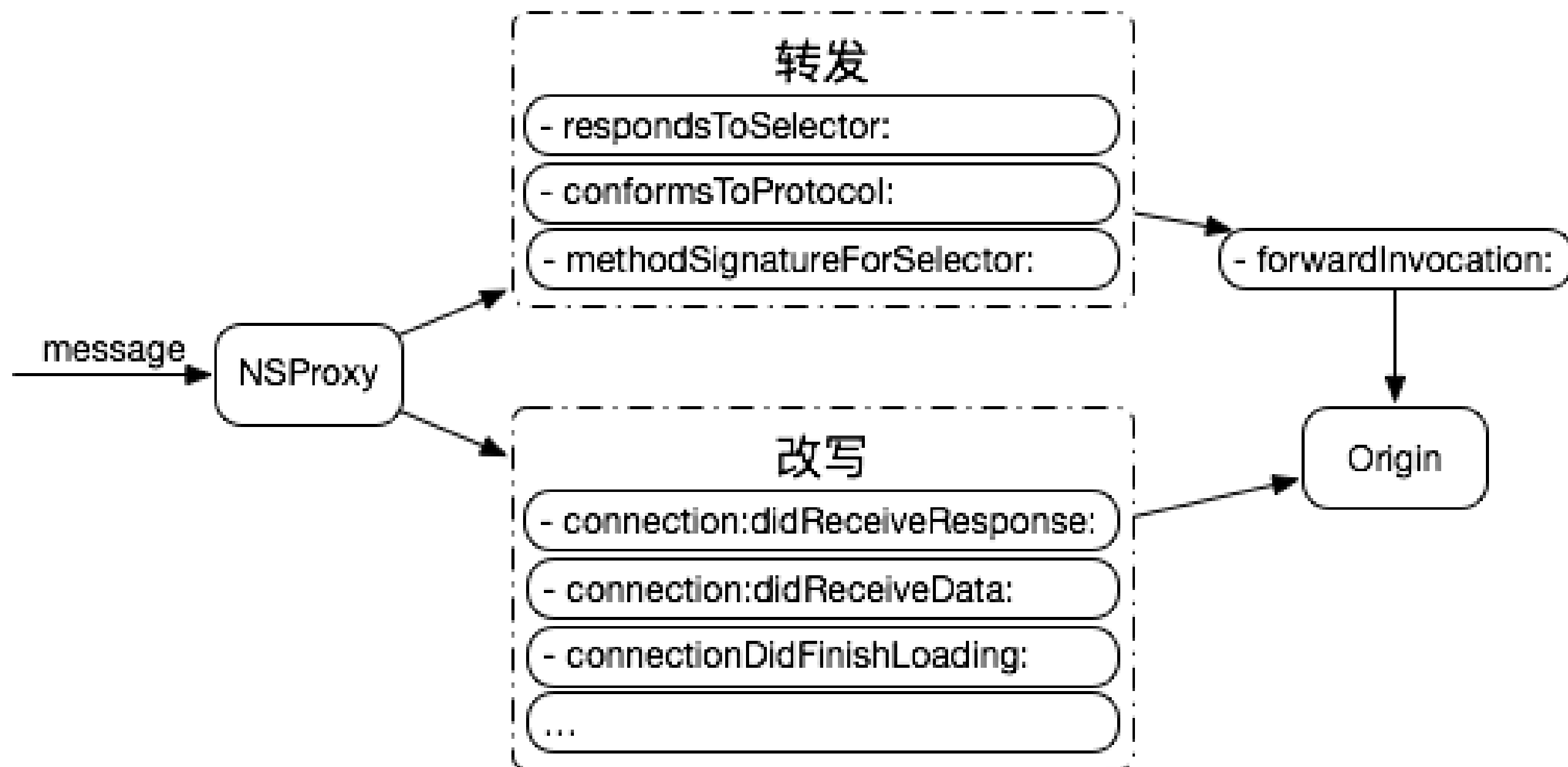
database

alert

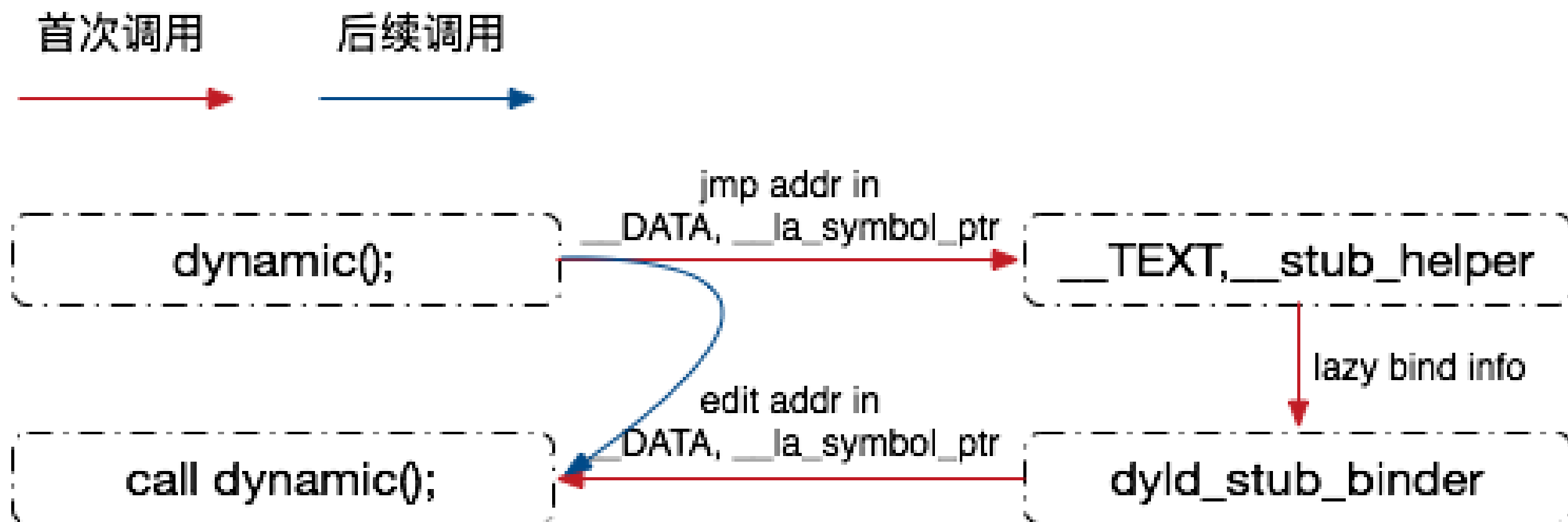
方式一：method Swizzling

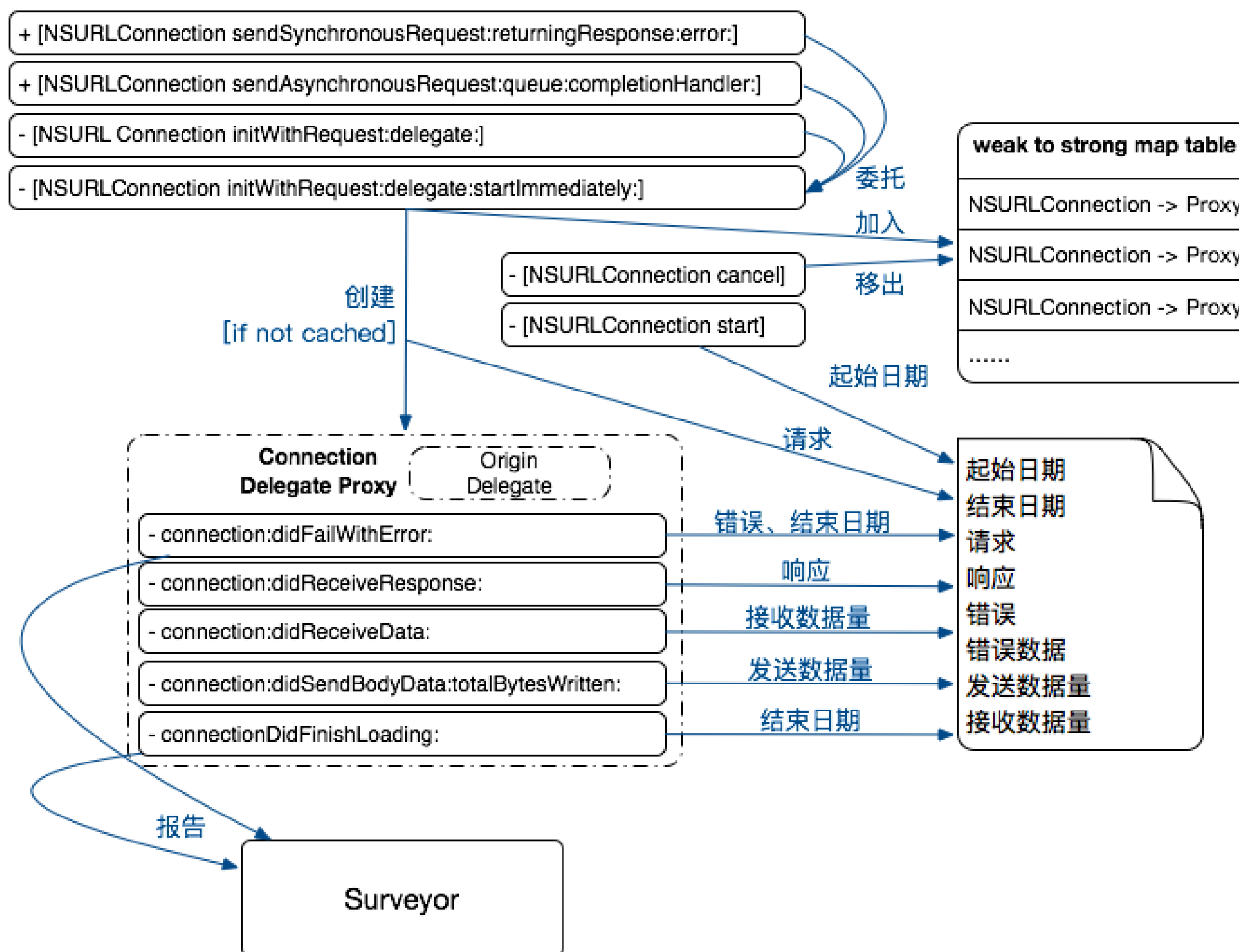


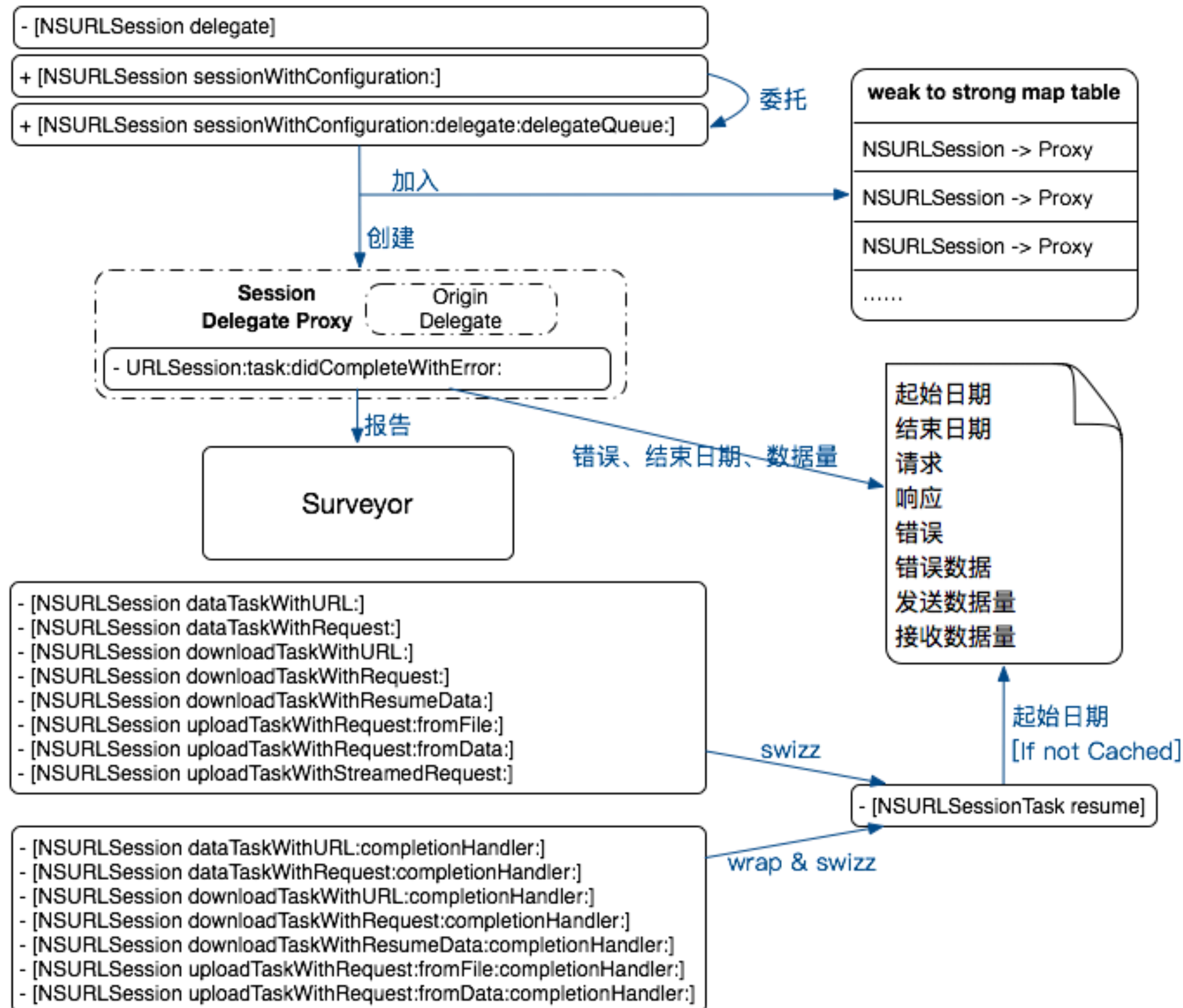
方法二：Proxy转发

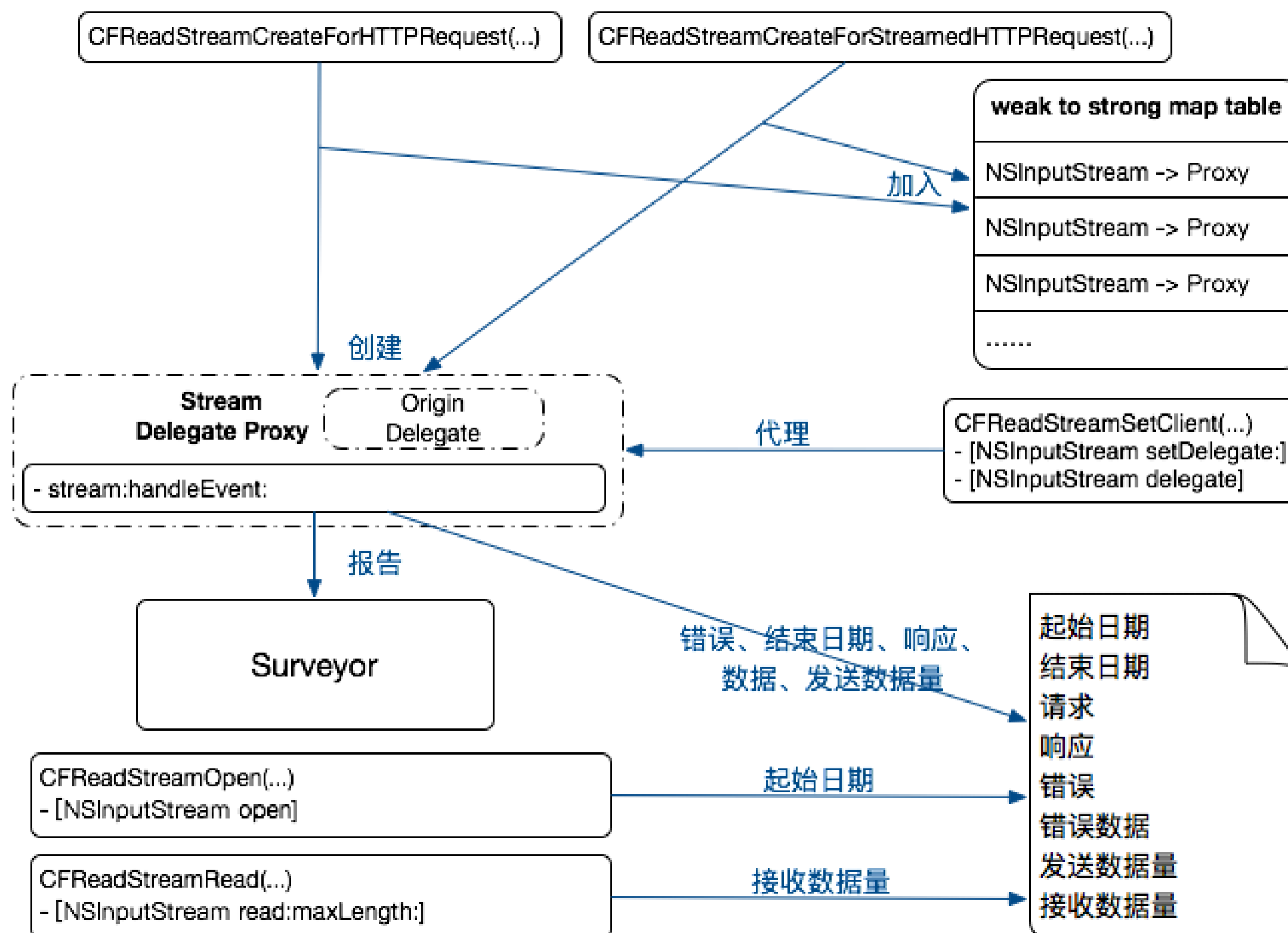


方法三：动态binding









错误分析

首页 / 网络 / 错误分析

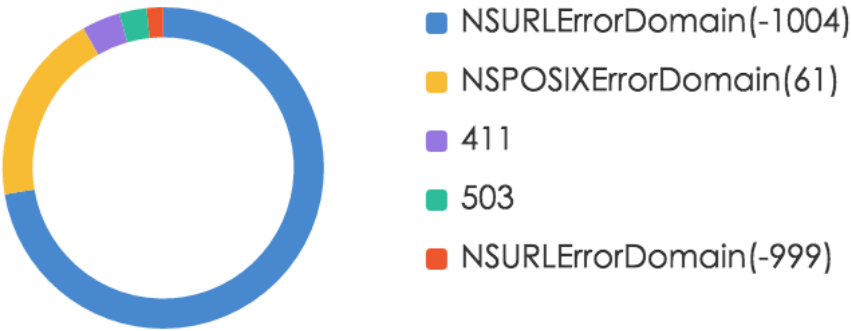
应用版本: 全部版本

统计时间: 最近6小时

错误总览 Top5

错误URL	错误率	发生次数
http://127.0.0.1:12306/apm/post.htm	100%	75
http://127.0.0.1:12306/apm/get.htm	100%	60
http://127.0.0.1:12306/apm/timeout.htm	100%	12
http://127.0.0.1:12306/apm/put.htm	100%	9
http://127.0.0.1:12306/apm/latency.htm	100%	9

错误类型



错误列表

搜索url

错误URL	错误类型	发生次数	最新上报	首现时间
http://www.x.x	503	5	2016-08-04 11:17:10	2016-07-04 11:02:09
http://127.0.0.1:12306/apm/get.htm	NSPOSIXErrorDomain(61)	12	2016-08-04 11:17:10	2016-08-04 11:09:32
http://127.0.0.1:12306/apm/get.htm	NSURLErrorDomain(-1004)	45	2016-08-04 11:17:10	2016-06-21 17:36:21

错误分析

首页 / 网络 / 错误分

应用版本: 全部版本

错误总览 Top5

错误URL

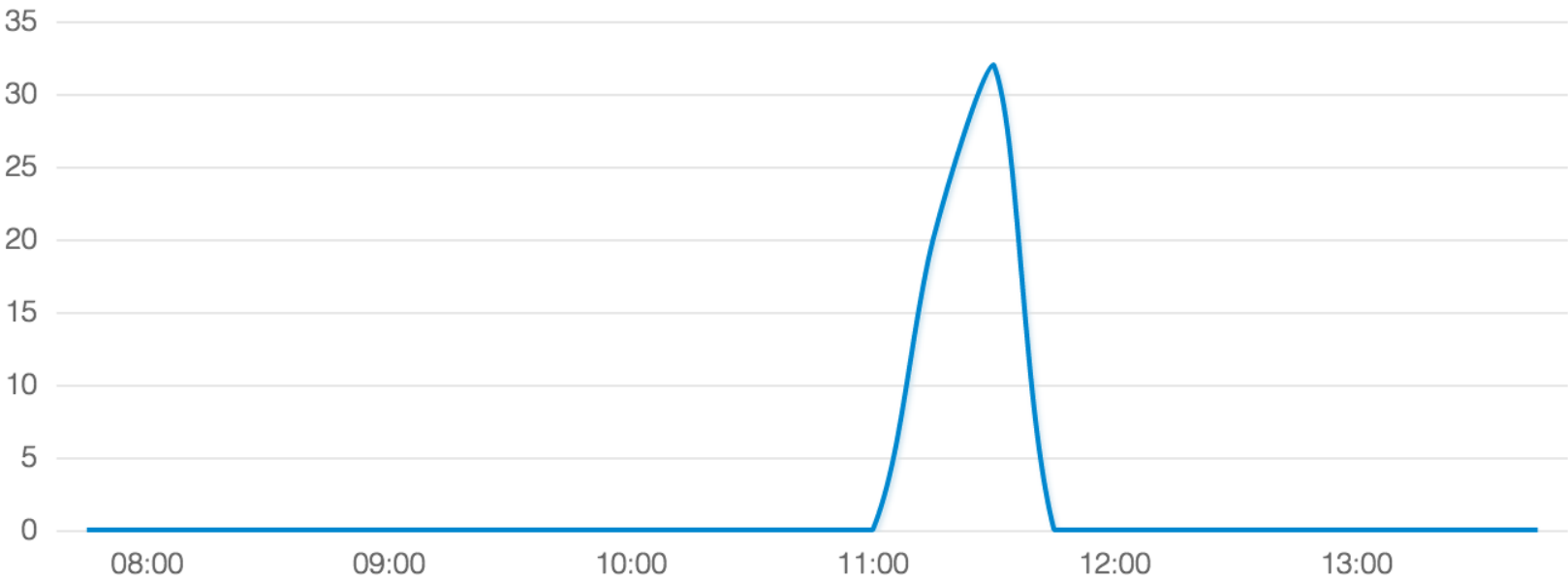
错误列表

搜索url



错误类型	发生次数	最新上报	首现时间
NSURLErrorDomain(-1004)	52	2016-08-04 11:17:10	2016-06-21 17:46:23

错误分布图



错误日志

发生时间: 2016-08-04 11:16:25

< 1 / 49 >

Response headers	Response body	Request headers	设备信息
{ "Content-Length": "27278"			

拓扑

首页 / 网络 / 拓扑

应用版本：

全部版本

统计时间：

最近14天

网络环境：

不限

运营商：

所有运营商

应用总览

- ①

平均响应时间：

27 ms
- ①

每分钟请求量：

0.01 次
- ①

错误率：

89.952 %

拓扑详情

服务	平均响应时间	每分钟请求量	错误率	操作
baichuan.com	0 ms	0 次	0 %	展开
qrcode.com	0 ms	0 次	0 %	展开
image69.360doc.com	335 ms	0 次	0 %	展开
qq.com	0 ms	0 次	0 %	展开
x.x	8 ms	0 次	100 %	展开
taobao.com	0 ms	0 次	0 %	展开
127.0.0.1:12306	7 ms	0.009 次	100 %	展开
baidu.com	150 ms	0 次	70 %	展开

http请求

首页 / 网络 / http请求

应用版本: 统计时间: 网络环境: 运营商:

URL列表

^

所有host

image69.360doc.com

x.x

127.0.0.1:12306

baidu.com

< 1 / 1 >

URL详情: http://image69.360doc.com/DownloadImg/2014/02/2709/39459401_2.jpg

平均响应时间

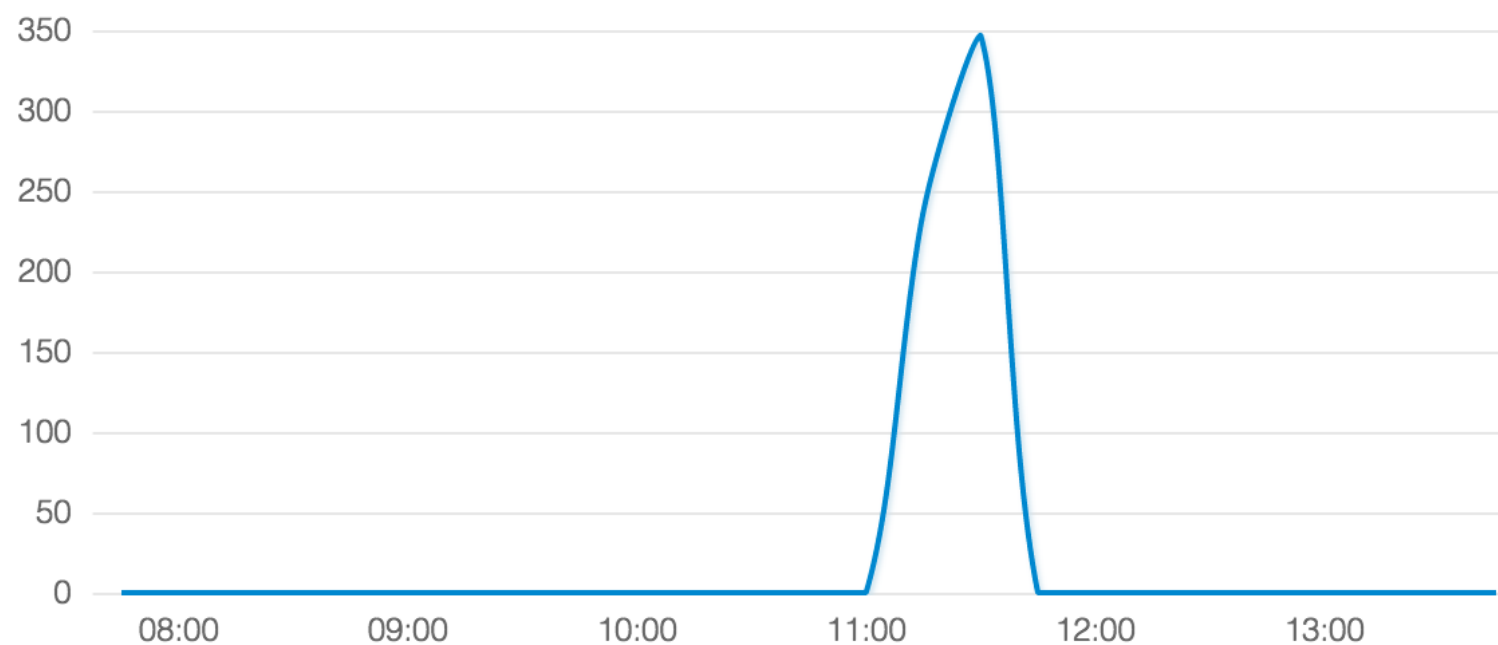
335 ms

每分钟请求量

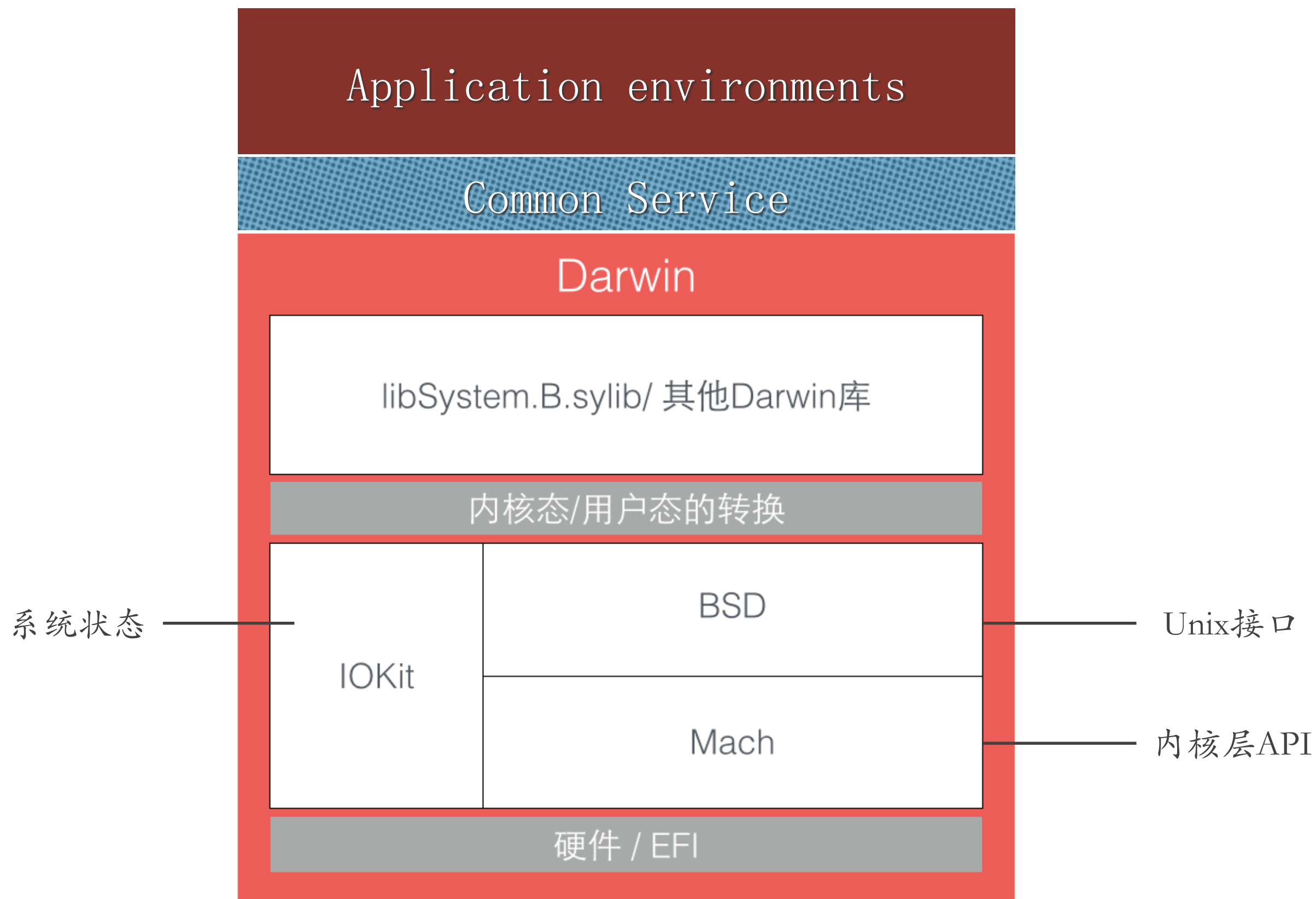
0.025 次

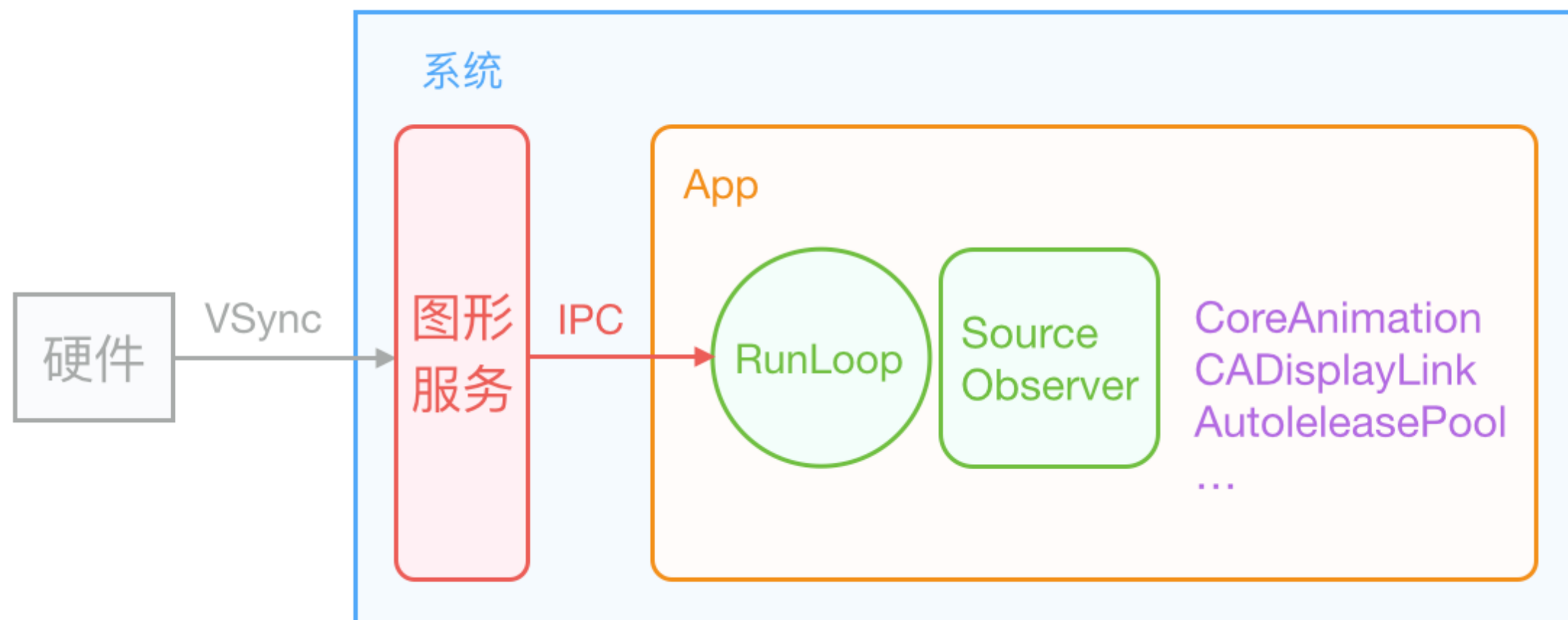
错误率 ↗

0 %



流畅的操作界面





方案一：监听FPS 降低 — CADisplayLink

方案二：主线程 Runloop 执行



```
do
{
    // 通知将要处理timer和source
    __CFRunLoopDoObservers(kCFRunLoopBeforeTimers);
    __CFRunLoopDoObservers(kCFRunLoopBeforeSources);

    __CFRunLoopDoBlocks(); //处理非延迟的主线程调用
    __CFRunLoopDoSource0(); //处理UIEvent事件

    //GCD dispatch main queue
    CheckIfExistsMessagesInMainDispatchQueue();

    // 即将进入休眠
    __CFRunLoopDoObservers(kCFRunLoopBeforeWaiting);

    // 等待内核mach_msg事件
    mach_port_t wakeUpPort = SleepAndWaitForWakingUpPorts();

    // Zzz...

    // 从等待中醒来
    __CFRunLoopDoObservers(kCFRunLoopAfterWaiting);

    // 处理因timer的唤醒
    if (wakeUpPort == timerPort)
        __CFRunLoopDoTimers();

    // 处理异步方法唤醒,如dispatch_async
    else if (wakeUpPort == mainDispatchQueuePort)
        __CFRUNLOOP_IS_SERVICING_THE_MAIN_DISPATCH_QUEUE__();

    // UI刷新,动画显示
    else
        __CFRunLoopDoSource1();

    // 再次确保是否有同步的方法需要调用
    __CFRunLoopDoBlocks();

} while (!stop && !timeout);
```

卡顿检测原理

主线程RunLoop

1. 通知Observer: 即将进入runLoop

2. 通知Observer: 即将处理timer

kCFRunLoopBefore
Timers

3. 通知Observer: 即将处理source0

kCFRunLoopBeforeS
ource

4. 处理 blocks & Source0

处理事件响应

5. 如果有Source1, 跳到9步

刷新UI

6. 通知Observer线程即将休眠

kCFRunLoopBefore
Waiting

7. 休眠等待唤醒

8. 通知Observer线程刚被唤醒

kCFRunLoopAfterW
aiting

9. 处理唤醒时收到的消息, 之后跳回2

刷新UI

10. 通知Observer即将退出runloop

监听线程

注册RunLoop回调

循环检测主线程runloop

runloop是否跑完

否

source的状态

计数器++

计数器>N dump 内存堆栈

卡顿检测的问题

问题:

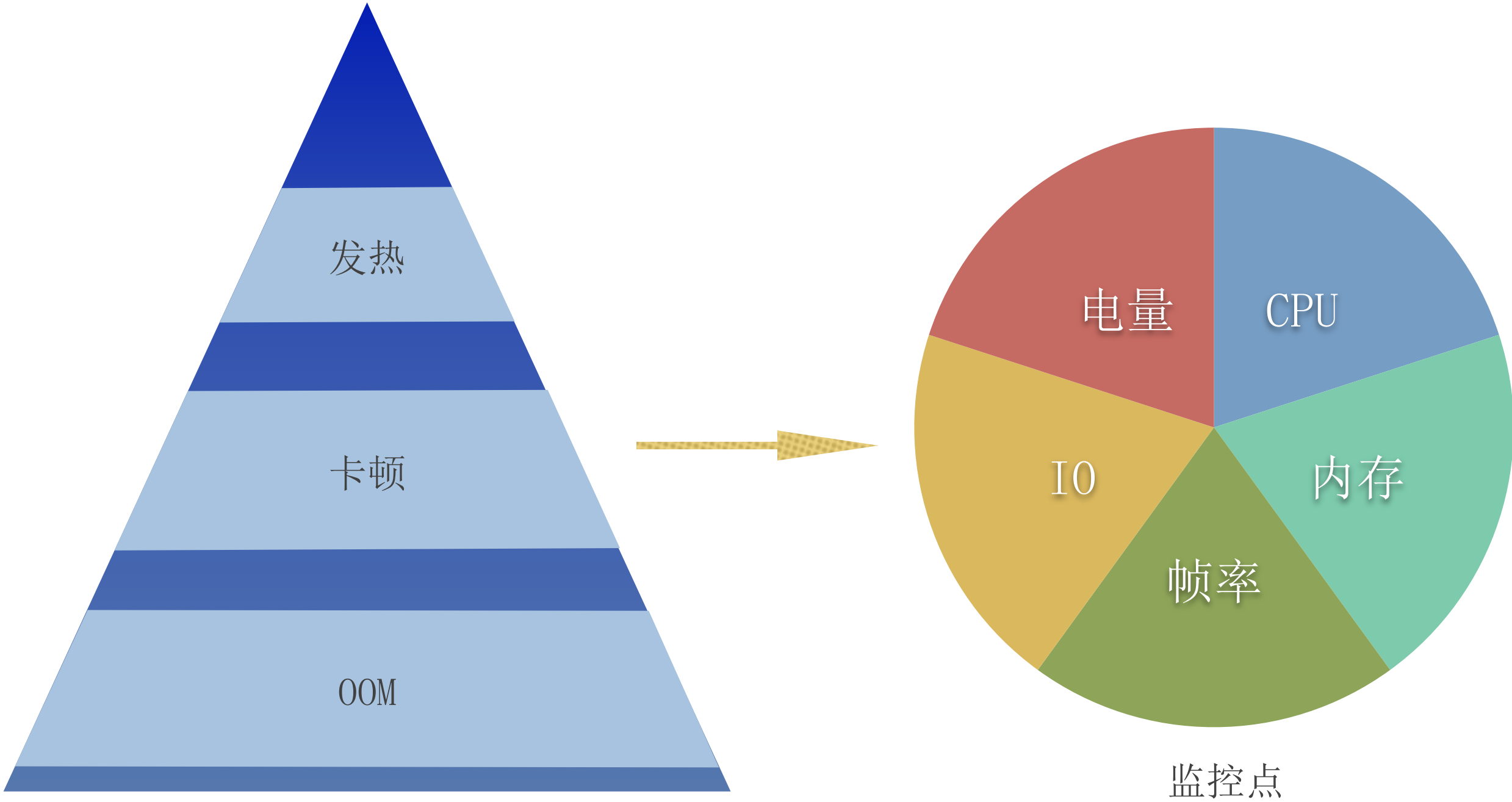
- 数据量大,实时处理有困难,数据采集可能影响到业务服务器的性能
- 一直check和dump堆栈比较耗电
- 启动和退后台卡顿

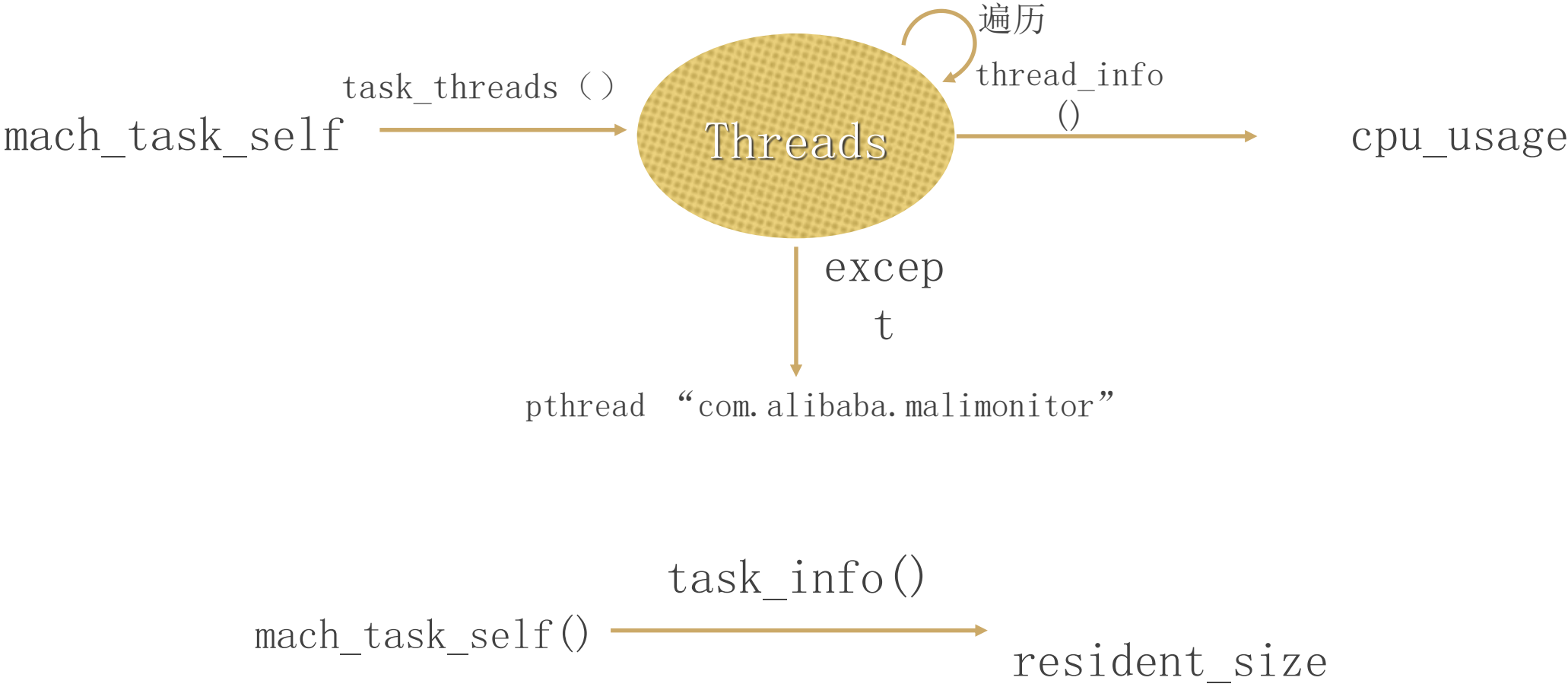
解决:

- 用户采样
- 按堆栈做聚合
- 触屏调整
- watch dog 动态调整

总体			0-10					10-20					
滑动速度		偏移量		帧率		滑动速度	偏移量		帧率	滑动速度	偏移量		帧率
平均值 ⚡	平均值 ⚡	90%区间 ⚡	平均值 ⚡	90%区间 ⚡	平均值 ⚡	平均值 ⚡	90%区间 ⚡	平均值 ⚡	90%区间 ⚡	平均值 ⚡	平均值 ⚡	90%区间 ⚡	平均值 ⚡
0.2185	2334.1	5500	4.41	5	0.3224	201.39	500	15.02	15	0.2459	298.13	500	25.01

尽量少的资源消耗





方案一

`[UIDevice currentDevice]`
得到BatteryState和BatteryLevel

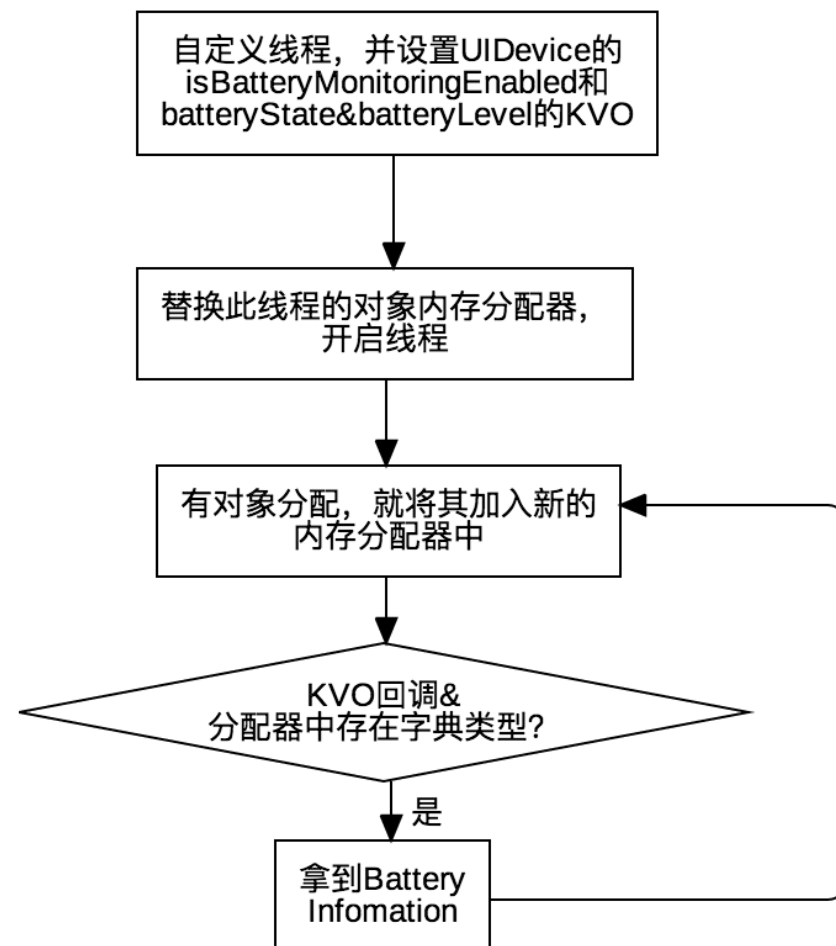
方案二

IOKit `path:/System/Library/Frameworks/IOKit.framework`

得到InstantAmperage（电流）和Voltage（电压）

电量获取方案

方案三 替换线程allocator



方案四

链接iOSDiagnosticsSupport私有库

反射出MBSDevice实例，并调用copyPowerLogsToDir:方法，将日志信息拷贝到指定磁盘

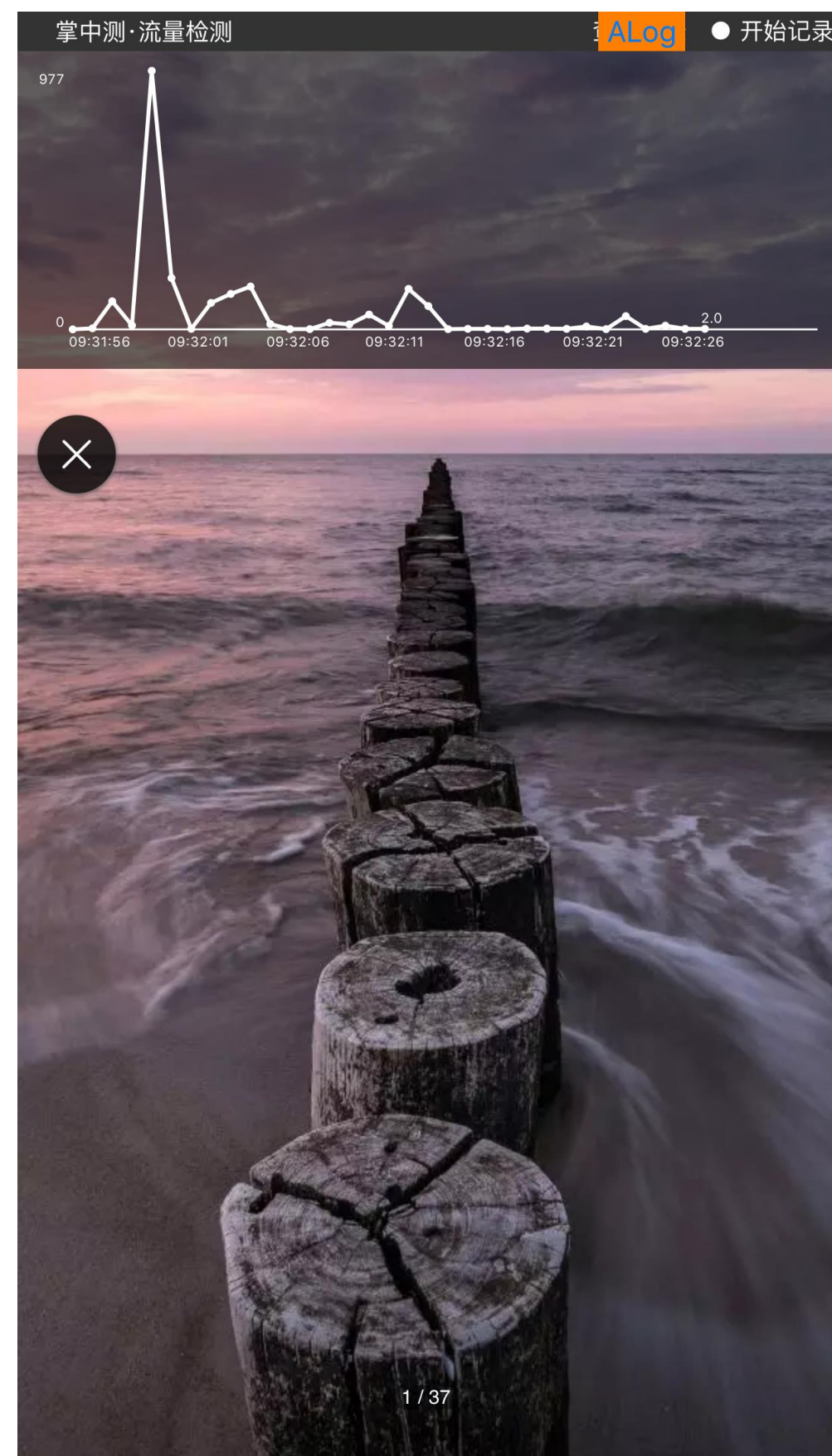


这些表中包含了iOS系统采集的所有应用的小时级别的耗电量

电量获取方案

方案	优点	缺点
UIDevice属性	简单	粗粒度，不符合要求
IOKit	可以拿到当前的电流电压	无法精确到应用级别
替换线程Allocator	拿到数据同上	除了以上，还有活的数据的时机依赖于KVO
越狱方案	可以拿到每个应用小时耗电量	小时级别, 不符合要求

百川掌中测截图



漂亮的视觉交互



向上一个像素

颜色再深一点

字体小一号

你妹...



上午9:41

ALog100%

扫一扫

Q运动会金牌好货

消息

极有家



家有宝贝
好材质 高颜值
送优惠 抢红包

天猫

聚划算

天猫国际

外卖

天猫超市

充值中心

阿里旅行

领金币

到家

分类

淘宝头条

热议

你适合直发还是卷发，明星告诉你！有图有...

热评

鲁豫换发型年轻了7岁，简直美成郭碧婷！

淘宝便利店
1小时达

水果买一赠一
领券满25减5,1小时达



淘抢购

汇全网尖货 抢半价

00:07:44



99元
限量抢

有好货

每日精选200款
品质生活指南



爱逛街

可爱的你会喜欢

必买清单

都帮您整理好啦

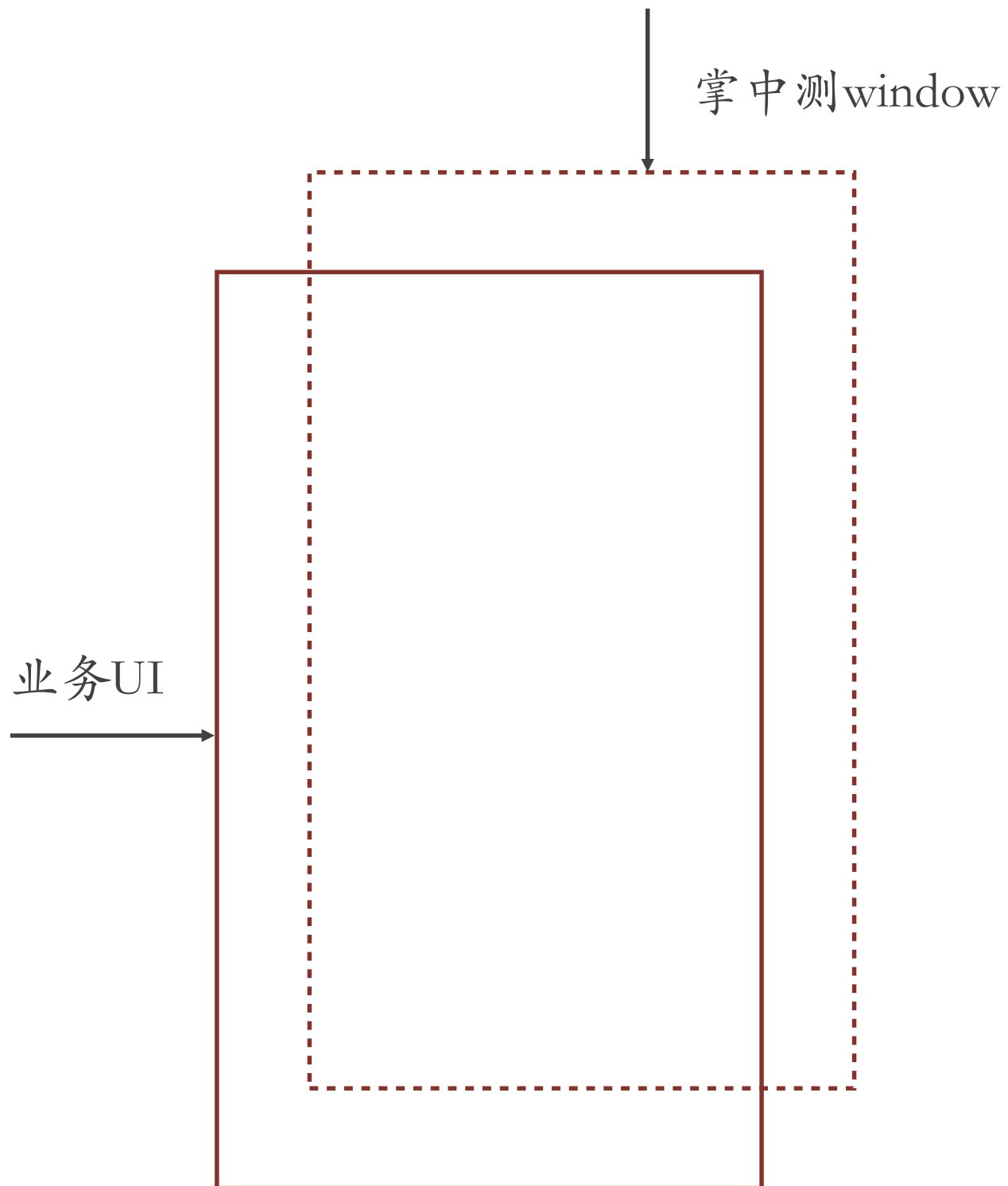
首页

微淘

社区

购物车

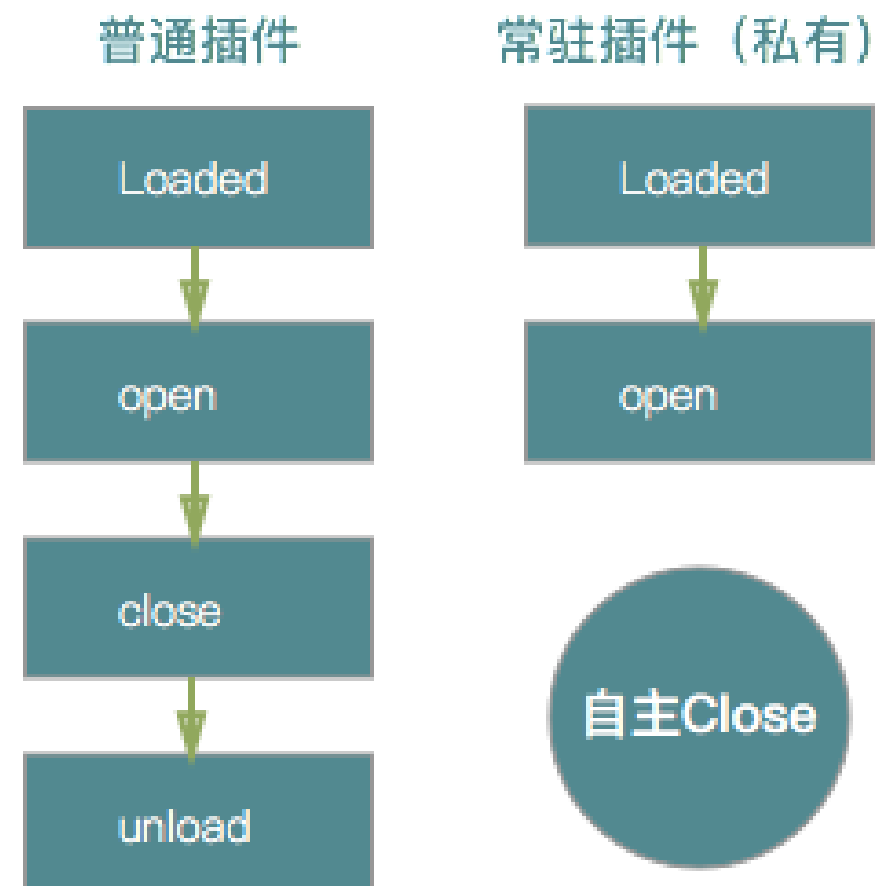
我的淘宝



拦截手势实现交互独立

```
- (BOOL)pointInside:(CGPoint)point withEvent:(UIEvent *)event
{
    BOOL pointInside = NO;
    if ([self.touchDelegate shouldHandleTouchAtPoint:point]) {
        pointInside = [super pointInside:point withEvent:event];
    }
    return pointInside;
}
```

自定义插件和生命周期管理



Thank you

