Caesium
iOS启动时间监控

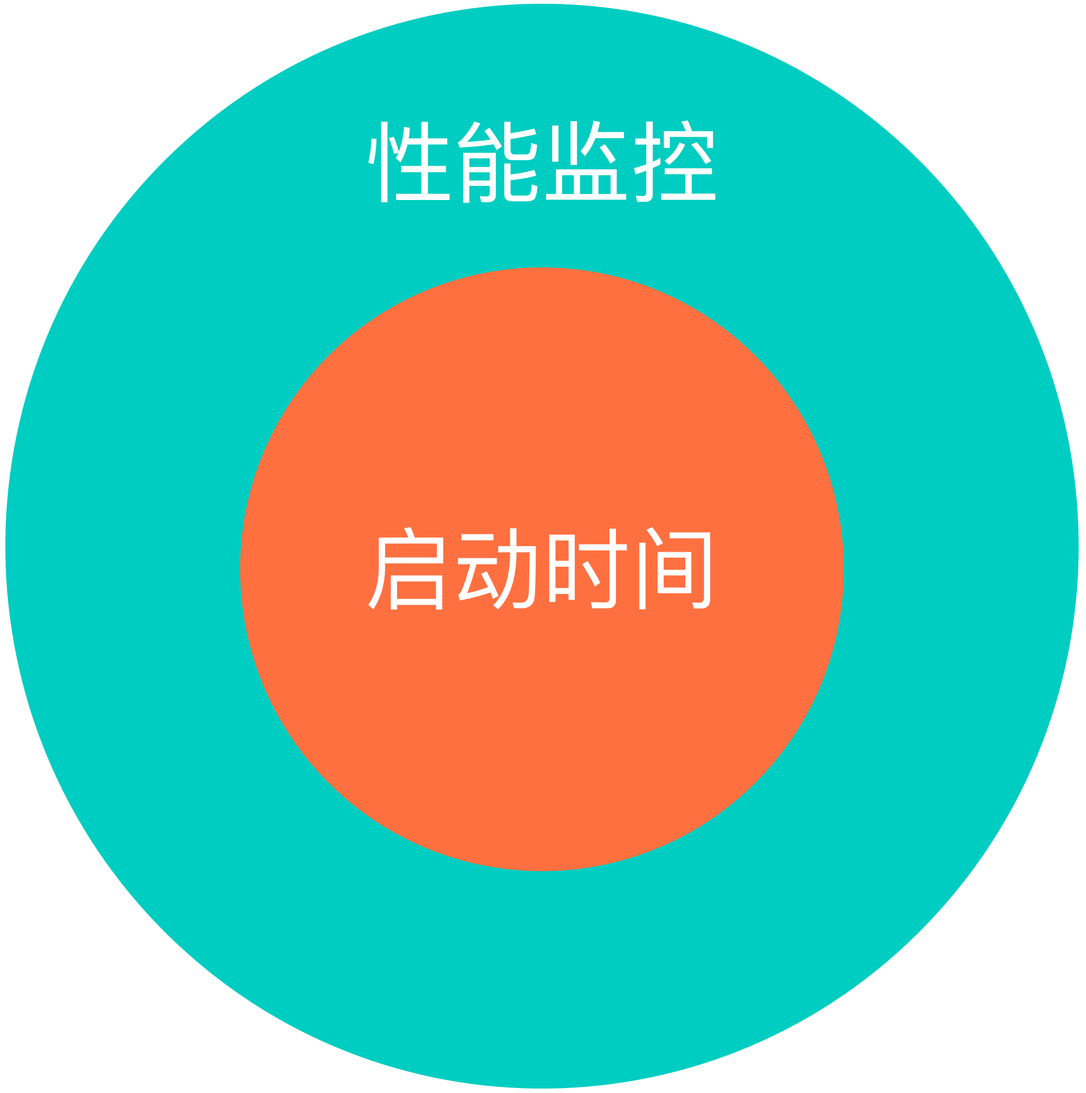
吴君阳

美团点评

美团平台I技术研发部

吴君阳，美团点评iOS工程师。
2017年2月加入美团点评，负责性能平台和性能优化相关工作。

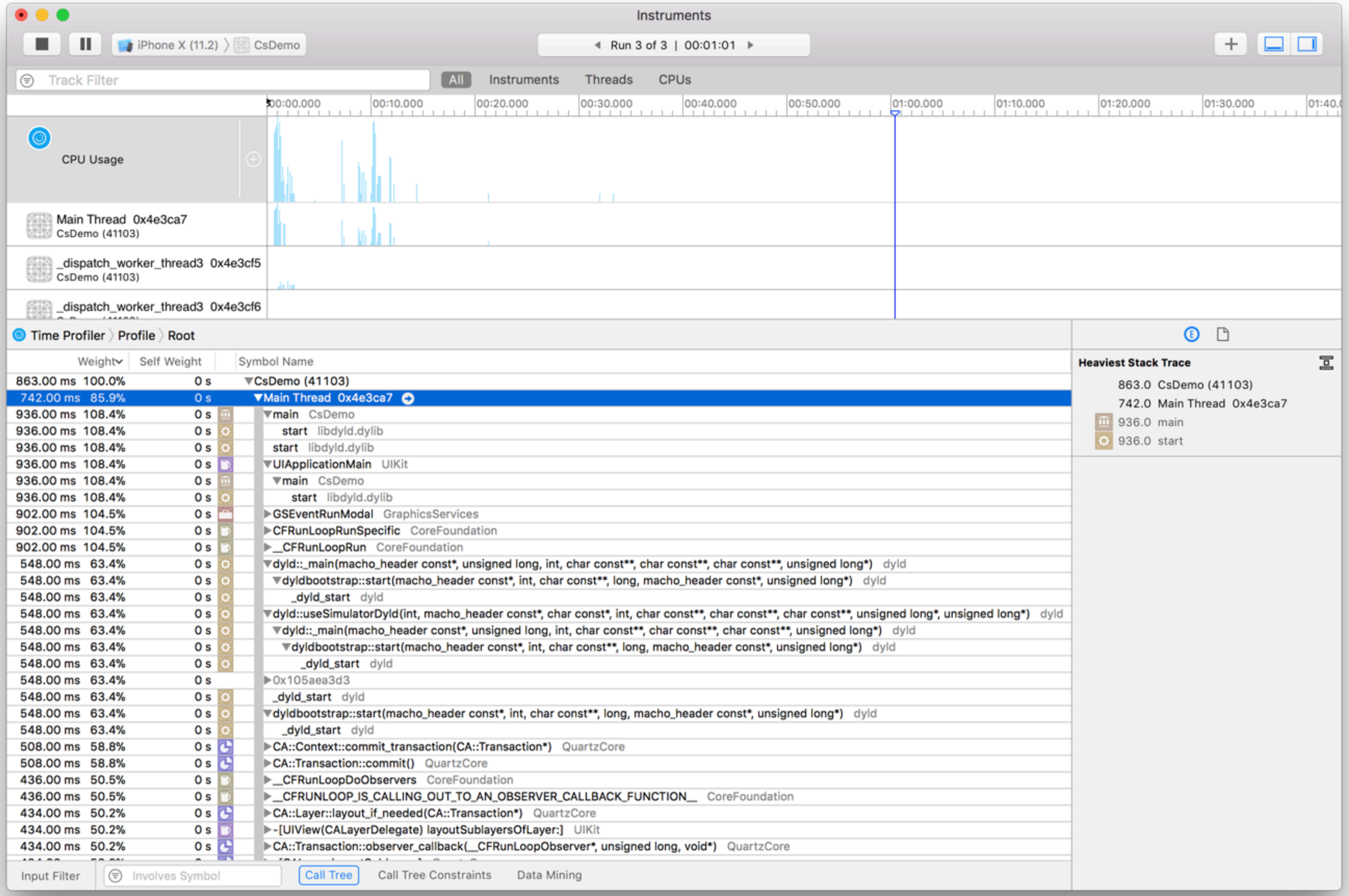
背景



传统做法

Instruments

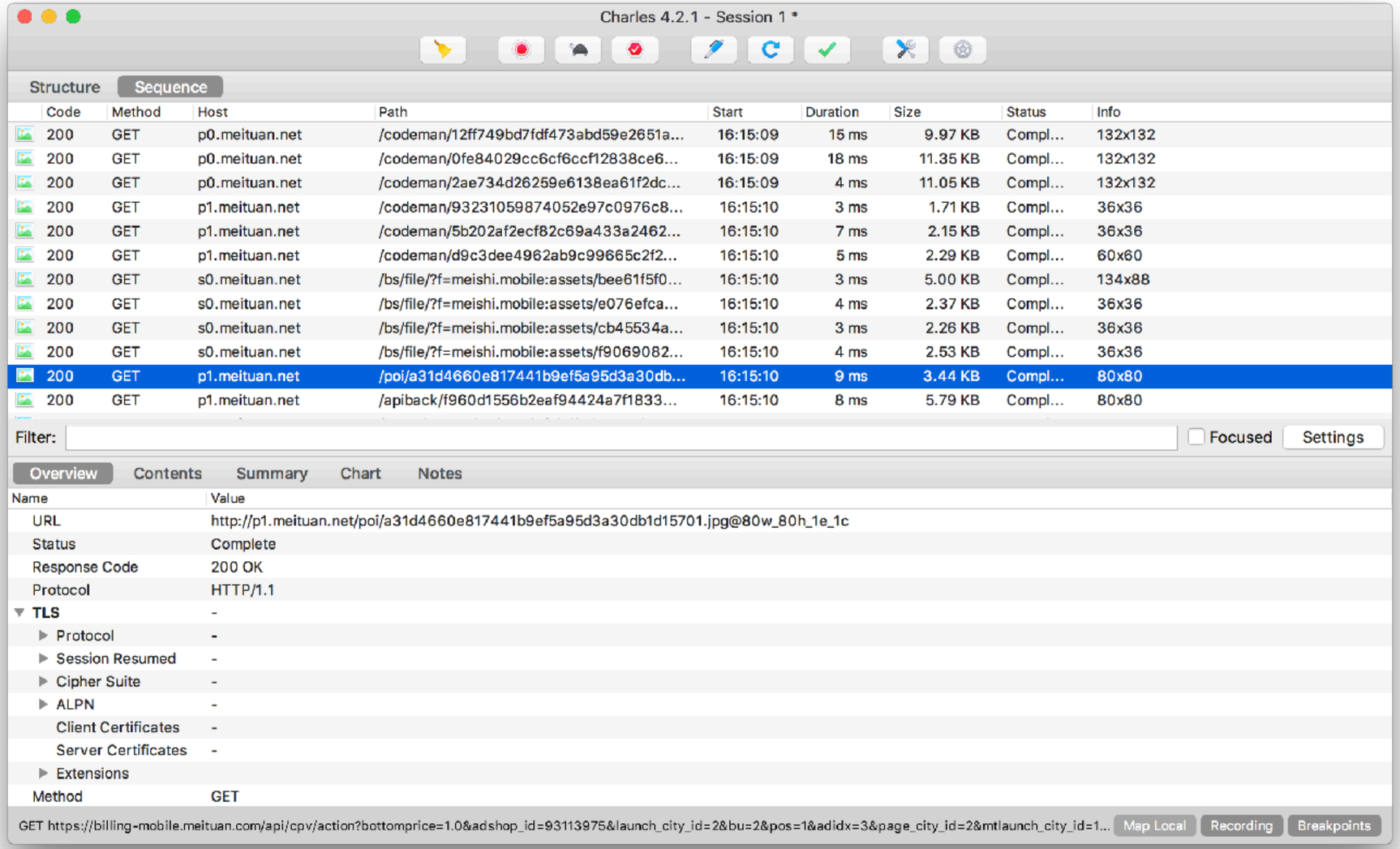
传统做法


Instruments
iPhone X (11.2) 〉 CsDemo
Run 3 of 3 | 00:01:01
Track Filter
All
Instruments
Threads
CPUs
00:00.000
00:10.000
00:20.000
00:30.000
00:40.000
00:50.000
01:00.000
01:10.000
01:20.000
01:30.000
CPU Usage
Main Thread 0x4e3ca7
CsDemo (41103)
_dispatch_worker_thread3 0x4e3cf5
CsDemo (41103)
_dispatch_worker_thread3 0x4e3cf6
Time Profiler 〉 Profile 〉 Root
Weight
Self Weight
Symbol Name
863.00 ms 100.0% 0 s CsDemo (41103)
742.00 ms 85.9% 0 s Main Thread 0x4e3ca7
936.00 ms 108.4% 0 s main CsDemo
936.00 ms 108.4% 0 s start libdyld.dylib
936.00 ms 108.4% 0 s start libdyld.dylib
936.00 ms 108.4% 0 s UIApplicationMain UIKit
936.00 ms 108.4% 0 s main CsDemo
936.00 ms 108.4% 0 s start libdyld.dylib
902.00 ms 104.5% 0 s GSEventRunModal GraphicsServices
902.00 ms 104.5% 0 s CFRunLoopRunSpecific CoreFoundation
902.00 ms 104.5% 0 s __CFRunLoopRun CoreFoundation
548.00 ms 63.4% 0 s dyld::_main(macho_header const*, unsigned long, int, char const**, char const**, char const**, unsigned long*) dyld
548.00 ms 63.4% 0 s dyldbootstrap::start(macho_header const*, int, char const**, long, macho_header const*, unsigned long*) dyld
548.00 ms 63.4% 0 s _dyld_start dyld
548.00 ms 63.4% 0 s dyld::useSimulatorDyld(int, macho_header const*, char const*, int, char const**, char const**, char const**, unsigned long*, unsigned long*) dyld
548.00 ms 63.4% 0 s dyld::_main(macho_header const*, unsigned long, int, char const**, char const**, char const**, unsigned long*) dyld
548.00 ms 63.4% 0 s dyldbootstrap::start(macho_header const*, int, char const**, long, macho_header const*, unsigned long*) dyld
548.00 ms 63.4% 0 s _dyld_start dyld
548.00 ms 63.4% 0 s 0x105aea3d3
548.00 ms 63.4% 0 s _dyld_start dyld
548.00 ms 63.4% 0 s dyldbootstrap::start(macho_header const*, int, char const**, long, macho_header const*, unsigned long*) dyld
548.00 ms 63.4% 0 s _dyld_start dyld
508.00 ms 58.8% 0 s CA::Context::commit_transaction(CA::Transaction*) QuartzCore
508.00 ms 58.8% 0 s CA::Transaction::commit() QuartzCore
436.00 ms 50.5% 0 s __CFRunLoopDoObservers CoreFoundation
436.00 ms 50.5% 0 s __CFRUNLOOP_IS_CALLING_OUT_TO_AN_OBSERVER_CALLBACK_FUNCTION__ CoreFoundation
434.00 ms 50.2% 0 s CA::Layer::layout_if_needed(CA::Transaction*) QuartzCore
434.00 ms 50.2% 0 s -[UIView(CALayerDelegate) layoutSublayersOfLayer:] UIKit
434.00 ms 50.2% 0 s CA::Transaction::observer_callback(__CFRunLoopObserver*, unsigned long, void*) QuartzCore
Heaviest Stack Trace
863.0 CsDemo (41103)
742.0 Main Thread 0x4e3ca7
936.0 main
936.0 start
Input Filter
Involves Symbol
Call Tree
Call Tree Constraints
Data Mining

传统做法

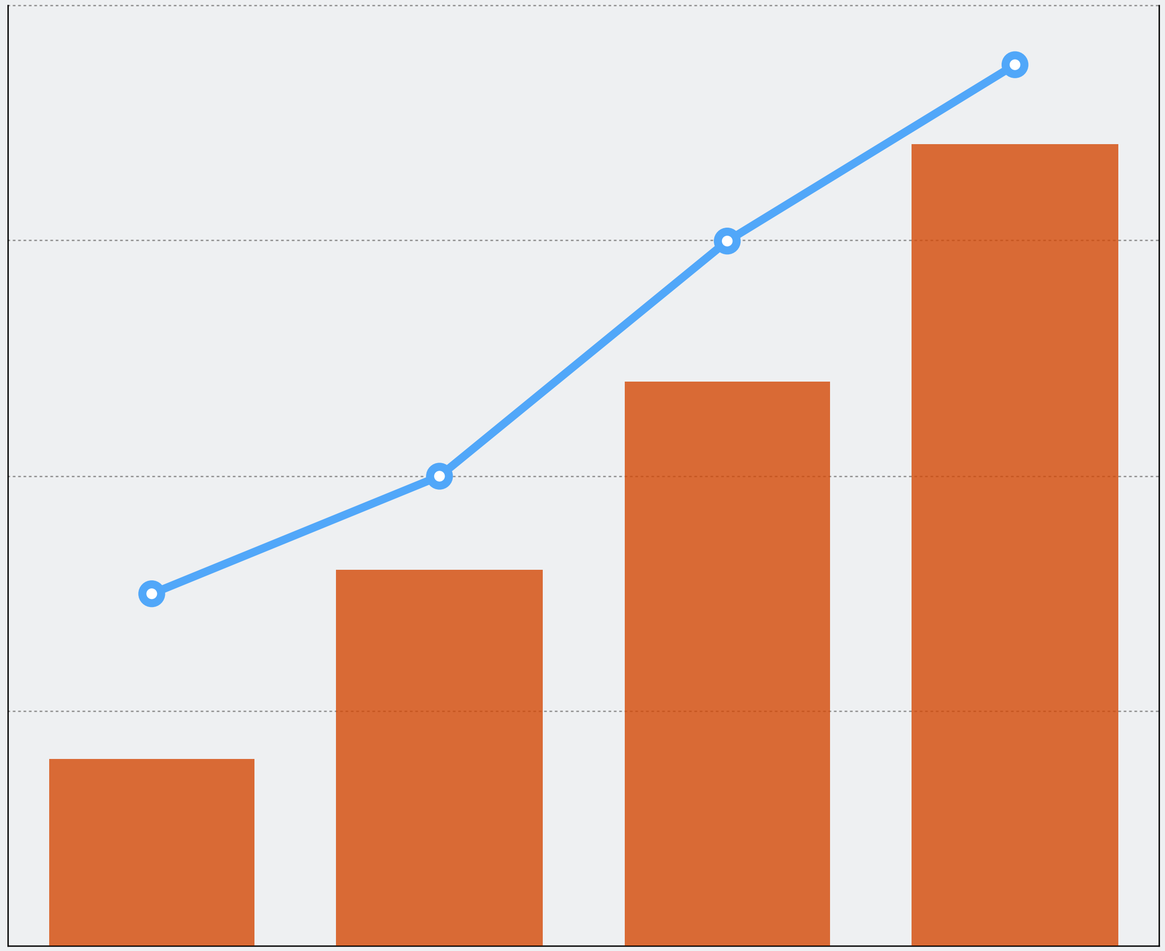
Charles

传统做法


Charles 4.2.1 - Session 1 *
Structure
Sequence
Code Method Host Path Start Duration Size Status Info
200 GET p0.meituan.net /codeman/12ff749bd7fdf473abd59e2651a... 16:15:09 15 ms 9.97 KB Compl... 132x132
200 GET p0.meituan.net /codeman/0fe84029cc6cf6ccf12838ce6... 16:15:09 18 ms 11.35 KB Compl... 132x132
200 GET p0.meituan.net /codeman/2ae734d26259e6138ea61f2dc... 16:15:09 4 ms 11.05 KB Compl... 132x132
200 GET p1.meituan.net /codeman/93231059874052e97c0976c8... 16:15:10 3 ms 1.71 KB Compl... 36x36
200 GET p1.meituan.net /codeman/5b202af2ecf82c69a433a2462... 16:15:10 7 ms 2.15 KB Compl... 36x36
200 GET p1.meituan.net /codeman/d9c3dee4962ab9c99665c2f2... 16:15:10 5 ms 2.29 KB Compl... 60x60
200 GET s0.meituan.net /bs/file/?f=meishi.mobile:assets/bee61f5f0... 16:15:10 3 ms 5.00 KB Compl... 134x88
200 GET s0.meituan.net /bs/file/?f=meishi.mobile:assets/e076efca... 16:15:10 4 ms 2.37 KB Compl... 36x36
200 GET s0.meituan.net /bs/file/?f=meishi.mobile:assets/cb45534a... 16:15:10 3 ms 2.26 KB Compl... 36x36
200 GET s0.meituan.net /bs/file/?f=meishi.mobile:assets/f9069082... 16:15:10 4 ms 2.53 KB Compl... 36x36
200 GET p1.meituan.net /poi/a31d4660e817441b9ef5a95d3a30db... 16:15:10 9 ms 3.44 KB Compl... 80x80
200 GET p1.meituan.net /apiback/f960d1556b2eaf94424a7f1833... 16:15:10 8 ms 5.79 KB Compl... 80x80
Filter:
Focused
Settings
Overview
Contents
Summary
Chart
Notes
Name
Value
URL
http://p1.meituan.net/poi/a31d4660e817441b9ef5a95d3a30db1d15701.jpg@80w_80h_1e_1c
Status
Complete
Response Code
200 OK
Protocol
HTTP/1.1
TLS
Protocol
Session Resumed
Cipher Suite
ALPN
Client Certificates
Server Certificates
Extensions
Method
GET
GET https://billing-mobile.meituan.com/api/cpv/action?bottomprice=1.0&adshop_id=93113975&launch_city_id=2&bu=2&pos=1&adidx=3&page_city_id=2&mtlaunch_city_id=1...
Map Local
Recording
Breakpoints

当传统做法遇到大型App

监控大型App



监控耗时久

代码量大

编译运行一次耗时很长

问题定位难

启动时函数调用量大

数据不够直观

无监控体系

无法及时暴露问题

成果难保持

难以成为一项常规工作

理想的方案

01 数据收集成本低

02 无侵入

03 信息全面

04 数据展示直观

05 体系化

06 对性能影响小

Caesium是什么

在现行国际单位制下，在1967年召开的第13届国际度量衡大会对秒的定义是：铯（Caesium）133原子基态的两个超精细能阶间跃迁对应辐射的9,192,631,770个周期的持续时间。 第一个精确的原子钟便是根据铯-133的迁越制成的

Caesium是什么

全方位的 精准的 iOS 应用性能分析工具

Caesium

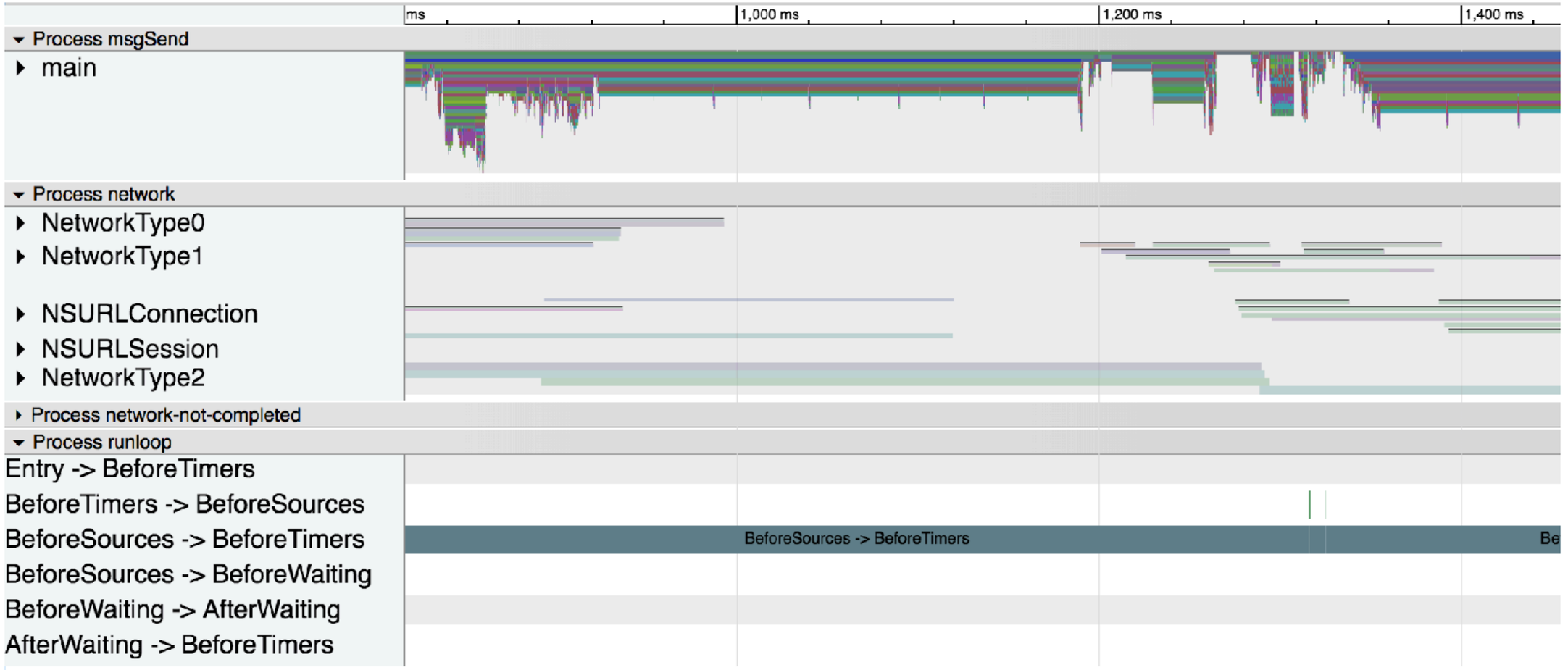
功能

方法调用/网络/Runloop可视化

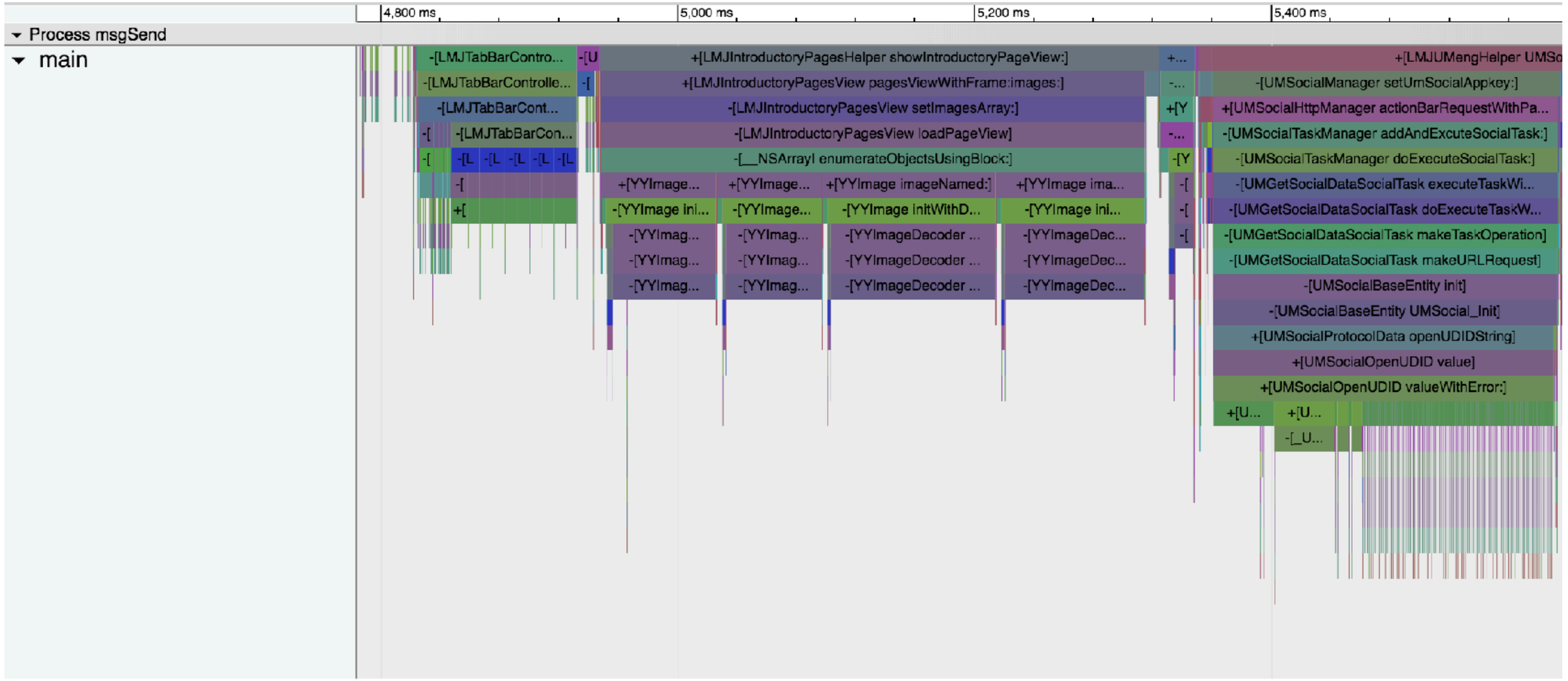
数据可对比

A Glance

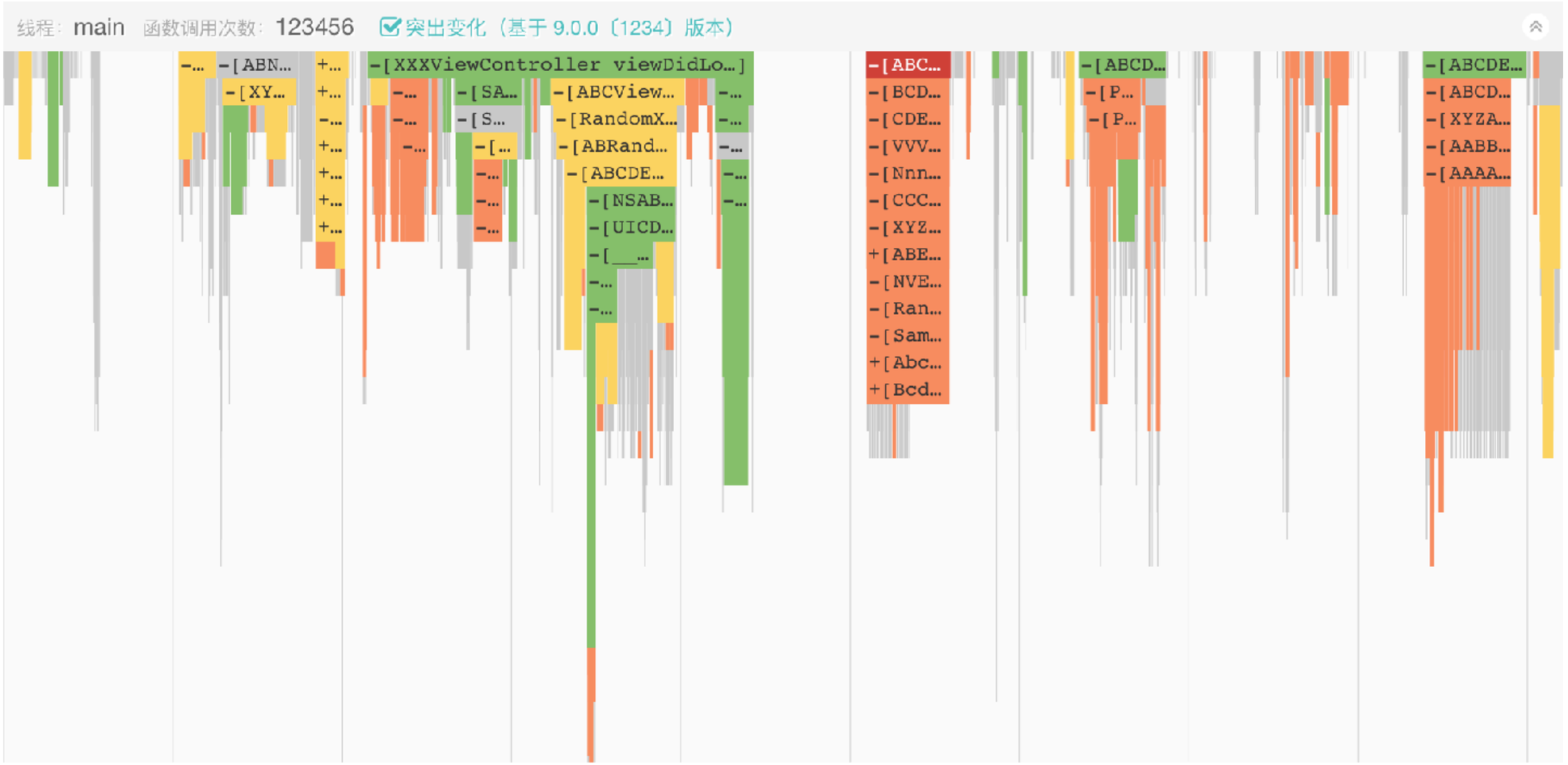
A Glance


ms
1,000 ms
1,200 ms
1,400 ms
Process msgSend
main
Process network
NetworkType0
NetworkType1
NSURLConnection
NSURLSession
NetworkType2
Process network-not-completed
Process runloop
Entry -> BeforeTimers
BeforeTimers -> BeforeSources
BeforeSources -> BeforeTimers
BeforeSources -> BeforeWaiting
BeforeWaiting -> AfterWaiting
AfterWaiting -> BeforeTimers
BeforeSources -> BeforeTimers
Be

A Glance


4,800 ms
5,000 ms
5,200 ms
5,400 ms
Process msgSend
main
-[LMJTabBarContro...
-[U
+[LMJIntroductoryPagesHelper showIntroductoryPageView:]
+...
+[LMJUMengHelper UMSo
-[LMJTabBarControlle...
-[
+[LMJIntroductoryPagesView pagesViewWithFrame:images:]
-...
-[UMSocialManager setUmSocialAppkey:]
-[LMJTabBarCont...
-[LMJIntroductoryPagesView setImagesArray:]
+[Y
+[UMSocialHttpManager actionBarRequestWithPa...
-[
-[LMJTabBarCon...
-[LMJIntroductoryPagesView loadPageView]
-...
-[UMSocialTaskManager addAndExcuteSocialTask:]
-[
-[L
-[L
-[L
-[L
-[L
-[__NSArrayI enumerateObjectsUsingBlock:]
-[Y
-[UMSocialTaskManager doExecuteSocialTask:]
-[
+[YYImage...
+[YYImage...
+[YYImage imageNamed:]
+[YYImage ima...
-[
-[UMGetSocialDataSocialTask executeTaskWi...
+[
-[YYImage ini...
-[YYImage...
-[YYImage initWithD...
-[YYImage ini...
-[
-[UMGetSocialDataSocialTask doExecuteTaskW...
-[YYImag...
-[YYImag...
-[YYImageDecoder ...
-[YYImageDec...
-[
-[UMGetSocialDataSocialTask makeTaskOperation]
-[YYImag...
-[YYImag...
-[YYImageDecoder ...
-[YYImageDec...
-[UMGetSocialDataSocialTask makeURLRequest]
-[YYImag...
-[YYImag...
-[YYImageDecoder ...
-[YYImageDec...
-[UMSocialBaseEntity init]
-[UMSocialBaseEntity UMSocial_Init]
+[UMSocialProtocolData openUDIDString]
+[UMSocialOpenUDID value]
+[UMSocialOpenUDID valueWithError:]
+[U...
+[U...
-[_U...

A Glance


线程: main
函数调用次数: 123456
突出变化（基于 9.0.0（1234）版本）
-[XXXViewController viewDidLo…]
-[ABN…
-[XY…
-[SA…
-[S…
-[ABCView…
-[RandomX…
-[ABRand…
-[ABCDE…
-[NSAB…
-[UICD…
-[ABC…
-[BCD…
-[CDE…
-[VVV…
-[Nnn…
-[CCC…
-[XYZ…
+[ABE…
-[NVE…
-[Ran…
-[Sam…
+[Abc…
+[Bcd…
-[ABCD…
-[P…
-[P…
-[ABCDE…
-[ABCD…
-[XYZA…
-[AABB…
-[AAAA…

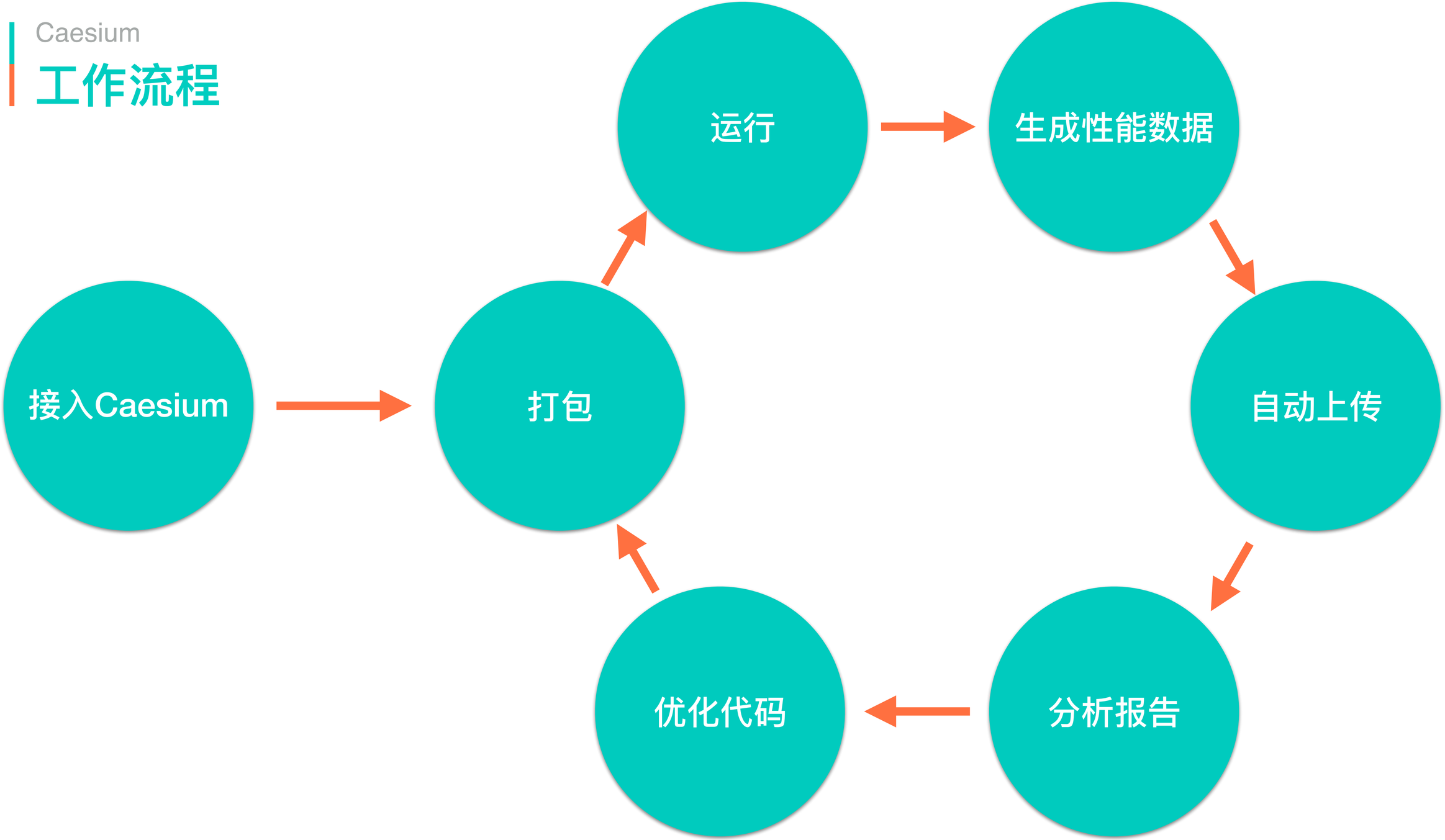
A Glance

函数名	总耗时		耗时变化
-[Note allTheMethodsBelow:areRandomlyGenerated:]	1139.546	338.060	29.7%
-[XXXTask start]	251.215	251.215	100.0%
-[SomeClass doSthWithThis:that:]	234.748	114.282	48.7%
+[SomeManager setup]	99.875	69.069	69.2%
-[SomeClassA doSomething:]	112.148	59.269	52.8%
-[NSNotificationCenter postNotificationName:object:userInfo:]	70.233	40.332	57.4%
+[SomeModule setup]	38.098	38.098	100.0%
-[__NSArrayI enumerateObjectsUsingBlock:]	27.895	27.895	100.0%
-[ABCViewController updateData]	60.682	25.088	41.3%
-[XYZView loadData:]	24.005	24.005	100.0%
-[ALongLongClassName aLongLongMethodName:]	19.546	19.546	100.0%
+[SampleClass showSample]	17.825	17.825	100.0%
-[RandomRandom generateSomethingRandom:]	15.595	15.595	100.0%
-[HahahaManager manageSomething]	15.407	15.407	100.0%
-[SomeStrangeView setData:]	14.040	14.040	100.0%
-[SampleObject reloadAllData]	54.862	13.716	25.0%
-[XXXViewController doSthSlow:]	55.612	13.715	24.7%
+[HugeSingleton sharedInstance]	13.630	13.630	100.0%
-[AAATableViewController tableView:cellForRowAtIndexPath:]	12.994	12.994	100.0%
-[ThisIsAModule setupModule]	12.779	12.779	100.0%
-[ABCDemoManager manageSth:finished:]	12.579	12.579	100.0%
-[AnotherSampleItem initWithType:]	57.063	12.447	21.8%
-[AhaSection init]	57.064	12.447	21.8%
-[UnknownDataObject whatIsThis:andThat:]	73.785	12.017	16.3%
-[NSData initWithThis:]	11.508	11.508	100.0%
-[NSArray initWithMood:]	11.309	11.309	100.0%
-[Wow nice:]	10.150	10.150	100.0%
+[RandomString randomRandom:]	31.878	10.026	31.5%
-[AlmostAnModule loadDataIfNeeded]	18.790	8.128	43.3%
-[FakeClass printEverything]	20.034	6.828	34.1%

» 显示更多

Caesium

工作流程



实现方案

```
[someObj someMessage];
```

```
[XXRecordSystem recordStartWithClass:get_class(someObj)
                                 cmd:@"someMessage"
                                time:currentTime];

[someObj someMessage];

[XXRecordSystem recordEndWithClass:get_class(someObj)
                               cmd:@"someMessage"
                              time:currentTime];
```

```
[someObj1 someMessage1];
[someObj2 someMessage2];
[someObj3 someMessage3];
[someObj4 someMessage4];
[someObj5 someMessage5];
[someObj6 someMessage6];
[someObj7 someMessage7];
[someObj8 someMessage8];
...
```

```
objc_msgSend(id self, SEL _cmd, ...)
```

替换objc_msgSend

objc_msgSend

new_objc_msgSend 

1. 记录 class, cmd, startTime
2. 调用 orig_objc_msgSend
3. 记录 endTime

替换objc_msgSend

objc_msgSend 调用约定

寄存器 r0, r1 ... 栈

new_objc_msgSend

汇编

new_objc_msgSend

保存寄存器 x0 - x8, q0 - q7 到栈上

记录 class, cmd, startTime

恢复寄存器 x0 - x8, q0 - q7

调用原始 objc_msgSend

记录 endTime

……？？？

new_objc_msgSend

lr 寄存器（Link Register）

子方法执行完毕后将会被执行的地址

new_objc_msgSend

保存寄存器 x0 - x8, q0 - q7 和 lr 到栈上

记录 class, cmd, startTime

恢复寄存器 x0 - x8, q0 - q7, 以及 lr

调用原始 objc_msgSend，记录 lr

保存寄存器 x0 - x8, q0 - q7 和 lr 到栈上

记录 endTime

恢复寄存器 x0 - x8, q0 - q7, 以及 lr

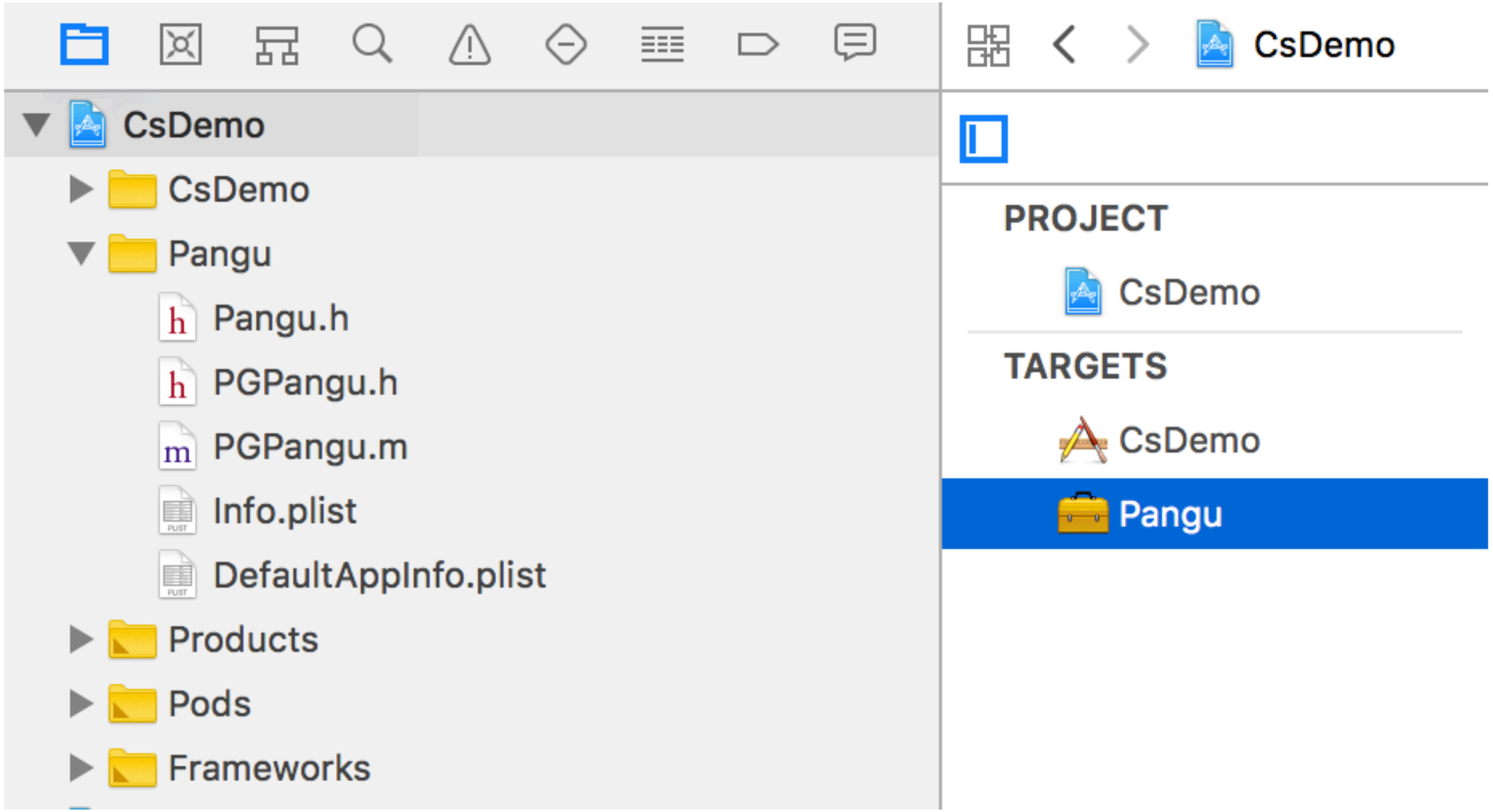
替换时机

+ (void)load

调用顺序

Pangu

替换objc_msgSend



替换objc_msgSend

dyld, Mach-O

fishhook

Tracing Network

Swizzle Network Request

Empty Record

开始时间

类别

地址

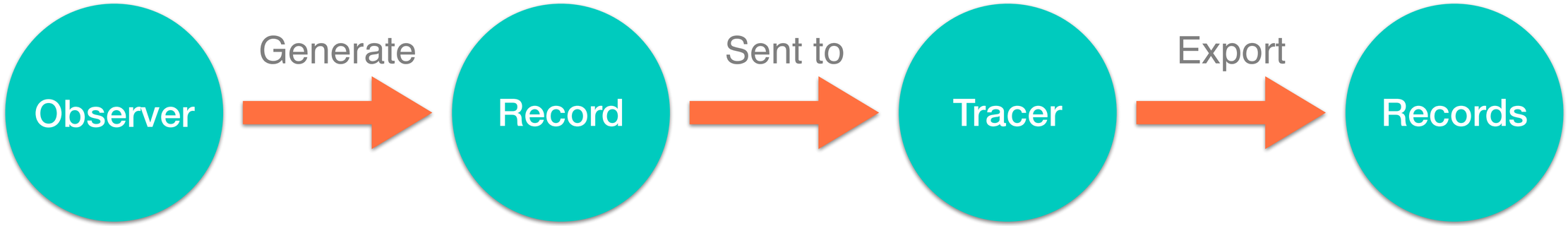
参数

记录时间

记录时间

Full Record

Tracing Runloop



Caesium vs 传统方案

优势

01 数据收集成本低

02 无侵入

03 信息全面

04 数据展示直观

05 体系化

06 对性能影响小

优势

01 数据收集成本低

02 无侵入

03 信息全面

04 数据展示直观

05 体系化

06 对性能影响小

优势

01 数据收集成本低

02 无侵入

03 信息全面

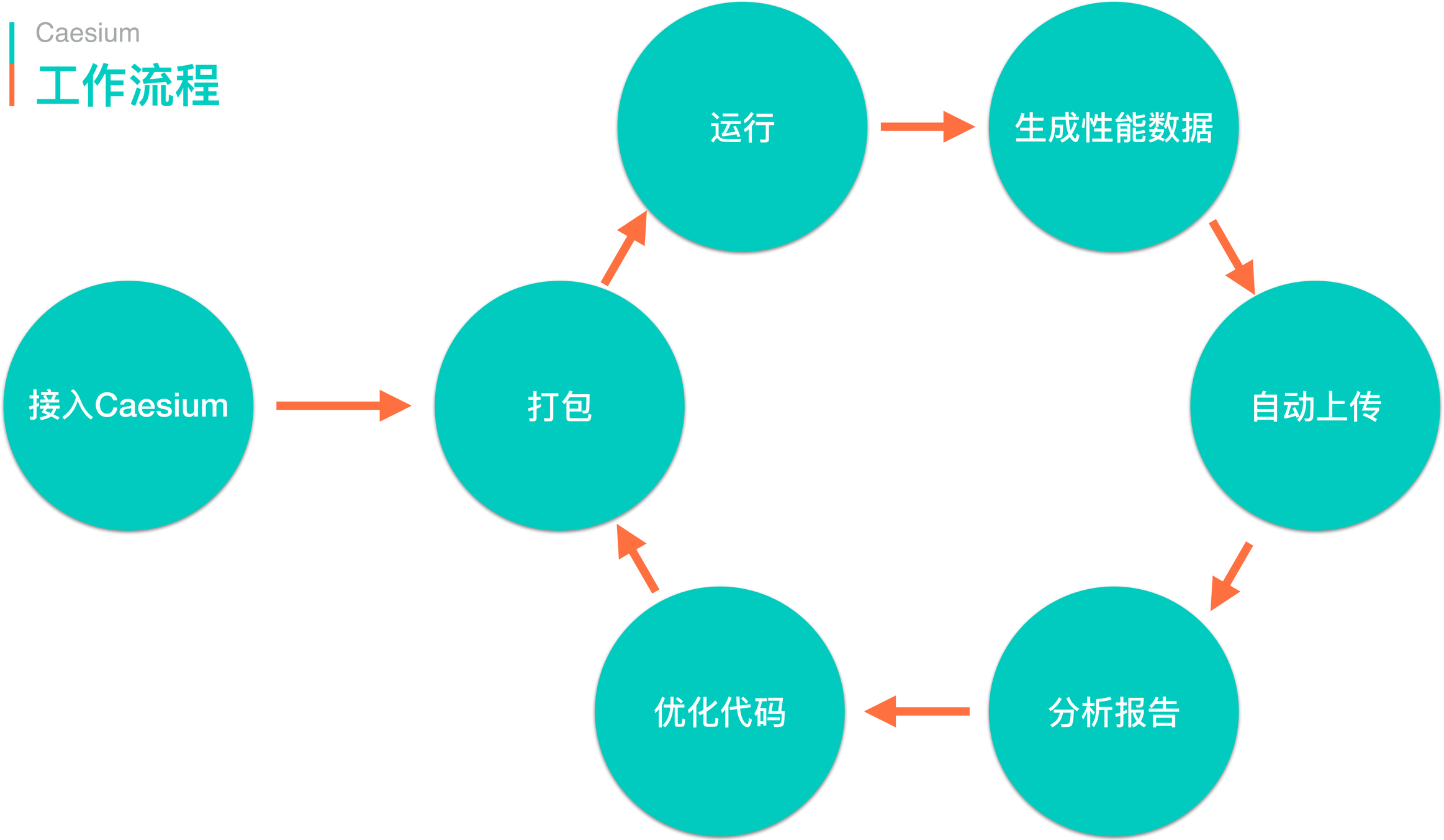
04 数据展示直观

05 体系化

06 对性能影响小

Caesium

工作流程



优势

01 数据收集成本低

02 无侵入

03 信息全面

04 数据展示直观

05 体系化

06 对性能影响小

优势

01 数据收集成本低

02 无侵入

03 信息全面

04 数据展示直观

05 体系化

06 对性能影响小

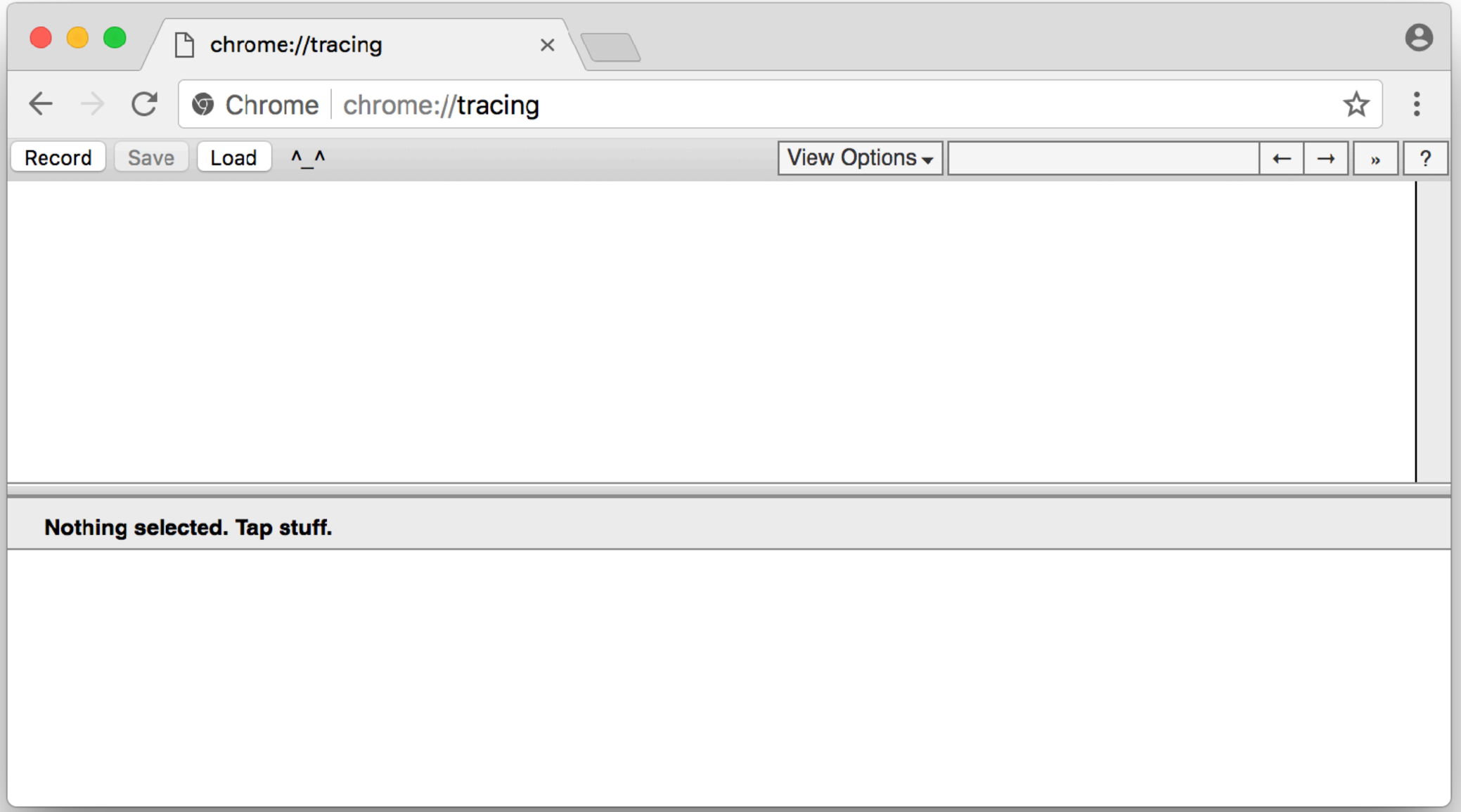
本地结果展示

Chrome 性能展示工具

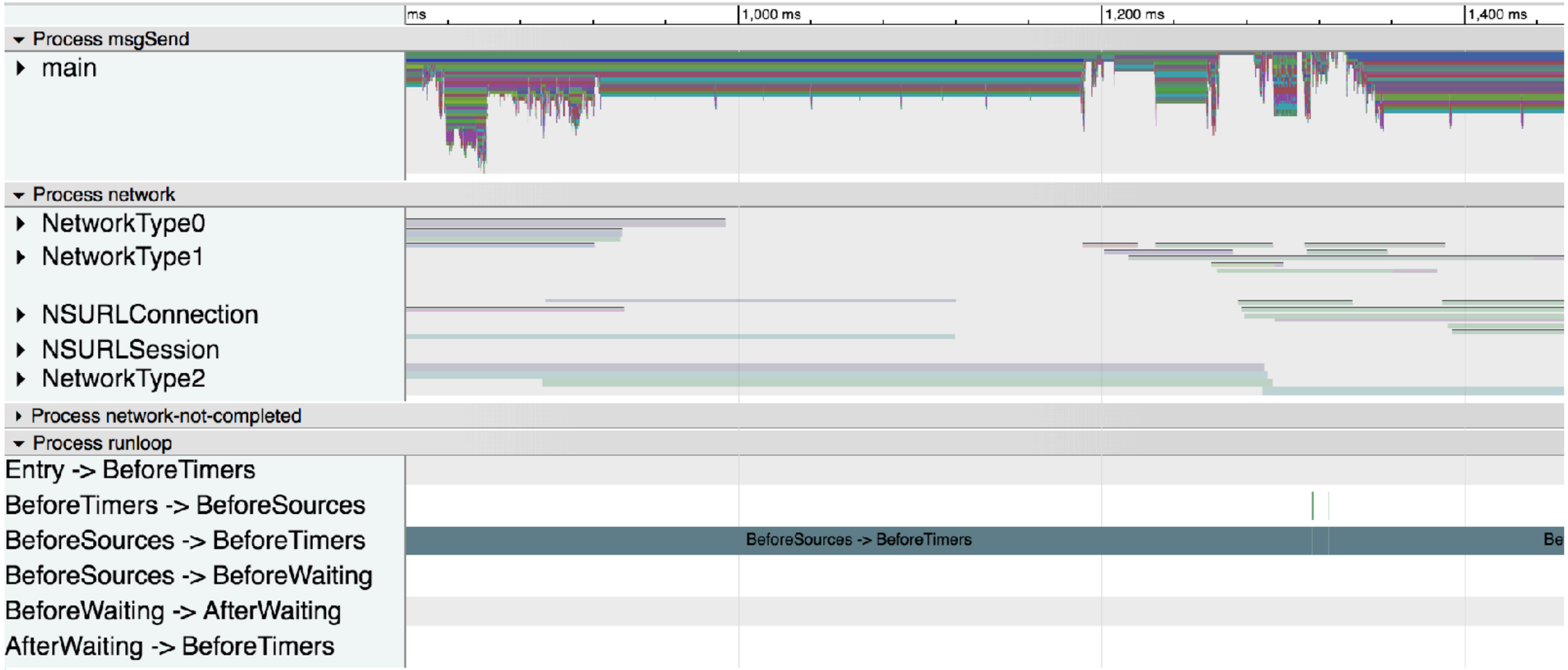
chrome://tracing

标准 JSON 格式

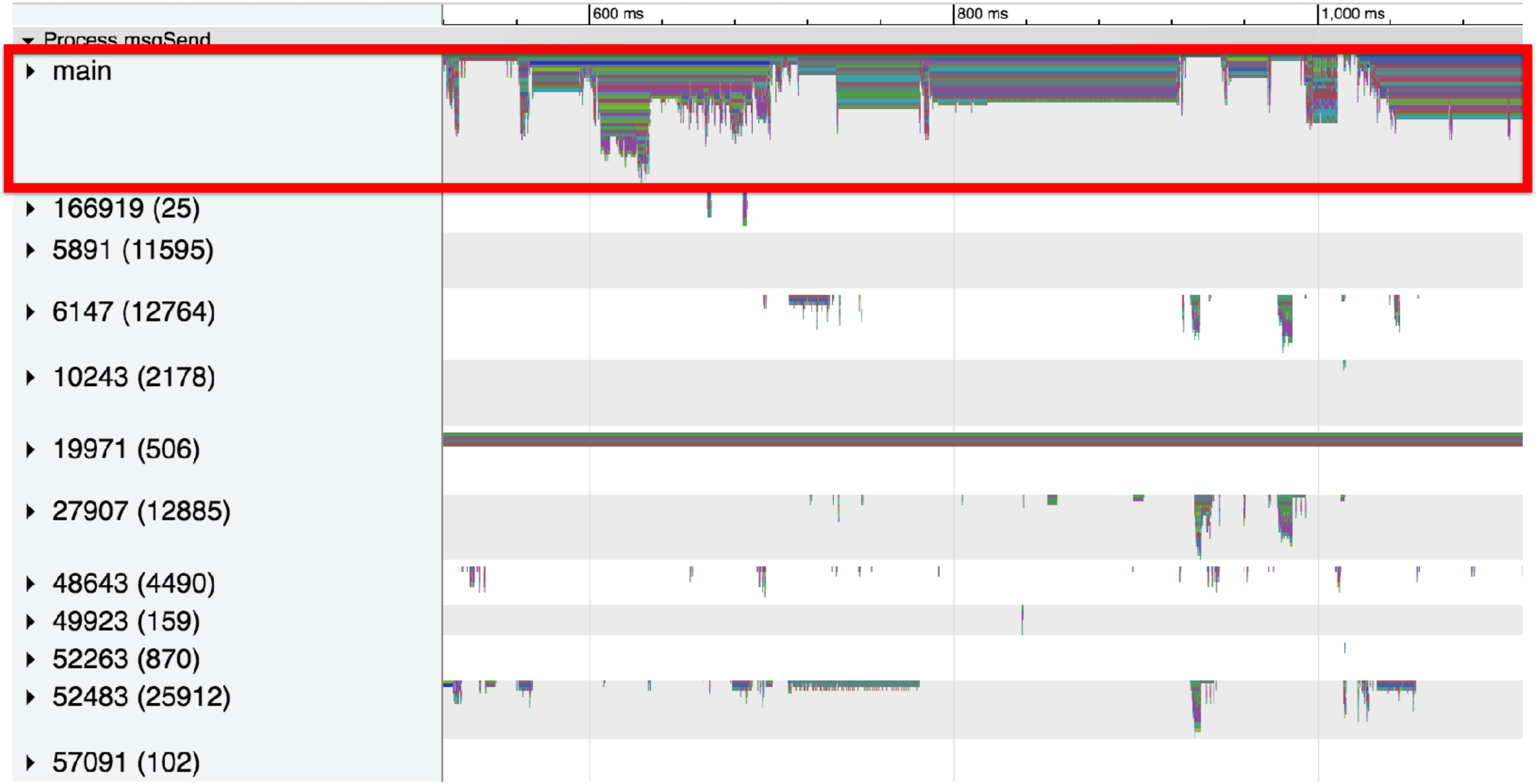
本地结果展示


chrome://tracing
Chrome
chrome://tracing
Record
Save
Load
^_^
View Options
Nothing selected. Tap stuff.

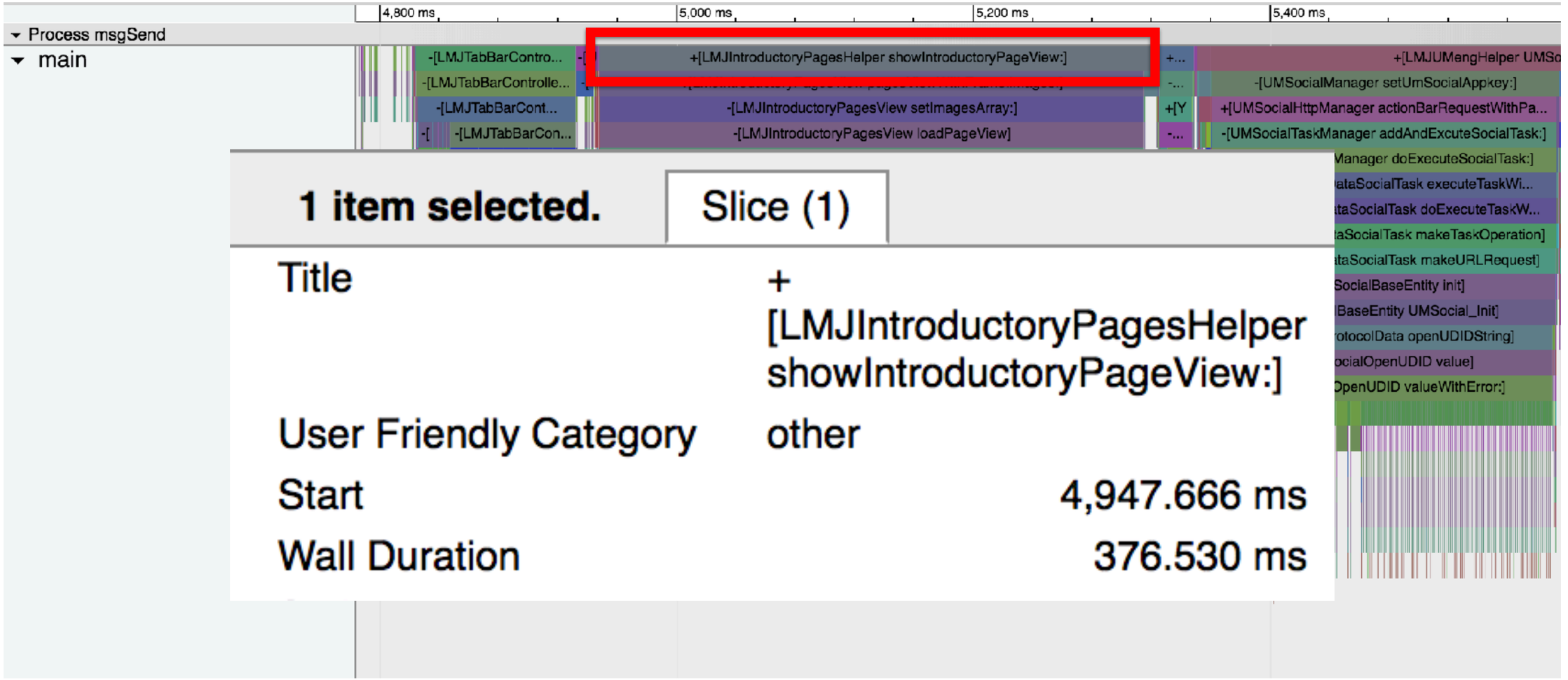
本地结果展示


ms
1,000 ms
1,200 ms
1,400 ms
Process msgSend
main
Process network
NetworkType0
NetworkType1
NSURLConnection
NSURLSession
NetworkType2
Process network-not-completed
Process runloop
Entry -> BeforeTimers
BeforeTimers -> BeforeSources
BeforeSources -> BeforeTimers
BeforeSources -> BeforeWaiting
BeforeWaiting -> AfterWaiting
AfterWaiting -> BeforeTimers
BeforeSources -> BeforeTimers
Be

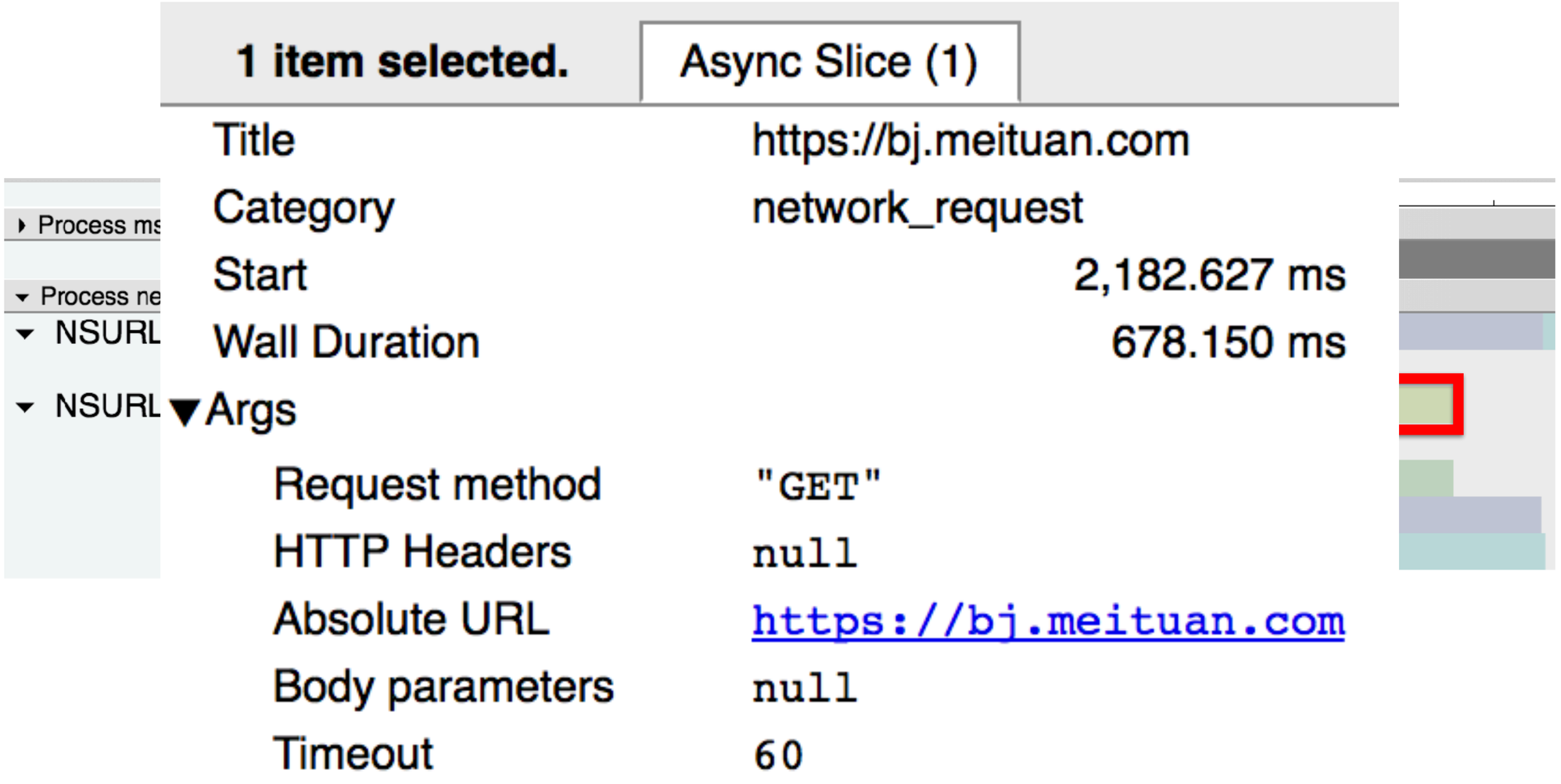
本地结果展示


600 ms
800 ms
1,000 ms
Process msgSend
main
166919 (25)
5891 (11595)
6147 (12764)
10243 (2178)
19971 (506)
27907 (12885)
48643 (4490)
49923 (159)
52263 (870)
52483 (25912)
57091 (102)

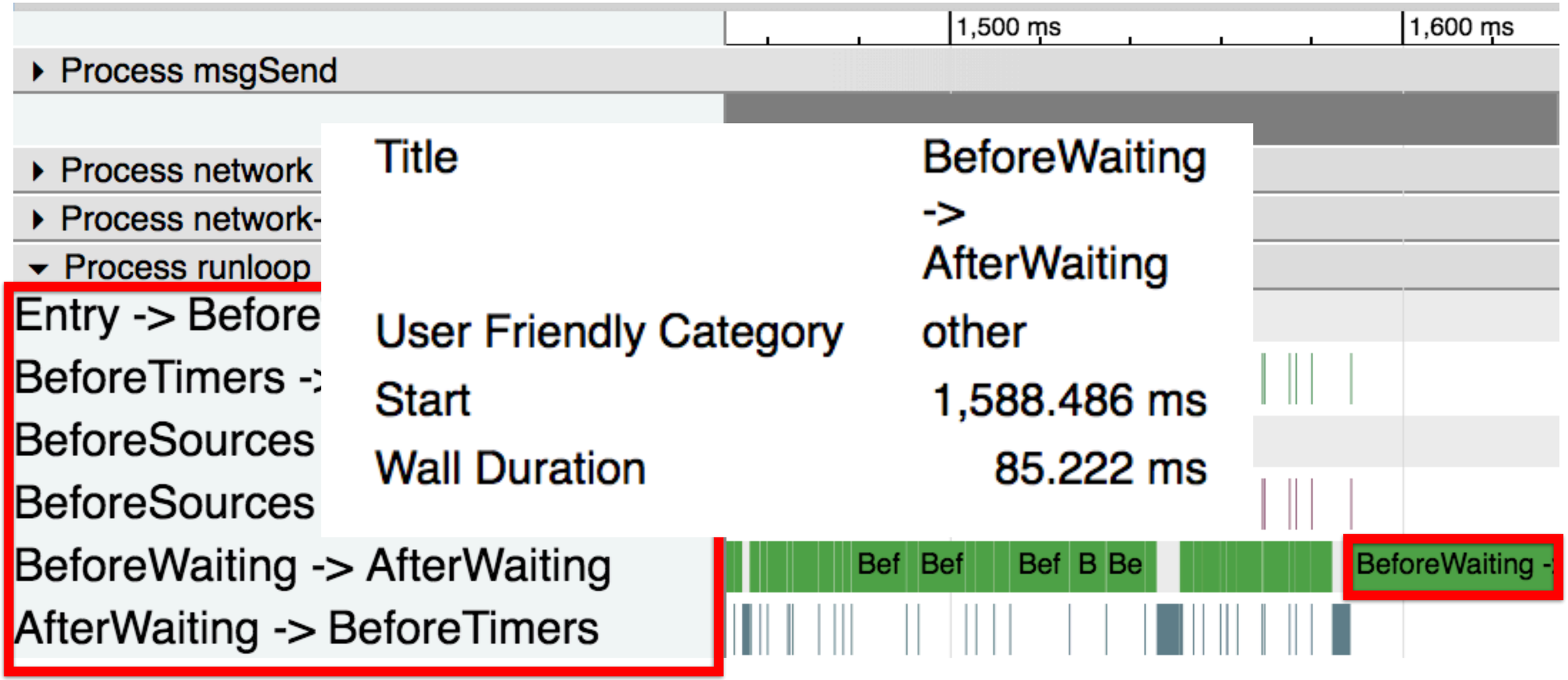
本地结果展示


4,800 ms
5,000 ms
5,200 ms
5,400 ms
Process msgSend
main
-[LMJTabBarContro...
+[LMJIntroductoryPagesHelper showIntroductoryPageView:]
+[LMJUMengHelper UMSo
-[LMJTabBarControlle...
-[UMSocialManager setUmSocialAppkey:]
-[LMJTabBarCont...
-[LMJIntroductoryPagesView setImagesArray:]
+[UMSocialHttpManager actionBarRequestWithPa...
-[LMJTabBarCon...
-[LMJIntroductoryPagesView loadPageView]
-[UMSocialTaskManager addAndExcuteSocialTask:]
1 item selected.
Slice (1)
Title
+ [LMJIntroductoryPagesHelper showIntroductoryPageView:]
User Friendly Category
other
Start
4,947.666 ms
Wall Duration
376.530 ms

本地结果展示


1 item selected.
Async Slice (1)
Title
https://bj.meituan.com
Category
network_request
Start
2,182.627 ms
Wall Duration
678.150 ms
Args
Request method
"GET"
HTTP Headers
null
Absolute URL
https://bj.meituan.com
Body parameters
null
Timeout
60
Process ms
Process ne
NSURL
NSURL

本地结果展示


1,500 ms
1,600 ms
Process msgSend
Process network
Process network-
Process runloop
Entry -> Before
BeforeTimers ->
BeforeSources
BeforeSources
BeforeWaiting -> AfterWaiting
AfterWaiting -> BeforeTimers
Title
BeforeWaiting -> AfterWaiting
User Friendly Category
other
Start
1,588.486 ms
Wall Duration
85.222 ms
Bef
Bef
Bef
B
Be
BeforeWaiting -

优势

01 数据收集成本低

02 无侵入

03 信息全面

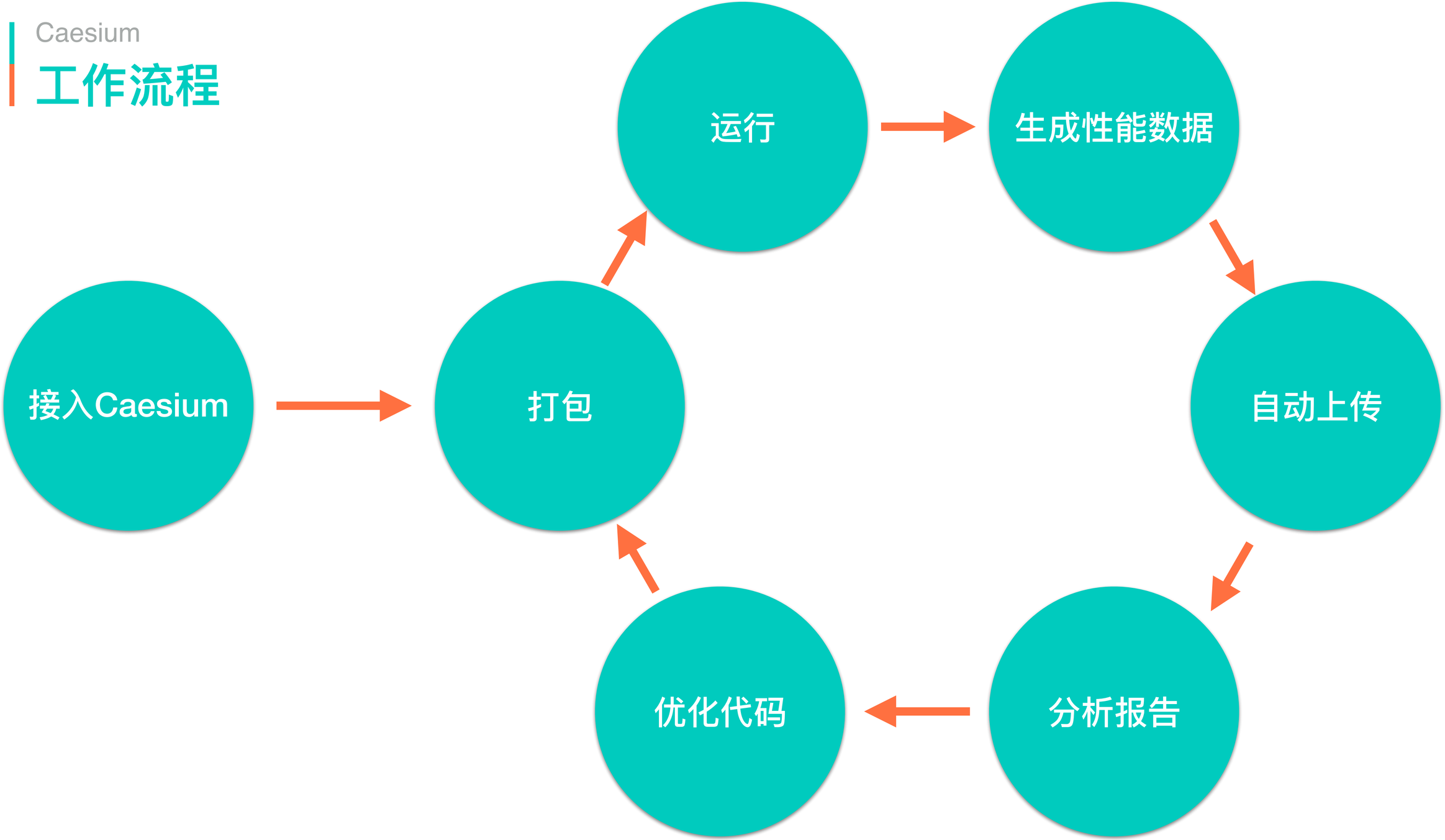
04 数据展示直观

05 体系化

06 对性能影响小

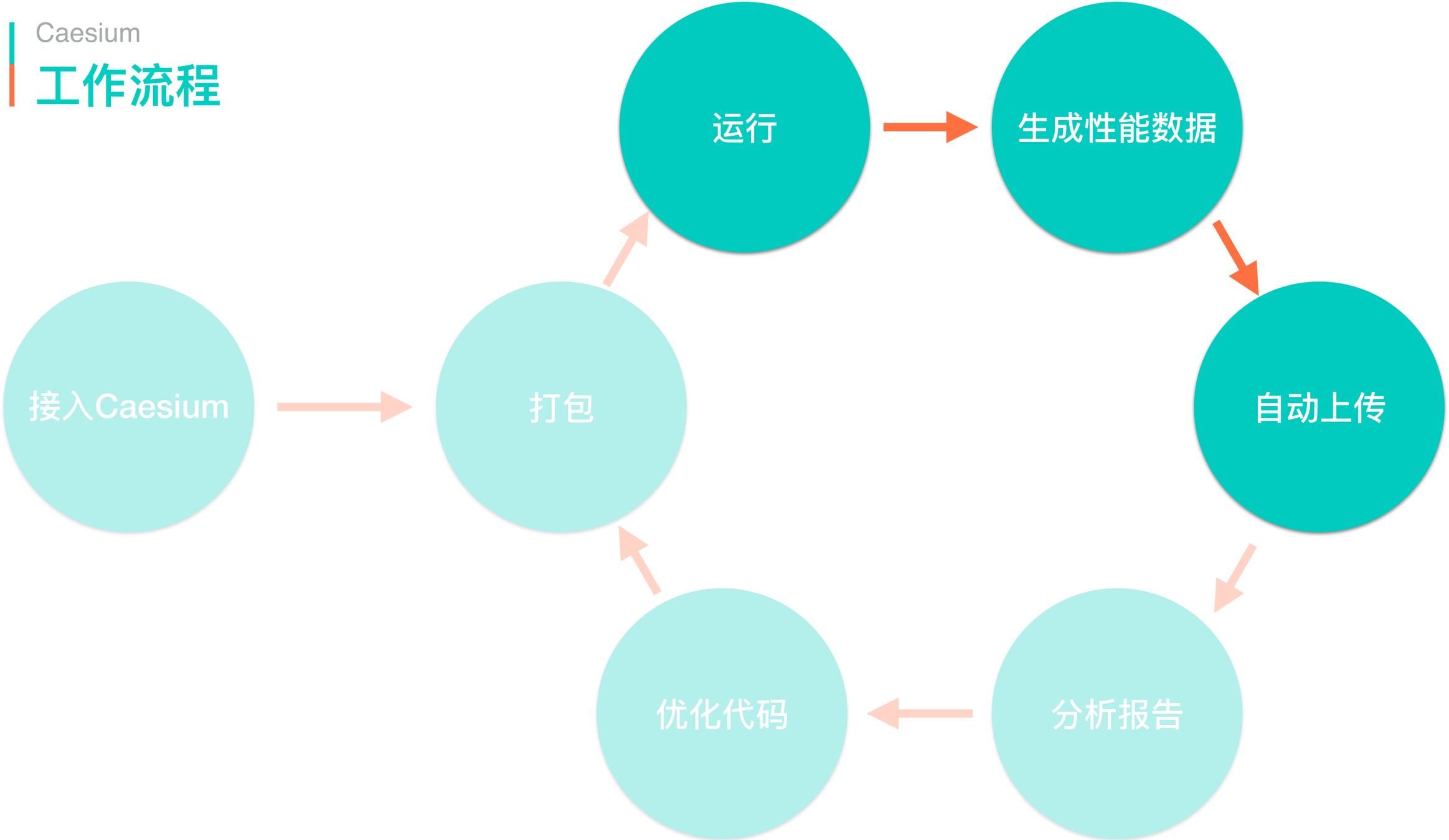
Caesium

工作流程

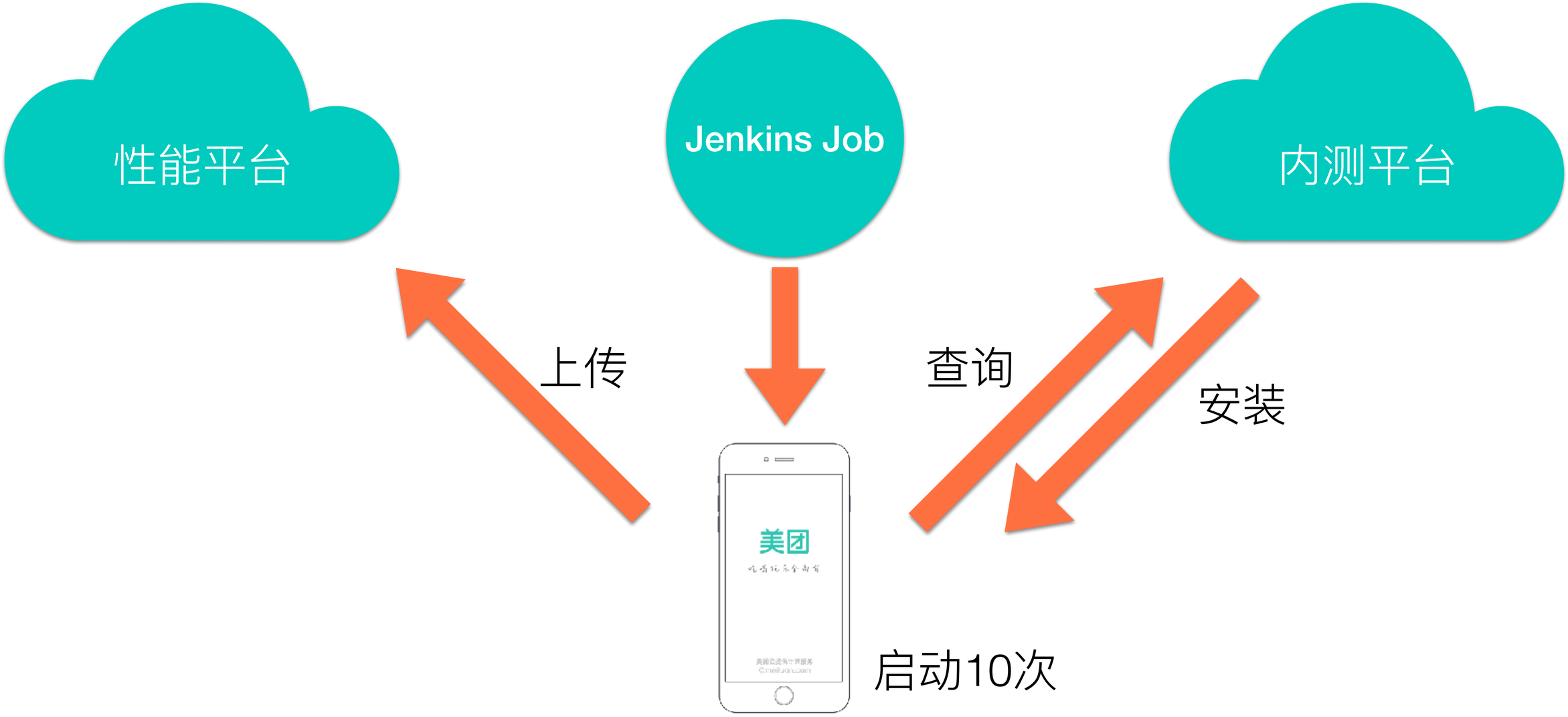


Caesium

工作流程

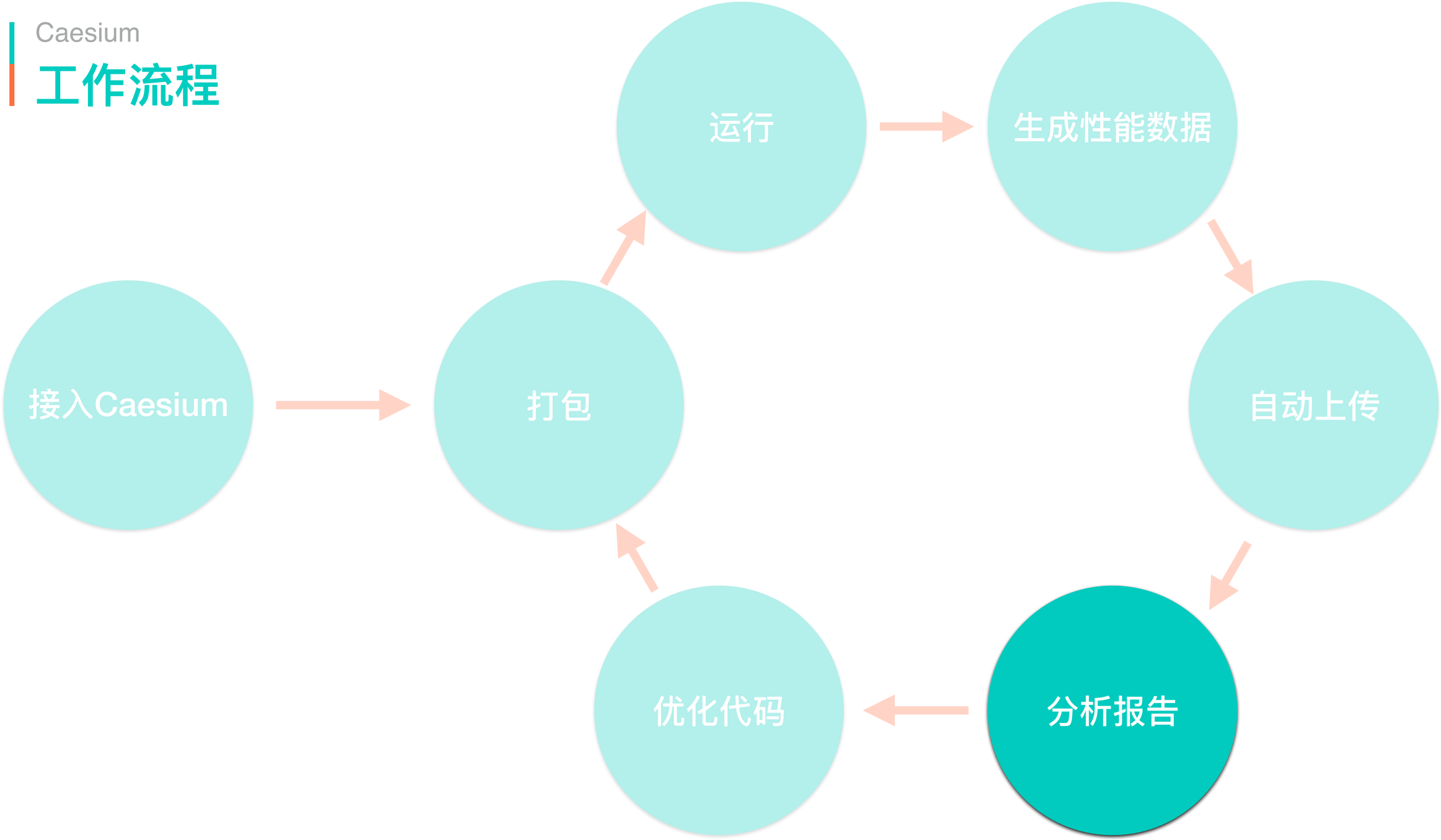


运行

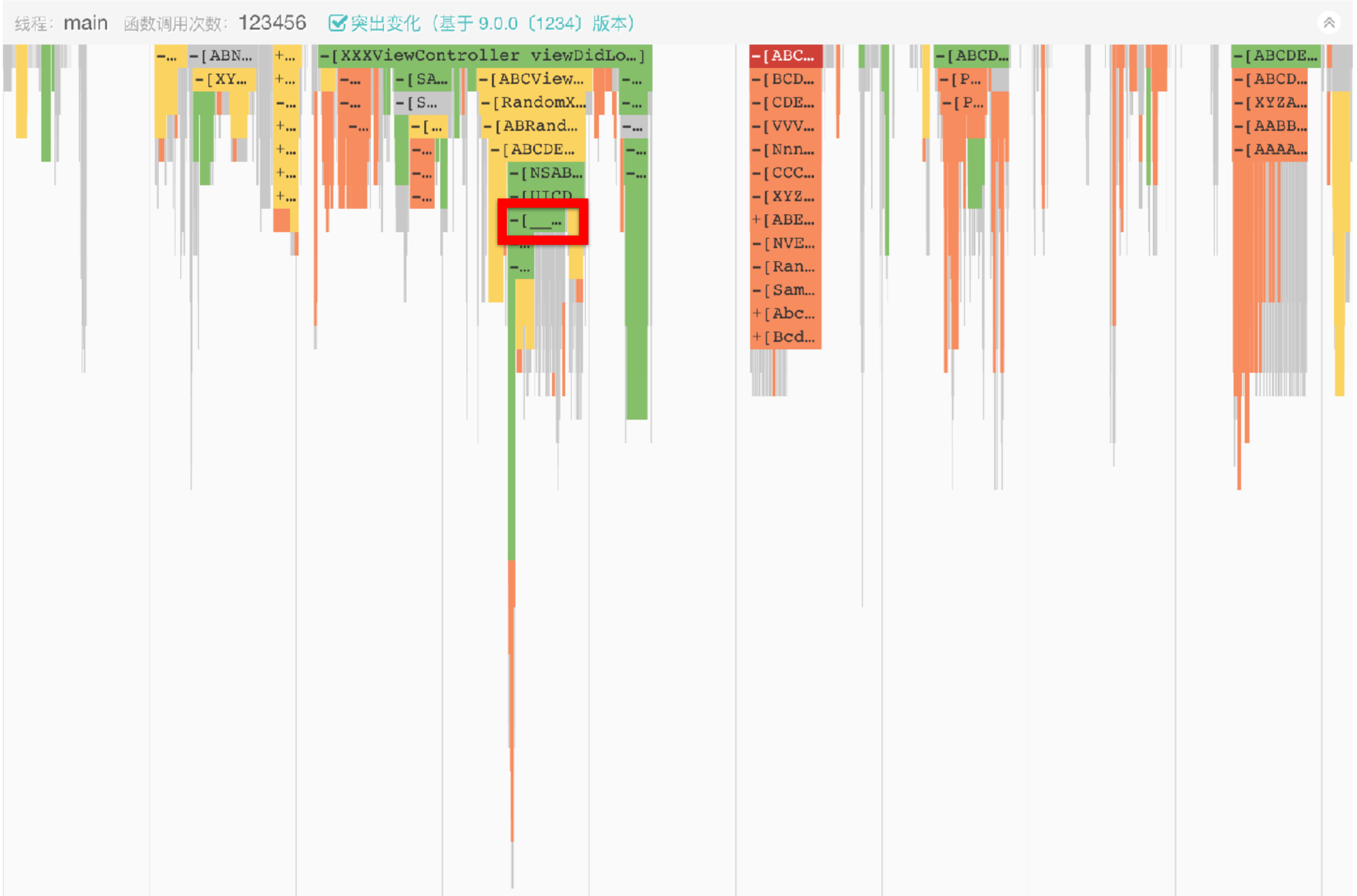


Caesium

工作流程



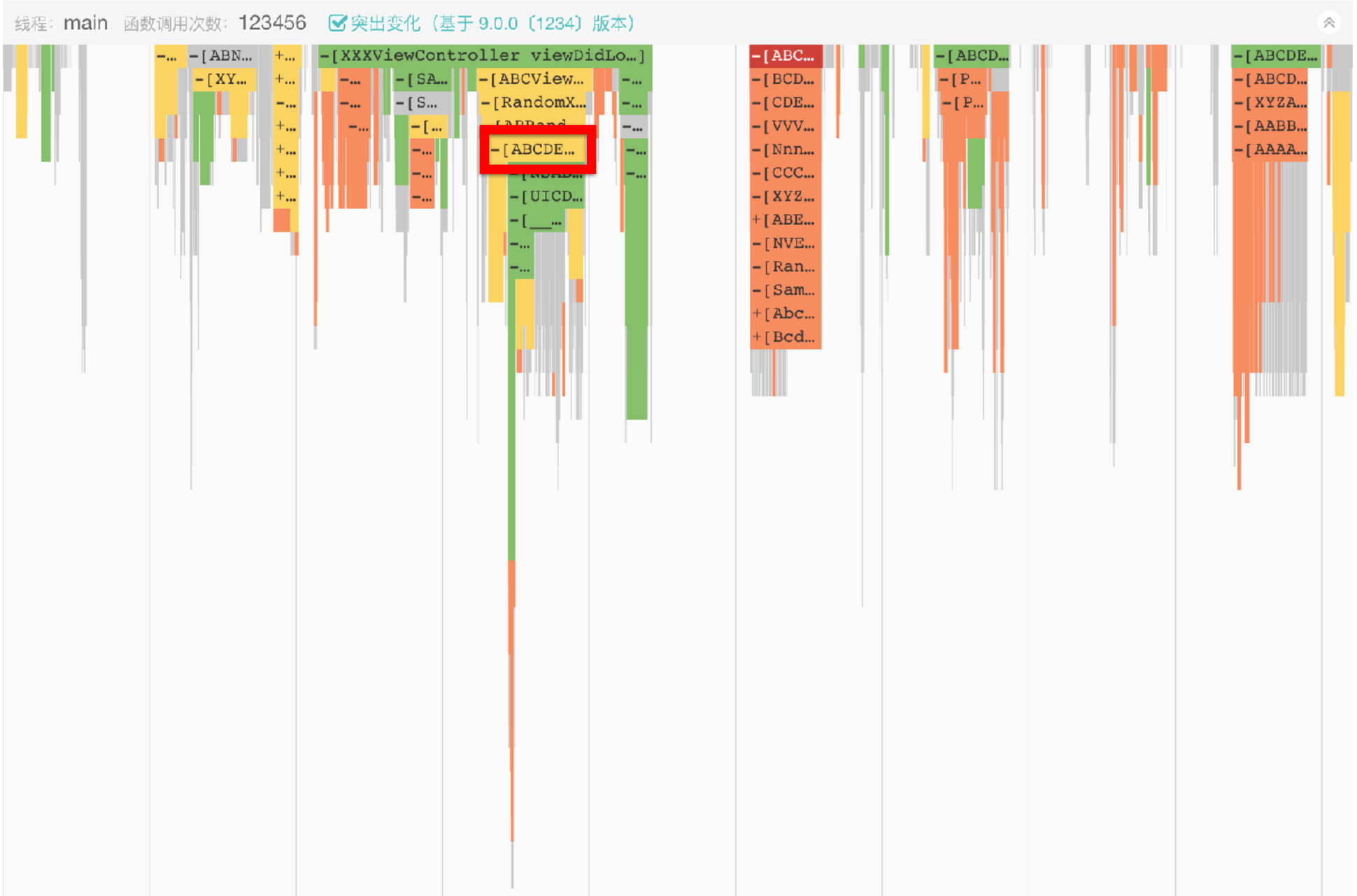
展示变化


线程: main
函数调用次数: 123456
突出变化（基于 9.0.0（1234）版本）
-[XXXViewController viewDidLo...]
-[ABN...
-[XY...
-[SA...
-[S...
-[ABCView...
-[RandomX...
-[ABRand...
-[ABCDE...
-[NSAB...
-[___...
-[ABC...
-[BCD...
-[CDE...
-[VVV...
-[Nnn...
-[CCC...
-[XYZ...
+[ABE...
-[NVE...
-[Ran...
-[Sam...
+[Abc...
+[Bcd...
-[ABCD...
-[P...
-[P...
-[ABCDE...
-[ABCD...
-[XYZA...
-[AABB...
-[AAAA...

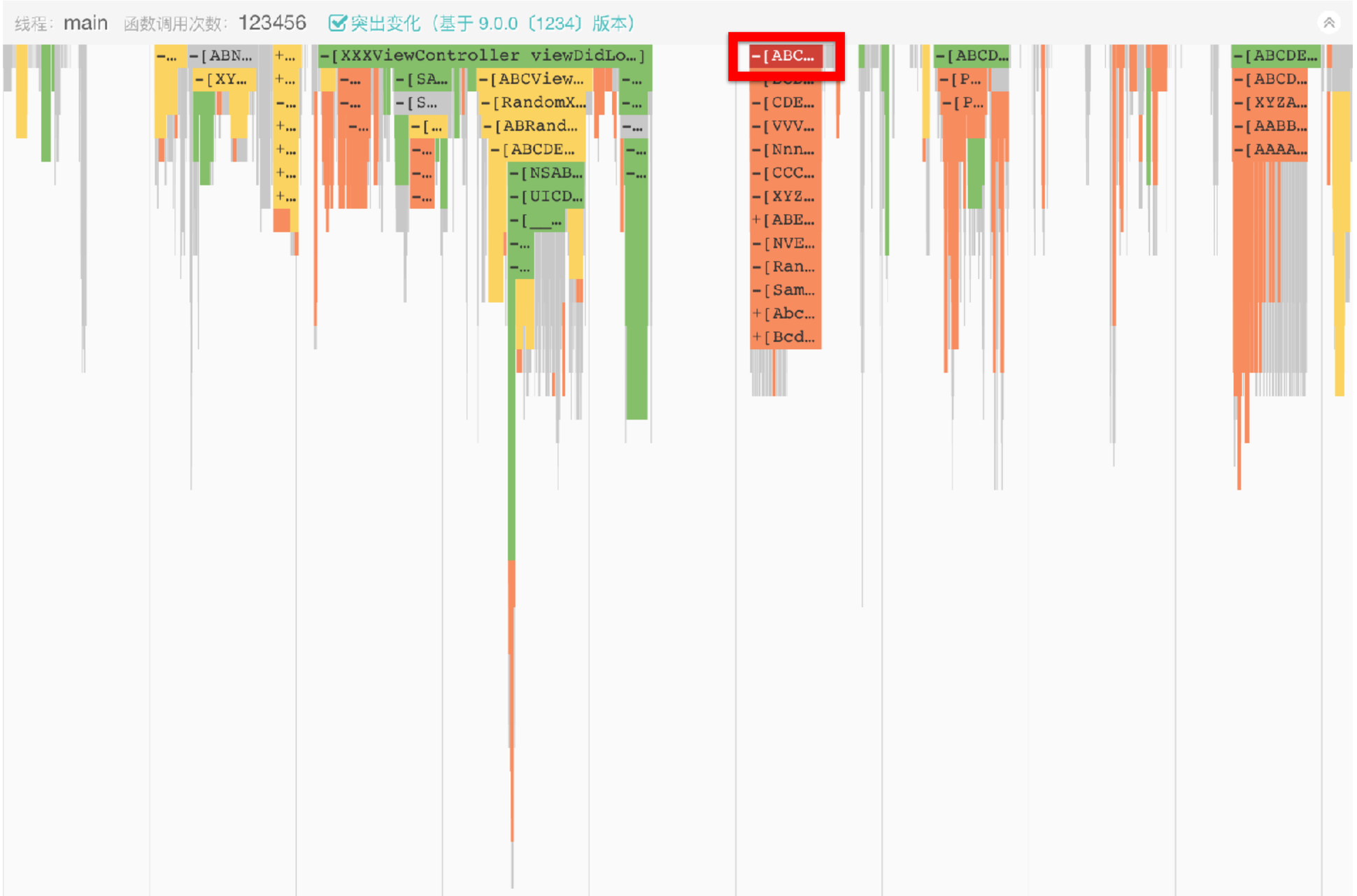
展示变化


线程: main
函数调用次数: 123456
突出变化（基于 9.0.0（1234）版本）
-[XXXViewController viewDidLo...]
-[ABCView...
-[RandomX...
-[ABRand...
-[ABCDE...
-[NSAB...
-[UICD...
-[ABC...
-[BCD...
-[CDE...
-[VVV...
-[Nnn...
-[CCC...
-[XYZ...
+[ABE...
-[ABCD...
-[P...
-[P...
-[ABCDE...
-[ABCD...
-[XYZA...
-[AABB...
-[AAAA...
函数
-[__NSArrayI enumerateObjectsUsingBlock:]
开始 1721.264ms
结束 1920.297ms
时长 199.033ms
变化 ↓ 24.711ms
点击时间段矩形以缩放

展示变化


线程: main
函数调用次数: 123456
突出变化（基于 9.0.0（1234）版本）
-[XXXViewController viewDidLo…]
-[ABCView…
-[RandomX…
-[ABCDE…
-[UICD…
-[ABC…
-[BCD…
-[CDE…
-[VVV…
-[Nnn…
-[CCC…
-[XYZ…
+[ABE…
-[NVE…
-[Ran…
-[Sam…
+[Abc…
+[Bcd…
-[ABCD…
-[P…
-[P…
-[ABCDE…
-[ABCD…
-[XYZA…
-[AABB…
-[AAAA…

展示变化

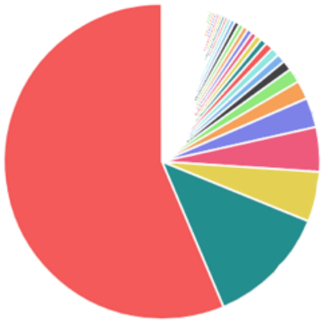

线程: main
函数调用次数: 123456
突出变化（基于 9.0.0（1234）版本）
-[XXXViewController viewDidLo...]
-[ABN...
-[XY...
-[SA...
-[S...
-[ABCView...
-[RandomX...
-[ABRand...
-[ABCDE...
-[NSAB...
-[UICD...
-[ABC...
-[CDE...
-[VVV...
-[Nnn...
-[CCC...
-[XYZ...
+[ABE...
-[NVE...
-[Ran...
-[Sam...
+[Abc...
+[Bcd...
-[ABCD...
-[P...
-[P...
-[ABCDE...
-[ABCD...
-[XYZA...
-[AABB...
-[AAAA...

展示变化

函数名	总耗时		耗时变化
-[Note allTheMethodsBelow:areRandomlyGenerated:]	1139.546	338.060	29.7%
-[XXXTask start]	251.215	251.215	100.0%
-[SomeClass doSthWithThis:that:]	234.748	114.282	48.7%
+[SomeManager setup]	99.875	69.069	69.2%
-[SomeClassA doSomething:]	112.148	59.269	52.8%
-[NSNotificationCenter postNotificationName:object:userInfo:]	70.233	40.332	57.4%
+[SomeModule setup]	38.098	38.098	100.0%
-[__NSArrayI enumerateObjectsUsingBlock:]	27.895	27.895	100.0%
-[ABCViewController updateData]	60.682	25.088	41.3%
-[XYZView loadData:]	24.005	24.005	100.0%
-[ALongLongClassName aLongLongMethodName:]	19.546	19.546	100.0%
+[SampleClass showSample]	17.825	17.825	100.0%
-[RandomRandom generateSomethingRandom:]	15.595	15.595	100.0%
-[HahahaManager manageSomething]	15.407	15.407	100.0%
-[SomeStrangeView setData:]	14.040	14.040	100.0%
-[SampleObject reloadAllData]	54.862	13.716	25.0%
-[XXXViewController doSthSlow:]	55.612	13.715	24.7%
+[HugeSingleton sharedInstance]	13.630	13.630	100.0%
-[AAATableViewController tableView:cellForRowAtIndexPath:]	12.994	12.994	100.0%
-[ThisIsAModule setupModule]	12.779	12.779	100.0%
-[ABCDemoManager manageSth:finished:]	12.579	12.579	100.0%
-[AnotherSampleItem initWithType:]	57.063	12.447	21.8%
-[AhaSection init]	57.064	12.447	21.8%
-[UnknownDataObject whatIsThis:andThat:]	73.785	12.017	16.3%
-[NSData initWithThis:]	11.508	11.508	100.0%
-[NSArray initWithMood:]	11.309	11.309	100.0%
-[Wow nice:]	10.150	10.150	100.0%
+[RandomString randomRandom:]	31.878	10.026	31.5%
-[AlmostAnModule loadDataIfNeeded]	18.790	8.128	43.3%
-[FakeClass printEverything]	20.034	6.828	34.1%

分组件统计

pod库耗时 饼图



pod库名称	所耗时间比(%)
SlowSlowPod	56.36
ABCModule	12.47
AnotherModule	5.12
JustAPod	4.56
YetAnotherPod	3.05
XYZManager	1.89
PowerfulFeature	1.47
LegacyCode	1.01
SomeDatabase	0.9
OurNetworkModule	0.85

1 2 3 4 5

分组件统计

SlowSlowPod内函数耗时 饼图

函数名称	所耗时间比(%)
-[ABCViewController viewDidAppear:]	57.12
-[XXXViewController viewDidLoad]	13.57
-[SomeManager loadData]	4.03
-[XXXCenter runWithThis:andThat:]	3.48
-[RandomDataGenerator giveMeFive]	3.02
-[Coffee explode]	2.37
-[JokingAgent laughWithStyle:volume:]	2.22
-[SuperHero savePlanet:retry:]	1.49
+[TechSalon isExcellent]	1.42
-[EndingViewController viewDidDisappear:]	1.35

1 2 3 4 5

优势

01 数据收集成本低

02 无侵入

03 信息全面

04 数据展示直观

05 体系化

06 对性能影响小

和传统方案的对比

vs Instruments

	Caesium	Instruments
数据收集成本	低	高
时间轴	有	无
结构化数据	是	否
可定制性	深度定制	简单设置
可靠性	高	偶尔卡死
多语言支持	只支持OC	支持多种语言

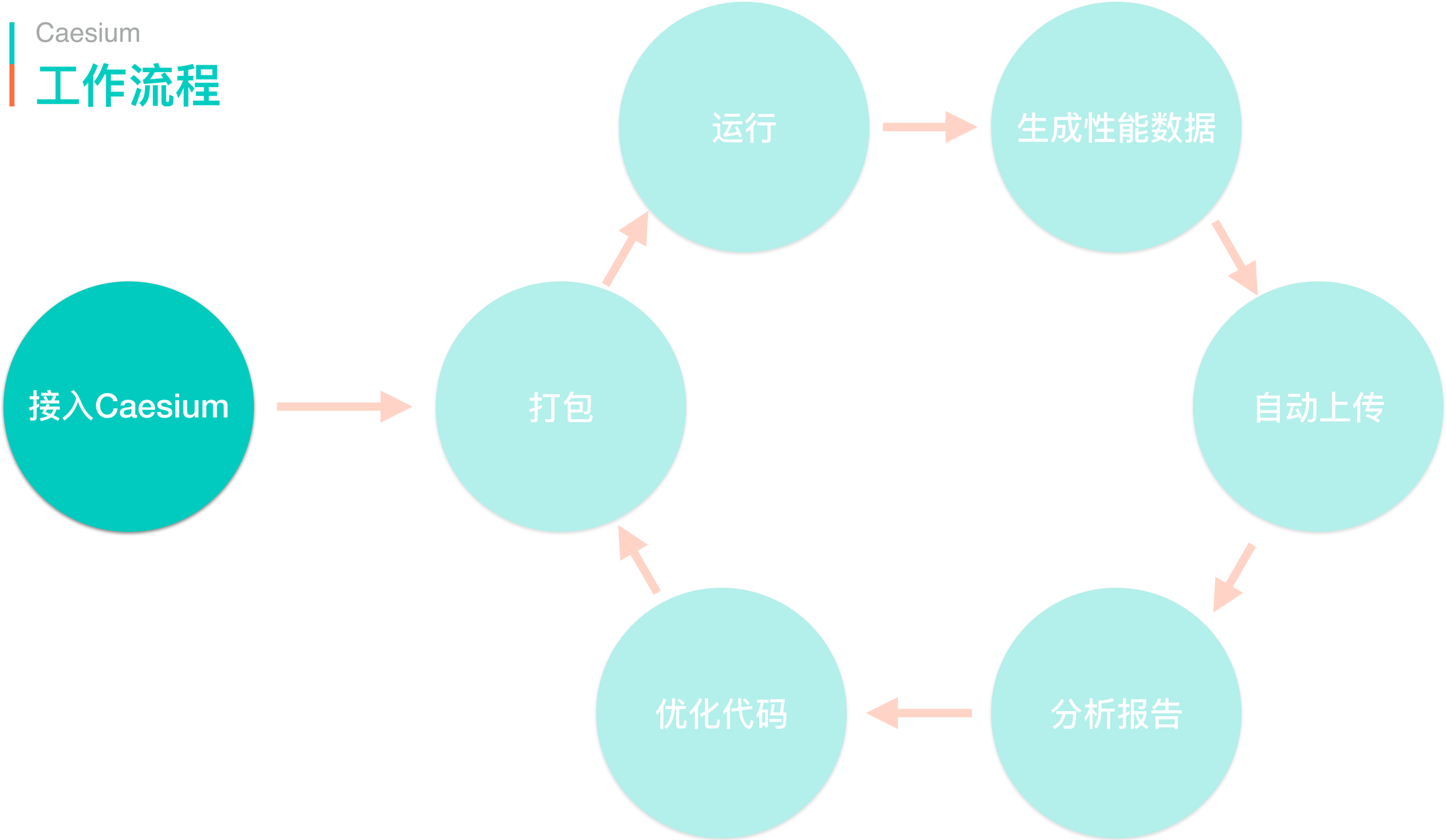
和传统方案的对比

vs 自定义插桩

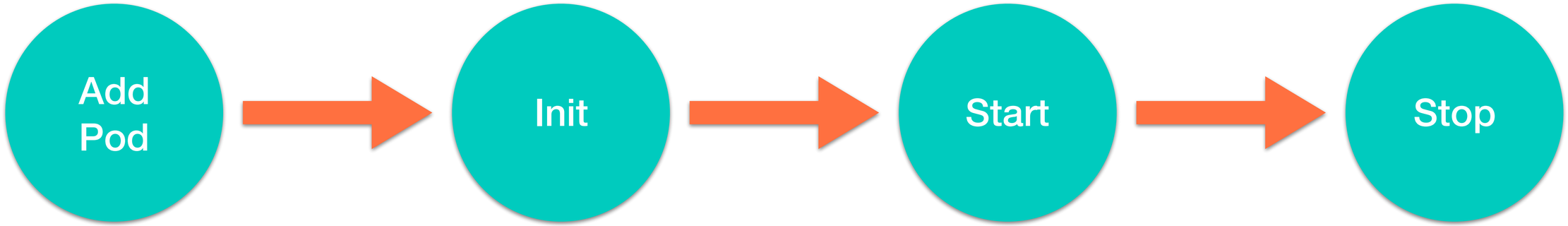
	Caesium	自定义插桩
代码覆盖范围	较全面	有限
侵入性	极低	高
数据收集成本	低	高

Caesium

工作流程



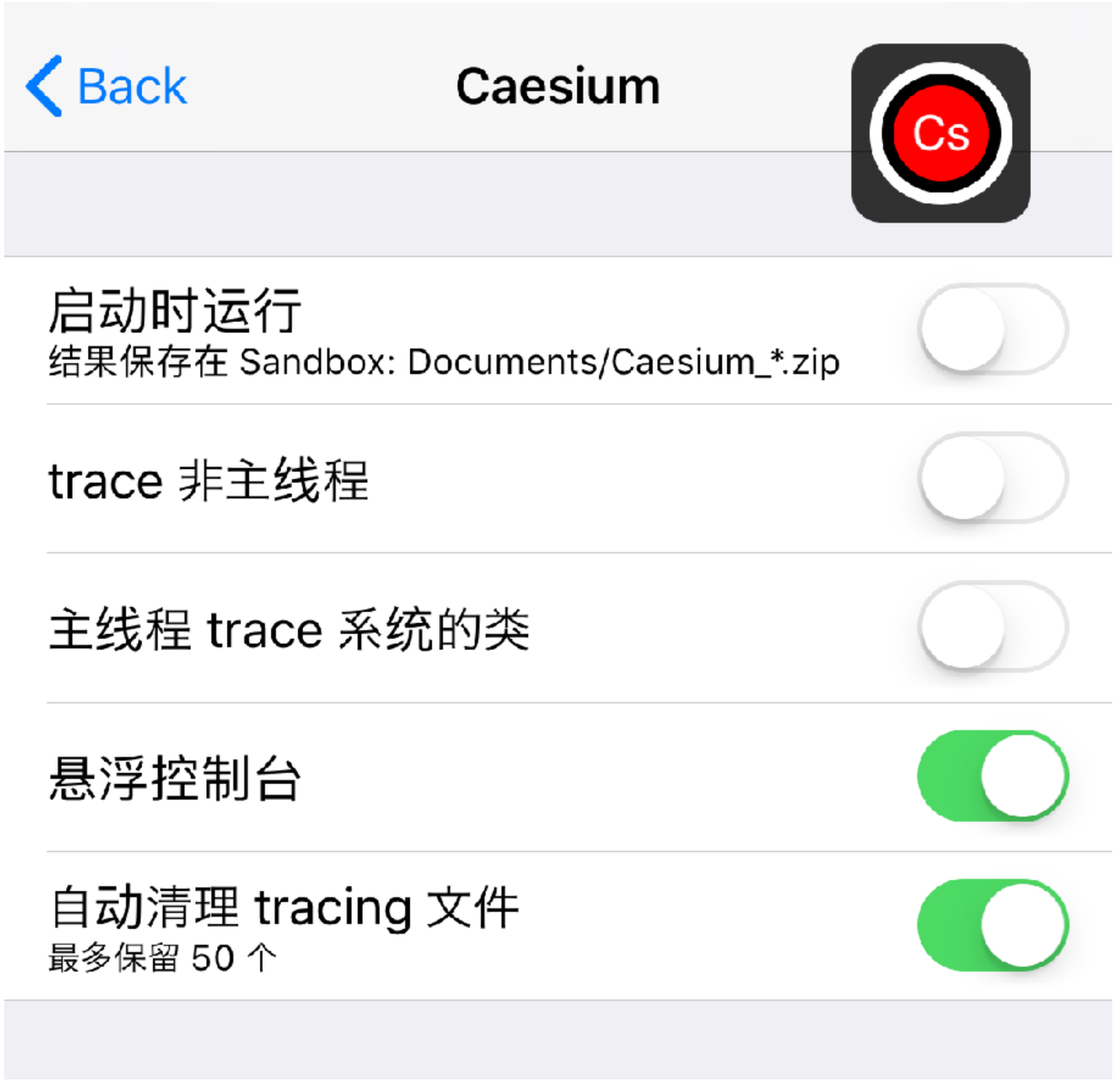
接入流程



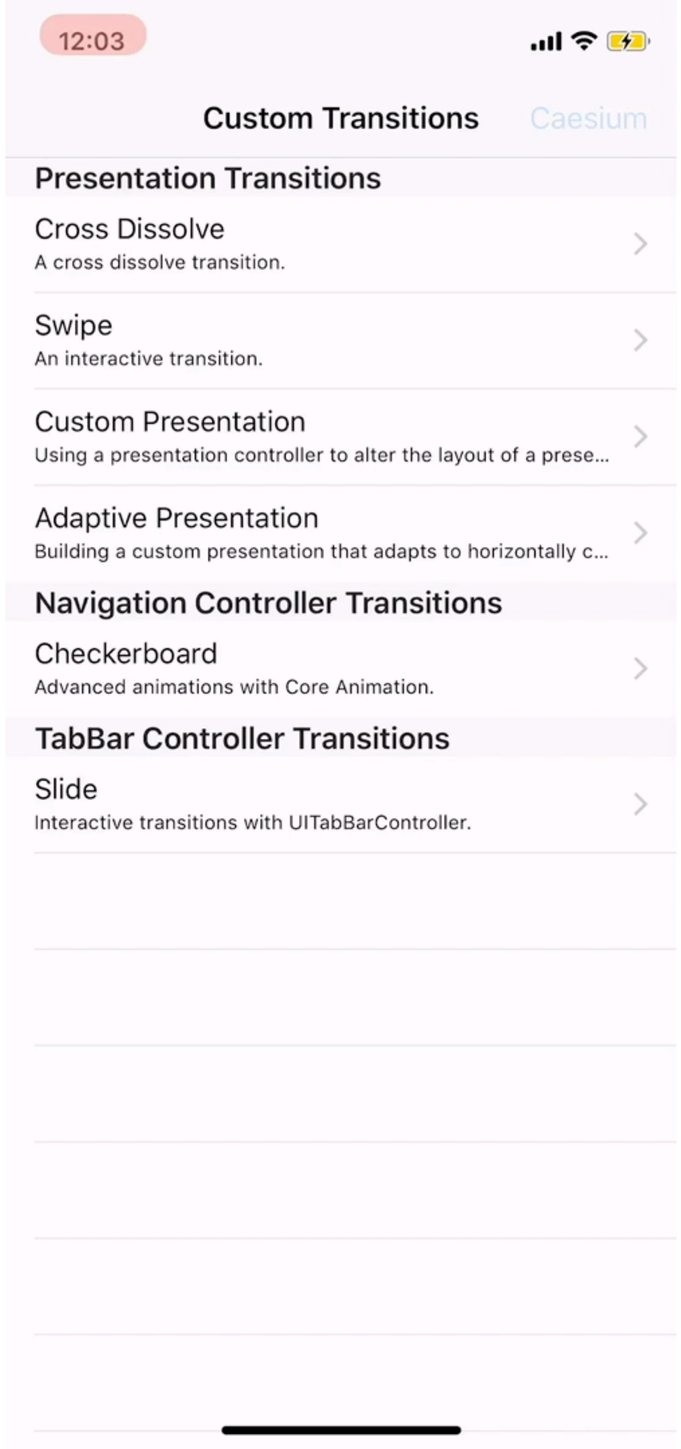
Caesium

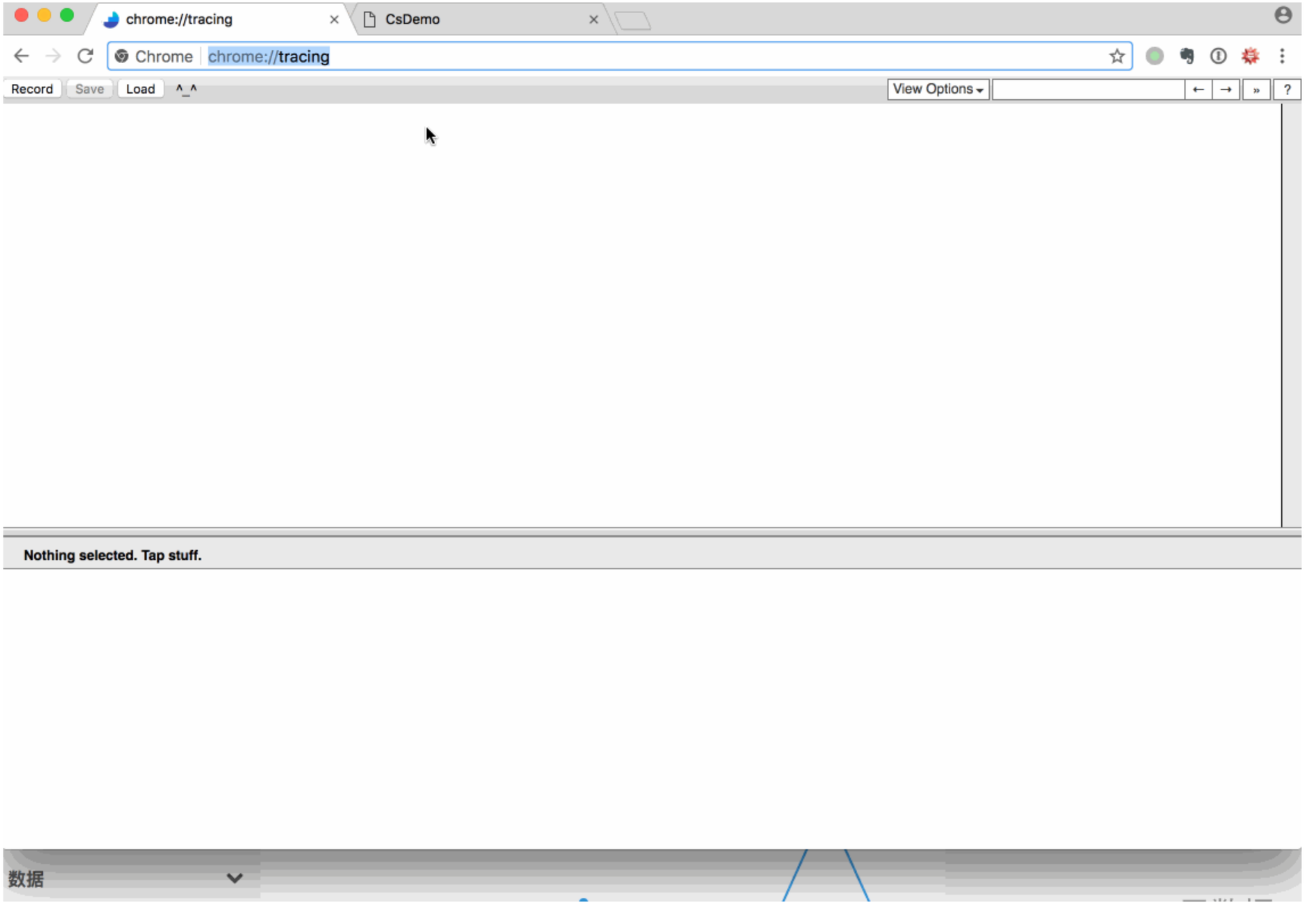
接入&运行

不仅是启动


Back
Caesium
Cs
启动时运行
结果保存在 Sandbox: Documents/Caesium_*.zip
trace 非主线程
主线程 trace 系统的类
悬浮控制台
自动清理 tracing 文件
最多保留 50 个

不仅是启动


12:03
Custom Transitions
Caesium
Presentation Transitions
Cross Dissolve
A cross dissolve transition.
Swipe
An interactive transition.
Custom Presentation
Using a presentation controller to alter the layout of a prese...
Adaptive Presentation
Building a custom presentation that adapts to horizontally c...
Navigation Controller Transitions
Checkerboard
Advanced animations with Core Animation.
TabBar Controller Transitions
Slide
Interactive transitions with UITabBarController.


chrome://tracing
CsDemo
Chrome
chrome://tracing
Record
Save
Load
^_^
View Options
Nothing selected. Tap stuff.
数据

美团App启动优化

2016

“农业时代”

纯手工打造

2017 Q2

“工业时代”

Caesium首次辅助启动优化
清理无用代码

2017 Q4

“互联网时代”

较完善的监控体系
网络请求优化
推迟非紧急任务
+load治理

启动时间缩短20%以上

Q&A

Thanks

Eat better,Live better.

美团点评