

## 原 GC详解及Minor GC和Full GC触发条件总结

2016年09月17日 22:09:26 逸辰查 阅读数：40263 更多

版权声明：本文为博主原创文章，未经博主允许不得转载。 [https://blog.csdn.net/YHYR\\_YCY/article/details/52566105](https://blog.csdn.net/YHYR_YCY/article/details/52566105)

GC，即就是Java垃圾回收机制。目前主流的JVM（HotSpot）采用的是分代收集算法。与C++不同的是，Java采用的是类似于树形结构还存在引用。即：从gcroot开始，把所有可以搜索得到的对象标记为存活对象。

### GC机制

要准确理解Java的垃圾回收机制，就要从：“什么时候”，“对什么东西”，“做了什么”三个方面来具体分析。

第一：“什么时候”即就是GC触发的条件。GC触发的条件有两种。（1）程序调用System.gc时可以触发；（2）系统自身来决定GC触发的时机。

系统判断GC触发的依据：根据Eden区和From Space区的内存大小来决定。当内存大小不足时，则会启动GC线程并停止应用线程。

第二：“对什么东西”笼统的认为是Java对象并没有错。但是准确来讲，GC操作的对象分为：通过可达性分析法无法搜索到的对象和可以搜索到的对象。对于搜索不到的方法进行标记。

第三：“做了什么”最浅显的理解为释放对象。但是从GC的底层机制可以看出，对于可以搜索到的对象进行复制操作，对于搜索不到的对象，调用finalize()方法进行释放。

具体过程：当GC线程启动时，会通过可达性分析法把Eden区和From Space区的存活对象复制到To Space区，然后把Eden Space和From Space区的对象释放掉。当GC轮训扫描To Space区一定次数后，把依然存活的对象复制到老年代，然后释放To Space区的对象。

**对于用可达性分析法搜索不到的对象，GC并不一定会回收该对象。要完全回收一个对象，至少需要经过两次标记的过程。**

第一次标记：对于一个没有其他引用的对象，筛选该对象是否有必要执行finalize()方法，如果没有执行必要，则意味可直接回收。（筛选依据：是否复写或执行过finalize()方法；因为finalize方法只能被执行一次）。

第二次标记：如果被筛选判定有必要执行，则会放入FQueue队列，并自动创建一个低优先级的finalize线程来执行释放操作。如果在一个对象释放前被其他对象引用，则该对象会被移除FQueue队列。

### GC过程中用到的回收算法：



22



14

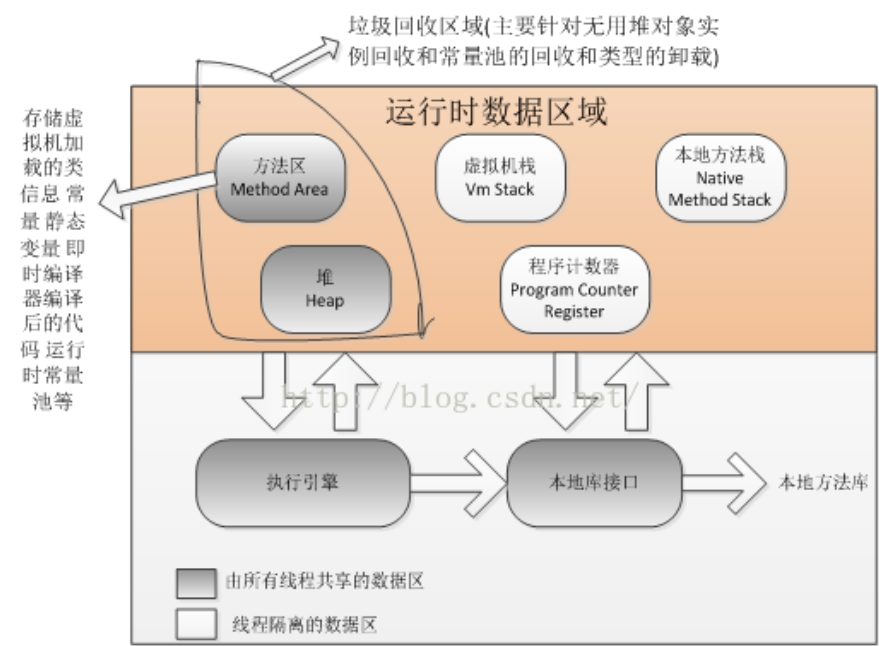


可达性分析法来判断对象是否

通过上面的GC过程不难看出，Java堆中的年轻代和老年代采用了不同的回收算法。年轻代采用了复制法；而老年代采用了标记-整理法

具体各种回收算法的详解参考：<http://www.cnblogs.com/dolphin0520/p/3783345.html>

JVM内存空间图解



Java虚拟机运行时数据区

程序计数器：线程私有。是一块较小的内存，是当前线程所执行的字节码的行号指示器。是Java虚拟机规范中唯一没有规定OOM（OutOfMemoryError）的区域。

Java栈：线程私有。生命周期和线程相同。是Java方法执行的内存模型。执行每个方法都会创建一个栈帧，用于存储局部变量和操作数（对象引用）。局部变量所需要的内存空间大小在编译期间完成分配。所以栈帧的大小不会改变。存在两种异常情况：若线程请求深度大于栈的深度，抛StackOverflowError。若栈在动态扩展时无法请求足够内存，抛OOM。

Java堆：所有线程共享。虚拟机启动时创建。存放对象实例和数组。所占内存最大。分为新生代（Young区），老年代（Old区）。新生代分Eden区，Servior区。Servior区又分为From space区和To Space区。Eden区和Servior区的内存比为8:1。当扩展内存大于可用内存，抛OOM。

方法区：所有线程共享。用于存储已被虚拟机加载的类信息、常量、静态变量等数据。又称为非堆（Non – Heap）。方法区又称“永久代”。GC很少在这个区域进行，但不代表不会回收。这个区域回收目标主要是针对常量池的回收和对类型的卸载。当内存申请大于实际可用内存，抛OOM。

22
14

本地方法栈：线程私有。与Java栈类似，但是不是为Java方法（字节码）服务，而是为本地非Java方法服务。也会抛StackOverflowError和OOM。

Minor GC , Full GC 触发条件

Minor GC触发条件：当Eden区满时，触发Minor GC。

Full GC触发条件：

- (1) 调用System.gc时，系统建议执行Full GC，但是不必然执行
- (2) 老年代空间不足
- (3) 方法去空间不足
- (4) 通过Minor GC后进入老年代的平均大小大于老年代的可用内存
- (5) 由Eden区、From Space区向To Space区复制时，对象大小大于To Space可用内存，则把该对象转存到老年代，且老年代的可用内存小于该对象大小

22

14


Python正确的学习路线，你一定不知道的薪资翻倍秘诀

如何从8K提至20K月薪，你要掌握学习那些技能



想对作者说点什么

 **申明霜**: 感谢分享，很清楚 (09-01 19:45 #12楼)

 **王啸tr1912**: 很清楚，总结的很好 (08-19 08:22 #11楼)

 **大白快跑8**: 很清楚，感谢分享 (08-11 09:15 #10楼)

[查看 14 条热评](#)

Minor GC、Major GC和Full GC之间的区别？

 548

Minor GC Minor GC指新生代GC，即发生在新生代（包括Eden区和Survivor区）的垃圾回收操作，当新生代无法为新生对象分配内存...

面试题：“你能不能谈谈，java GC是在什么时候，对什么东西，做了什么事情？”  1.9万

面试题目： 地球人都知道，Java有个东西叫垃圾收集器，它让创建的对象不需要像c/cpp那样delete、free掉，你能不能谈谈，GC是在...



客户关系管理系统解决方案

百度广告

22

14

Java GC工作原理以及Minor GC、Major GC、Full GC简单总结  1.4万

名词解释： GC：垃圾收集器 Minor GC：新生代GC，指发生在新生代的垃圾收集动作，所有的Minor GC都会触发全世界的暂停（stop...

Minor GC、Major GC和Full GC之间的区别  3251

对Minor GC和Major GC做点笔记 新生代 GC（Minor GC）：指发生在新生代的垃圾收集动作，因为 Java 对象大多都具 备朝生夕灭...

线上FullGC频繁的排查  1万

线上FullGC频繁的排查问题

第一次FullGC优化实战  4298

关于FullGC，博主只是在一些书中或者博客中，看别人调优过，今天兴起，亲自在本地调了一把本地的项目： 第一步：我首先打...



适合3-9岁小孩的英语课，不再为英语而苦恼

幼儿英语 美式互动英语课堂,激发英语兴趣,建立英语思维.

JVM架构和GC垃圾回收机制(JVM面试不用愁)  2万

JVM架构和GC垃圾回收机制 JVM架构图分析 JVM被分为三个主要的子系统： 1. 类加载器子系统 2. 运行时数据区 3. 执行引擎 1. 类...

Minor GC和Full GC区别  8364

概念： 新生代 GC（Minor GC）：指发生在新生代的垃圾收集动作，因为 Java 对象大多都具 备朝生夕灭的特性，所以 Minor GC 非...

聊聊JVM（四）深入理解Major GC, Full GC, CMS  441

很多人都分不清Major GC, Full GC的概念，事实上我查了下资料，也没有查到非常精确的Major GC和Full GC的概念定义。分不清这两...

jvm什么时候会触发full gc

6792

除直接调用System.gc外，触发Full GC执行的情况有如下四种。 1. 旧生代空间不足 旧生代空间只有在新生代对象转入及创建为大对象...

相关热词

-gc 并行gc并发gc dc和gc yum安装gc gc收集器

个人资料



逸辰杳

关注

原创 34 粉丝 23 喜欢 31 评论 27

等级： 博客 4 访问： 11万+  
积分： 1065 排名： 5万+  
勋章： 恒



1点点加盟要求



最新文章

Pandas.read\_json()踩坑总结 & 源码初探

基于Python初探Linux下的僵尸进程和孤儿进程(三)

基于Python初探Linux下的僵尸进程和孤儿进程(二)

基于Python初探Linux下的僵尸进程和孤儿进程(一)

查看Python依赖包及其版本号信息

个人分类

Java	4篇
JavaWeb	3篇
Linux	8篇
Hadoop	5篇
MongoDB	1篇

展开

热门文章

GC详解及Minor GC和Full GC触发条件总结  
阅读量：40017

Java实现排列组合算法  
阅读量：20737

基于Python操作ElasticSearch  
阅读量：14800

MyEclipse 2015实现JavaScript代码自动提示（不需要额外安装插件）  
阅读量：7126

22
14

使用Myeclipse + SVN + TaoCode 免费实现  
项目版本控制的详细教程  
阅读量： 3999

最新评论

- GC详解及Minor GC和Ful...

u010711441: [reply]u011738824[/reply] 是都回收  
当时如果是cms 会更复杂点
- GC详解及Minor GC和Ful...


sms15732621690: 感谢分享, 很清楚
- GC详解及Minor GC和Ful...

tr1912: 很清楚, 总结的很好
- 基于Spring Boot实现My...

READlgg: 有一个疑问, 我发现在测试过程中, 我  
不手动指定数据源, 仍然可以正常运行, 比如我注  
释掉DataSou...
- GC详解及Minor GC和Ful...

zt15732625878: 很清楚, 感谢分享

联系我们



请扫描二维码联系客服

✉ webmaster@csdn.net

☎ 400-660-0108

🗣 QQ客服    🗣 客服论坛

关于    招聘    广告服务    网站地图  
©2018 CSDN版权所有 京ICP证09002463号  
🔍 百度提供搜索支持



经营性网站备案信息

网络110报警服务

中国互联网举报中心

北京互联网违法和不良信息举报中心

22
14

22
14