

将者，智、信、仁、勇、严也。

大智若愚，大巧若拙，大勇若怯！

## 公告

昵称: bonelee

园龄: 2年

粉丝: 78

关注: 0

+加关注

< 2018年9月 >

日	一	二	三	四	五	六
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

## 搜索

<input type="text"/>	找找看
<input type="text"/>	谷歌搜索

## 常用链接

博客园 首页 新随笔 联系 管理 订阅 XML

随笔 - 1361 文章 - 0 评论 - 581

## java old GC和young GC

### Java内存分配机制

摘自: <http://www.cnblogs.com/zhguang/p/3257367.html>

这里所说的内存分配，主要指的是在堆上的分配，一般的，对象的内存分配都是在堆上进行，但现代技术也支持将对象拆成标量类型（标量类型即原子类型，表示单个值，可以是基本类型或String等），然后在栈上分配，在栈上分配的很少见，我们这里不考虑。

Java内存分配和回收的机制概括的说，就是：分代分配，分代回收。对象将根据存活的时间被分为：年轻代（Young Generation）、年老代（Old Generation）、永久代（Permanent Generation，也就是方法区）。如下图（来源于《成为JavaGC专家part I》，<http://www.importnew.com/1993.html>）：

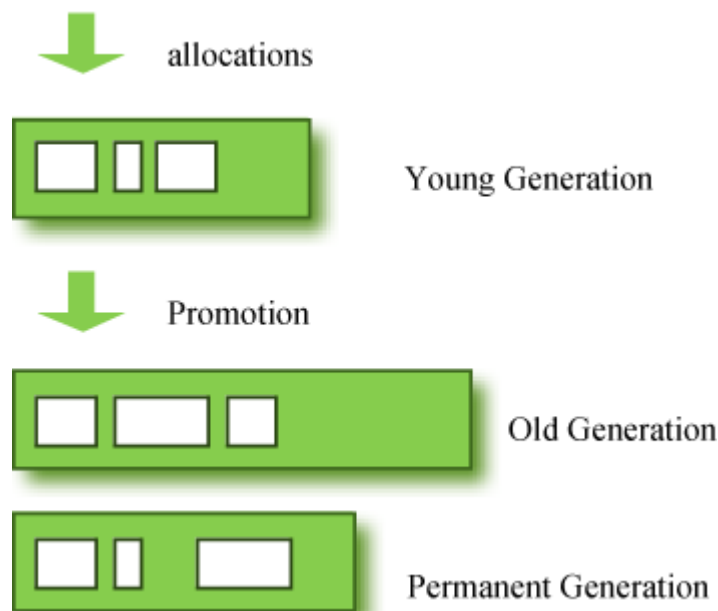
我的随笔  
我的评论  
我的参与  
最新评论  
我的标签

## 我的标签

安全分析(212)  
机器学习(198)  
leetcode(163)  
elasticsearch(113)  
数据库(108)  
搜索引擎(85)  
其他(69)  
算法(65)  
python(65)  
tensorflow(56)  
更多

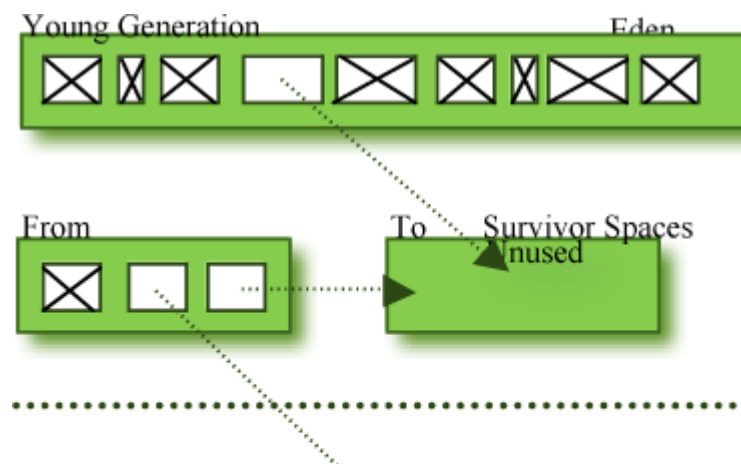
## 随笔档案

2018年9月 (7)  
2018年8月 (17)  
2018年7月 (41)  
2018年6月 (62)  
2018年5月 (85)  
2018年4月 (66)  
2018年3月 (113)  
2018年2月 (33)  
2018年1月 (46)  
2017年12月 (37)  
2017年11月 (81)  
2017年10月 (25)



**年轻代 (Young Generation)**：对象被创建时，内存的分配首先发生在年轻代（大对象可以直接被创建在年老代），大部分的对象在创建后很快就不再使用，因此很快变得不可达，于是被年轻代的GC机制清理掉（IBM的研究表明，98%的对象都是很快消亡的），这个GC机制被称为Minor GC或叫Young GC。注意，Minor GC并不代表年轻代内存不足，它事实上只表示在Eden区上的GC。

年轻代上的内存分配是这样的，年轻代可以分为3个区域：Eden区（伊甸园，亚当和夏娃偷吃禁果生娃娃的地方，用来表示内存首次分配的区域，再贴切不过）和两个存活区（Survivor 0、Survivor 1）。内存分配过程为（来源于《成为JavaGC专家part I》，<http://www.importnew.com/1993.html>）：



2017年9月 (37)  
 2017年8月 (43)  
 2017年7月 (53)  
 2017年6月 (40)  
 2017年5月 (76)  
 2017年4月 (50)  
 2017年3月 (94)  
 2017年2月 (113)  
 2017年1月 (107)  
 2016年12月 (80)  
 2016年11月 (38)  
 2016年10月 (11)  
 2016年9月 (3)  
 2016年8月 (3)

## 最新评论

### 1. Re:识别TLS加密恶意流量

基于加密握手过程信息、加密通信过程中的传输信息以及加密流量的背景流量信息特征，利用随机森林分类模型实现恶意加密流量的检测

--bonelee

### 2. Re:lucene字典实现原理——FST

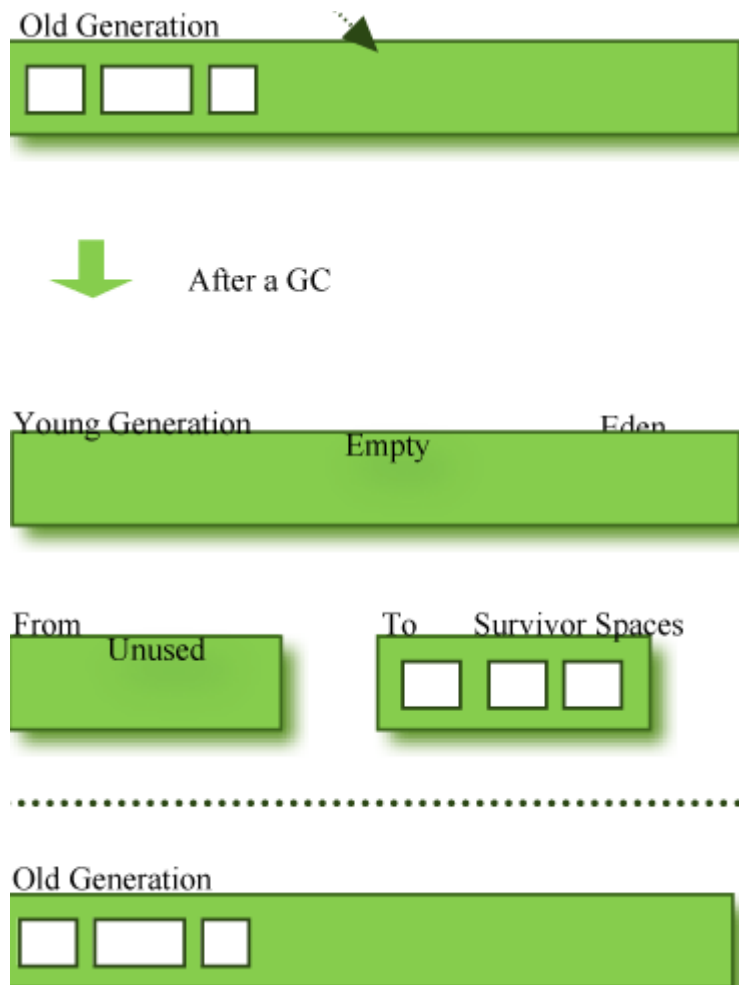
图中再体现一下权重就好了。

--夜无痕星

### 3. Re:核心网信令风暴问题根因分析

Diameter信令网信令风暴抑制方案  
 研究Diameter协议是一种新的面向IP的AAA协议,它从Radius协议演进并且有逐渐的取代Radius协议的

java old GC和young GC - bonelee - 博客园



1. 绝大多数刚创建的对象会被分配在Eden区，其中的大多数对象很快就会消亡。Eden区是连续的内存空间，因此在其上分配内存极快；
2. 最初一次，当Eden区满的时候，执行Minor GC，将消亡的对象清理掉，并将剩余的对象复制到一个存活区Survivor0（此时，Survivor1是空白的，两个Survivor总有一个是空白的）；
3. 下次Eden区满了，再执行一次Minor GC，将消亡的对象清理掉，将存活的对象复制到Survivor1中，然后清空Eden区；
4. 将Survivor0中消亡的对象清理掉，将其中可以晋级的对象晋级到Old区，将存活的对象也复制到Survivor1区，然后清空Survivor0区；

趋势。Diameter协议包括基本协议和扩展协议.....

--bonelee

#### 4. Re:核心网信令风暴问题根因分析

如果是核心网的信令过载第三章从信令网的出口节点控制过载出发点, 结合在移动研究院所做的工作。□□研究了基于D i a m e t e r 信令网的出口□D R A节点的拥塞控制机制并提出了抑制边缘□□出口□D R A节点无效路.....

--bonelee

#### 5. Re:核心网信令风暴问题根因分析

上面总结的只是提到状态转换, 实际上另外一个文章总结得更全面: L T E 网络系统可能出现信令风暴的□□原因, 大致可以总结出以下几点: □□1. 网络架构的变化, 导致4 G核心网信令流量较2 G / 3 G大幅增加□□a) .....

--bonelee

### 阅读排行榜

1. 安装和使用访问暗网(21881)
2. SQL数据分析概览——Hive、Impala、Spark SQL、Drill、HAWQ 以及Presto+druid(13864)
3. sklearn的train\_test\_split, 果然很好用啊! (12892)

5. 当两个存活区切换了几次 (HotSpot虚拟机默认15次, 用-XX:MaxTenuringThreshold控制, 大于该值进入老年代, 但这只是个最大值, 并不代表一定是这个值) 之后, 仍然存活的对象 (其实只有一小部分, 比如, 我们自己定义的对象), 将被复制到老年代。

从上面的过程可以看出, Eden区是连续的空间, 且Survivor总有一个为空。经过一次GC和复制, 一个Survivor中保存着当前还活着的对象, 而Eden区和另一个Survivor区的内容都不再需要了, 可以直接清空, 到下一次GC时, 两个Survivor的角色再互换。因此, 这种方式分配内存和清理内存的效率都极高, 这种垃圾回收的方式就是著名的“**停止-复制 (Stop-and-copy)**”**清理法 (将Eden区和一个Survivor中仍然存活的对象拷贝到另一个Survivor中)**, 这不代表着停止复制清理法很高效, 其实, 它也只在这种情况下高效, 如果在老年代采用停止复制, 则挺悲剧的。

在Eden区, HotSpot虚拟机使用了两种技术来加快内存分配。分别是bump-the-pointer和TLAB (Thread-Local Allocation Buffers), 这两种技术的做法分别是: 由于Eden区是连续的, 因此bump-the-pointer技术的核心就是跟踪最后创建的一个对象, 在对象创建时, 只需要检查最后一个对象后面是否有足够的内存即可, 从而大大加快内存分配速度; 而对于TLAB技术是对于多线程而言的, 将Eden区分为若干段, 每个线程使用独立的一段, 避免相互影响。TLAB结合bump-the-pointer技术, 将保证每个线程都使用Eden区的一段, 并快速的分配内存。

**年老代 (Old Generation)**: 对象如果在年轻代存活了足够长的时间而没有被清理掉 (即在几次Young GC后存活了下来), 则会被复制到年老代, 年老代的空间一般比年轻代大, 能存放更多的对象, 在年老代上发生的GC次数也比年轻代少。当年老代内存不足时, 将执行Major GC, 也叫 Full GC。

可以使用-XX:+UseAdaptiveSizePolicy开关来控制是否采用动态控制策略, 如果动态控制, 则动态调整Java堆中各个区域的大小以及进入老年代的年龄。

如果对象比较大 (比如长字符串或大数组), Young空间不足, 则大对象会直接分配到老年代上 (大对象可能触发提前GC, 应少用, 更应避免使用短命的大对象)。用-XX:PretenureSizeThreshold来控制直接升入老年代的对象大小, 大于这个值的对象会直接分配在老年代上。

可能存在年老代对象引用新生代对象的情况, 如果需要执行Young GC, 则可能需要查询整个老年代以确定是否可以清理回收, 这显然是低效的。解决的方法是, 年老代中维护一个512 byte的块——“card table”, 所有老年代对象引用新生代对象的记录都记录在这里。Young GC时, 只要查这里即可, 不用再去查全部老年代, 因此性能大大提高。

### Java GC机制

GC机制的基本算法是: 分代收集, 这个不用赘述。下面阐述每个分代的收集方法。

4. numpy中的ndarray方法和属性 (12869)
5. spark shuffle内在原理说明 (8464)

## 评论排行榜

1. CC 攻击检测研究现状(44)
2. 网络流量预测 国内外研究现状【见评论】——传统的ARIMA、HMM模型，目前LSTM、GRU、CNN应用较多，貌似小波平滑预处理步骤非常关键(42)
3. 信令风暴研究现状总结(40)
4. ES业界优秀实践案例汇总(28)
5. 国外DDoS产品的一些调研——Akamai Arbor Networks Cloudflare DOSarrest F5 Fastly Imperva Link11 Neustar Nexusguard Oracle (Dyn) Radware Verisign(22)

## 推荐排行榜

1. MySQL索引实现(3)
2. 机器学习 LR中的参数迭代公式推导——极大似然和梯度下降(3)
3. spark 任务运行原理(2)
4. lucene底层数据结构——FST，针对field使用列存储，delta encode压缩doc ids数组，LZ4压缩算法(2)
5. Spark任务提交底层原理(1)

## 年轻代：

事实上，在上一节，已经介绍了新生代的主要垃圾回收方法，在新生代中，使用“停止-复制”算法进行清理，将新生代内存分为2部分，1部分 Eden区较大，1部分Survivor比较小，并被划分为两个等量的部分。每次进行清理时，将Eden区和一个Survivor中仍然存活的对象拷贝到 另一个Survivor中，然后清理掉Eden和刚才的Survivor。

这里也可以发现，停止复制算法中，用来复制的两部分并不总是相等的（传统的停止复制算法两部分内存相等，但新生代中使用1个大的Eden区和2个小的Survivor区来避免这个问题）

由于绝大部分的对象都是短命的，甚至存活不到Survivor中，所以，Eden区与Survivor的比例较大，HotSpot默认是 8:1，即分别占新生代的80%，10%，10%。如果一次回收中，Survivor+Eden中存活下来的内存超过了10%，则需要将一部分对象分配到 老年代。用-XX:SurvivorRatio参数来配置Eden区域Survivor区的容量比值，默认是8，代表Eden: Survivor1: Survivor2=8:1:1。

## 老年代：

老年代存储的对象比年轻代多得多，而且不乏大对象，对老年代进行内存清理时，如果使用停止-复制算法，则相当低效。一般，老年代用的算法是标记-整理算法，即：标记出仍然存活的对象（存在引用的），将所有存活的对象向一端移动，以保证内存的连续。

在发生Minor GC时，虚拟机会检查每次晋升进入老年代的大小是否大于老年代的剩余空间大小，如果大于，则直接触发一次Full GC，否则，就查看是否设置了-XX:+HandlePromotionFailure（允许担保失败），如果允许，则只会进行MinorGC，此时可以容忍内存分配失败；如果不允许，则仍然进行Full GC（这代表着如果设置-XX:+HandlePromotionFailure，则触发MinorGC就会同时触发Full GC，哪怕老年代还有很多内存，所以，最好不要这样做）。

标签: [java](#)

好文要顶

关注我

收藏该文



[bonelee](#)

[关注 - 0](#)

[粉丝 - 78](#)

[+加关注](#)

0

0

« 上一篇: [ES JVM使用如果超过75%就会GC较多，导致ES索引性能下降](#)

» 下一篇: [dns tunnel工具地址](#)

posted on 2017-12-19 17:55 [bonelee](#) 阅读(1977) 评论(0) [编辑](#) [收藏](#)

Copyright ©2018 bonelee

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。



#### 最新IT新闻:

- 不到4年估值60亿美元 商汤科技值吗?
- 年轻的奥秘在于复杂性: 人体日渐衰老生理过程变简单
- 人人车推出二手车严选商城 将开设上百家线下店
- 马云“禅位”: 阿里巴巴真的可以没有他么?
- 上海楼市后半场, 到底有多艰难?
- » 更多新闻...



华为全联接大会 | 上海 | 2018.10.10-12

「大会门票+云服务器」专属套餐0.35折起



#### 最新知识库文章:

- 为什么说 Java 程序员必须掌握 Spring Boot ?
- 在学习中, 有一个比掌握知识更重要的能力
- 如何招到一个靠谱的程序员
- 一个故事看懂“区块链”
- 被踢出去的用户
- » 更多知识库文章...

#### 历史上的今天:

2016-12-19 JS模块式开发

