ctrip tech 携程技术中心

ctrip tech 携程技术中心 | IT大咖说 知识分享平台

携程技术沙龙

随时随地掌中测技术

王尧波

讲师介绍

王尧波

- 负责携程App基础产品的研发和测试相关工作
- 关注App开发和测试框架，推送效能，基础通用组件
- 专注于Android底层的相关技术

目录

CONTENTS

我们遇到的问题

为什么只有这个
机器会崩溃？

页面怎么卡住了，
点击没反应啊？

内存占用怎么
越来越大？

不知道

一脸懵懂

能否有个工具能够解决这些问题

掌测APP&SDK

掌测具备的职能

1. 全真的测试现场
2. 发现非表象问题
3. 深入到开发代码
4. 数据指标一目了然

掌测模块构成

Crash监控模块

Crash问题一网打尽，偶现的问题也难逃手掌

卡顿监控模块

针对耗时的UI操作进行代码诊断，定位耗时的代码堆栈

内存泄露模块

对复杂的内存泄露问题进行监控，通过内存引用分析定位引发问题的代码

性能指标模块

应用内查看CPU、内存、帧率、流量，每个页面的性能指标一目了然

工作方式

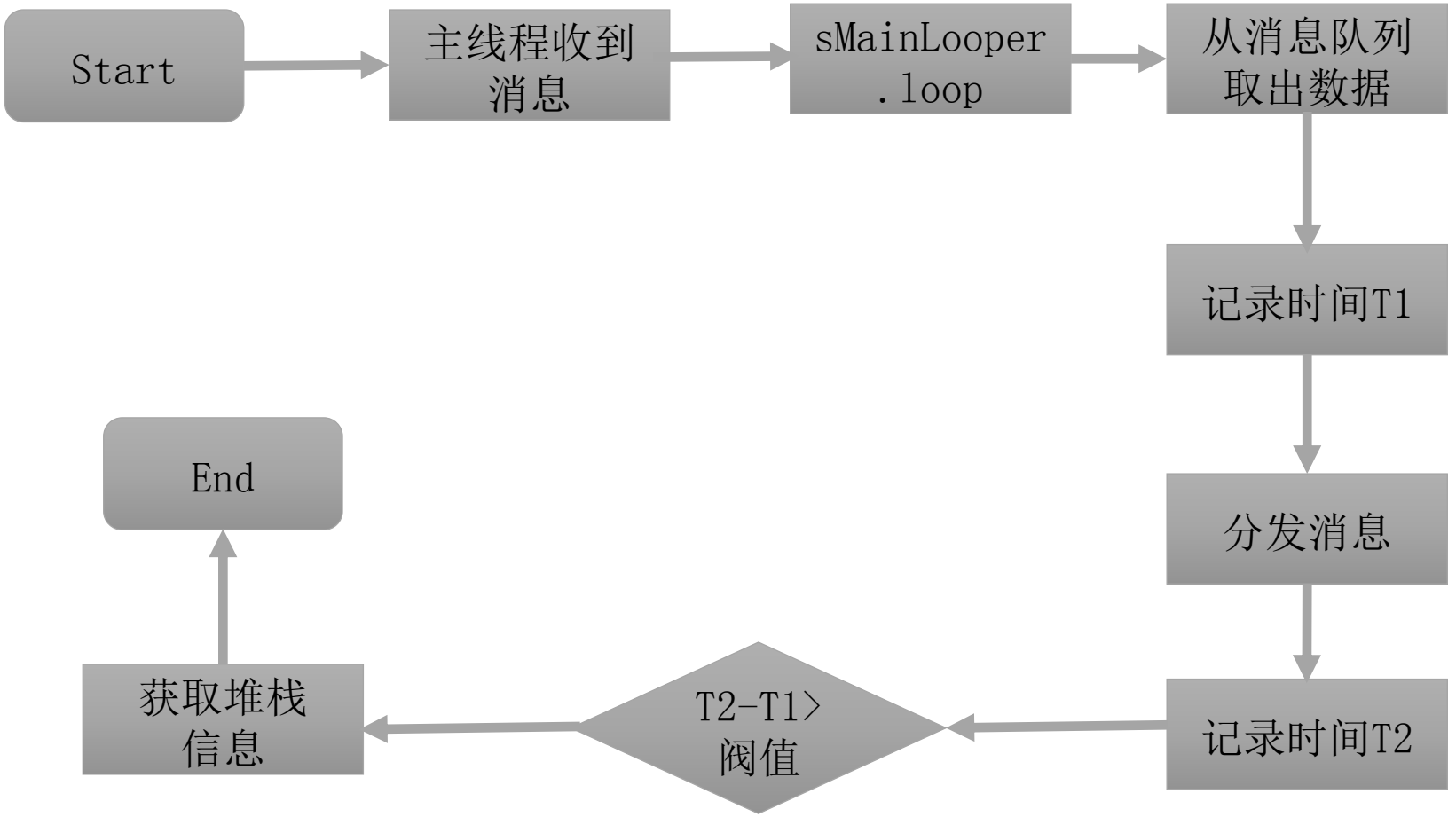
内存

```
** MEMINFO in pid 10432 [ctrip.android.view] **
                   Pss  Private  Private  Swapped     Heap     Heap     Heap
                 Total    Dirty    Clean    Dirty     Size    Alloc     Free
                ------   ------   ------   ------   ------   ------   ------
  Native Heap        0        0        0        0    50315    50315    20340
  Dalvik Heap    51351    51248        0    14144    67700    59657     8043
 Dalvik Other      508      508        0        0
        Stack      872      872        0        0
      Ashmem       128      128        0        0
    Other dev    19779    15348       20        0
     .so mmap    11064      484     7732     1544
     .jar mmap       2        0        0        0
     .apk mmap     334        0      132        0
     .ttf mmap     651        0      496        0
     .dex mmap   17698        0    15196        0
    code mmap     4164        0     2632        0
   image mmap     2576     1540      484      612
   Other mmap     1211        4      640        0
     Graphics    42784    42784        0        0
           GL    28556    28556        0        0
      Unknown    45941    45856       12     2728
        TOTAL   227619   187328    27344    19028   118015   109972    28383
```

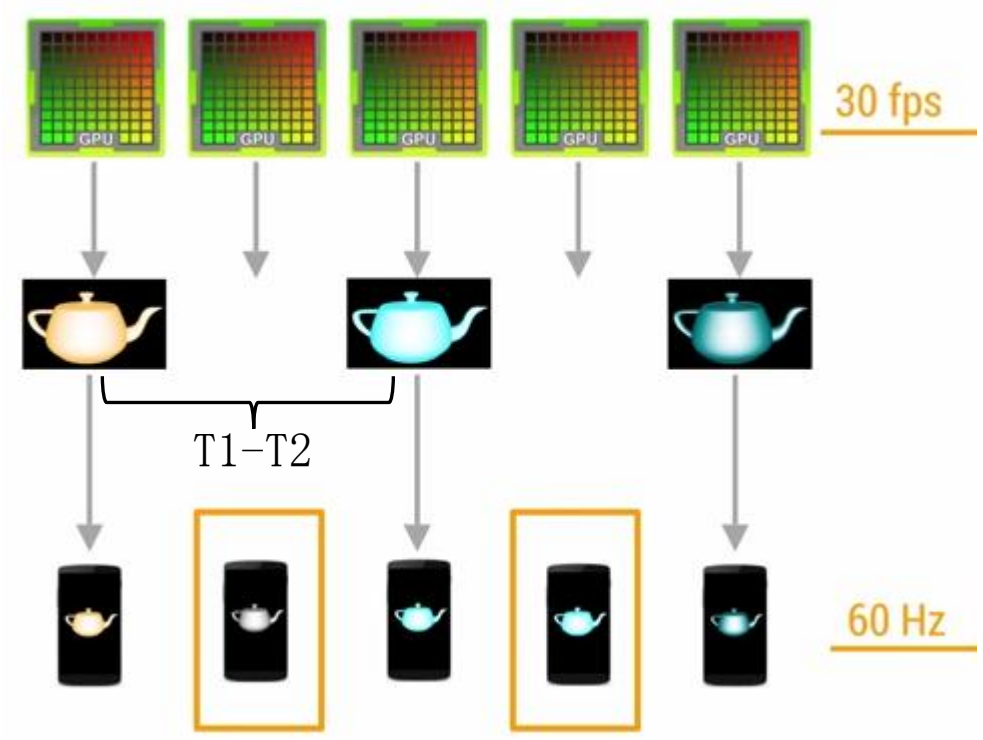
Pss：内存所实际占用的值
Private Dirty：进程独占内存

采用Private Dirty计算方式作为内存指标

卡顿检测

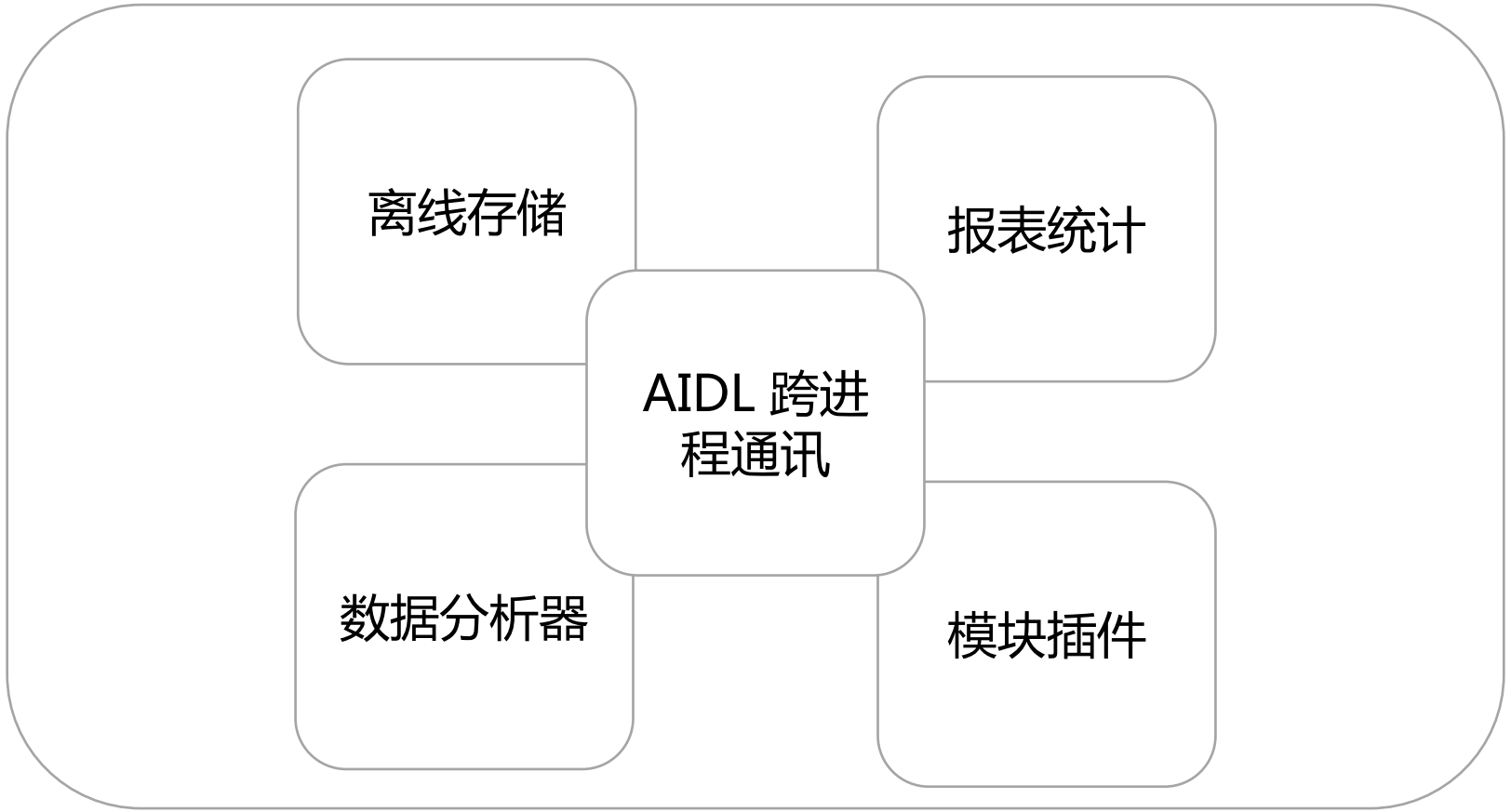

Start
主线程收到消息
sMainLooper.loop
从消息队列取出数据
记录时间T1
分发消息
记录时间T2
T2-T1>阀值
获取堆栈信息
End

帧率 60fps




```
Choreographer.FrameCallback{
        doFrame(time)
}
```

SDK模块组成

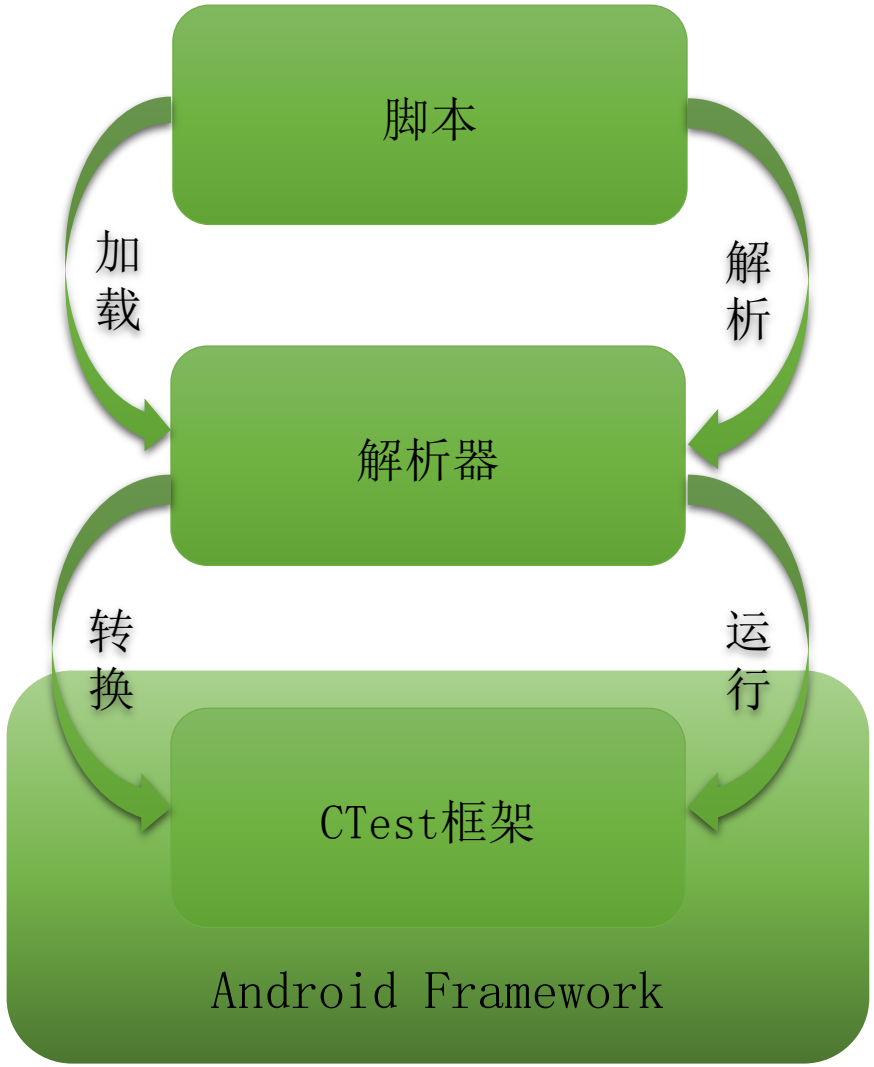


接入方法

一行代码接入SDK

```
CTestSDKManager.builder()
        .setApplicationContext(this)
        .setBlockCanaryEnable(true)
        .setBlockThreshold(200)
        .setLeakCanaryEnable(true)
        .setCrashEnable(true)
        .build().startMonitor();
```

机器人




```
Screenshot{
}
Click{
resource_id:test.ctrip.testapp
$id/robot_test;
}
Wait{
time:1000;
}
Screenshot{
}
Type{
index:0;
value:ctrip test;
}
Screenshot{
}
```

未来计划-CMonkey

准确的识别控件，避免无效点击

遍历UI测试，执行效率更高

对于Crash路径记忆并能够优先执行

ctrip tech 携程技术中心

ctrip tech 携程技术中心 | IT大咖说 知识分享平台

THANK YOU!

Q&A