

Android内存优化

阿里巴巴集团 移动事业群-浏览器研发部 李跃辉
邮箱：liyuehui.lyh@alibaba-inc.com



Geekbang>

极客邦科技

全球领先的技术人学习和交流平台

扫我，码上开启新世界



Geekbang>

InfoQ | EGO NETWORKS | StuQ

InfoQ

专注中高端技术人员
的社区媒体

EGO NETWORKS

EXTRA GEEKS' ORGANIZATION
高端技术人员
学习型社交网络

StuQ

实践驱动的IT职业
学习和服务平台



促进软件开发领域知识与创新的传播



实践第一 案例为主

时间：2015年12月18-19日 / 地点：北京·国际会议中心

欢迎您参加ArchSummit北京2015, 技术因你而不同



ArchSummit北京二维码



【北京站】

2016年04月21日-23日



关注InfoQ官方信息
及时获取QCon演讲视频信息

自我介绍

全名：李跃辉

- 现在对应的产品--UC浏览器，负责国际版客户端研发
- 标准码农、属于那种不写代码闷的慌
- 13年工作经验，5年多Android。08年开始做手机软件开发，先后做过WinCE/Windows Mobile,Symbian,K-Java,iOS,Android。
- 擅长做新技术研究探索和数据分析



前有狼后有虎

- 四处乱窜的OOM
- 明明有剩余内存，但还是OOM
- 崩溃率长期居高不下
- 业务不断扩展需求



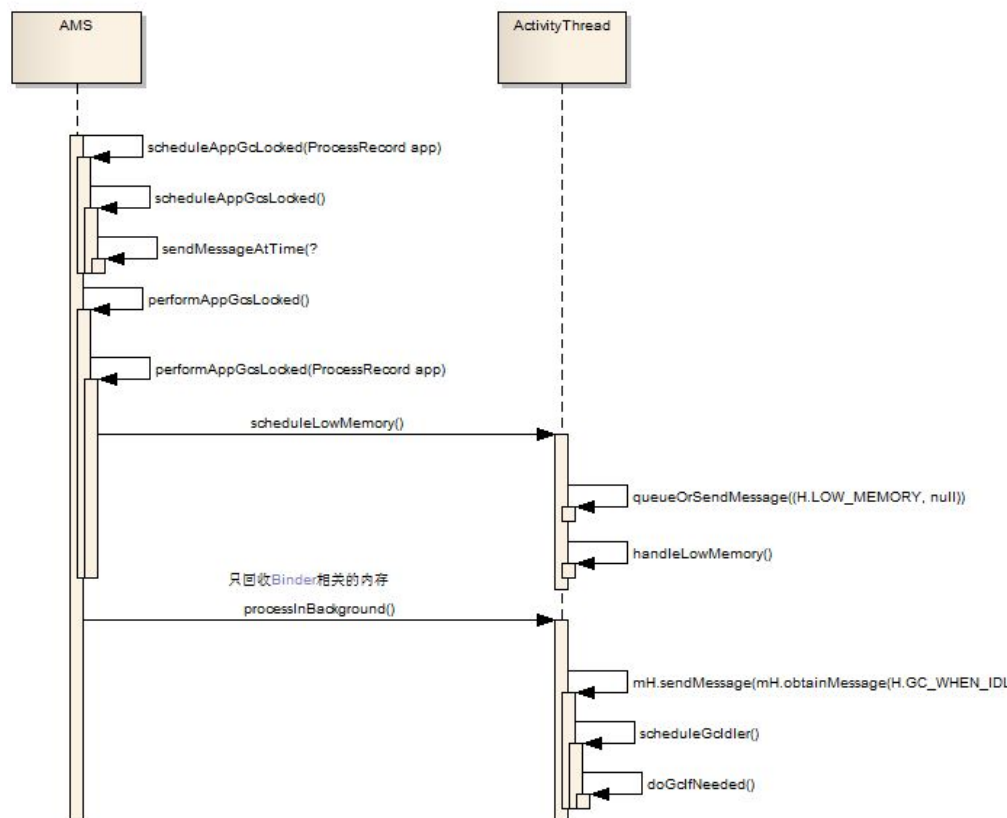
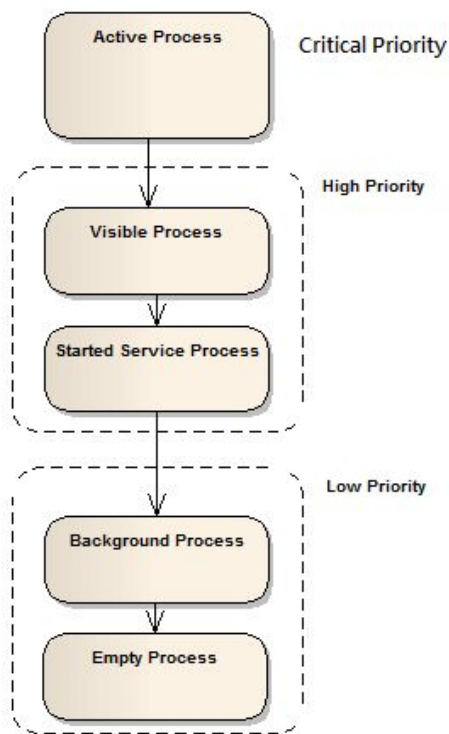
主要内容

- 系统内存基础知识
- 得到的优化启示
- 构建内存分析体系
- Native内存转换
- 其他内存优化心得



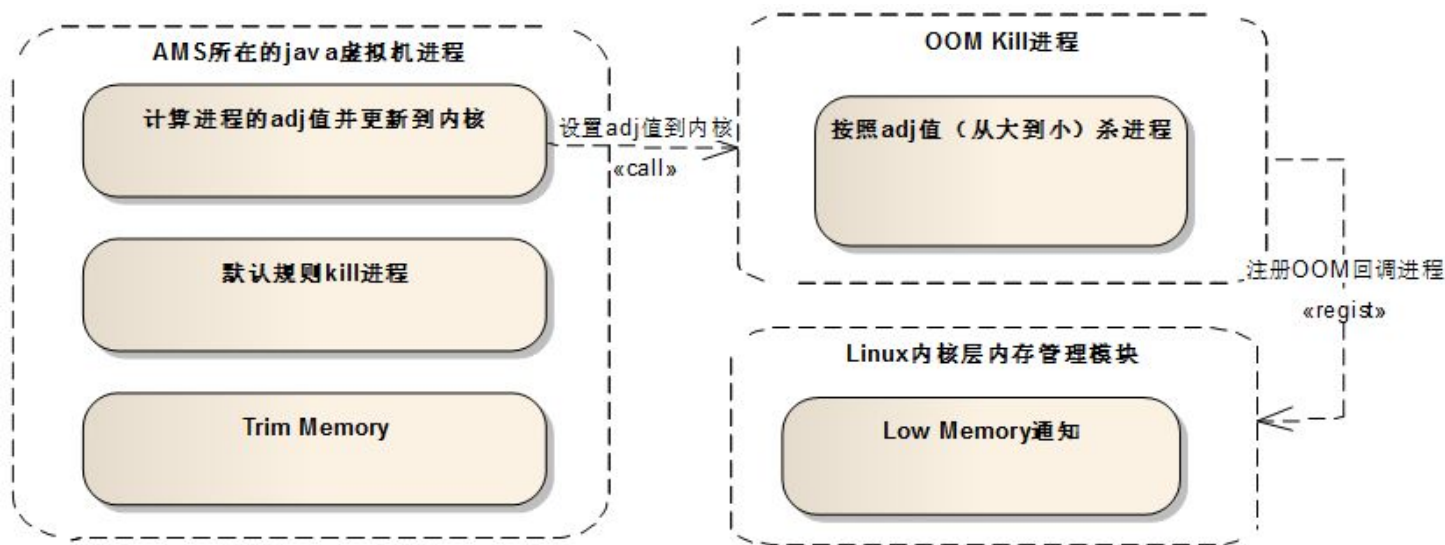
系统内存基础知识

- 内存回收机制
 - 回收基本原则、优先级
 - 应用程序内存回收



系统内存基础知识

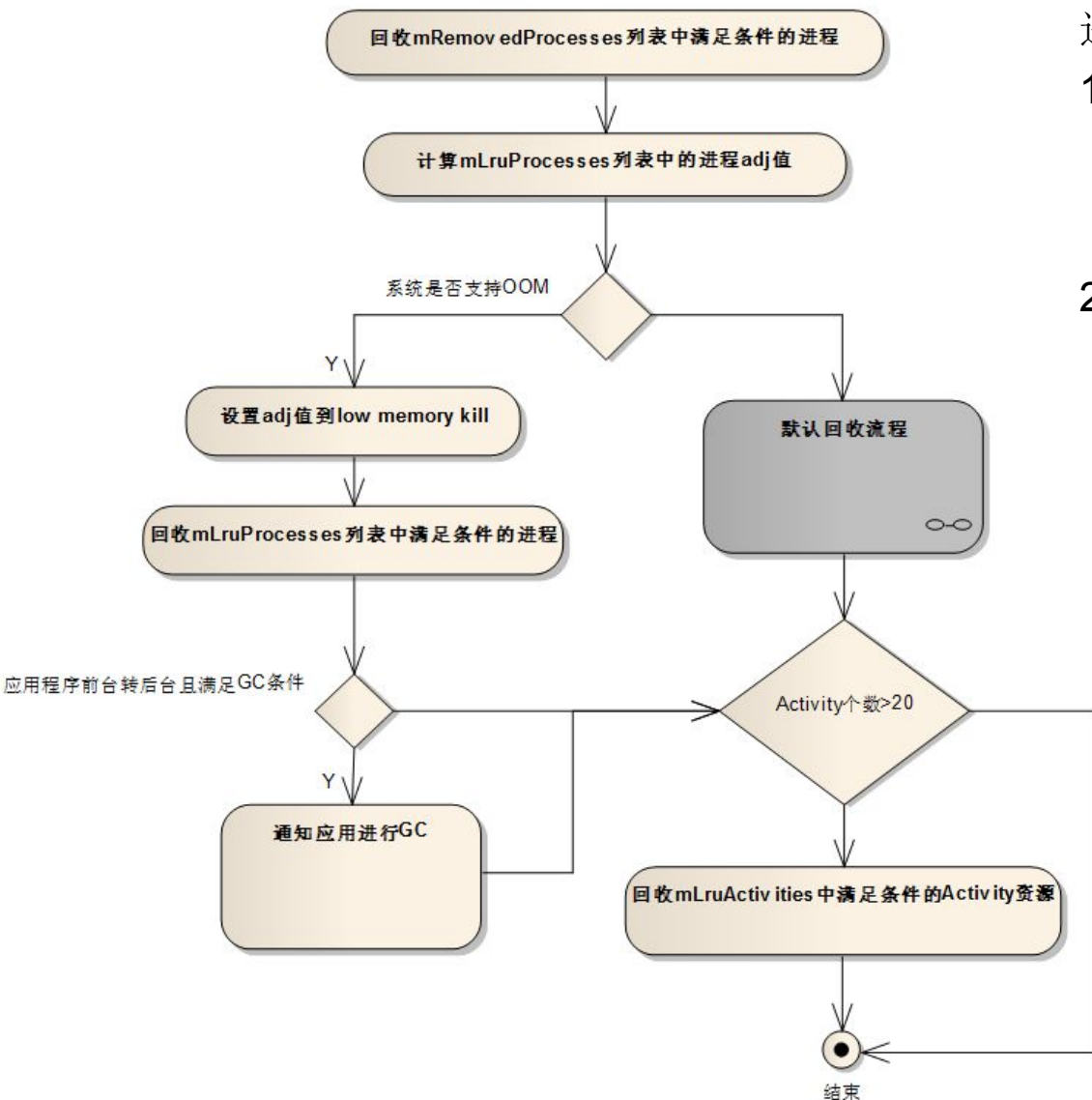
- 框架层 (framework)
 - 由 AMS完成
- 内核层 (kernel)
 - 由 low memory killer完成



Android 2.3内存回收流程

进程adj计算

- 1、初步计算：根据进程本身的一些比较容易判别的特性进行初步判断。在这一阶段，系统共分了10条规则进行计算。
- 2、进一步调整：有些进程可能具有更复杂的情况，如连接了service或包含content provider或依赖于其他进程等等。在这些情况下，进程的adj会受service或content provider或其他进程的影响，因此为更准确的计算进程的adj，还需针对这些情况对adj做进一步地调整。

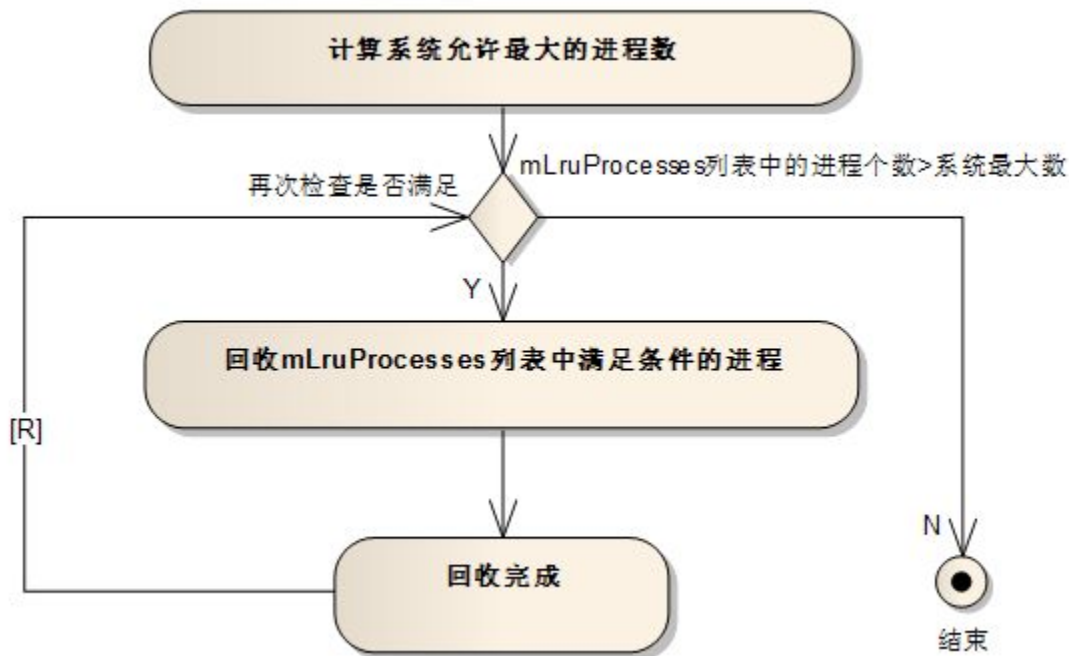


垃圾回收的四个触发时机：

- 1、当一个进程死亡的时
- 2、当启动指定的activity失败时
- 3、在更新adj值时
- 4、处理完当前所有广播时



默认回收机制

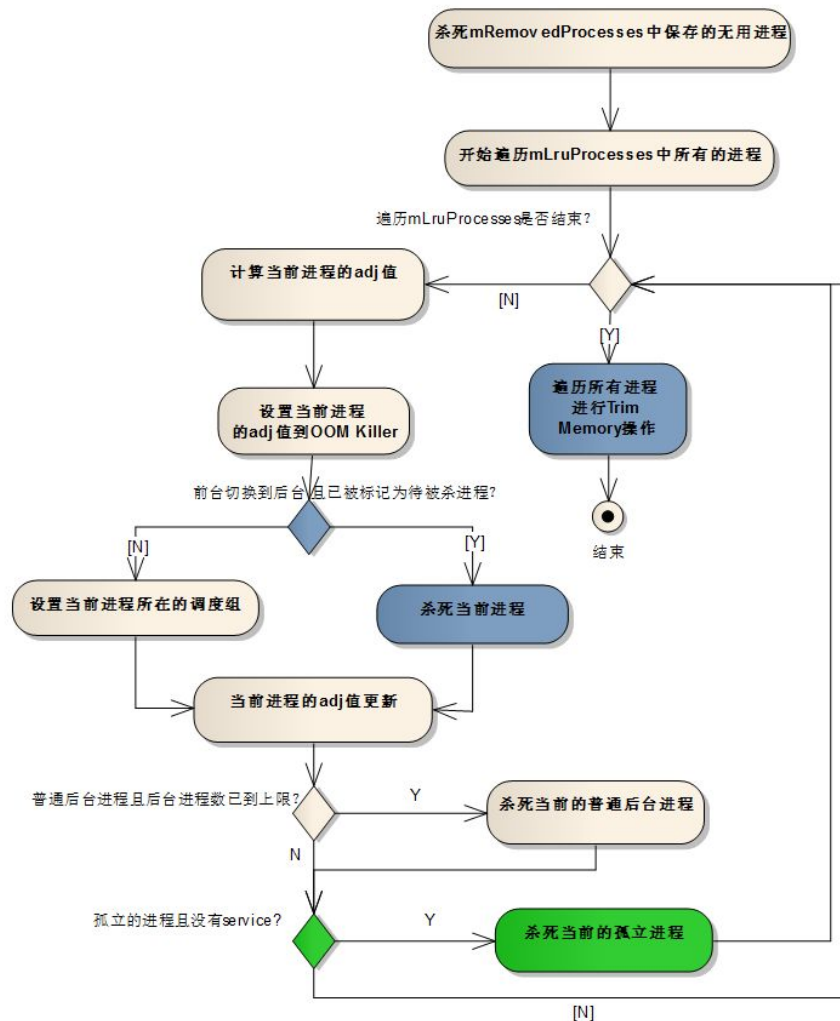


如果系统不支持OOM机制，则AMS会执行内存默认回收机制来回收内存，其实质就是根据最近使用过的应用进程个数是否超过最大进程数来判断是否要进行内存回收。



Android 4.x 内存回收流程

- 内存回收主要是通过：
 - kill 进程
 - trim Memory



mRemovedProcesses列表中主要包含了 crash 的进程、5 秒内没有响应并被用户选在强制关闭的进程、以及应用开发这调用 killBackgroundProcess 想要杀死的进程。这里无用进程是指该进程即没有activity也没有service 且无receiver。

trim Memory的逻辑主要是根据进程内存回收的级别调用handleTrimMemory进行内存回收工作



得到的优化启示

- Activity一直处于Foreground
- 后台service进程保活
- Notification常驻
-



构建内存分析体系

- 内存分级，内部回收
 - VeryGood, Good, Bad, VeryBad
- 工具分析
 - 内存自动化测试平台—UCQMS
 - AddressSanitizer(Asan)分析
 - DMRRunner 检测内存泄漏
 - Deep Memory Profiler



UCQMS

- 版本维度对比
- 场景化对比
 - 启动场景（首次 / 非首次）
 - 浏览网页
 - 看视频 / 读小说
 - 500个场景
- 多指标项对比
 - PSS,RSS,VSS
 - dalvik heap,mmap,rtf
 -



UCQMS

全选/取消全选

- | | | | |
|---|--|--|---|
| <input checked="" type="checkbox"/> psstotal_total | <input checked="" type="checkbox"/> rss | <input checked="" type="checkbox"/> libwebcore_uc_rss | <input checked="" type="checkbox"/> psstotal_nativeheap |
| <input checked="" type="checkbox"/> psstotal_dalvikheap | <input checked="" type="checkbox"/> psstotal_dalvikother | <input checked="" type="checkbox"/> psstotal_stack | <input checked="" type="checkbox"/> psstotal_otherdev |
| <input checked="" type="checkbox"/> psstotal_sommap | <input checked="" type="checkbox"/> psstotal_apkmmmap | <input checked="" type="checkbox"/> psstotal_ttfmmap | <input checked="" type="checkbox"/> psstotal_dexmmap |
| <input checked="" type="checkbox"/> psstotal_codemmap | <input checked="" type="checkbox"/> psstotal_imagemmap | <input checked="" type="checkbox"/> psstotal_othermmap | <input checked="" type="checkbox"/> psstotal_graphics |
| <input checked="" type="checkbox"/> psstotal_gl | <input checked="" type="checkbox"/> psstotal_unknown | | |

全选/取消全选

- | | | | |
|--|--|---|---|
| <input type="checkbox"/> HEAP_SO_P | <input type="checkbox"/> HEAP_DALVIK_P | <input type="checkbox"/> HEAP_NATIVE_P | <input type="checkbox"/> HEAP_UNKNOWN_DEV_P |
| <input type="checkbox"/> HEAP_UNKNOW_P | <input type="checkbox"/> PageCache_cnt | <input type="checkbox"/> WebWindowCount | <input type="checkbox"/> webview_cnt |
| <input type="checkbox"/> Images_size | <input type="checkbox"/> skia_bitmap_global_pool | <input type="checkbox"/> skia_glyph_cache | <input type="checkbox"/> ac_canvas_buffer |
| <input type="checkbox"/> GLLayerTexMem | <input type="checkbox"/> GLBaseLayerTexMem | <input type="checkbox"/> GLGbMem | <input type="checkbox"/> render_size |
| <input type="checkbox"/> v8_engine | | | |



UCQMS

场景: first_start

first_start

选择显示的数据类型

数据表单

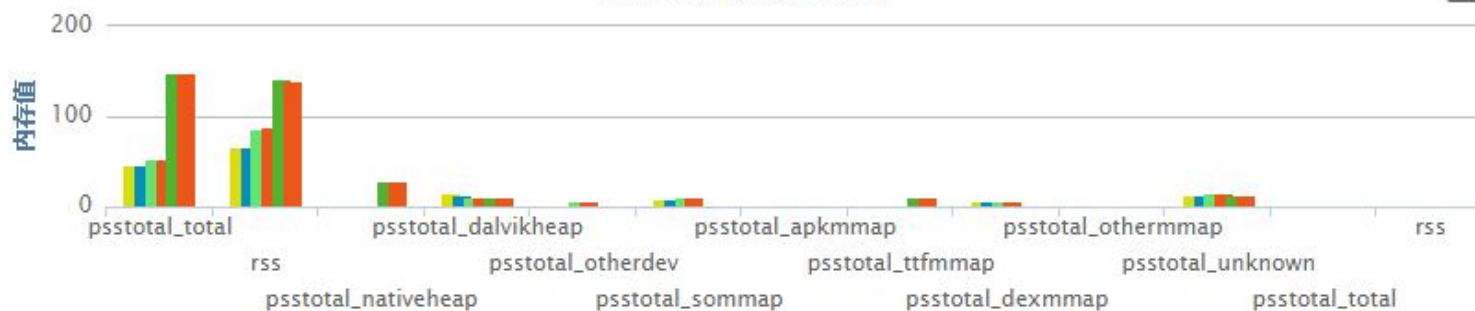
折线图

柱状图

圆饼图

CPU折线图

first_start_first_start



- 162774|UCBrowser_V10.7.0.636_android_pf145_(en-us)_qmsintl_(Build150921213644).apk|[内存]Nexus S_1
- 162774|UCBrowser_V10.7.0.636_android_pf145_(en-us)_qmsintl_(Build150921213644).apk|[内存]Nexus S_2
- 162774|UCBrowser_V10.7.0.636_android_pf145_(en-us)_qmsintl_(Build150921213644).apk|[内存]Nexus S_3
- 162774|UCBrowser_V10.7.0.636_android_pf145_(en-us)_qmsintl_(Build150921213644).apk|[内存]Nexus S_4
- 162774|UCBrowser_V10.7.0.636_android_pf145_(en-us)_qmsintl_(Build150921213644).apk|[内存]Nexus S_5
- 162775|UCBrowser_V10.7.2.645_android_pf145_(en-us)_qmsintl_(Build150921215937).apk|[内存]Nexus S_1
- 162775|UCBrowser_V10.7.2.645_android_pf145_(en-us)_qmsintl_(Build150921215937).apk|[内存]Nexus S_2
- 162775|UCBrowser_V10.7.2.645_android_pf145_(en-us)_qmsintl_(Build150921215937).apk|[内存]Nexus S_3
- 162775|UCBrowser_V10.7.2.645_android_pf145_(en-us)_qmsintl_(Build150921215937).apk|[内存]Nexus S_4

▲ 1/4 ▼

Highcharts.com



META:

Time: 1442548966
Reason: 10 sec since the last dump
CommandLine: com.UCMobile
RunID: localhost-linux-25295-1442548526
PageSize: 4096

MMAP_LIST:

```
3db9f000-3dcc0000 ---p 00000000 00:00 0
  (3db80000)- 3dcc0000 hooked 0 / 1183744 @ 8251
3dd00000-3dd09000 rw-p 00000000 00:00 0
  3dd00000 - (3dd2c000) hooked 36864 / 36864 @ 2173
40059000-4005a000 r--- 00128000 00:00 57382 /system/framework/secondary-framework.jar
  40059000 - 4005a000 unhooked 0 / 4096 @ 3769
4005a000-4005b000 r--- 00024000 00:00 66047 /data/local/tmp/asset_dex/src11.jar
  6a6b9000 - 6a7b8000 unhooked 8192 / 1044480 @ 2205
6a7b8000-6a7b9000 ---p 00000000 00:00 0
  6a7b8000 - 6a7b9000 unhooked 0 / 4096 @ 2205
6a7b9000-6a8b8000 rw-p 00000000 00:00 0 [stack:25515]
  6a7b9000 - 6a8b8000 unhooked 8192 / 1044480 @ 2205
```

GLOBAL_STATS:

total (258076672) == profiled-mmap (85737300) + nonprofiled-* (172339372)

	virtual	committed
total	1106137088	258076672
absent	0	102400
file-exec	96182272	43077632
file-nonexec	752414720	90353664
anonymous	150618112	98660352
stack	135168	45056
other	106786816	25837568
nonprofiled-total	708977043	172339372
nonprofiled-absent	0	0
nonprofiled-anonymous	28479488	23539712
nonprofiled-file-exec	96182272	43077632
nonprofiled-file-nonexec	477393299	79839404
nonprofiled-stack	135168	45056
nonprofiled-other	106786816	25837568
profiled-mmap	397160045	85737300
profiled-malloc	17039045	16479695

STACKTRACES:

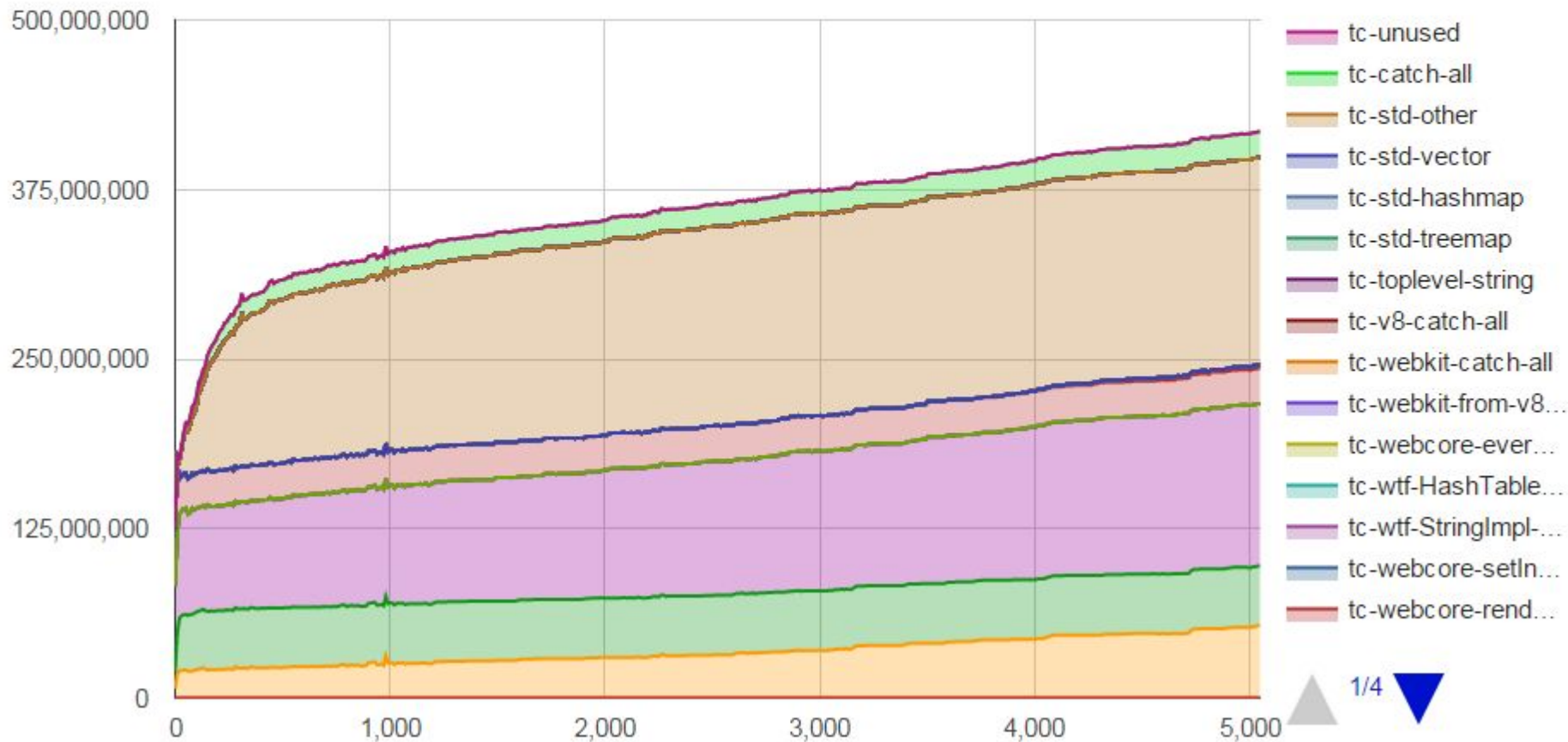
```
virtual committed
144 144 0 0 @ 4177
```



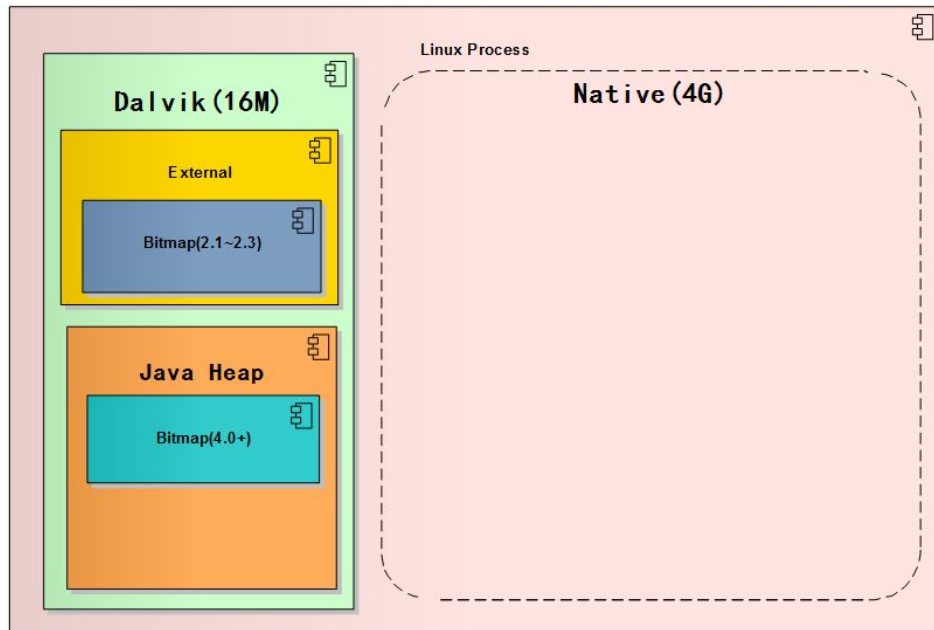
Deep Memory Profiler

t0 12 uc.prune uc.prune.plus android.webview uc

DMP Graph (Policy: I2)



图片使用native内存



分辨率	截屏 (ARGB8888)	占比
240*320	$240*320*4=0.3\text{M}$	$0.3/16=1.875\%$
320*480	$320*480*4=0.6\text{M}$	$0.6/16=3.75\%$
480*800	$480*800*4=1.5\text{M}$	$1.5/24=6.25\%$
720*1280	$720*1280*4=3.5\text{M}$	$3.5/48=7.3\%$

可以看到，虽然分辨率越高，上限越大，但由于图片也更大，实际上内存反而更加吃紧。

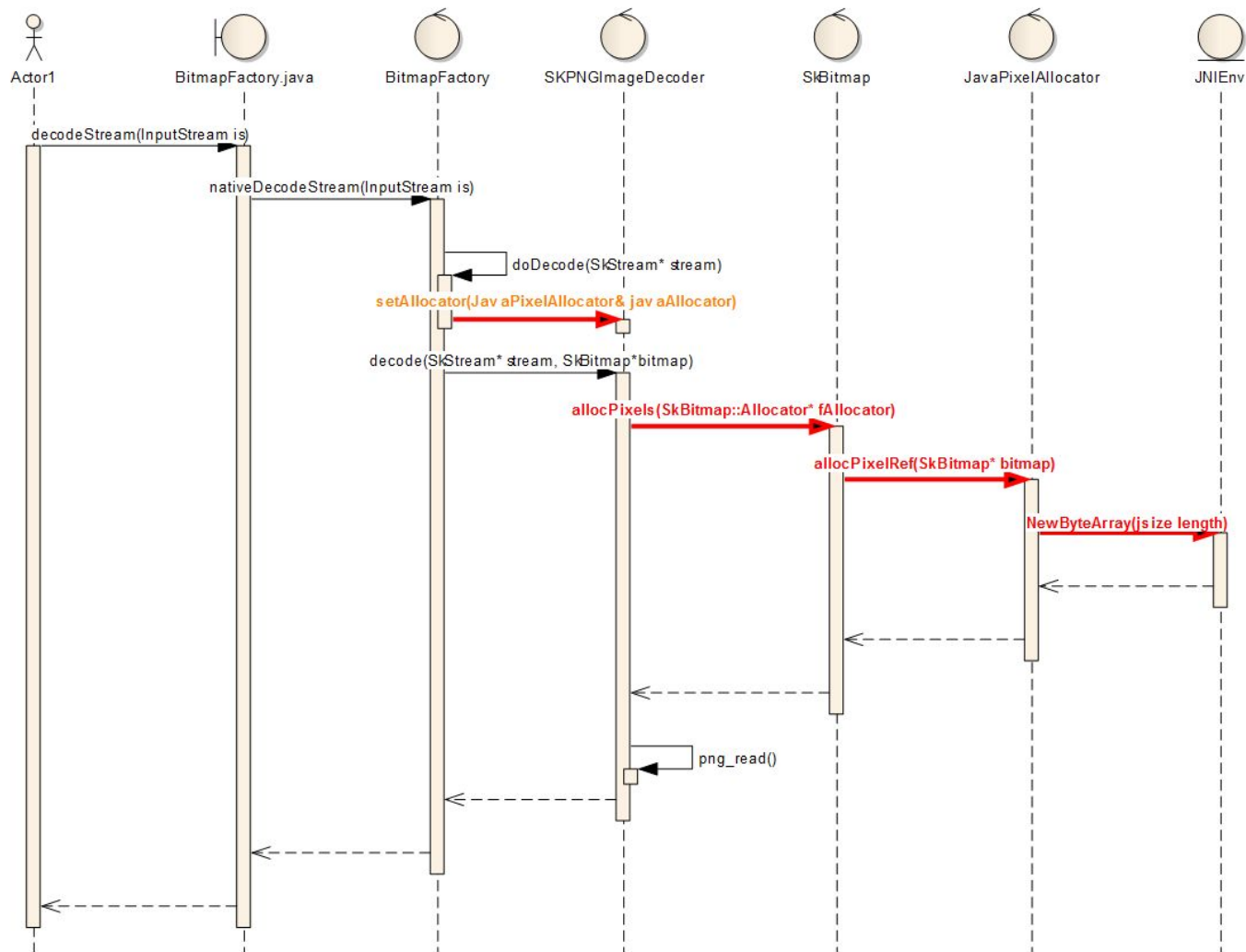
Native内存转换

- 图片使用Native内存
 - UC内置图片转Native
 - UC外部图片转Native
 - 2.x inNativeAlloc
 - 非240dpi手机缩放density
- 核心关键点
 - inNativeAlloc



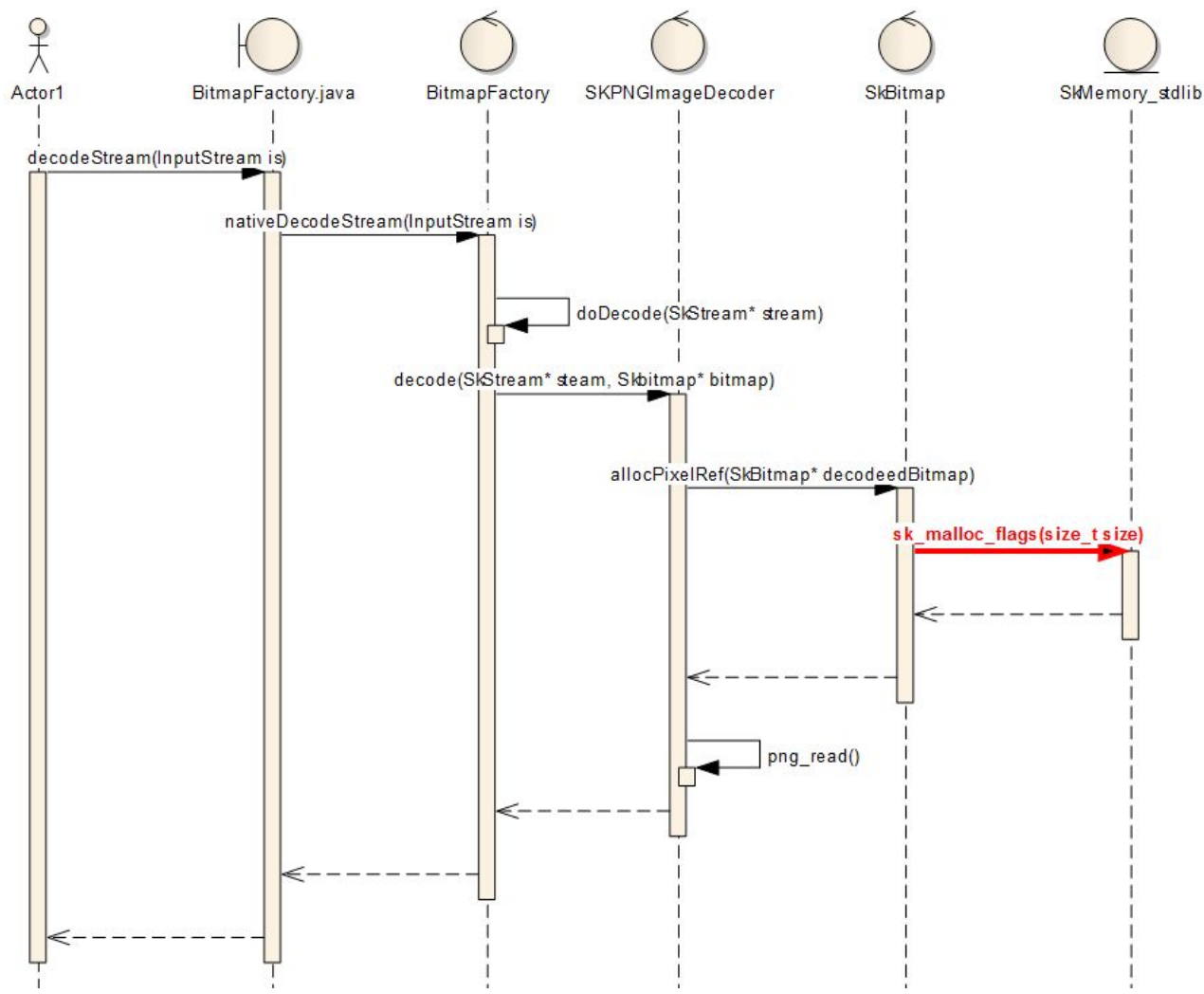
Bitmap decode原理

使用Dalvik虚拟机内存



Bitmap decode原理

使用Native内存



BitmapFactory.Options

- inNativeAlloc
- inPurgeable
- inInputShareable
- inDensity
- inTargetDensity
- inSampleSize




```
public Bitmap decodeFile(String filePath){
    Bitmap bitmap = null;
    BitmapFactory.Options options = new BitmapFactory.Options();
    options.inPurgeable = true;
    try {
        BitmapFactory.Options.class.getField("inNativeAlloc").setBoolean(opti
ons,true);
    } catch (IllegalArgumentException e) {
        e.printStackTrace();
    } catch (SecurityException e) {
        e.printStackTrace();
    } catch (IllegalAccessException e) {
        e.printStackTrace();
    } catch (NoSuchFieldException e) {
        e.printStackTrace();
    }
    if(mFilePath != null) {
        bitmap = BitmapFactory.decodeFile(mFilePath, options);
    }
    return bitmap;
}
```



其他内存优化心得

- Drawable.Callback引起的内存泄漏
 - 尽量不要在static成员中保存Drawable对象
 - 对于需要保存的Drawable对象，在需要时调用Drawable#setCallback(null).
- Context/Handler等泄漏
- GC Log
 - D/dalvikvm(745): GC_CONCURRENT
freed 199K, 53% free
3023K/6343K,external 0K/0K, paused
2ms+2ms



其他内存优化心得

- GC_CONCURRENT表示触发垃圾收集的原因，有以下几种：
 - GC_MALLOC, 内存分配失败时触发
 - GC_CONCURRENT, 当分配的对象大小超过384K时触发
 - GC_EXPLICIT, 对垃圾收集的显式调用(System.gc)
 - GC_EXTERNAL_ALLOC, 外部内存分配失败时触发





Thanks