

title: GOS项目系统整体应用性能问题

author: 欧杰

目录

0.声明

1.背景

2.内存监控数据分析

3.车机内存性能问题总结

4.优化建议

5.参考文献

0.声明

说明：

- 测试场景：测试同学正常测试各个功能，采集周期起始点并非重新开机。
- 以下监控的性能数据是测试2750个采集周期，每个采集周期10s，7.64小时的数据。
- 内存图横坐标是采集周期，单位：个；纵坐标是内存占用，单位：M
- CPU图横坐标是采集周期，单位：个；纵坐标是CPU占用，单位：%
- View图横坐标是采集周期，单位：个；纵坐标是当次采集界面中view的个数，单位：个
- 可用内存图横坐标是采集周期，单位：个；纵坐标是当前系统可用内存，单位：kb

1.背景

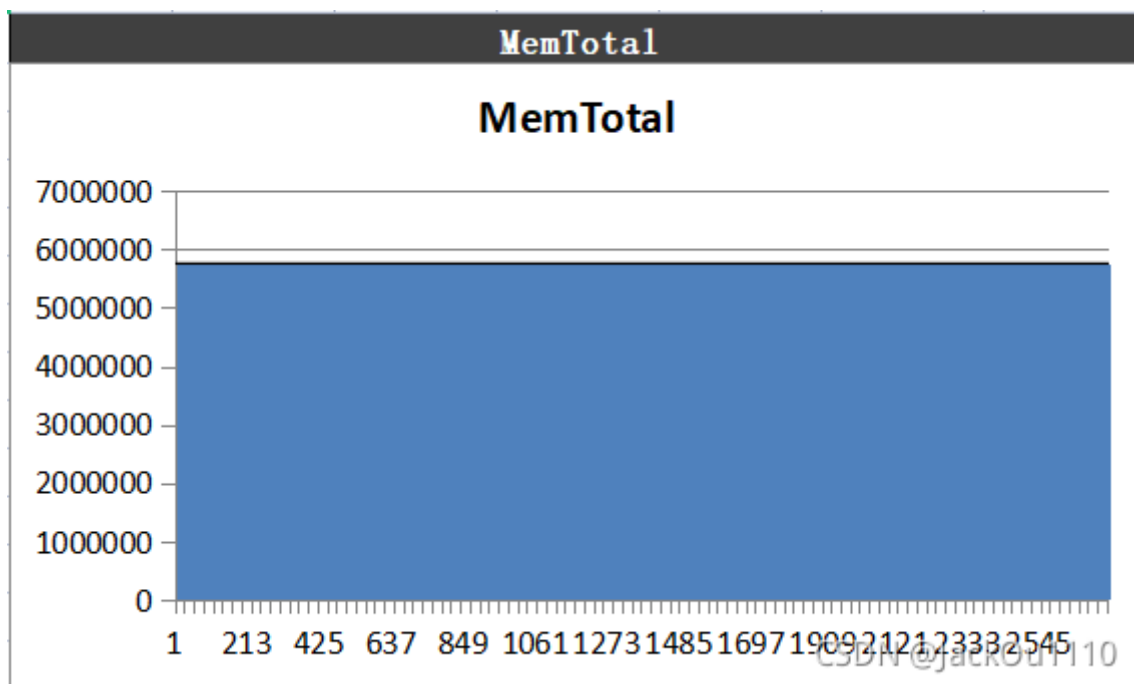
在以上测试周期日志中，发现有一次lowmemorykiller（LMK）杀进程的记录，通过杀oom_score_adj为906的进程来回收了约300M内存。从此项指标可以看出我们当前的系统长时间运行过程之后，会导致可用内存越来越少，最后触发LMK来回收内存资源。因此，此报告主要分析当前项目中各个应用暴露出来的内存问题。

```
.....
lowmemorykiller: Killing 'droid.deskclock' (23873) (tgid 23873), adj 906, \x0ato free 51492kB on behalf of 'kswapd0' (130) because \x0acache 321104kB is below limit 322560kB for oom score 906 \x0aFree memory is -37260kB above reserved \x0aF...
lowmemorykiller: Killing 'ti.diagnostics' (4462) (tgid 4462), adj 906, \x0ato free 51252kB on behalf of 'kswapd0' (130) because \x0acache 321644kB is below limit 322560kB for oom score 906 \x0aFree memory is -34988kB above reserved \x0aF...
lowmemorykiller: Killing 'id.printspooler' (19506) (tgid 19506), adj 906, \x0ato free 44428kB on behalf of 'kswapd0' (130) because \x0acache 321852kB is below limit 322560kB for oom score 906 \x0aFree memory is -31956kB above reserved \x0aF...
lowmemorykiller: Killing 'd.car.messenger' (4764) (tgid 4764), adj 906, \x0ato free 39004kB on behalf of 'kswapd0' (130) because \x0acache 322164kB is below limit 322560kB for oom score 906 \x0aFree memory is -28344kB above reserved \x0aF...
lowmemorykiller: Killing '.qcrilmsgtunnel' (2416) (tgid 2416), adj 906, \x0ato free 37412kB on behalf of 'kswapd0' (130) because \x0acache 322164kB is below limit 322560kB for oom score 906 \x0aFree memory is -23180kB above reserved \x0aF...
lowmemorykiller: Killing 'viders.calendar' (29748) (tgid 29748), adj 906, \x0ato free 56616kB on behalf of 'kswapd0' (130) because \x0acache 321476kB is below limit 322560kB for oom score 906 \x0aFree memory is -6488kB above reserved \x0aF...
```

2.内存监控数据分析

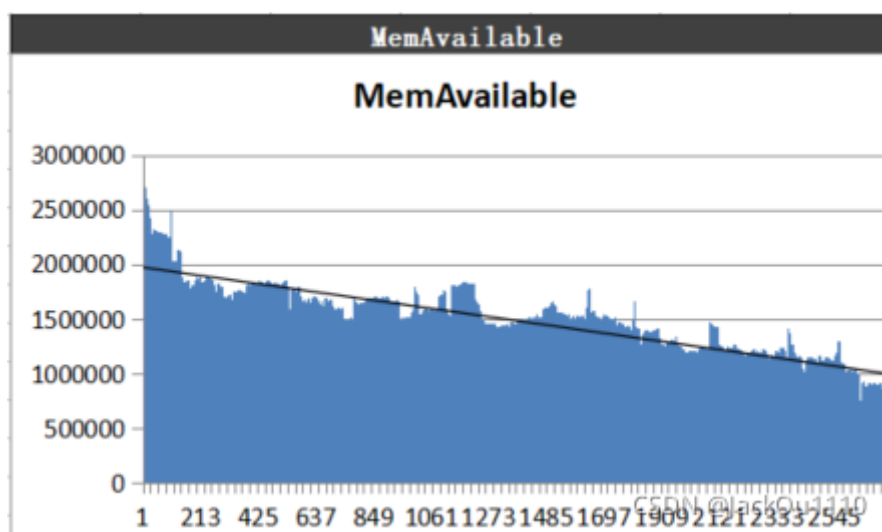
2.1 车机整体内存情况

2.1.1 系统总体可用内存



系统总体可用物理内存：5754876kb ≈ 5.49G

2.1.2 系统总体可用剩余物理内存



- 采集周期起始点所剩内存：2718080kb ≈ 2.59G
- 采集周期内最低所剩内存：724644 ≈ 690M
- (说明：此值是LMK回收后的所剩内存值。当系统内存少于315M时，系统就会自动把 /proc/<pid>/oom_score_adj 中oom_score_adj的值大于900，少于1000的进程杀掉)
- 从图中可以看出可用内存**在监控周期内持续降低**，出现这种情况有两个原因：
 - 一是启动了新应用，重新分配了内存导致可用内存下降。
 - 二是应用**内存泄露**，导致内存持续无法回收。初步判断可用内存持续降低会和内存泄露有关系，因为当前项目几乎全部应用都使用了平台签名，属于系统应用，oom_adj值比较低，正常情况下，所剩内存低于126M才会触发杀这些进程。所以可以近似认为系统应用进程是保活的，所以初步断定可用内存持续下降和应用内存泄露，内存持续无法被回收有关系。
- 从图中可以看出可用内存存在**一些时刻有突然陡降的现象**，出现这种情况有两种可能：

- 一是某些应用启动加载过多资源，导致内存在短时间内消耗过多，建议应用用时再初始化加载资源。
- 二是有进程崩溃或者被杀掉，释放出部分内存。

2.1.3 系统活跃/非活跃内存

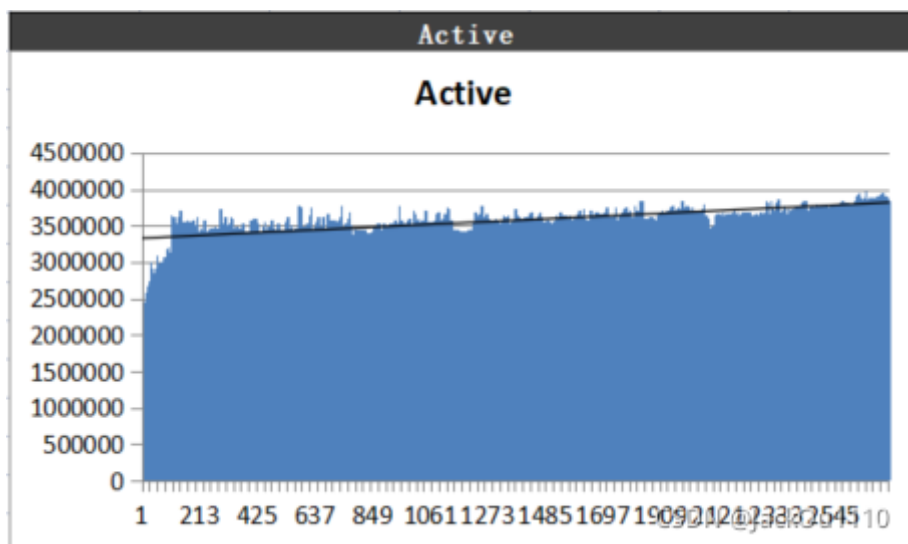
Active：最近使用的内存量，除非绝对必要，否则通常不会回收(以kb为单位)。

Inactive：最近使用较少且更有资格回收用于其他目的的内存量(以kb为单位)。

ActiveFile：自上次系统回收内存以来，处于活动使用状态或处于活动使用状态的文件缓存内存量(以kb为单位)

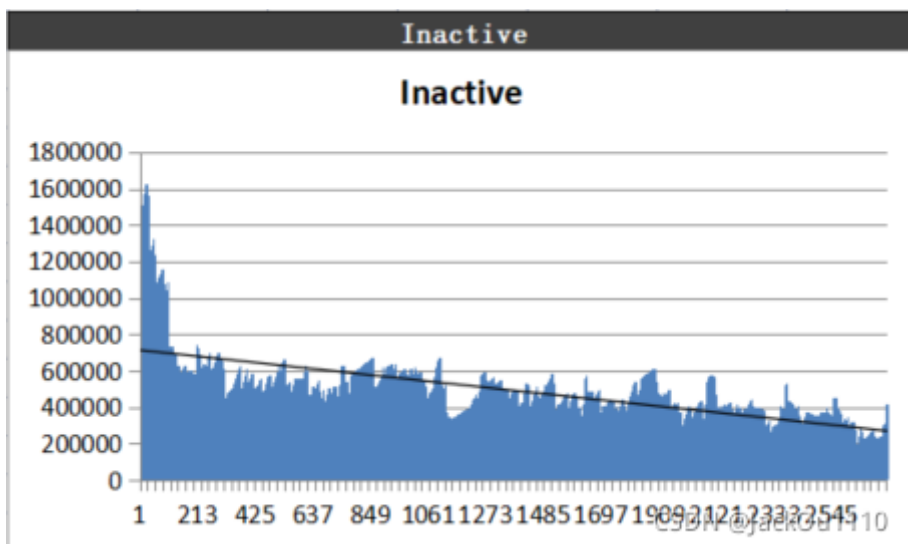
InactiveFile：从磁盘新加载的文件缓存内存量，或者是要回收的候选文件缓存内存量。(以kb为单位)

2.1.3.1 Active图



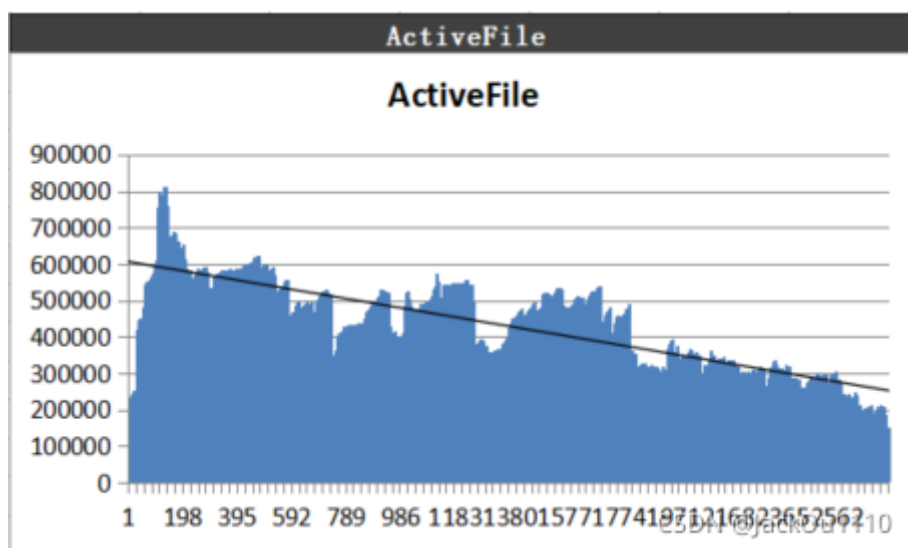
从Active内存图可以看出系统所有进程在整个监控周期内平均消耗：3571792kb \approx 3.41G，总共可用内存5.49G，平均使用率：**62%**

2.1.3.2 Inactive 图



从Inactive图中可以看出系统在监控周期内空闲的内存大小平均为：492299kb \approx 469M，空闲内存率：8.5%，此参数说明系统在特别需要内存的时候可以分配的内存。

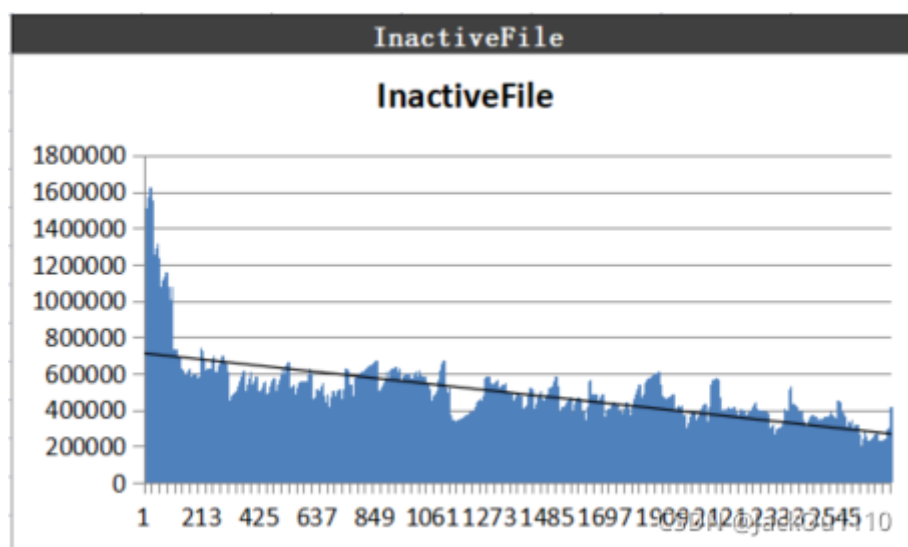
2.1.3.3 Active file图



从Active file图中可以看出系统在监控周期内,系统回收后剩余的内存越来越少，并且存在比较大的抖动。

缓存活跃文件内存最大：815492kb \approx 778M；最小：105736kb \approx 100.8M；平均：429188kb \approx 409.3M。可以看出在整个监控周期内，系统在持续压缩缓存文件的大小，并且期间存在进程崩溃情况。

2.1.3.4 InActive file图



从Inactive file图中可以看出系统中在前期有1626812kb \approx 1.55G 从磁盘新加载的文件缓存内存量，或者有待系统回收的内存！此处需要调查哪些应用启动会加载过多过大文件到内存。

2.2 应用内存消耗分析

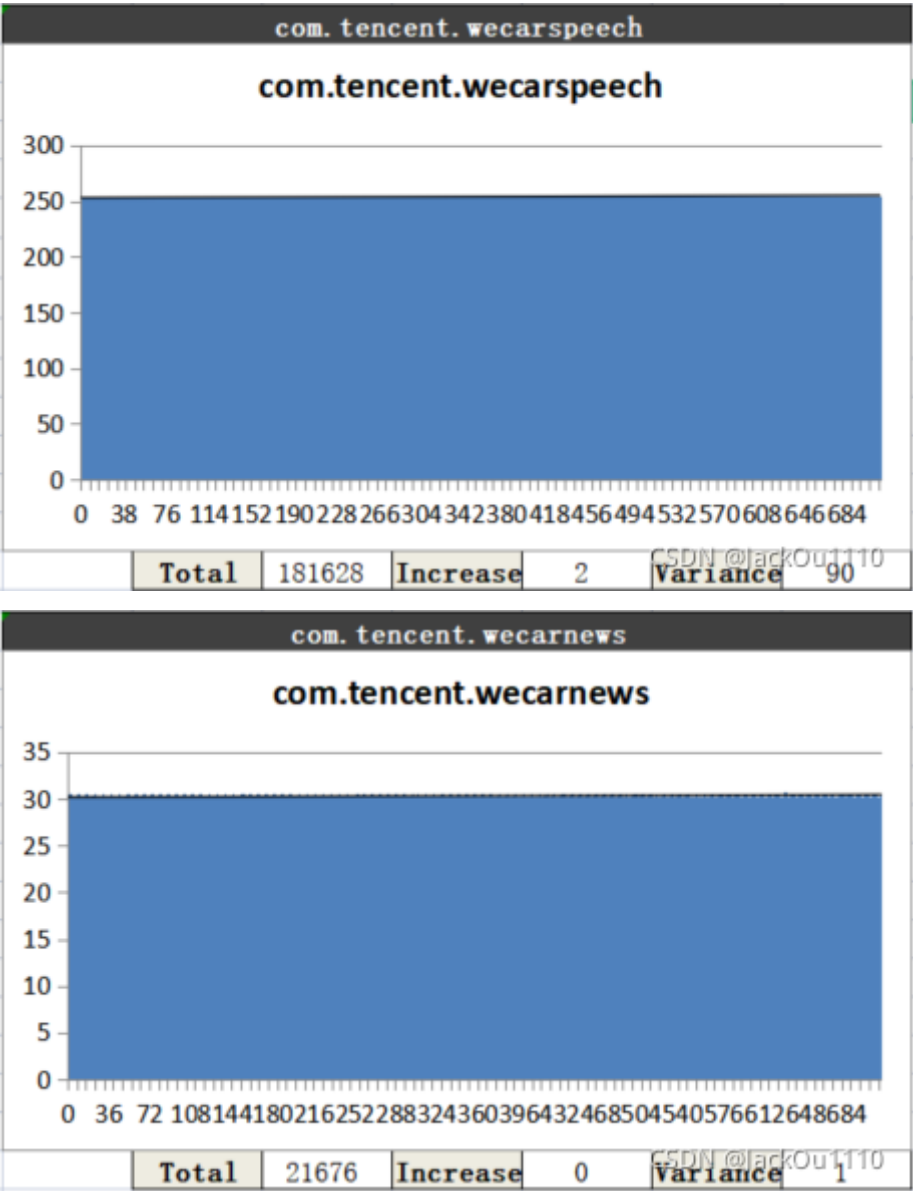
2.2.1 内存优化方向与目标

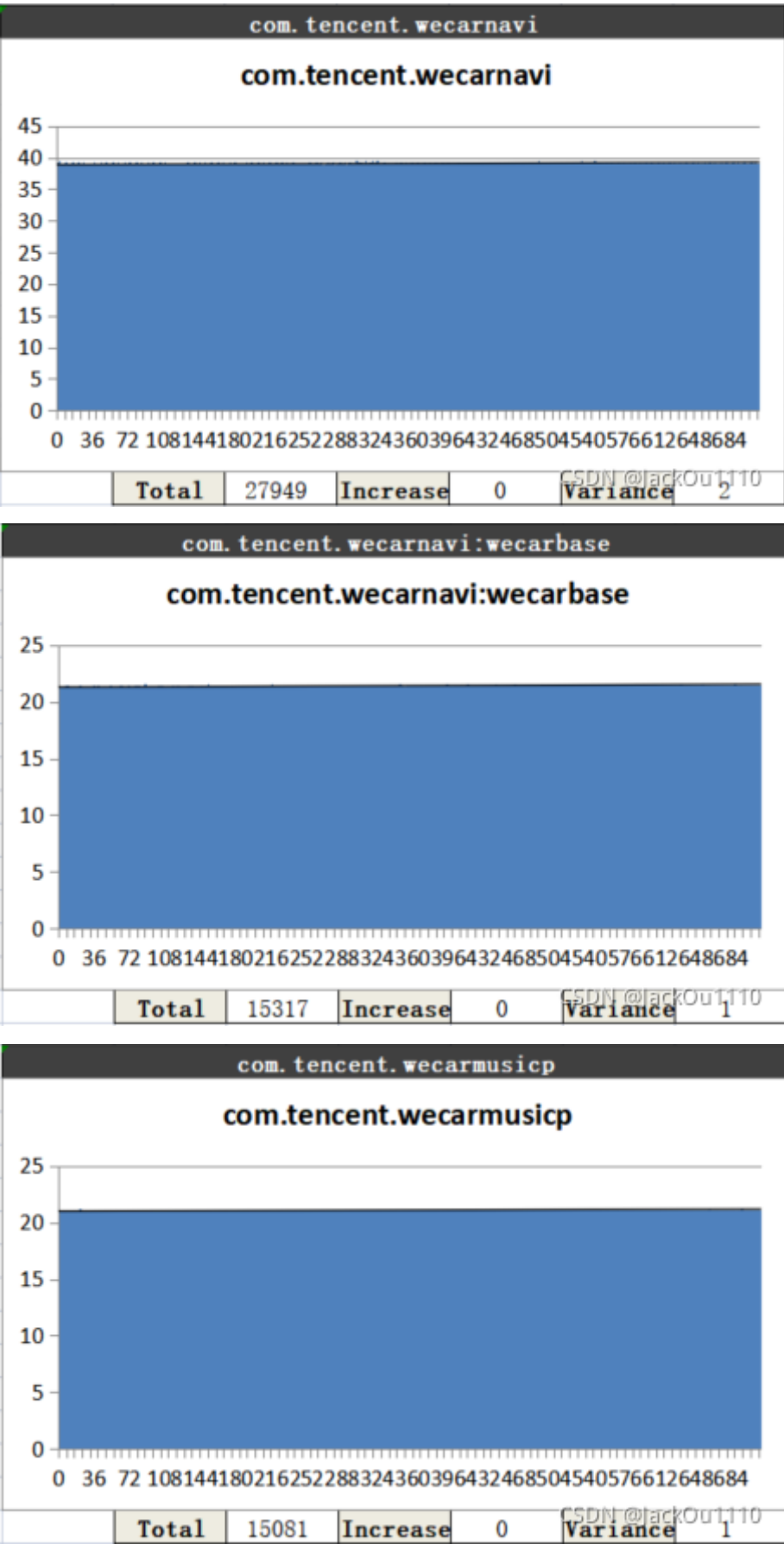
应用内存优化目标有四个方向：

- 无内存泄露
- 无内存抖动，小范围内内存抖动可接受
- 不存在阶梯内存(稳定性)
- 降低内存消耗

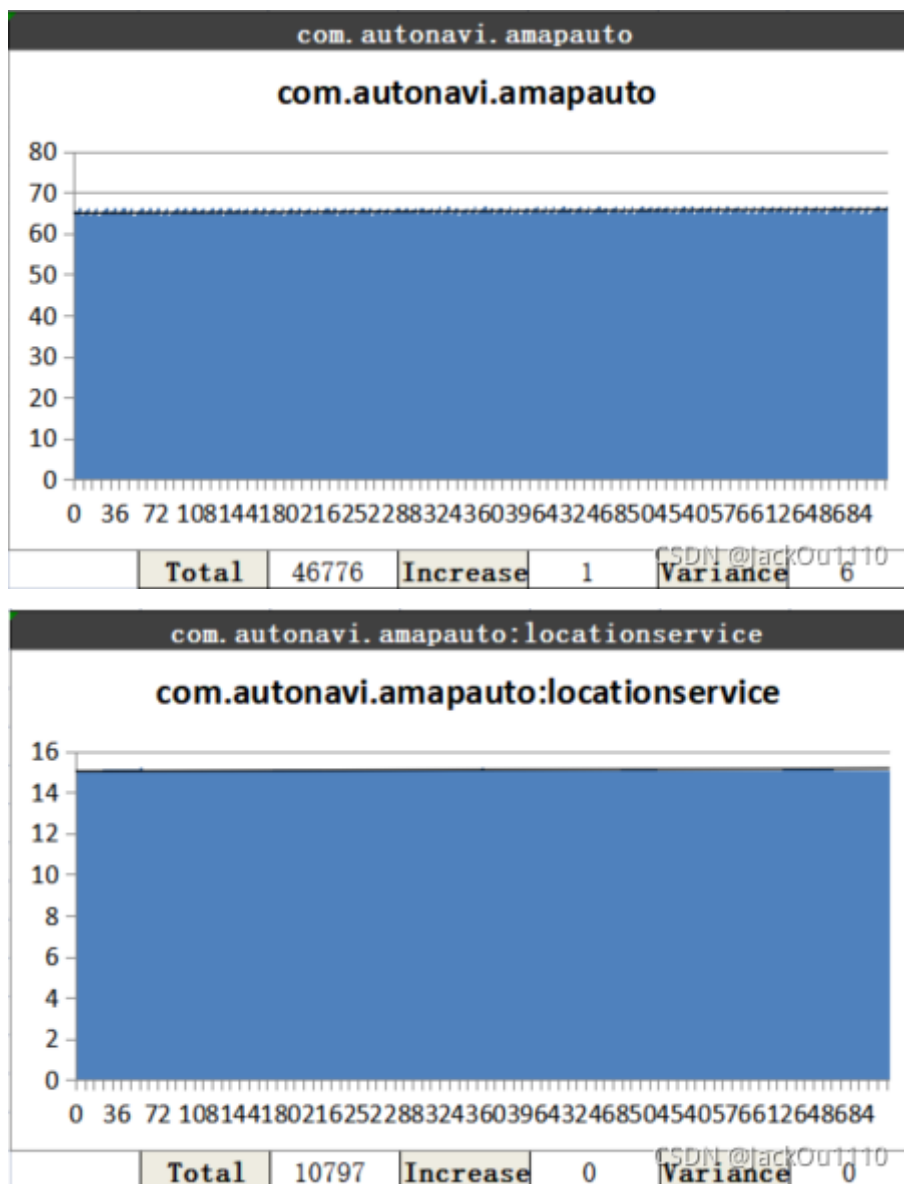
下面给出GOS一期应用内存监控结果，作为优化目标方向。

以下是腾讯系应用内存占用情况：

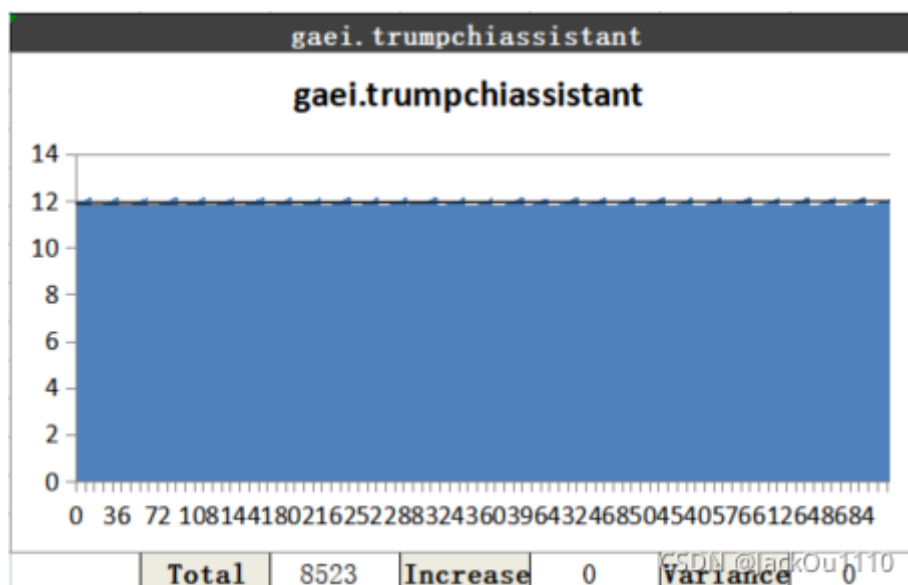


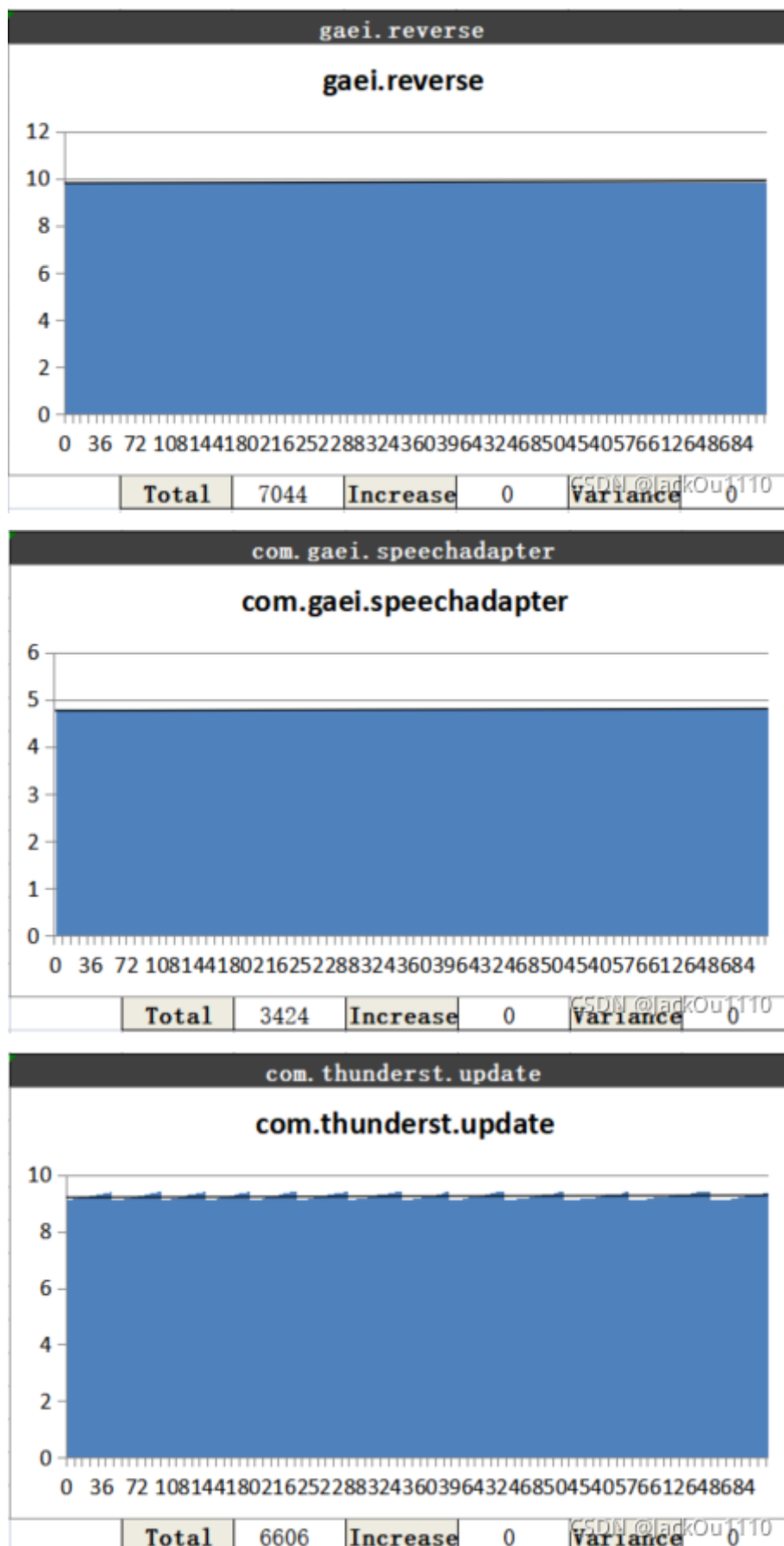


以下是高德系内存占用情况：



以下是中科创达系内存占用情况：



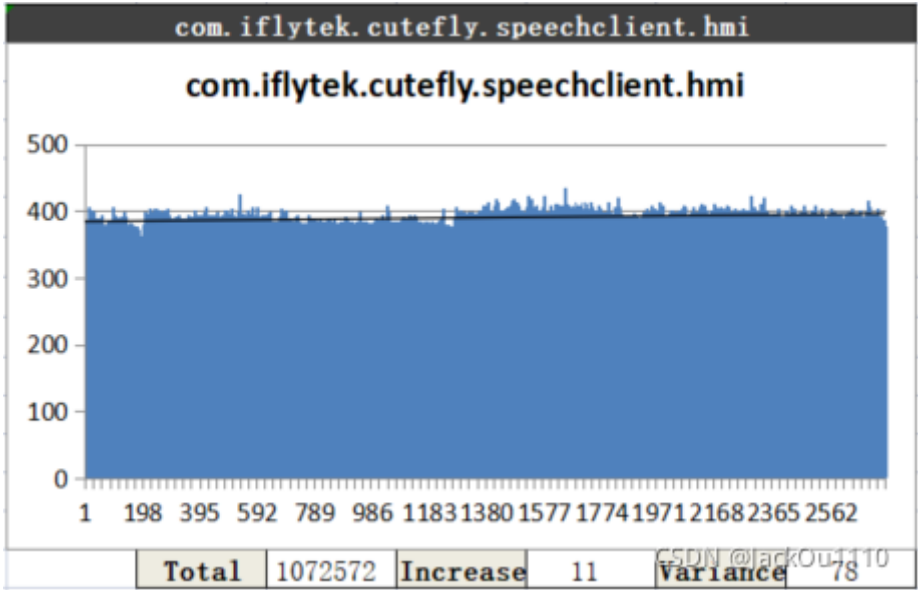


小结:

从上面图中可以看出在监控期间，腾讯系的应用内存最稳定，高德和创达的应用存在小幅波动，在可接受范围内。三家公司的应用总体内存消耗都比较低。

2.2.2 当前应用内存消耗现状

2.2.2.1 语音助理

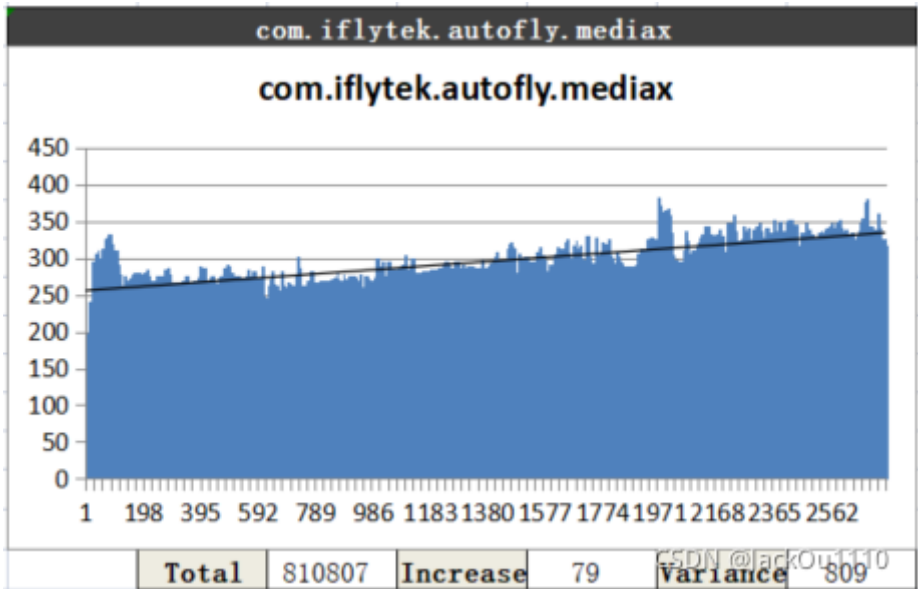


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
436M	356M	390M	存在	少量	少量

优化建议:

- 降低内存使用
- 增加对象复用，减少内存抖动

2.2.2.2 媒体



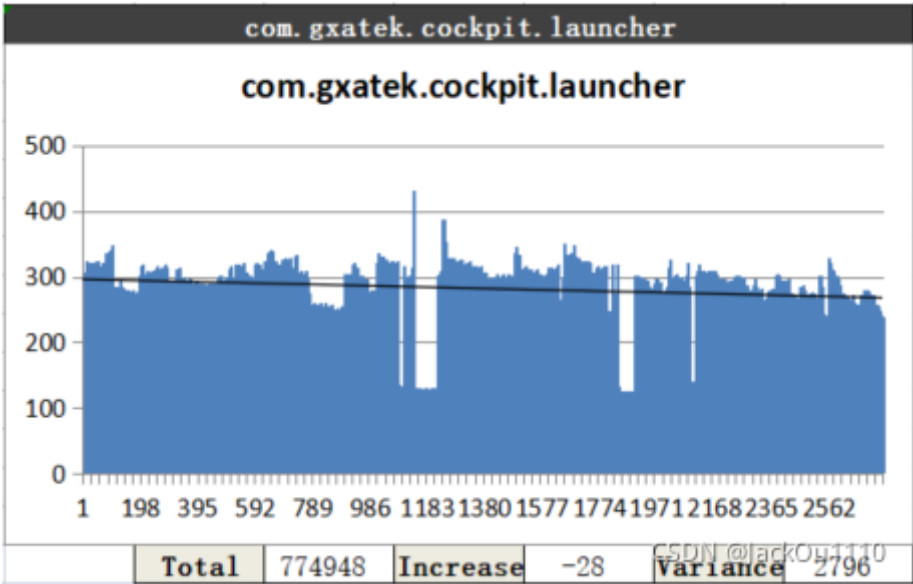
最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
383M	145M	295M	存在	存在	存在

优化建议:

- 控制对象创建个数，建立对象池实现对象复用

- 调查不能释放的内存
- 资源按需加载、调查加载占用大内存的资源原因，bitmap对象是否按空间尺寸压缩加载，预置图片资源放在适合屏幕尺寸的目录中。
- 看过之前卡顿分析报告，内存暂用大主要还是存在加载缓存bitmap过多所致。

2.2.2.3 主屏Launcher

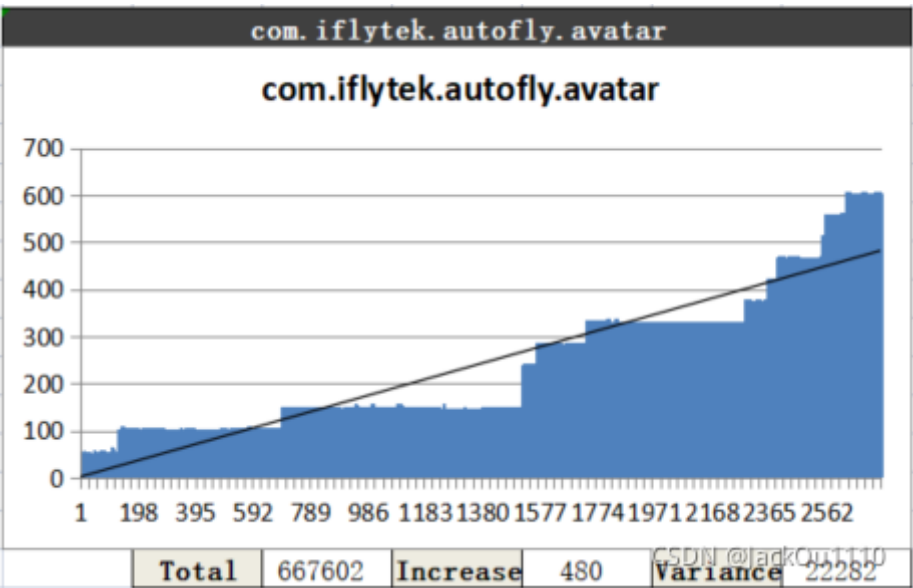


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
432M	118M	282M	存在	暂无	存在

优化建议：

- 出现阶梯内存，需要单独跑monkey调查崩溃原因，保证稳定性
- 存在内存抖动需要调查对象分配和复用情况

2.2.2.4 Avatar

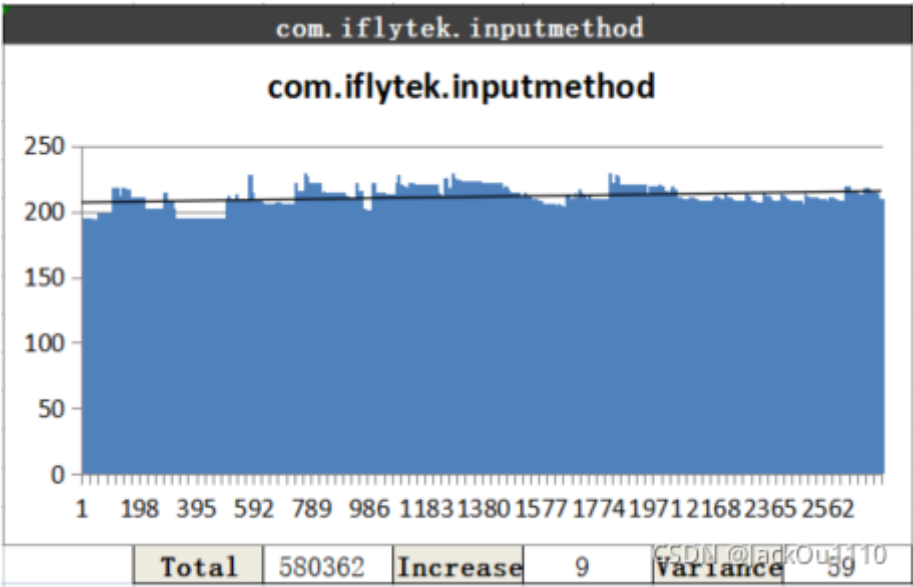


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
609M	33M	242M	存在	存在	存在

优化建议：

- 存在**严重内存泄露**，调查不能释放的对象
- 每次加载对象都在**50-100M**，需要调查为什么一次性需要加载那么多资源，bitmap对象是否按空间尺寸压缩加载

2.2.2.5 讯飞输入法

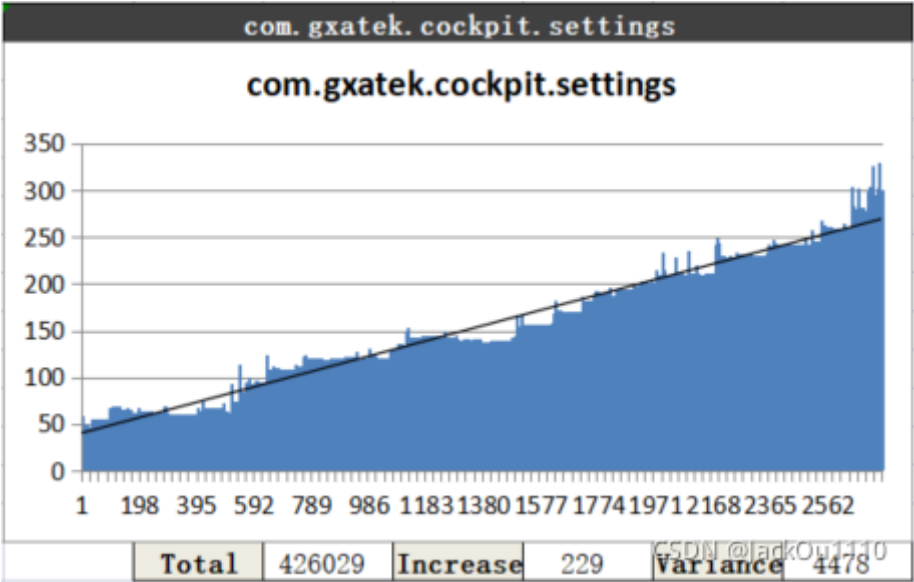


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
229M	194M	211M	少量	暂无	少量

优化建议：

- 调查部分对象创建使用完了就被回收的原因，建议对象复用减少内存抖动
- 是否可以减少部分资源的加载，输入法应用总体暂用内存比较高

2.2.2.6 系统设置

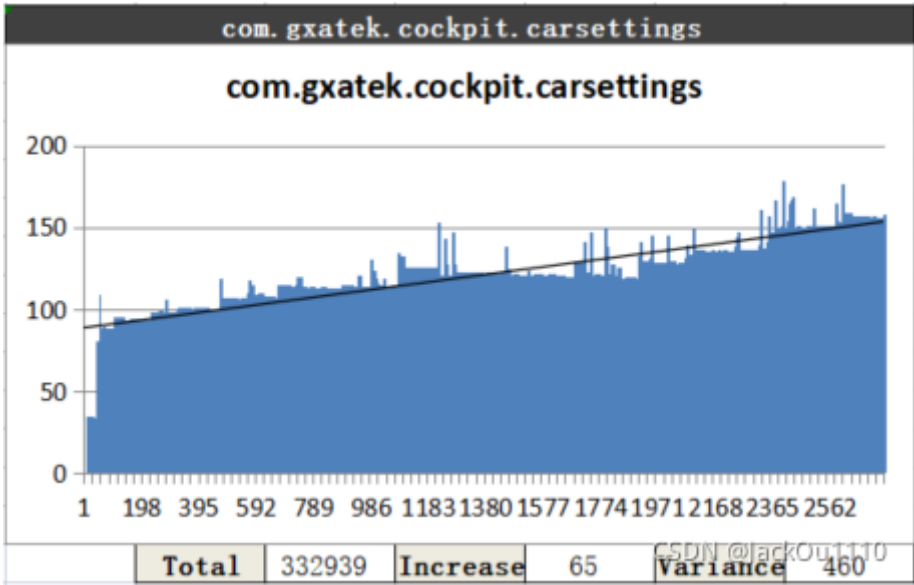


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
330M	47M	155M	少量	存在	存在

优化建议：

- 存在**严重内存泄露**，调查不能释放的对象
- 有部分**分配大内存**迹象需要调查

2.2.2.7 车辆设置

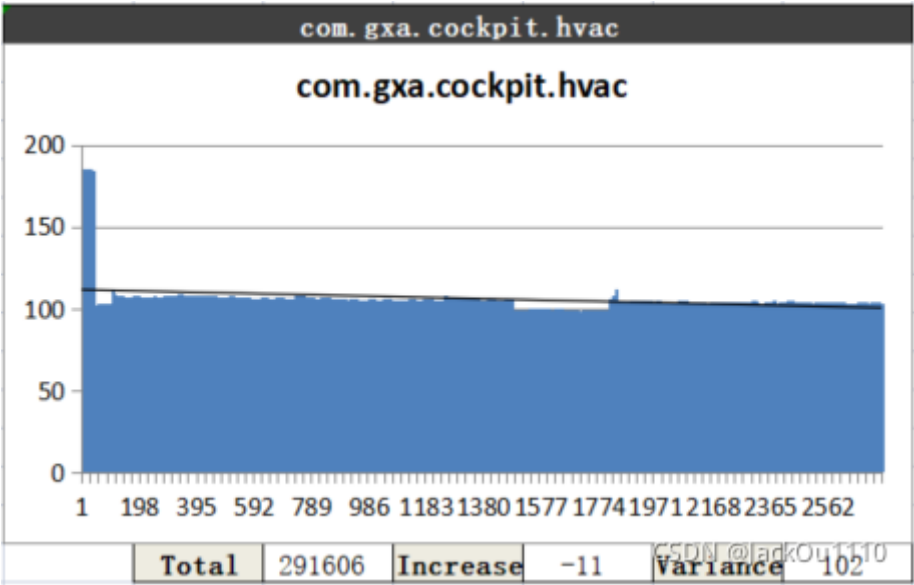


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
178M	34M	121M	少量	存在	存在

优化建议：

- 存在**严重内存泄露**，调查不能释放的对象
- 有部分**分配大内存**迹象需要调查

2.2.2.8 空调

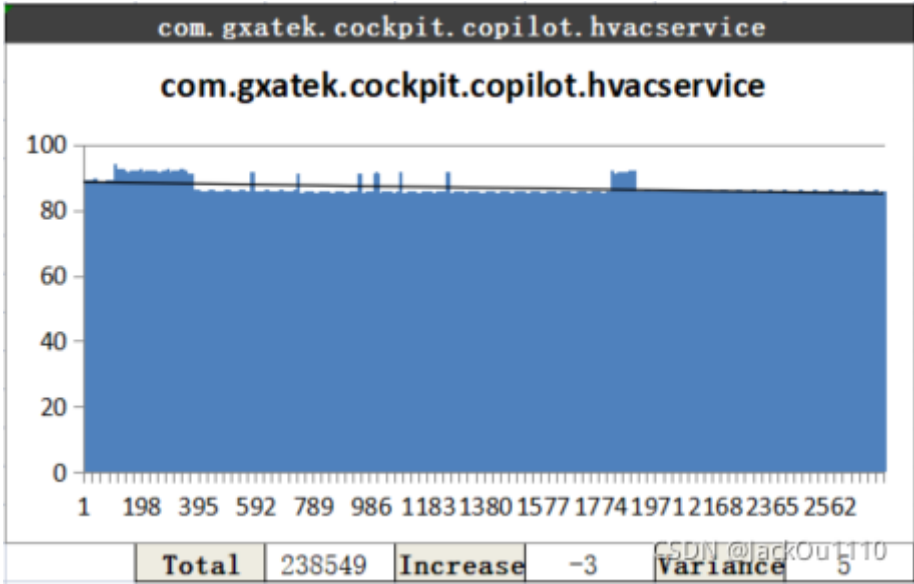


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
185M	99M	106M	否	否	存在

优化建议：

- 调查开始阶段是否crash了，需要单独monkey测试稳定性

2.2.2.9 空调服务

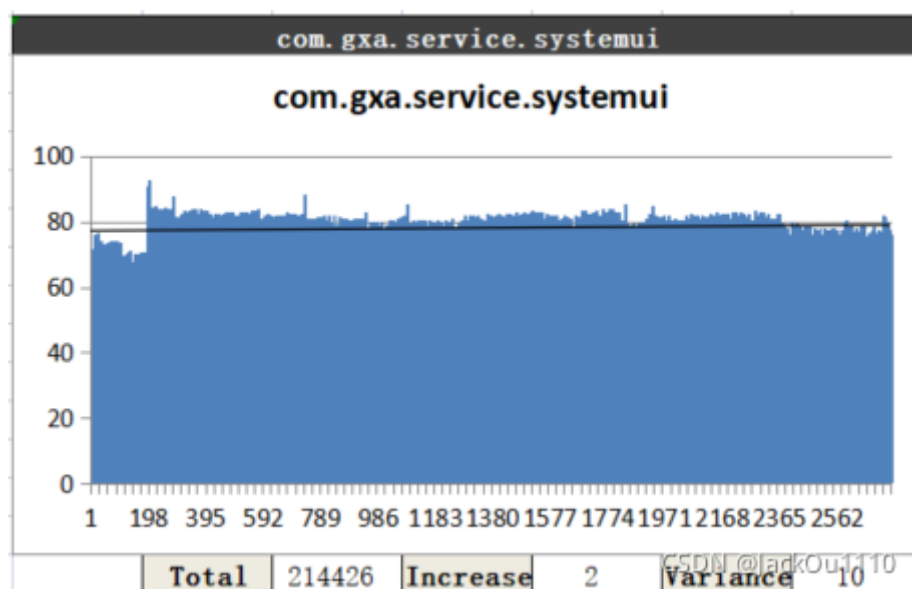


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
94M	85M	86M	否	否	少量

优化建议：

- 调查部分分配大内存情况

2.2.2.10 SystemUi

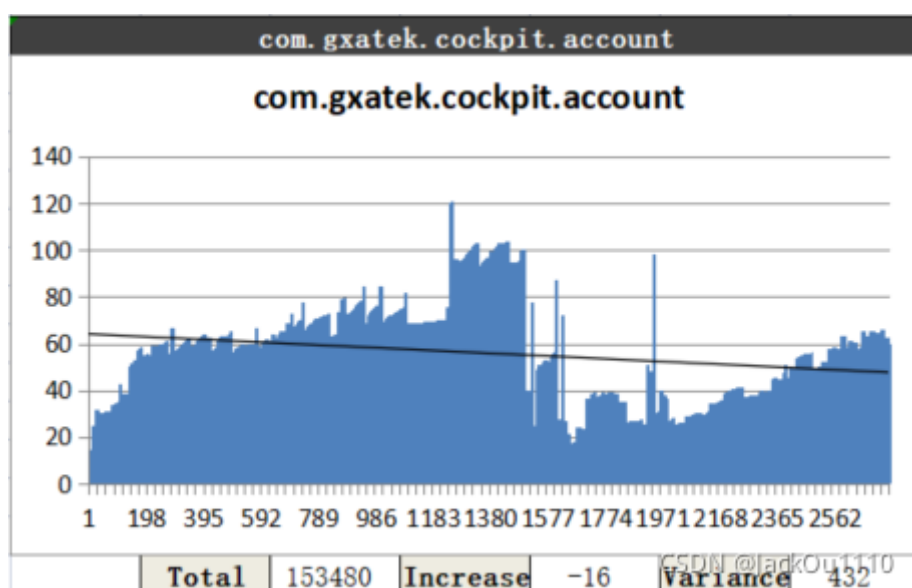


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
93M	65M	78M	少量	否	存在

优化建议:

- 监控前期出现内存陡增情况，需要调查原因

2.2.2.11 账号

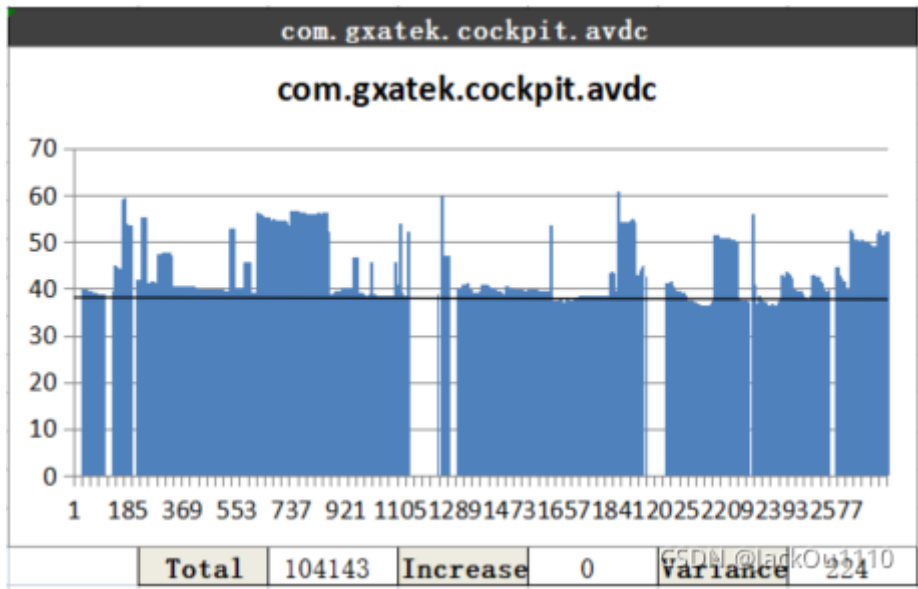


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
121M	14M	56M	存在	存在	存在

优化建议:

- 存在内存抖动需要调查内存复用
- 从两端内存图可以看出有内存泄露情况
- 出现严重阶梯，需要单独monkey测试稳定性

2.2.2.12 AVDC

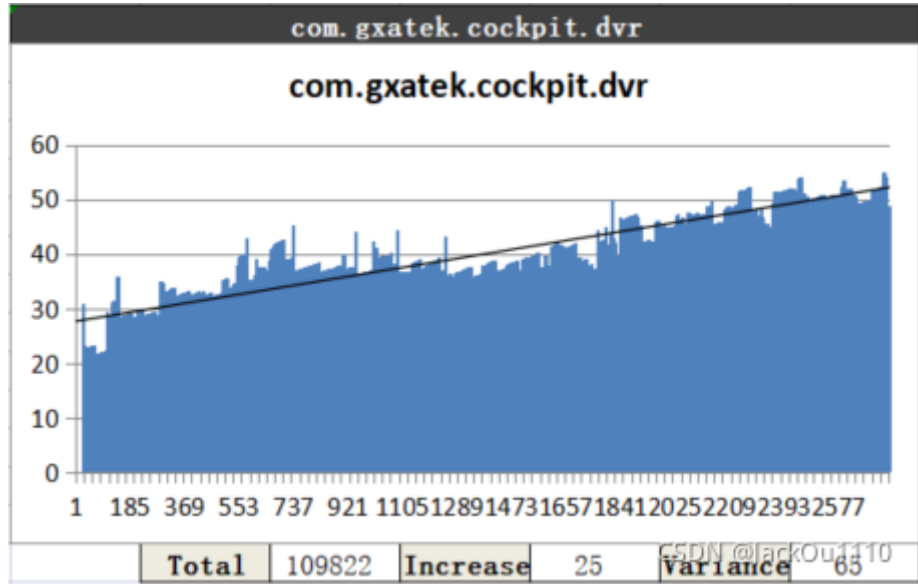


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
61M	36M	43M	存在	少量	存在

优化建议:

- 需要单独monkey调查该应用稳定性

2.2.2.13 DVR

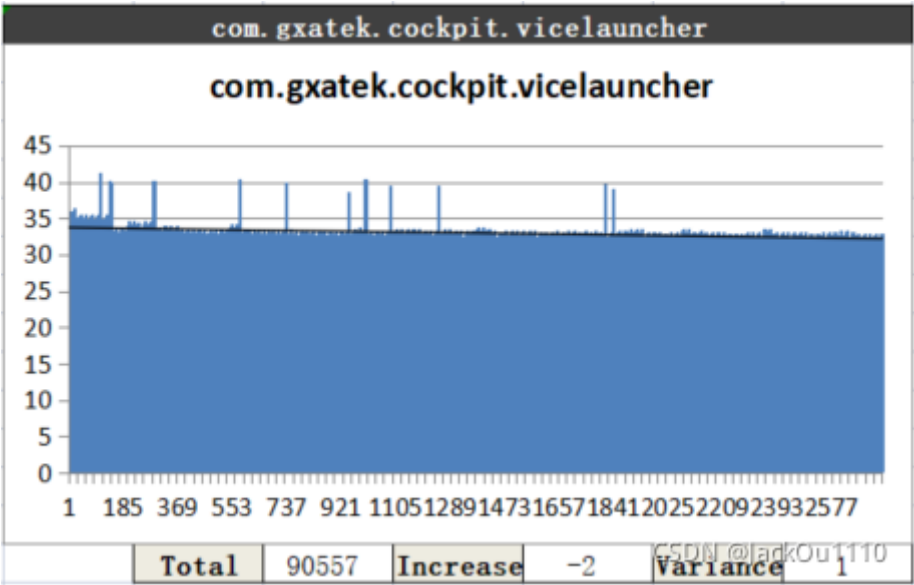


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
55M	21M	40M	存在	存在	存在

优化建议:

- 调查对象分配和回收，尽量对象复用
- 存在严重内存泄露，调查不能释放的对象

2.2.2.14 副屏launcher

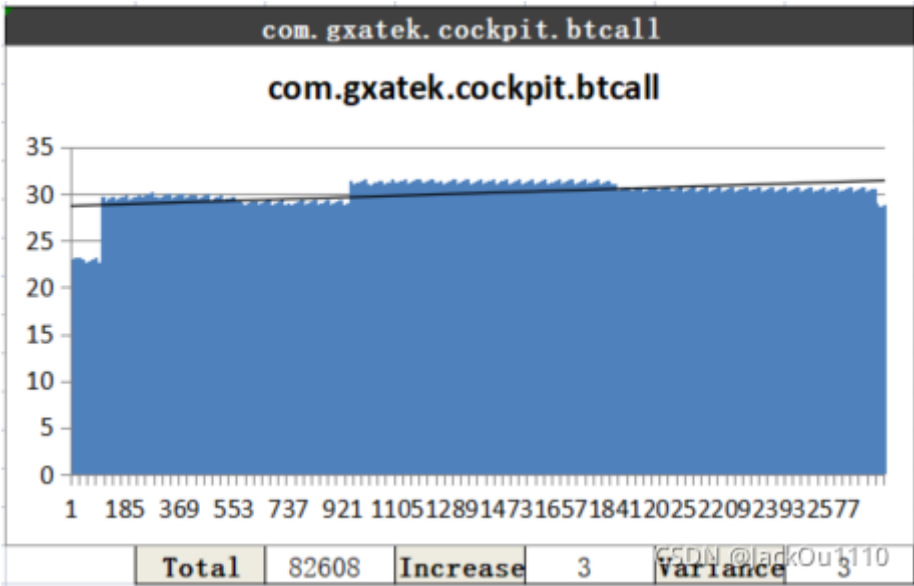


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
41M	32M	33M	少量	否	否

优化建议:

- 调查局部出现大对象分配场景

2.2.2.15 btcall

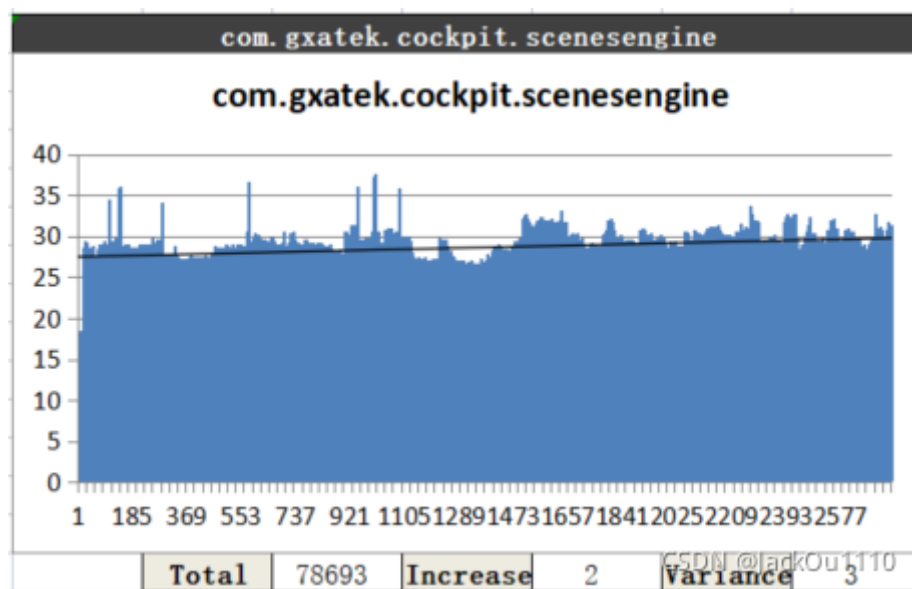


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
32M	23M	30M	否	否	少量

优化建议:

- 调查早期内存阶梯上升原因，如果因为启动应用，可以忽略。

2.2.2.16 场景引擎

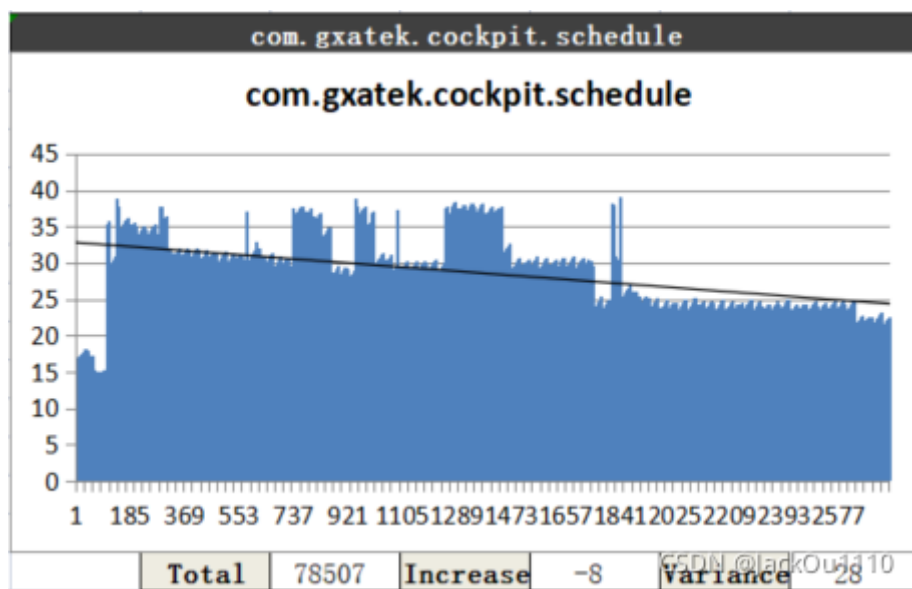


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
38M	16M	29M	存在	少量	存在

优化建议:

- 调查大对象分配场景
- 对于常用的对象采用对象复用机制

2.2.2.17 schedule

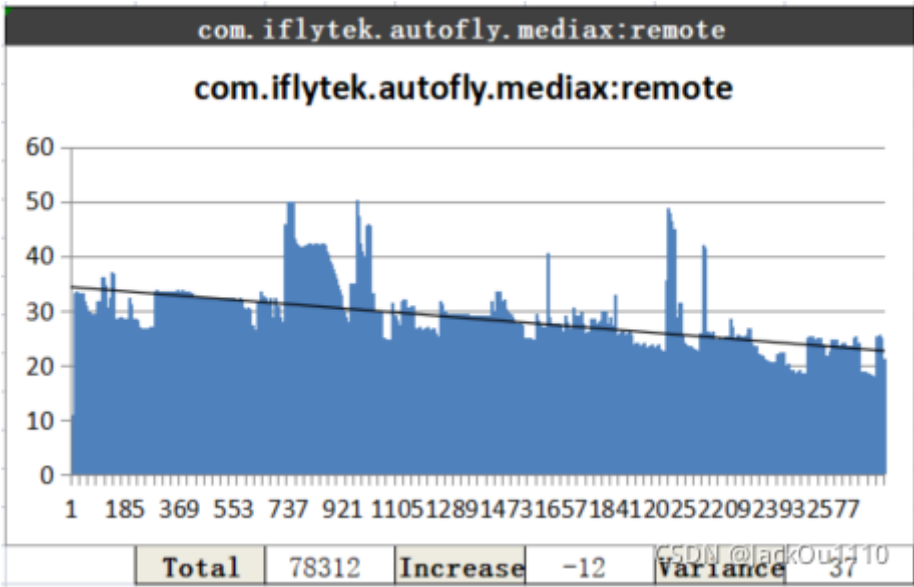


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
39M	14M	29M	少量	少量	存在

优化建议:

- 需要调查出现阶梯内存原因是进程被杀还是分配了大对象，从上图看，更像进程被杀的现象，建议先单独跑monkey查看稳定性

2.2.2.18 媒体子进程

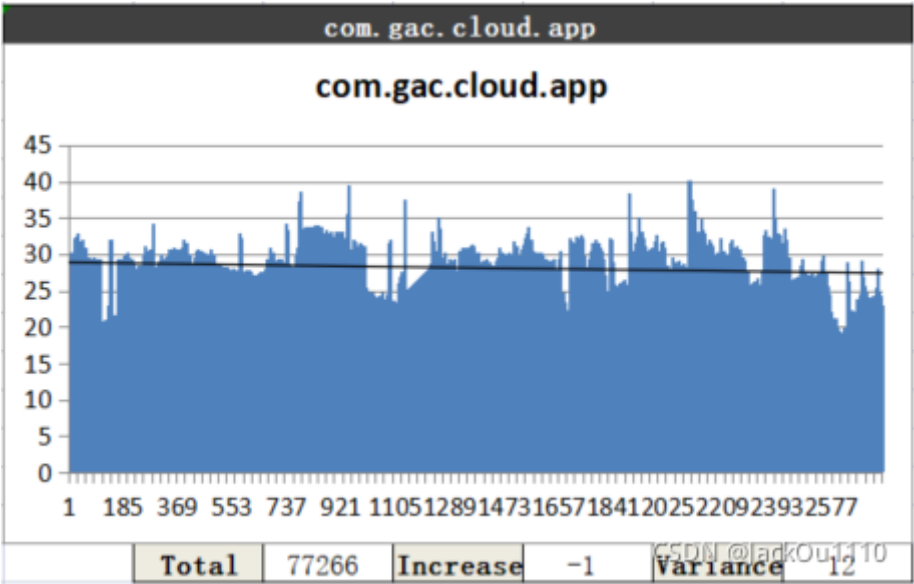


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
50M	10M	28M	存在	少量	存在

优化建议:

- 调查是否有大资源加载，中间有加载了又被回收的表象，建议按需加载资源

2.2.2.19 GCS

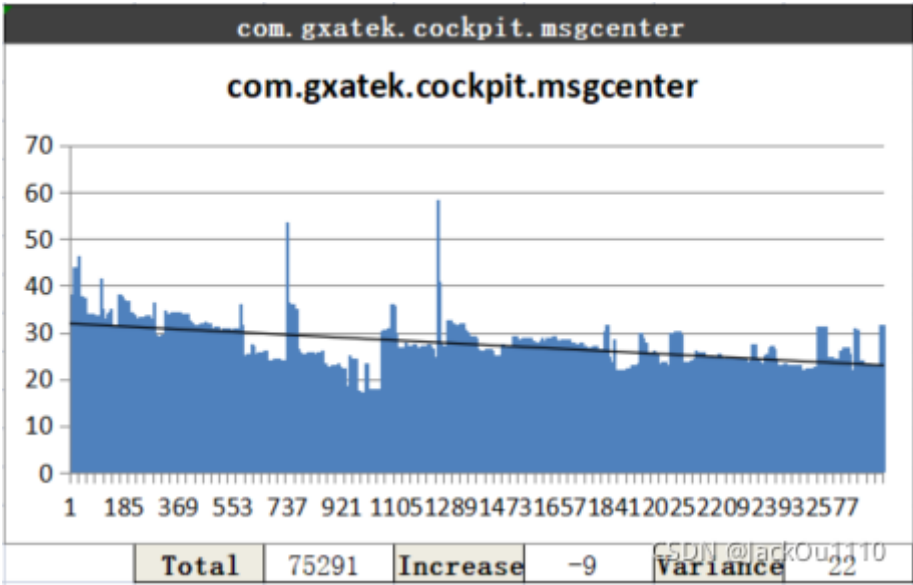


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
40M	14M	28M	存在	少量	存在

优化建议:

- 数据埋点模块加载数据上报的对象是否可以复用
- 账号模块的Bean对象是否可以复用
- 调查大对象创建和回收时机，尽量复用对象。

2.2.2.20 消息中心

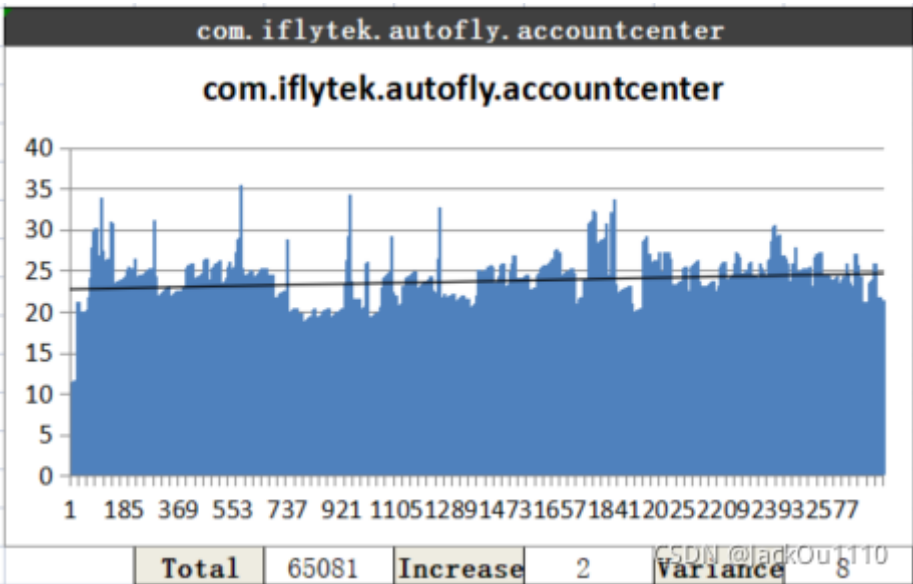


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
58M	17M	27M	存在	少量	存在

优化建议：

- 调查大对象创建情况，实现对象复用机制

2.2.2.21 讯飞账号中心

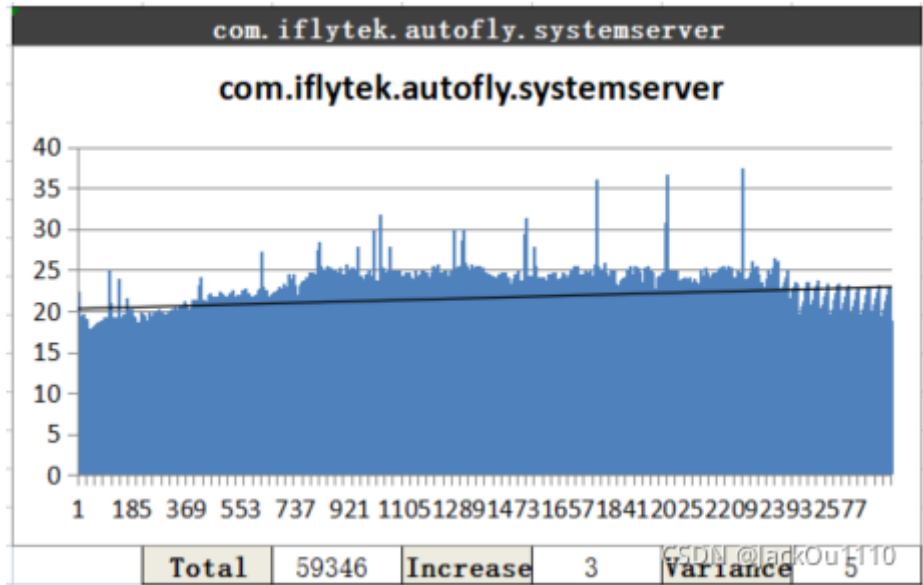


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
35M	11M	24M	存在	少量	存在

优化建议：

- 调查大对象分配与释放场景
- 复用内存减少内存抖动

2.2.2.22 讯飞SystemServer

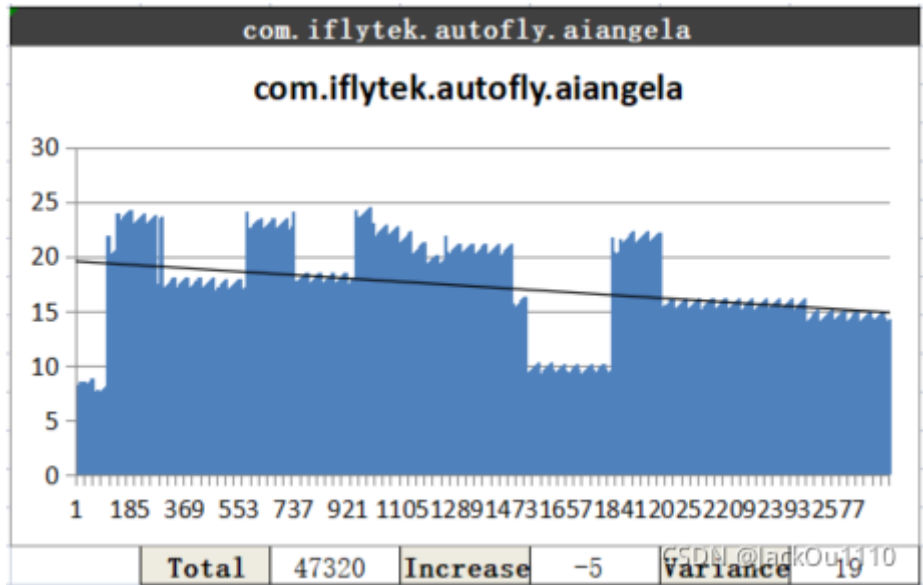


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
37M	16M	22M	少量	少量	少量

优化建议:

- 调查大对象创建场景是否在循环中创建对象，此对象是否可以new一个复用。

2.2.2.23 讯飞aiangela

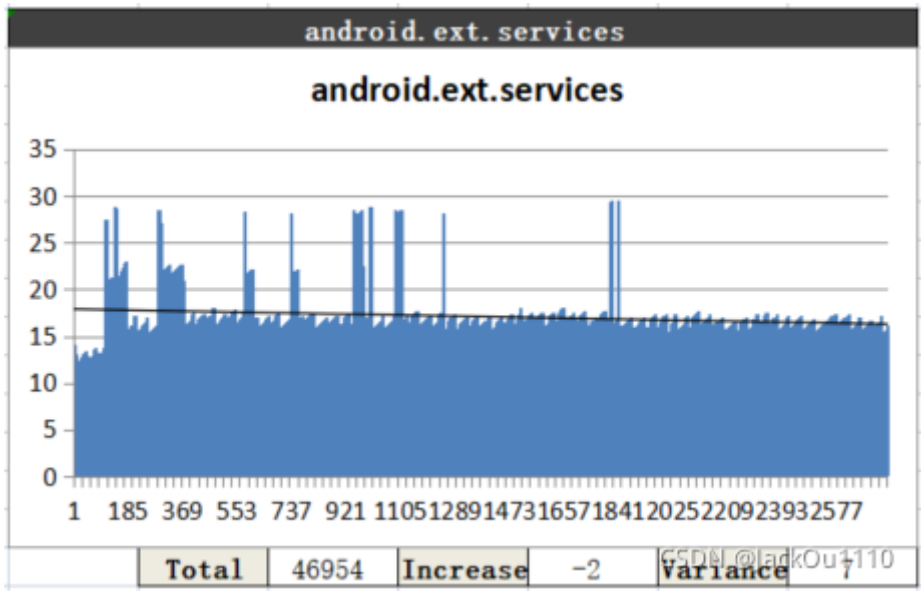


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
25M	7M	17M	少量	少量	存在

优化建议:

- 单独跑monkey,检测应用稳定性

2.2.2.24 扩展service

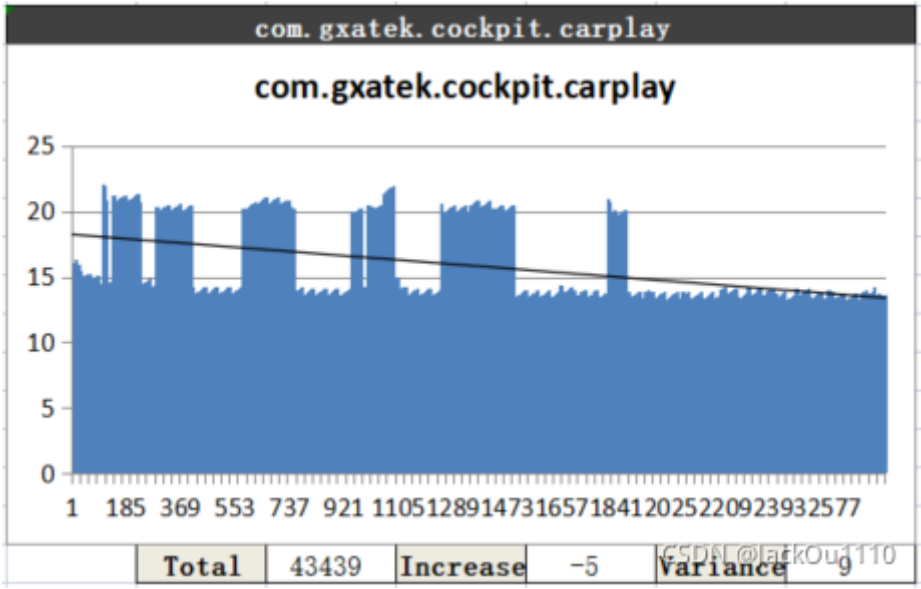


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
30M	12M	17M	存在	不存在	少量

优化建议:

- 调查大对象创建场景

2.2.2.25 carplay

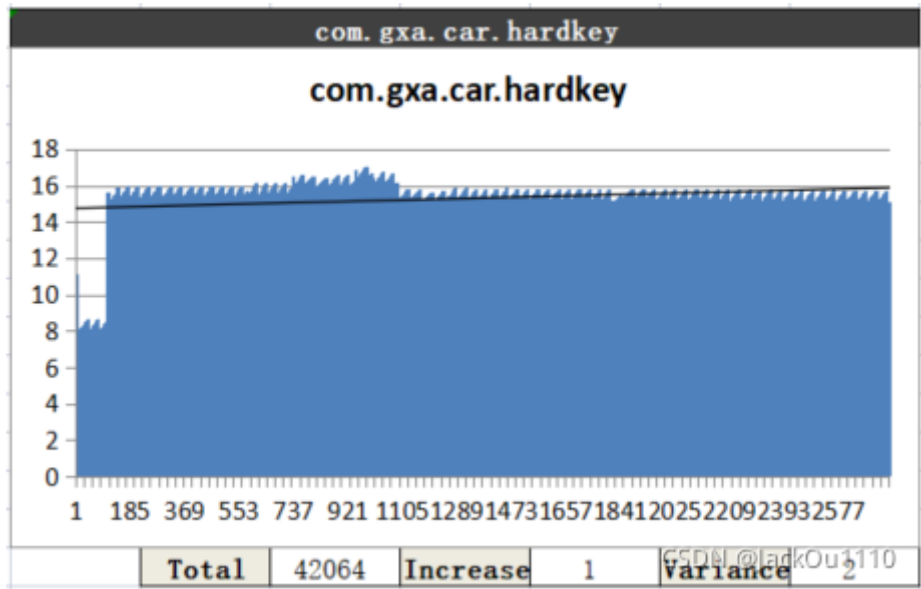


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
22M	13M	16M	少量可以接受	少量可接受	存在

优化建议:

- 单独跑monkey验证稳定性

2.2.2.26 hardkey

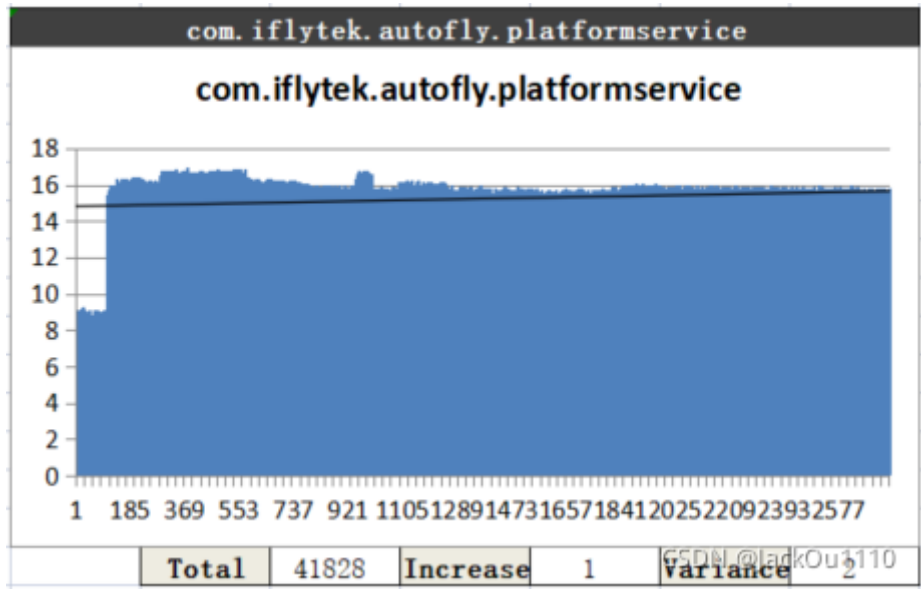


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
17M	8M	15M	少量可接受	少量可接受	存在

优化建议：

- 调查监控前期突然升高是否是启动服务，如果是可以暂时不优化。

2.2.2.27 讯飞平台服务

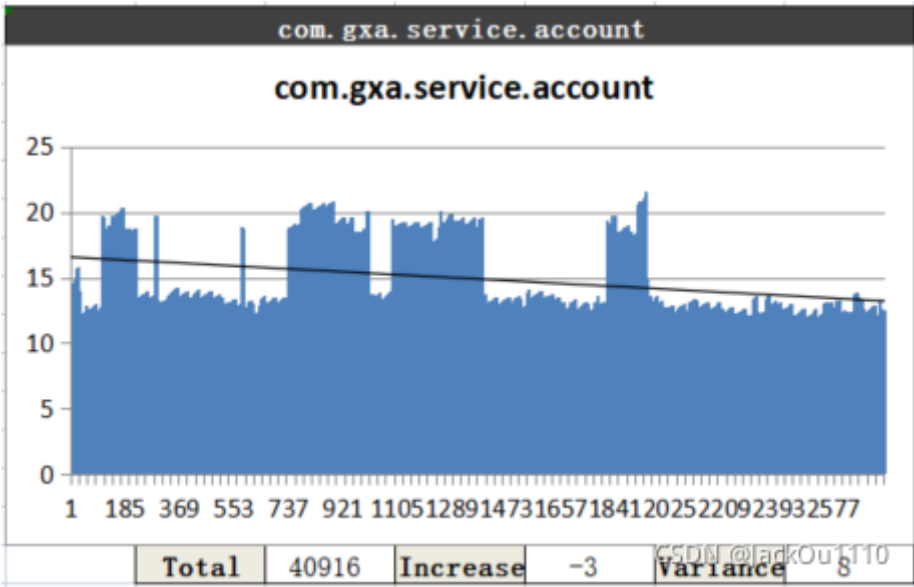


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
17M	8M	15M	少量可接受	暂无	存在

优化建议：

- 调查监控前期突然升高是否是启动服务，如果是可以暂时不优化。

2.2.2.28 账号服务端

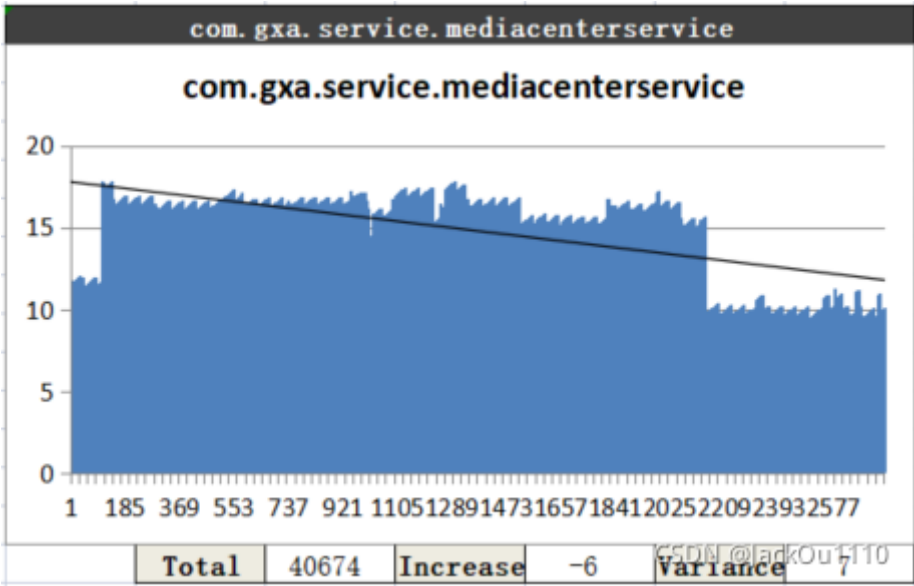


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
22M	11M	15M	少量可接受	暂无	存在

优化建议：

- 单独跑monkey验证稳定性

2.2.2.29 媒体中心服务

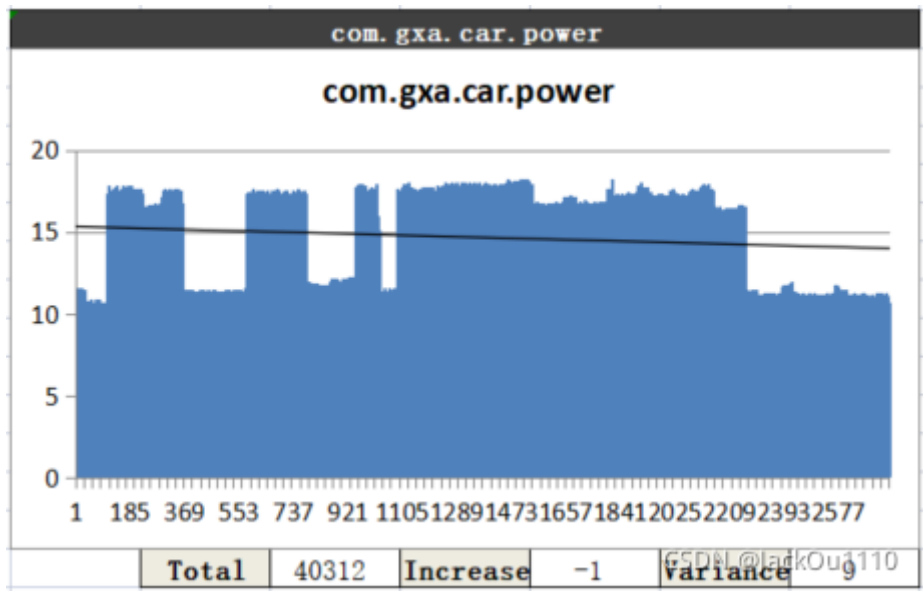


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
18M	9M	15M	少量	少量	存在

优化建议：

- 存在少量内存突然抖动场景，需要调查一下发生场景
- 内存图后半部分出现阶梯内存，需要单独跑monkey验证稳定性

2.2.2.30 Power

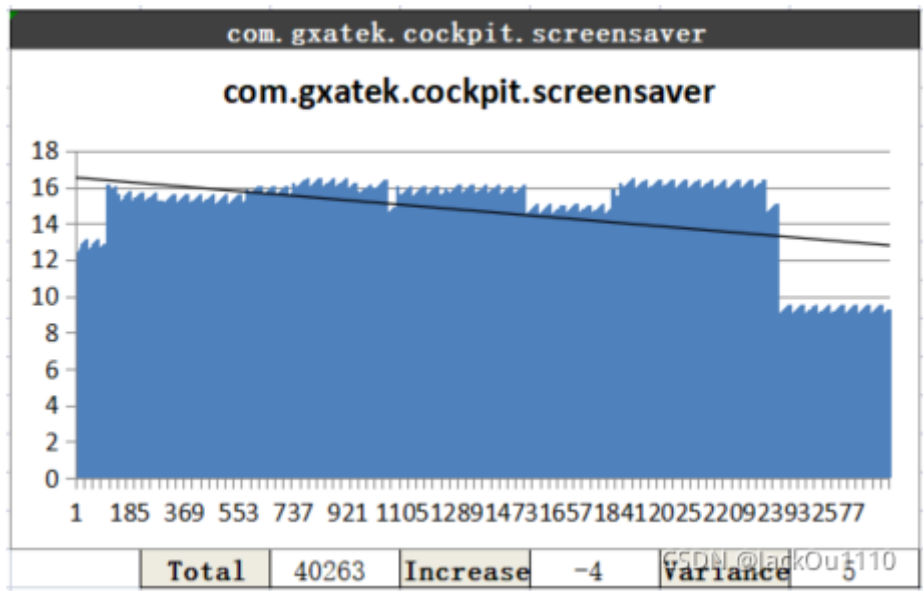


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
18M	10M	15M	少量	暂无	存在

优化建议：

- 图中看到大量阶梯内存，需要单独跑monkey验证稳定性

2.2.2.31 screensaver

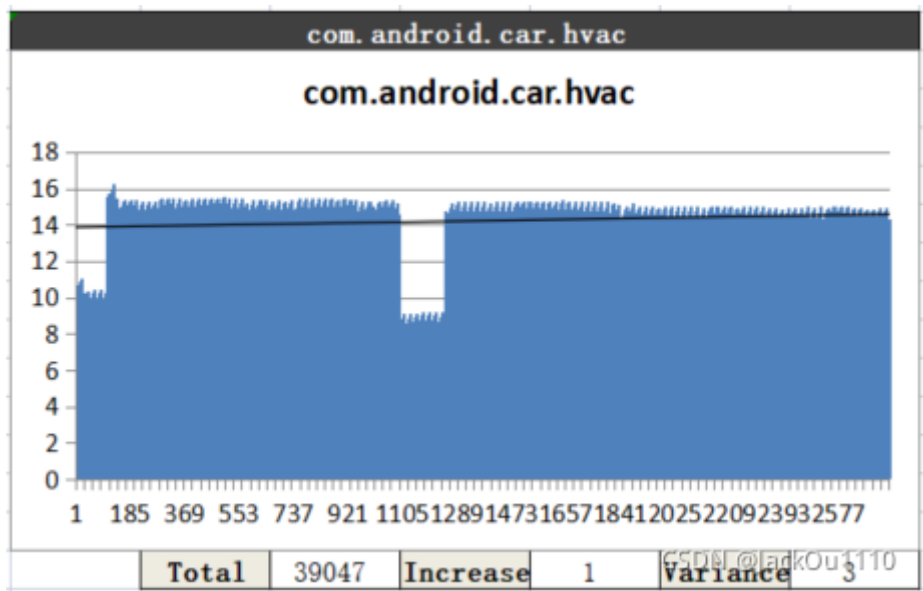


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
17M	9M	15M	少量	少量可接受	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.32 Car.hvac

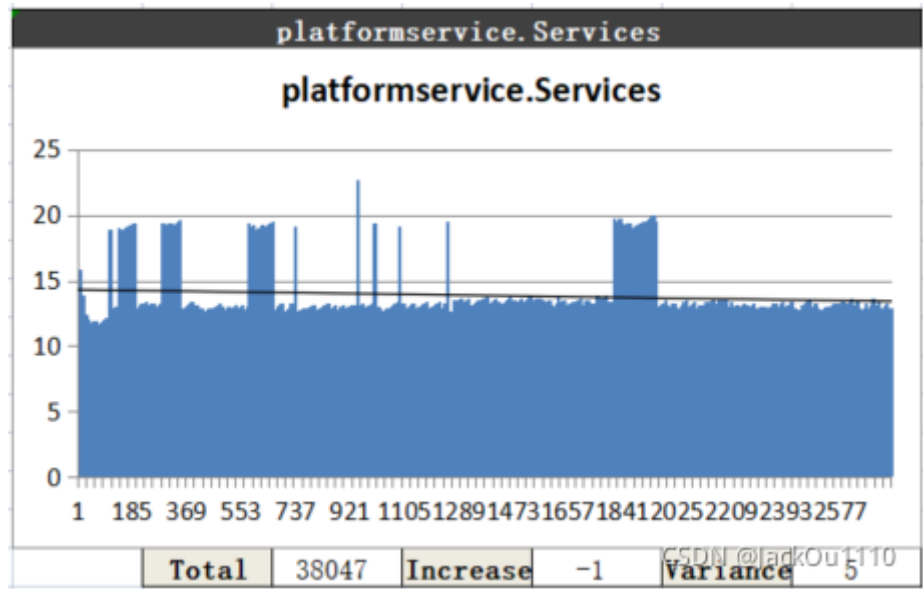


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
16M	8M	14M	少量可接受	少量可接受	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.33 platformservice.Services

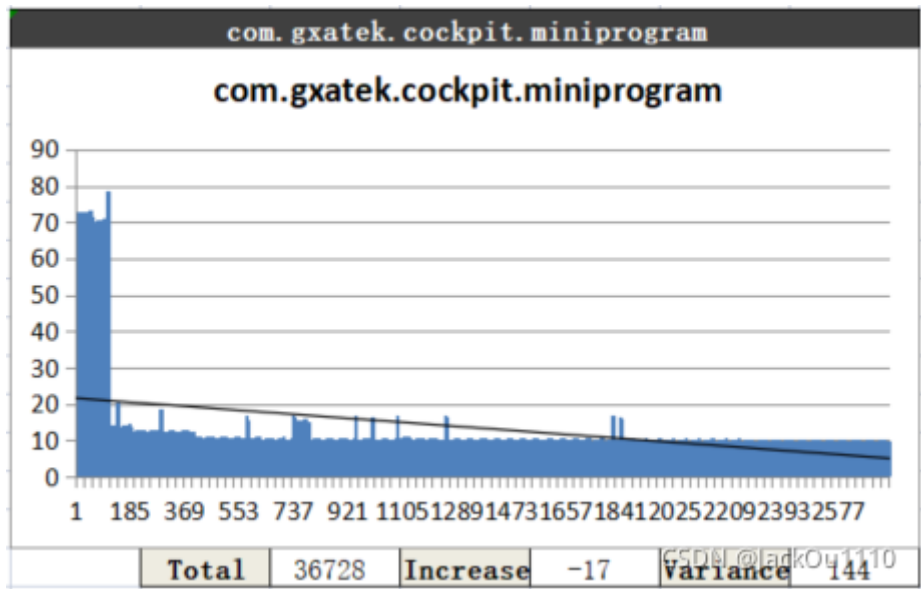


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
23M	11M	14M	少量可接受	暂无	存在

优化建议：

- 需要调查是稳定性还是在某些场景创建了大对象。

2.2.2.34 miniprogram

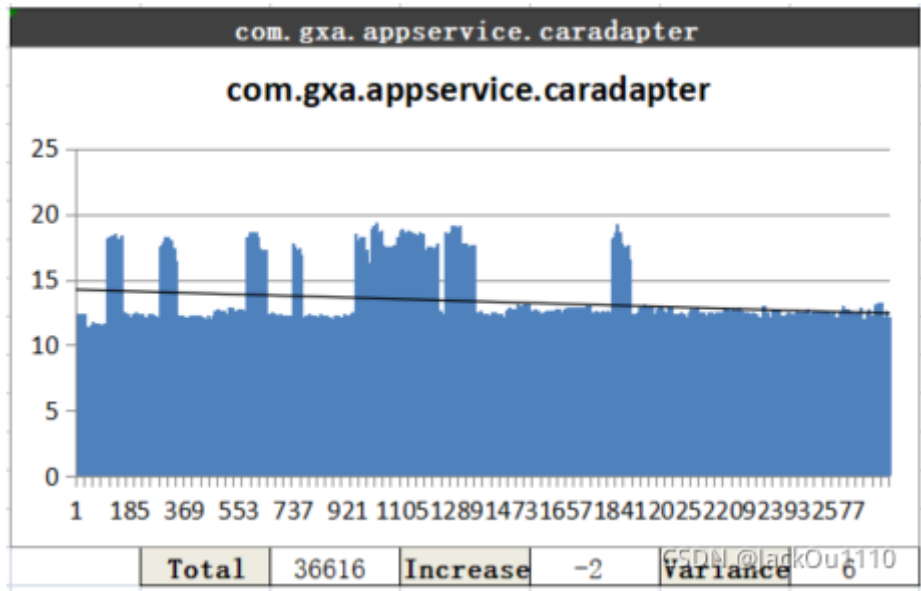


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
79M	10M	13M	少量可接受	暂无	存在

优化建议:

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.35 caradapter

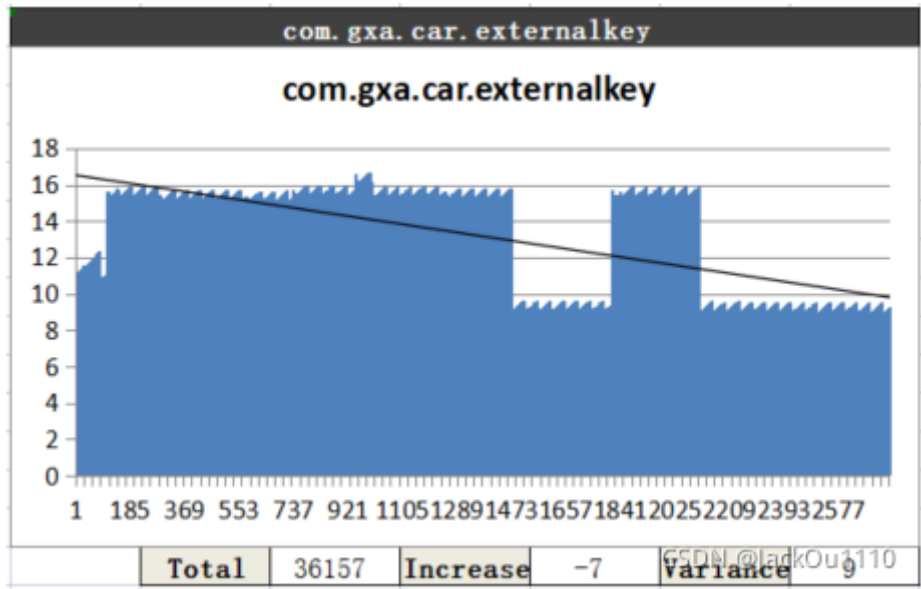


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
19M	10M	13M	少量可接受	少量可接受	存在

优化建议:

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性
- 调查某些场面分配大对象

2.2.2.36 externalkey

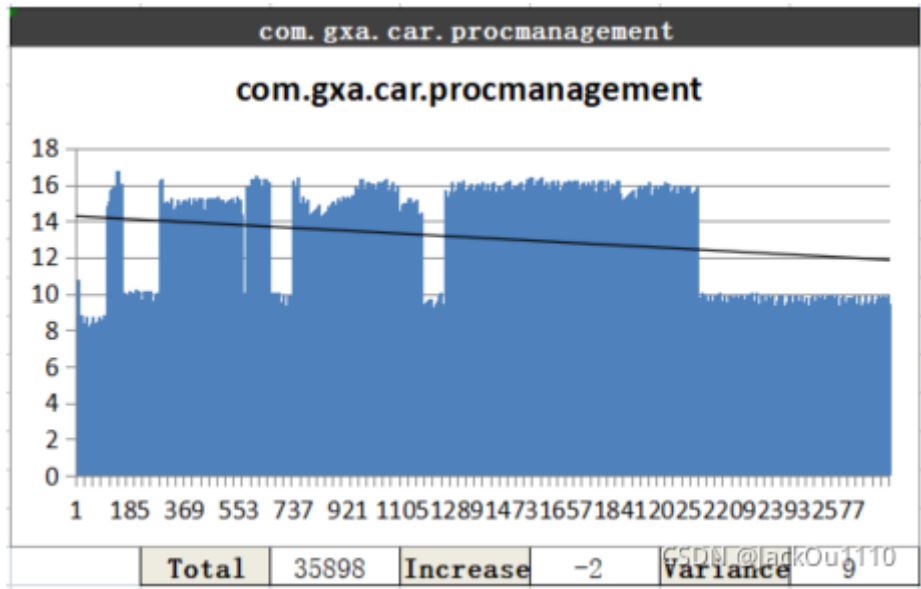


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
17M	9M	13M	少量可接受	暂无	存在

优化建议:

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.37 procmanagement

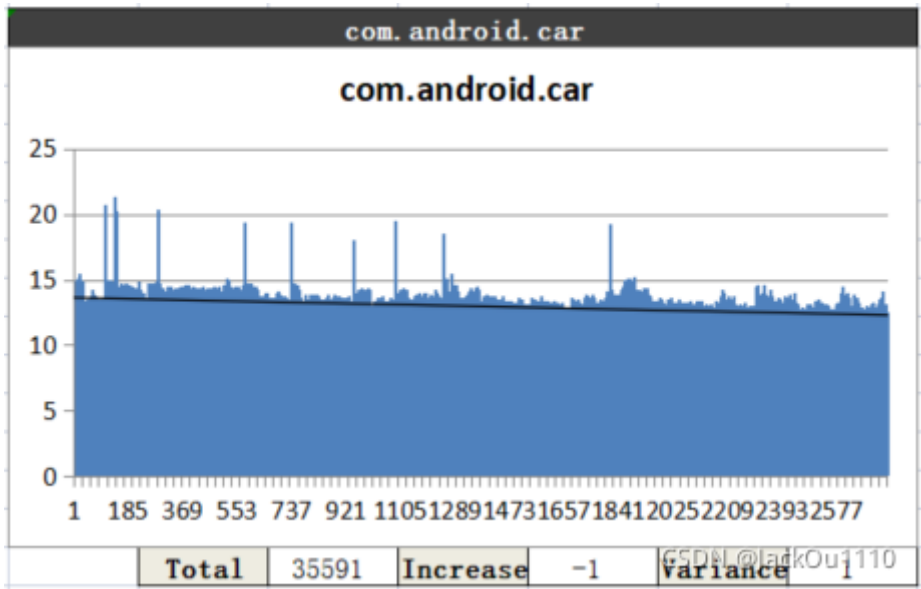


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
17M	8M	13M	少量可接受	暂无	存在

优化建议:

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.38 原生CAR

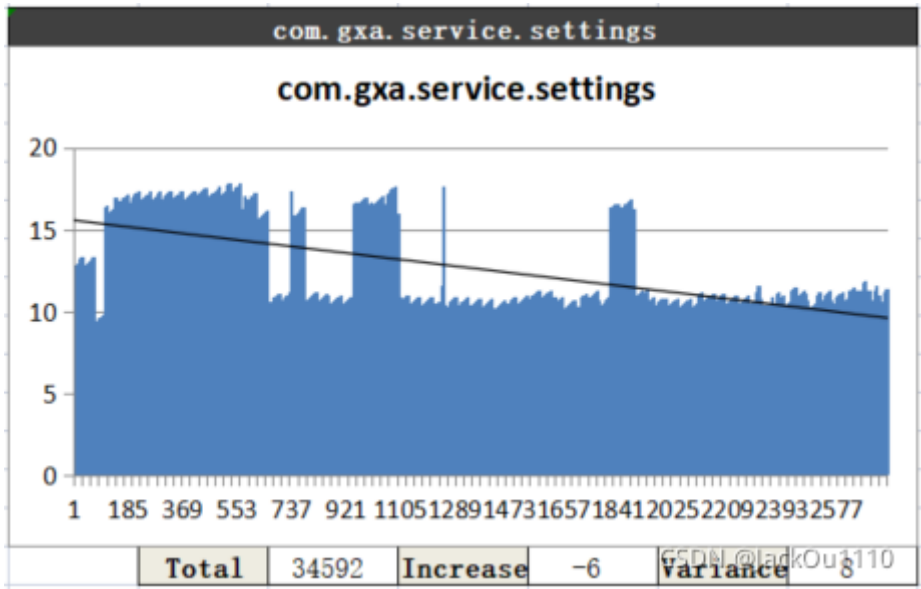


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
21M	11M	13M	少量	少量可接受	暂无

优化建议：

- 调查分配大对象场景

2.2.2.39 设置服务端

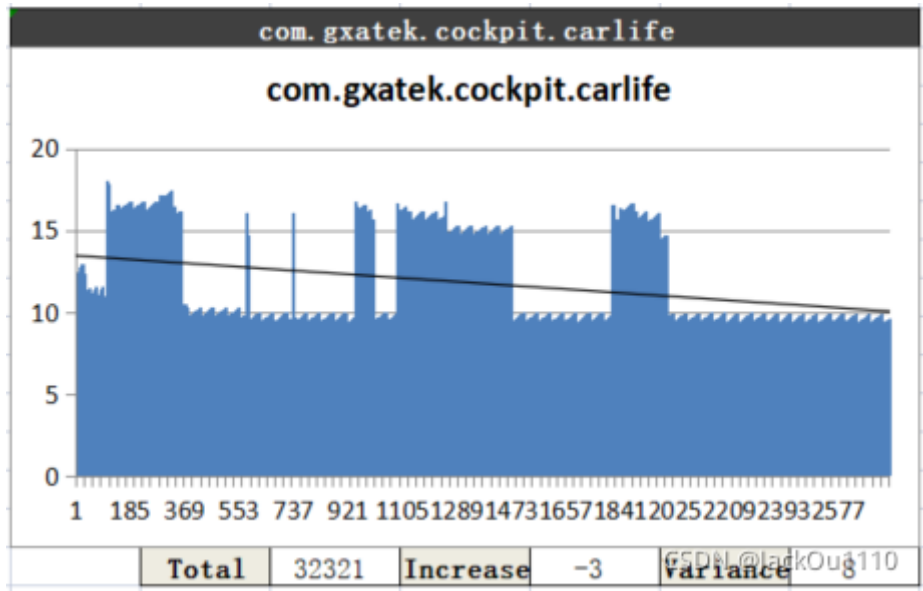


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
18M	9M	13M	少量	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.40 carlife

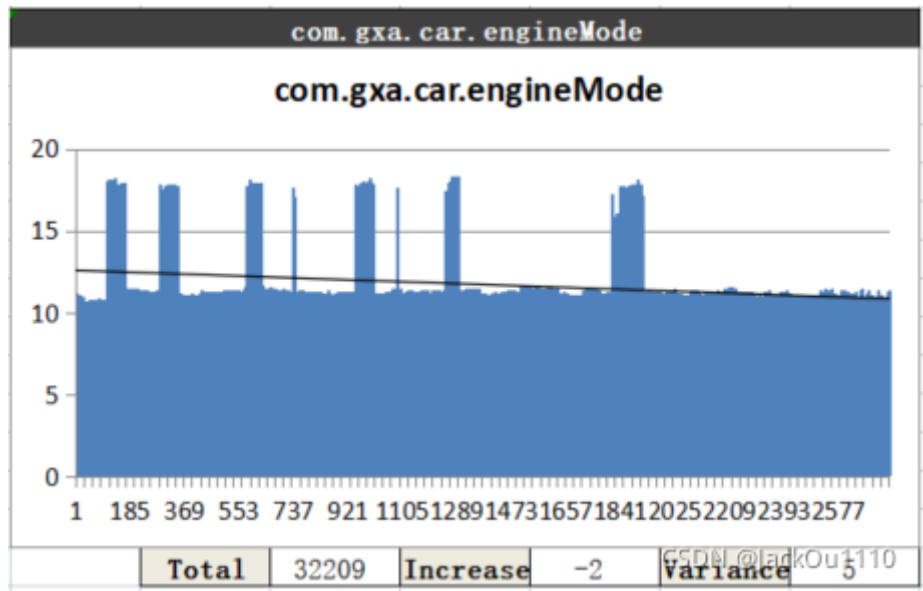


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
18M	9M	12M	少量可接受	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.41 engineMode

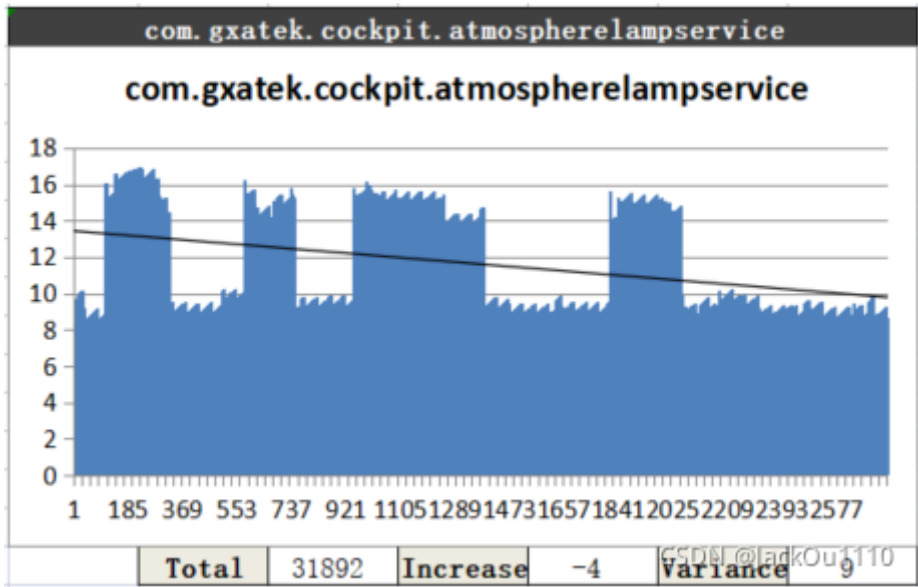


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
18M	10M	12M	暂无	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.42 氛围灯

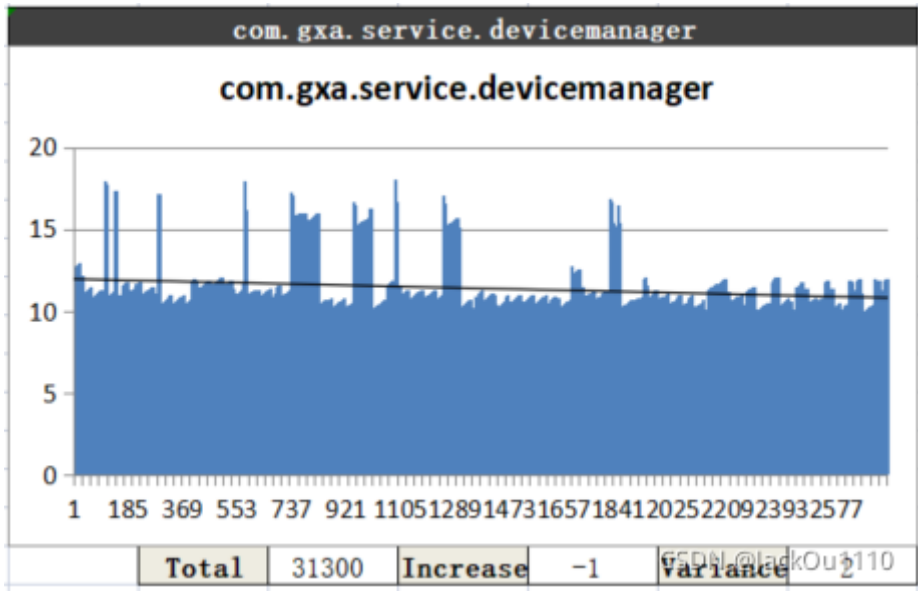


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
17M	9M	12M	少量可接受	少量可接受	存在

优化建议:

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.43 devicemanager

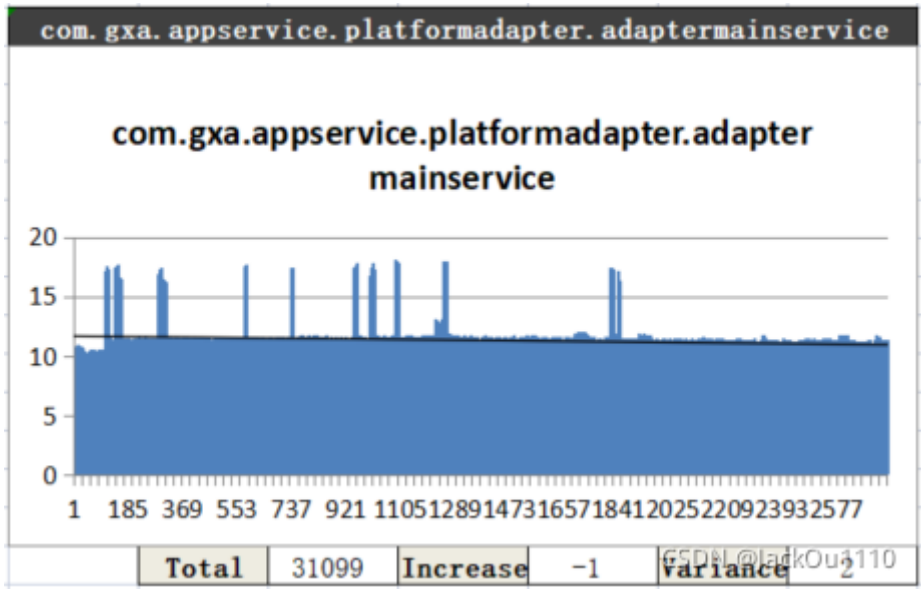


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
18M	10M	11M	少量可接受	暂无	存在

优化建议:

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性
- 调查部分场景分配大内存情况

2.2.2.44 adaptermainservice

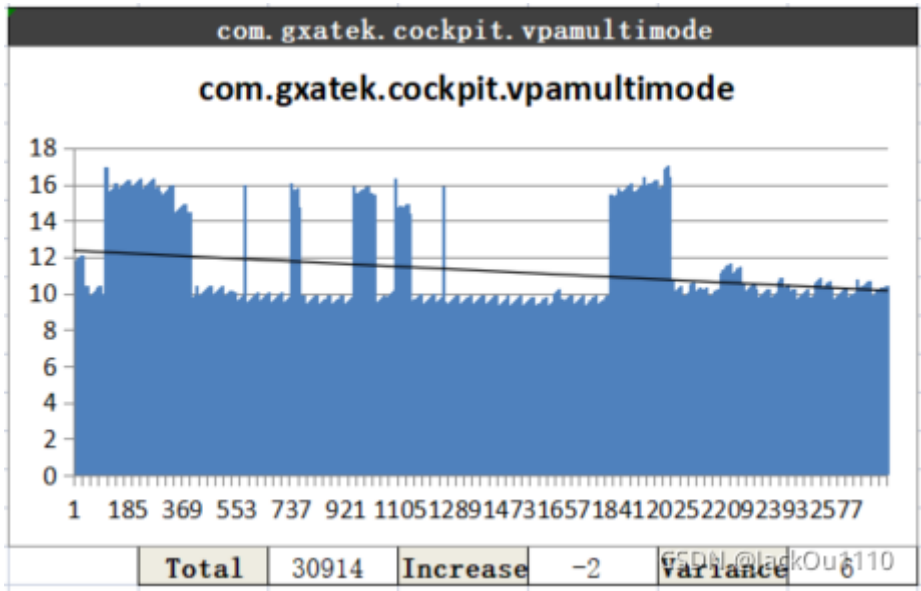


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
18M	9M	11M	少量	暂无	暂无

优化建议:

- 调查部分场景分配大内存情况

2.2.2.45 vpamultimode

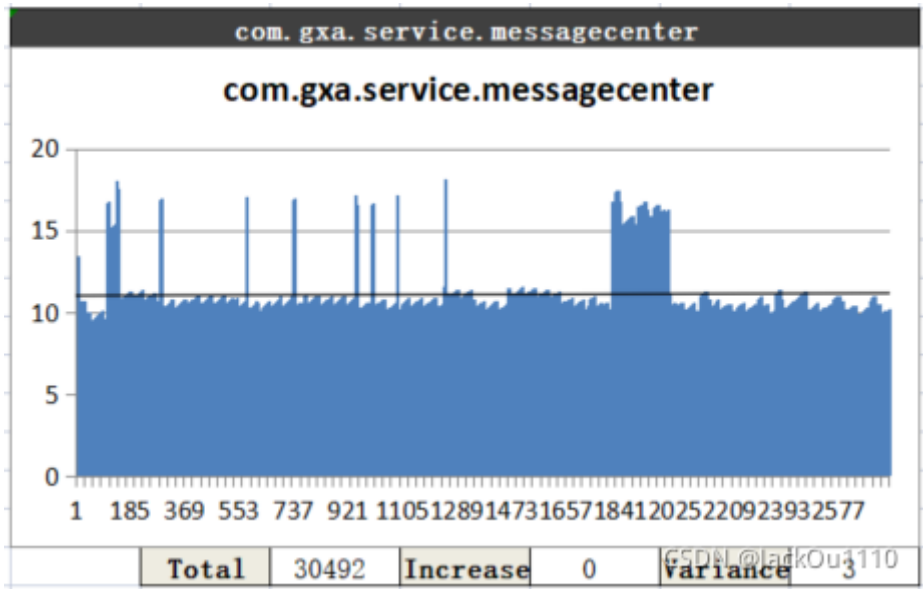


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
17M	9M	11M	少量	暂无	存在

优化建议:

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性
- 调查部分场景分配大内存情况

2.2.2.46 消息中心服务端

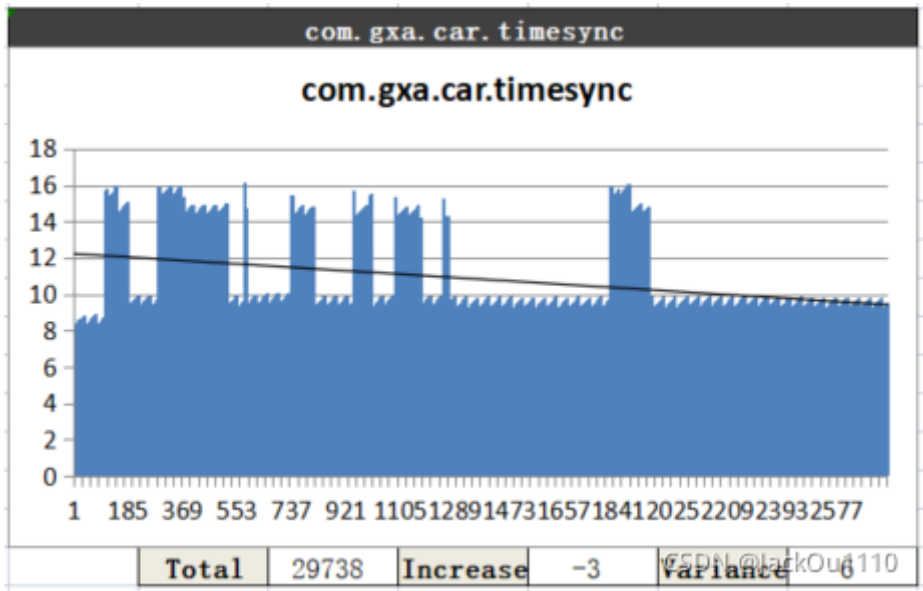


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
18M	9M	11M	少量	暂无	存在

优化建议:

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性
- 调查部分场景分配大内存情况

2.2.2.47 时钟同步

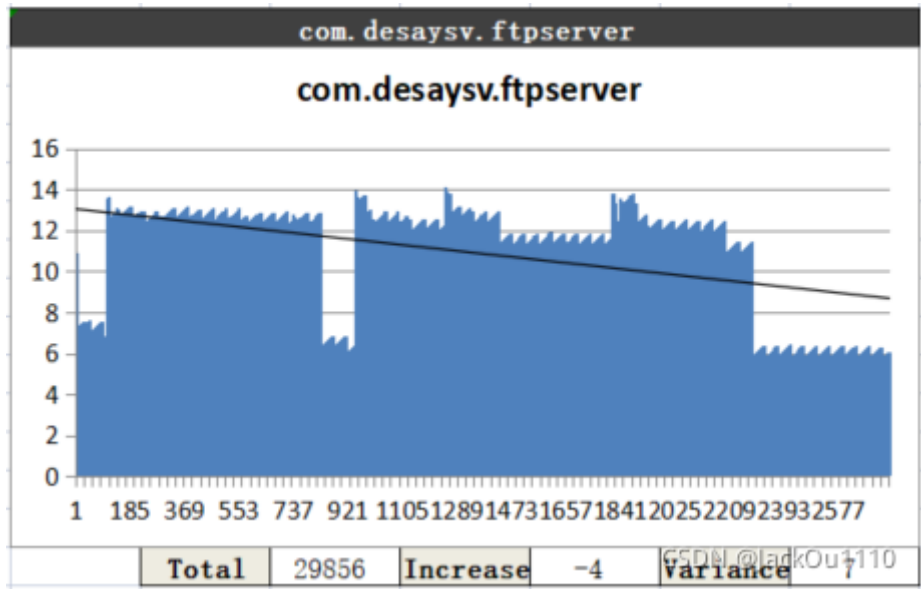


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
16M	8M	11M	少量	暂无	存在

优化建议:

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.48 德赛ftpserver

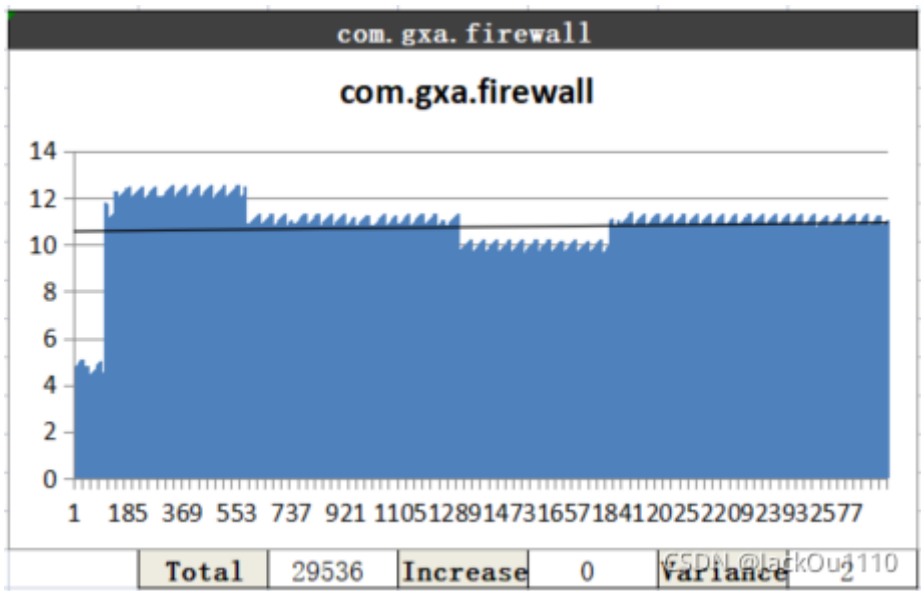


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
14M	6M	11M	存在	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性（已经提单稳定性问题）

2.2.2.49 防火墙

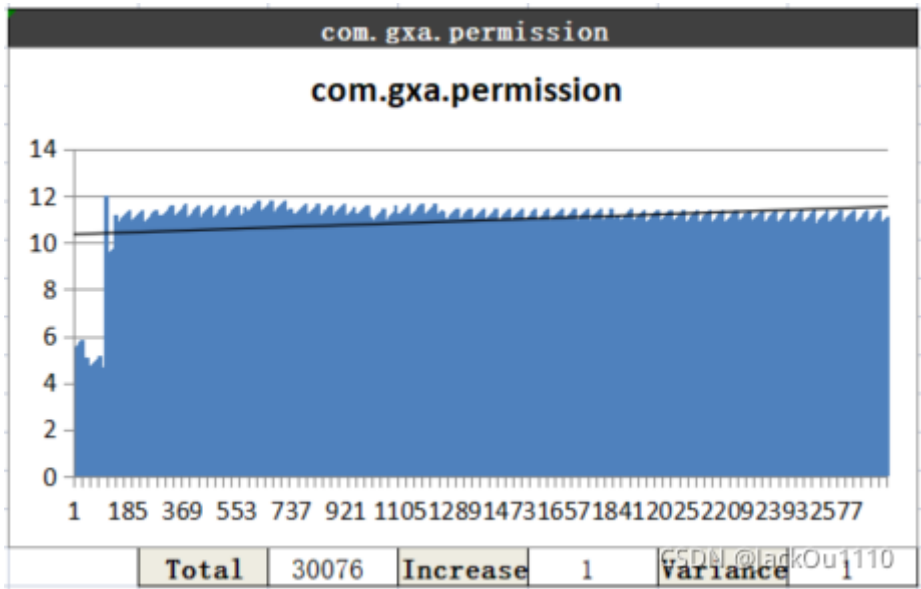


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
13M	4M	11M	少量	暂无	存在

优化建议：

- 需要调查图中前期的阶梯是否在启动服务

2.2.2.50 权限管理服务

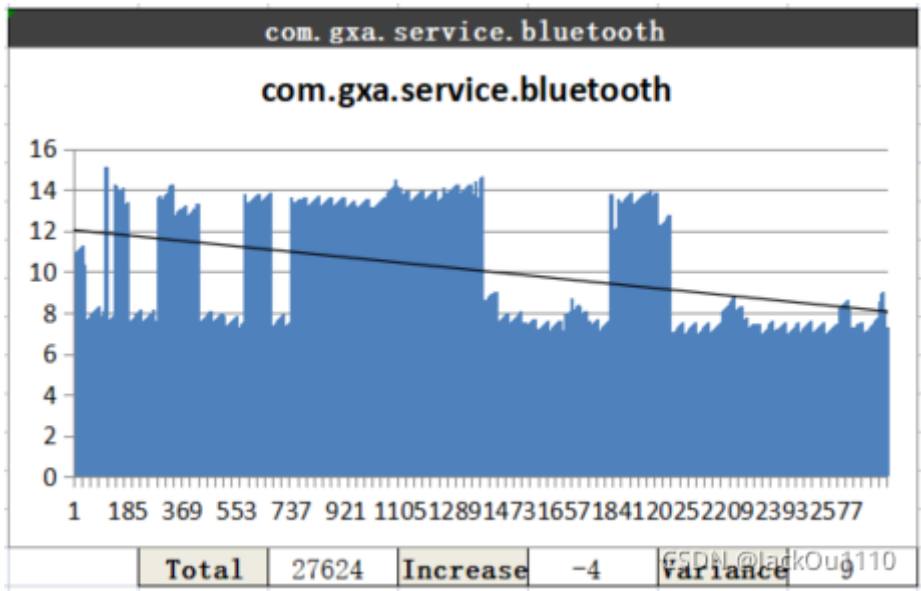


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
12M	5M	11M	少量可接受	暂无	存在

优化建议：

- 需要调查图中前期的阶梯是否在启动服务

2.2.2.51 蓝牙服务

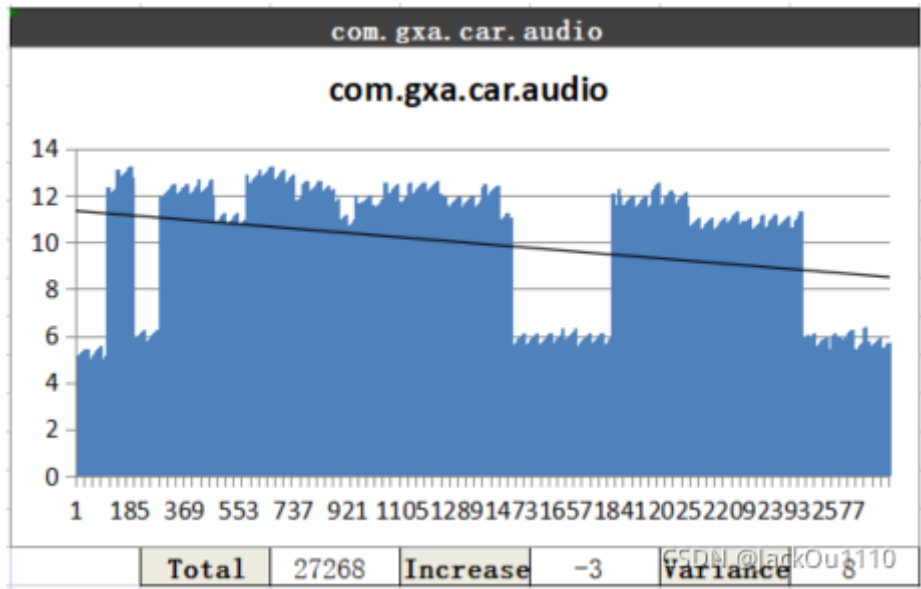


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
15M	7M	10M	少量	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.52 CarAudio

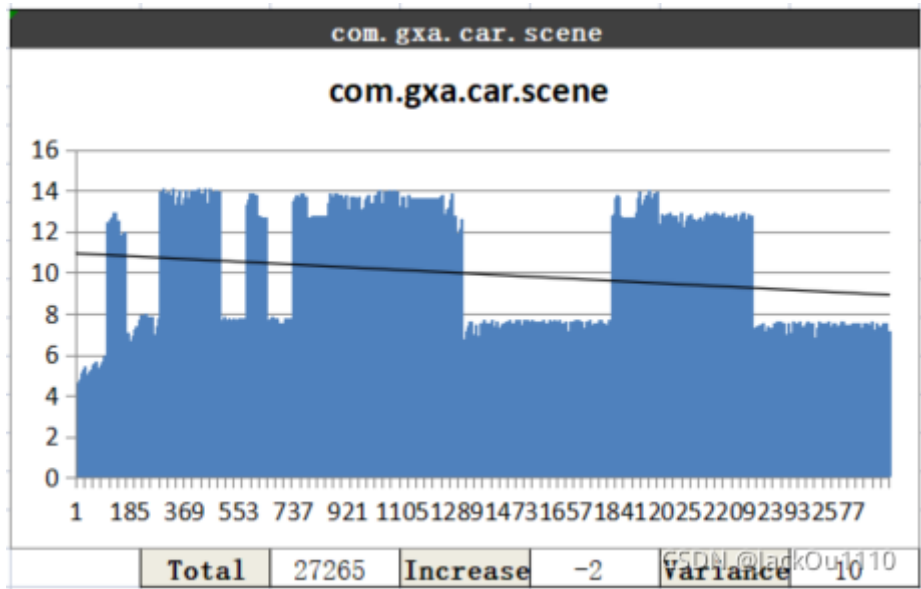


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
13M	5M	10M	少量	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.53 CarScene

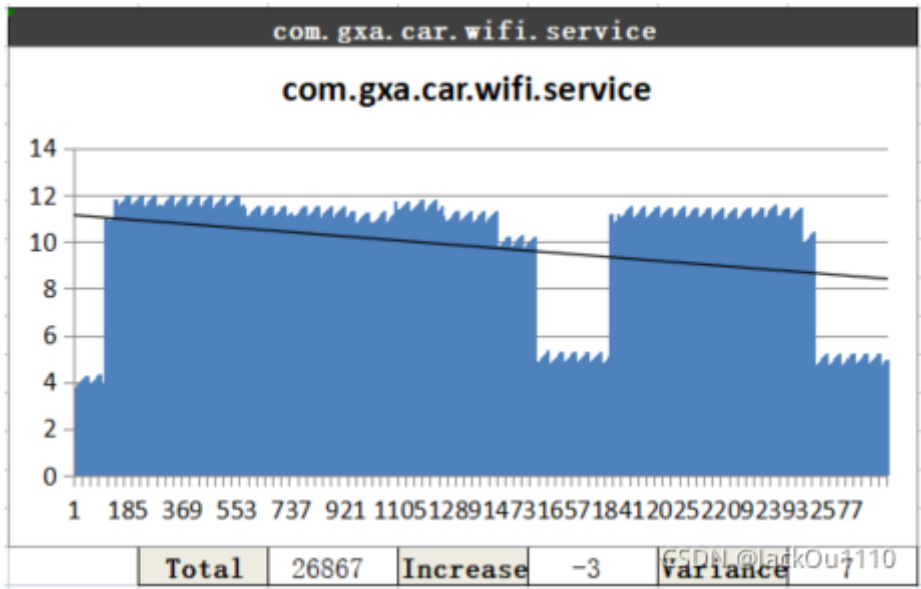


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
14M	4M	10M	少量	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.54 Wifi服务端

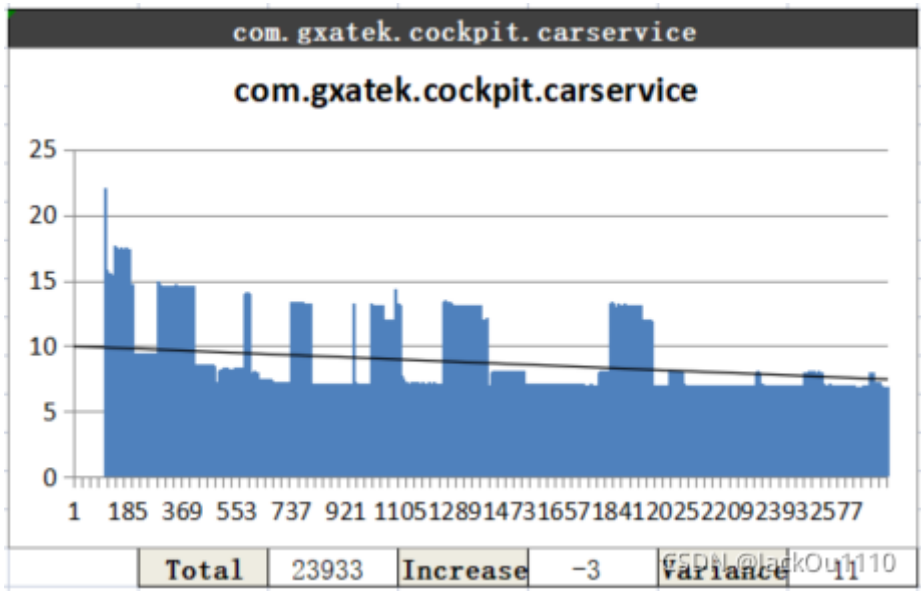


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
12M	4M	10M	少量	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.55 CarService

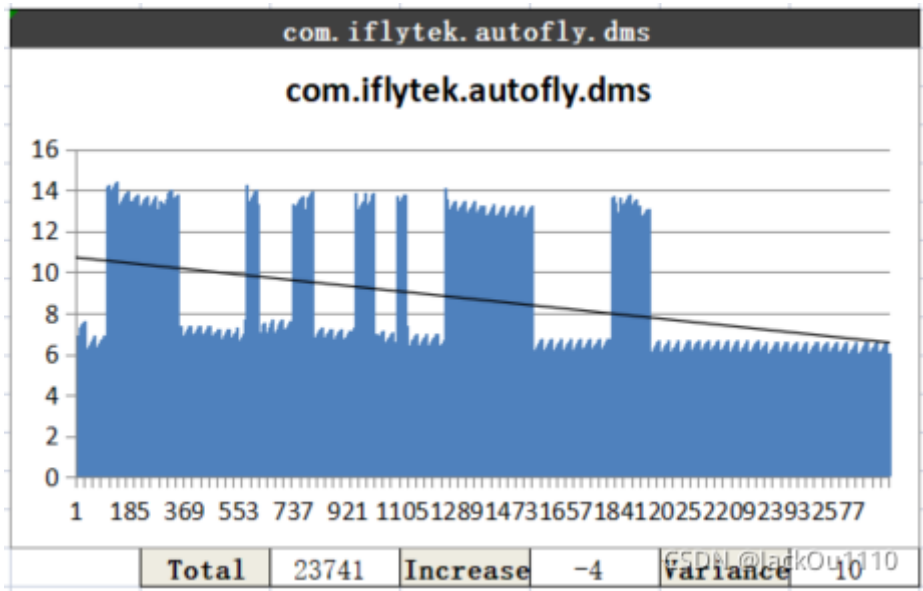


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
17M	7M	9M	少量	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.56 DMS

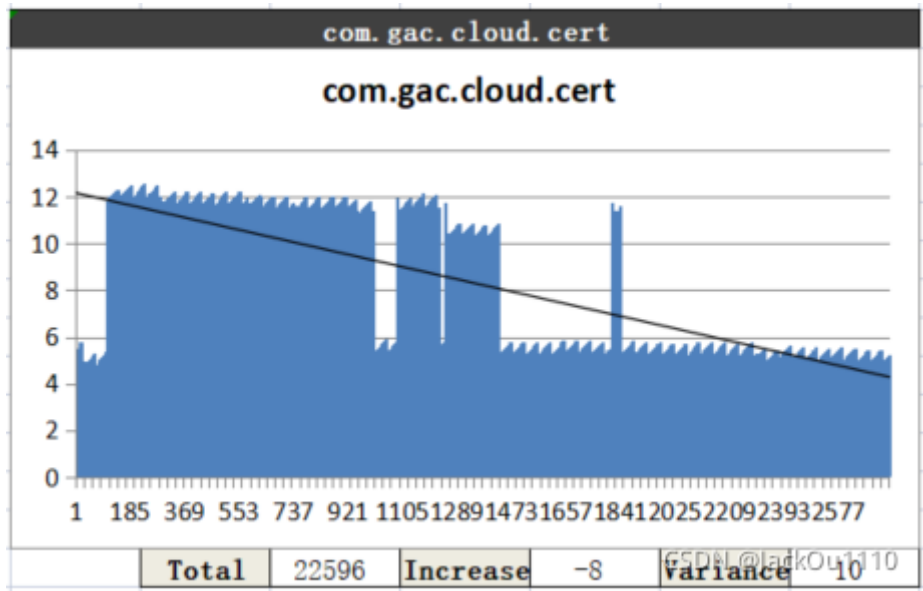


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
14M	6M	9M	少量	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.57 GCS证书管理

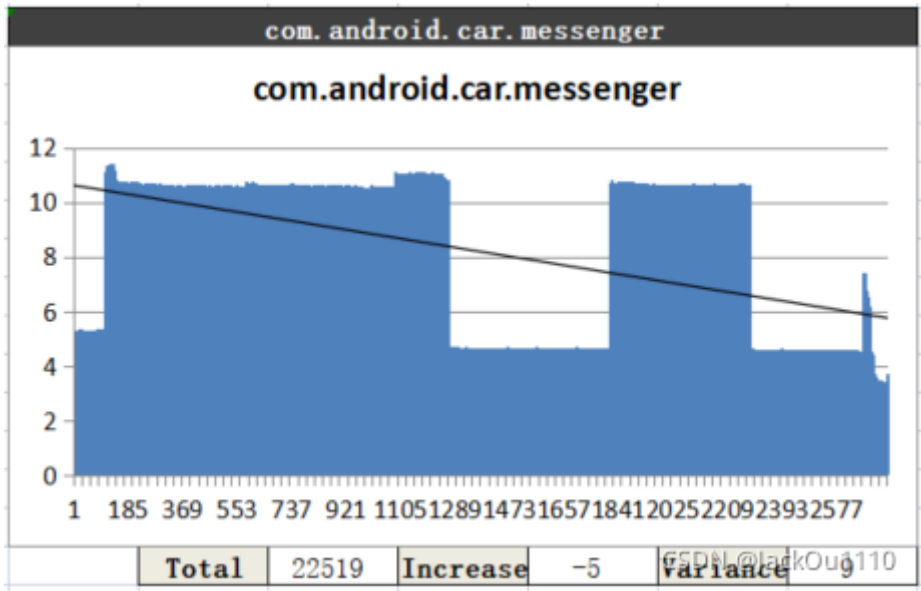


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
13M	5M	8M	少量	暂无	存在

优化建议：

- 不上此部分代码，此部分功能由pkisdk完成

2.2.2.58 CarMessenger

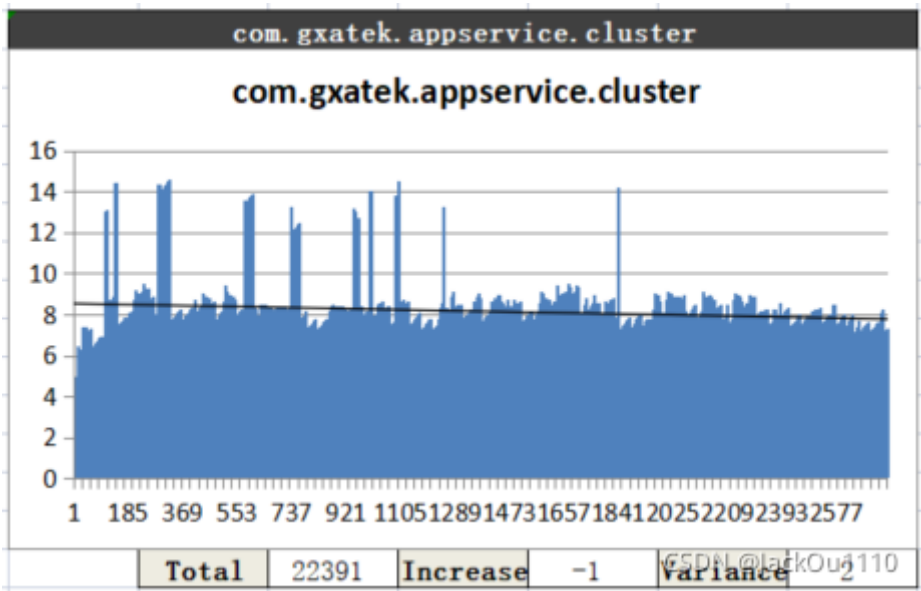


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
11M	3M	8M	暂无	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.59 仪表服务

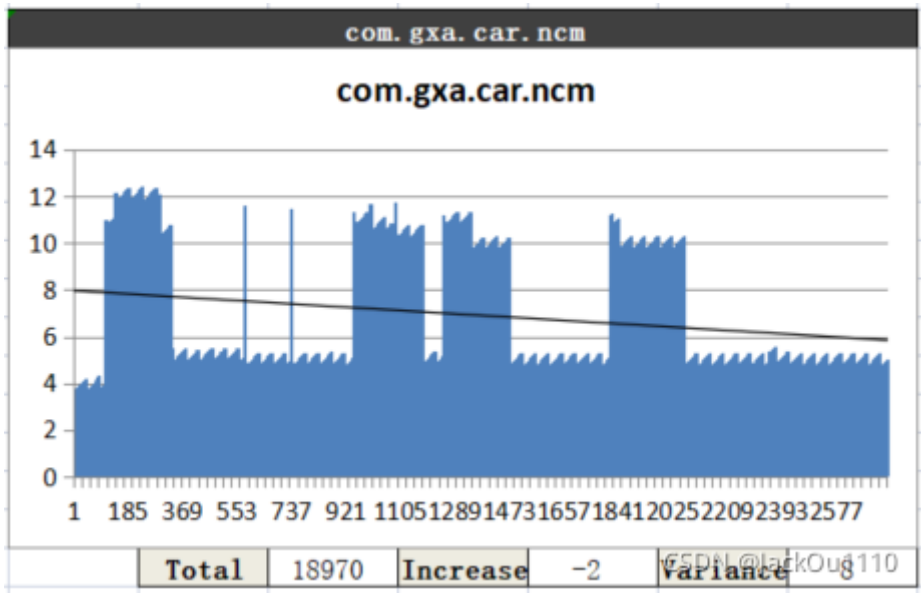


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
15M	4M	8M	存在	暂无	存在

优化建议：

- 调查大对象申请情况，实现对象复用

2.2.2.60 CarNcm

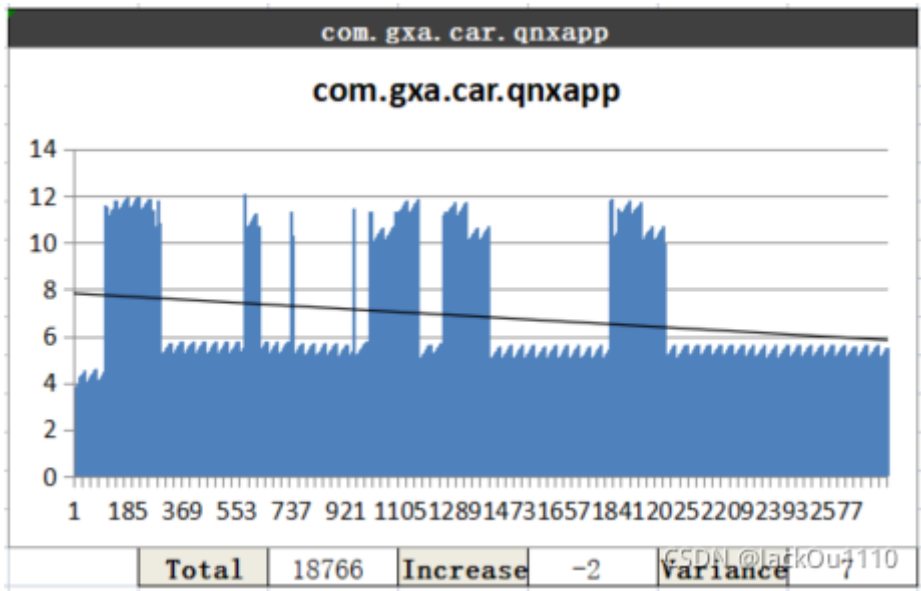


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
12M	3M	7M	少量	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.61 德赛QnxApp

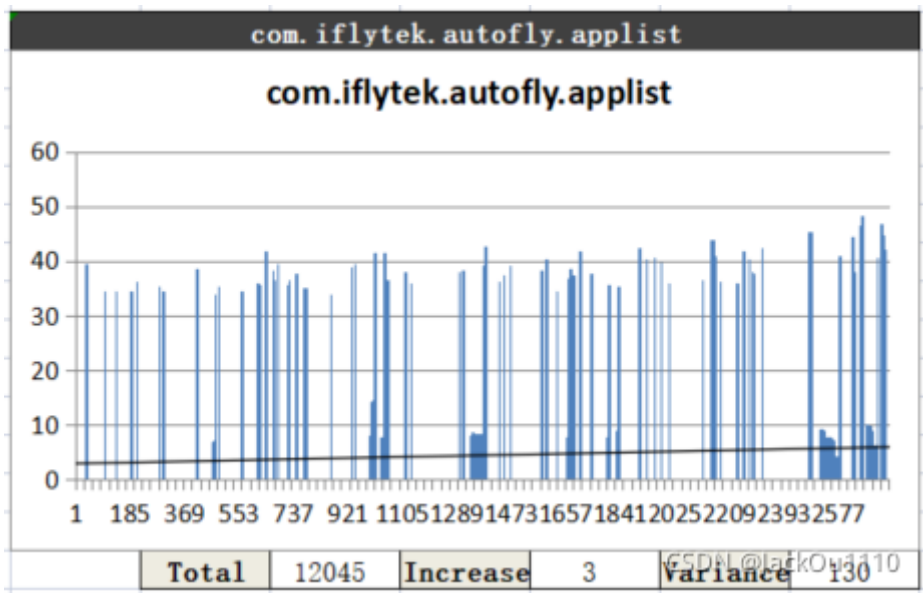


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
12M	4M	7M	少量	暂无	存在

优化建议：

- 图中看到存在阶梯内存，需要单独跑monkey验证稳定性

2.2.2.62 讯飞applist

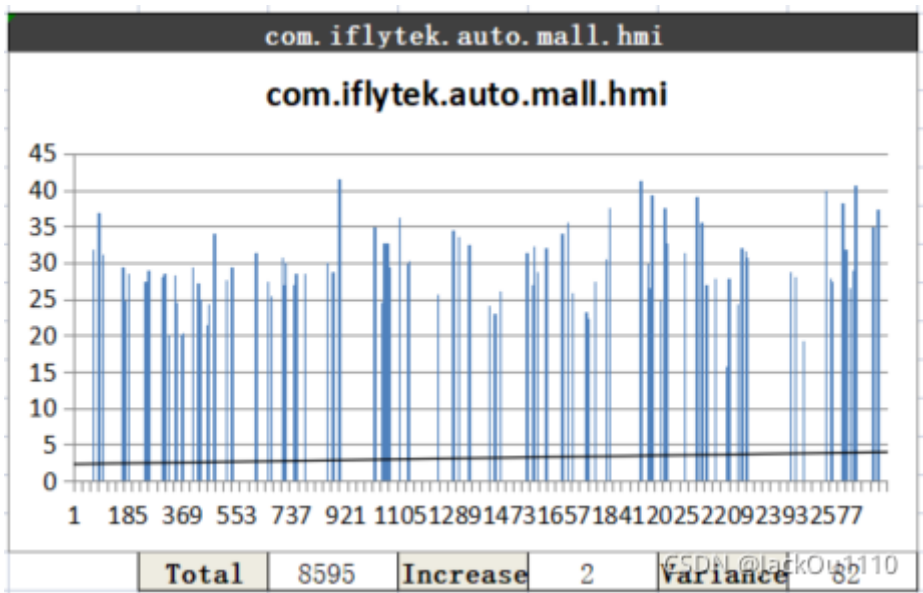


最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
48M	0M	4M	存在	少量	存在

优化建议：

- 需要调查该应用实现什么功能，是否可以合并到其他应用中。疑似此进程用完就被杀了

2.2.2.63 商城



最大内存	最小内存	平均内存	内存抖动状态	是否内存泄露	是否阶梯内存
41M	0M	3M	存在	少量	存在

优化建议：

- 需要调查该应用实现什么功能，是否可以合并到其他应用中。疑似此进程用完就被杀了

3 车机内存性能问题总结

从以上系统整体性能与各应用内存性能分析可以发现如下几个问题：

- 系统总体内存存在随车机运行时间加长，剩余内存逐步降低，最后触发了LMK杀进程机制
- 大部分应用内存存在内存抖动
- 大部分应用内存存在内存泄露
- 部分应用存在阶梯内存，存在稳定性问题
- 部分应用存在大内存分配情况
- 部分应用自身总体内存占用比较高

4.优化建议

针对以上问题，请各FO参考自身应用内存图情况，选择对应的工具进行调查，修复内存问题。针对不同问题，监控调查工具建议如下：

- 针对有界面的应用可以使用LeakCanary，动态检测应用是否由内存泄露

```
// 在build.gradle中添加以下依赖即可，也可以看看参考文献3和4
dependencies {
    // debugImplementation because LeakCanary should only run in debug builds.
    debugImplementation 'com.squareup.leakcanary:leakcanary-android:2.7'
}
```

- 针对内存抖动和大内存分配调查，可以使用Android Studio的Profile中memory模块来检测。使用方法见**参考文献5**
- 针对服务和LeakCanary检测不出来的，可以使用MAT（Memory Analyzer Tools, MAT），工具下载见**参考文献6**
- 针对阶梯内存，存在稳定性情况，可以使用monkey，晚上针对单个应用跑monkey

```
# -p参数后面填入自己应用的包名
monkey -p com.timnetwork.appstore --throttle 500 --bugreport --hprof --
ignore-crashes --ignore-timeouts --ignore-security-exceptions --ignore-
native-crashes --monitor-native-crashes -v -v -v 300000 1 >>
/mnt/sdcard/Stability_base_20211023.log 2>>
/mnt/sdcard/Stability_base_err_20211023.log &
```

参考文献：

[1. linux - what is Inactive\(file\) and Active\(file\) in /proc/meminfo? - Stack Overflow](#)

[2. /PROC/MEMINFO description](#)

[3.Getting Started - LeakCanary\(square.github.io\)](#)

[4.square/leakcanary: A memory leak detection library for Android.\(github.com\)](#)

[5.使用内存性能分析器查看应用的内存使用情况 | Android 开发者 | Android Developers \(google.cn\)](#)

[6.Eclipse Memory Analyzer Open Source Project | The Eclipse Foundation](#)

