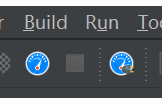
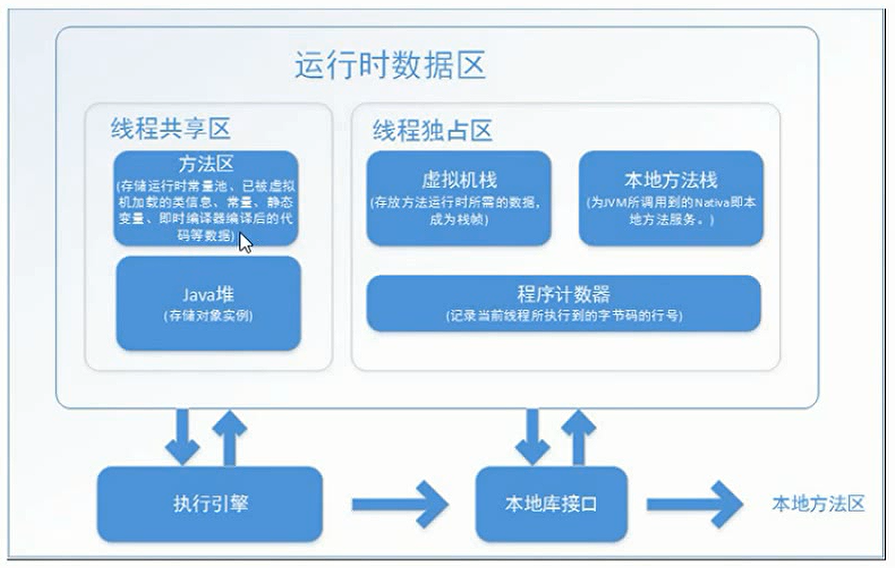
1. IDEA内存分析工具
   1. JProfiler

plugin安装好后，点击图标会弹出打开文件选项，这是什么情况呢，这是这个插件启动需要依赖一个可执行的文件，就是源生的JAVA PROFILER 去官网下载 [https://www.ej-technologies.com/products/jprofiler/overview.html](https://www.ej-technologies.com/products/jprofiler/overview.html" \t "https://blog.csdn.net/qq_22194659/article/details/_blank)

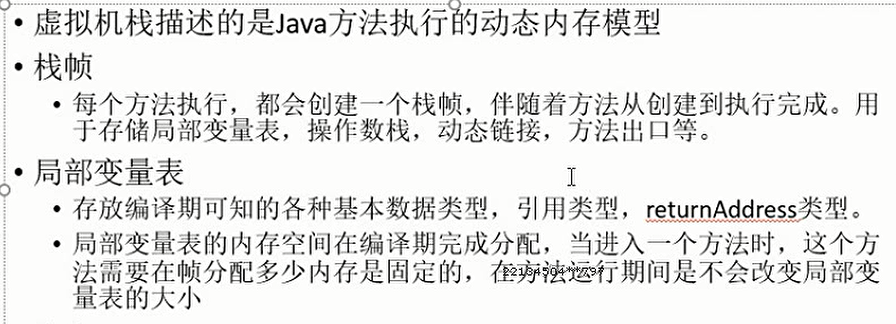
* 1. JDK自带检测工具 jconsole.exe在jdk/bin下

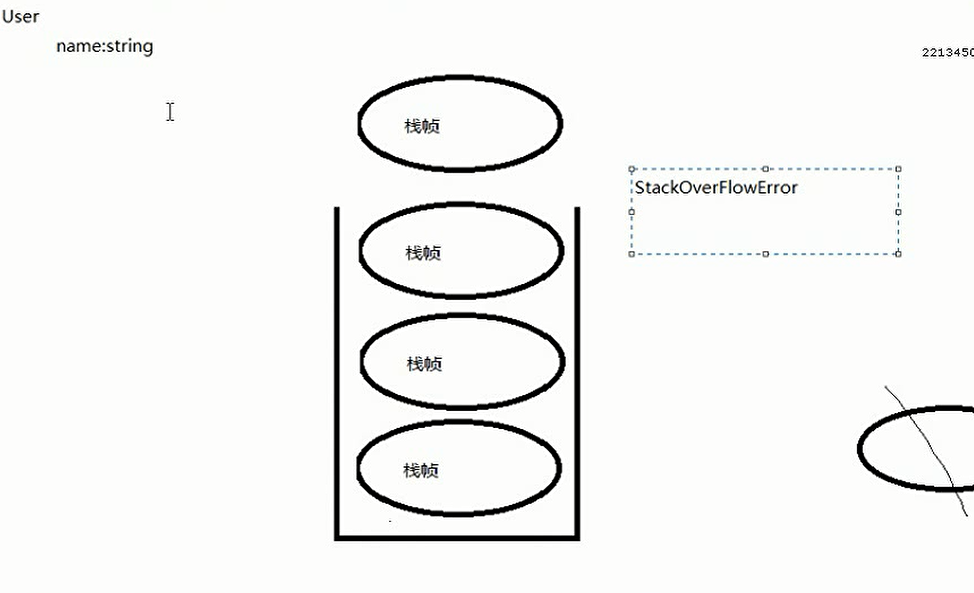
1. 虚拟机类型 一般都是HotSpotVM
2. 虚拟机内存管理



* 1. 栈

当要用到对象时，栈只存放对象的引用，所以运行期间，栈分配的内存是固定的。方法运行期间是不会改变的。





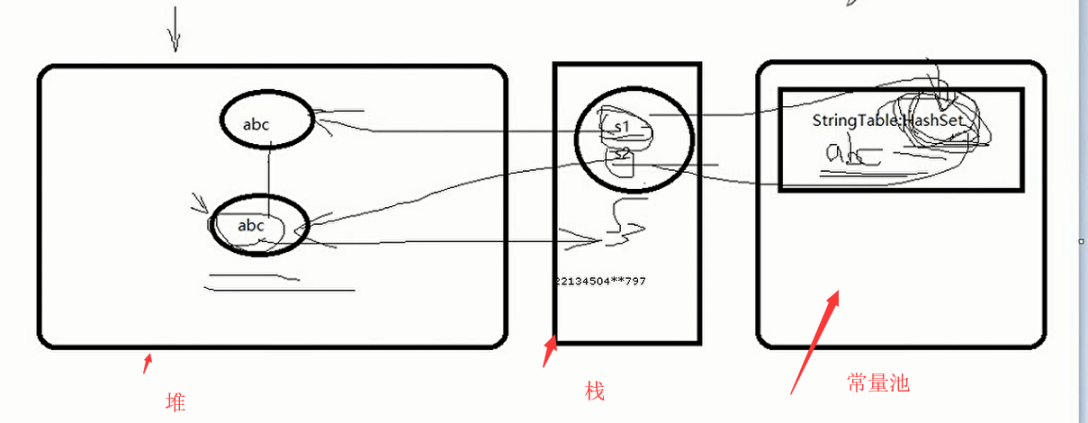
* 1. 堆

功能单一，只用来存放对象实例。所以也是GC的主要区域。

* 1. 方法区
     1. 存储虚拟机加载的类信息，常量，静态变量，即时编译器编译后的代码等数据

包括类的版本，字段，方法，接口

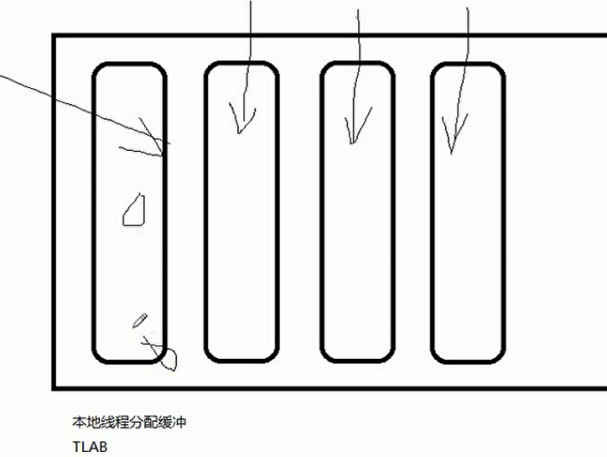
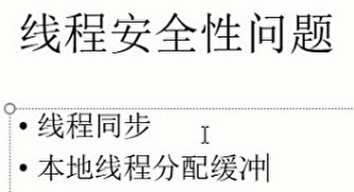
* + 1. 方法区和永久代并不等价。
    2. 运行池常量池是属于方法区的。



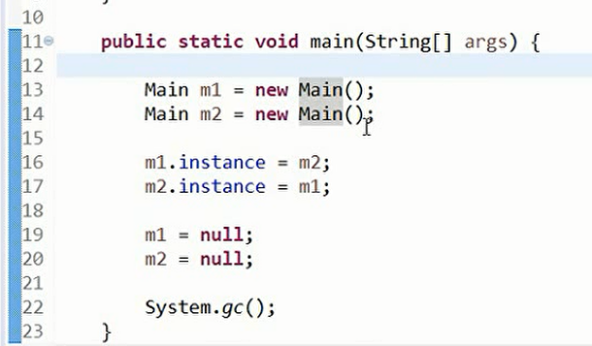
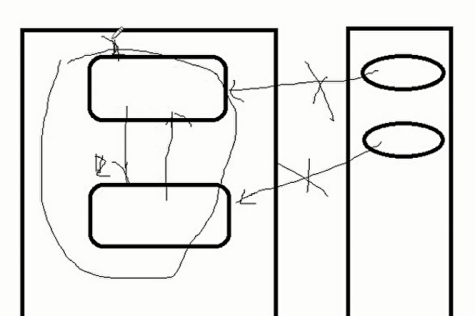


1. 对象创建



1. 对象结构
   1. header（对象头）
      1. 自身运行时数据(Mark word)
         1. HASH值，GC分代年龄，锁状态标志，线程持有的锁，偏向线程ID，偏向时间戳等
      2. 类型指针
   2. instancedata 保存实例数据
   3. padding 填充作用，没什么实际意义
2. 垃圾回收
   1. 打印垃圾回收日志
      1. 加参数-verbose:gc如果需要详细信息，可以再加个参数-XX:+PrintGcDetail
   2. 如何判定为垃圾对象
      1. 引用计数法
         1. 由于对象内部如果相互引用，但外边没有引用，计数器的值不会为0.所以会有问题。没有虚拟机使用这种方法



* + 1. 可达性分析法（主流虚拟机都采用此法）

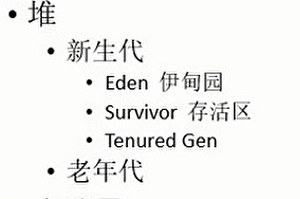


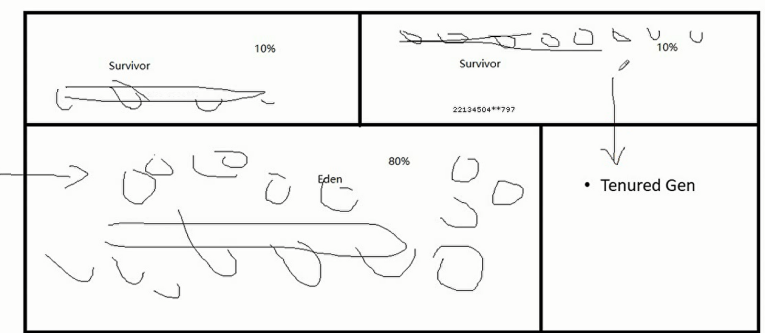
* 1. 如何回收
     1. 回收策略
        1. 标记-清除算法

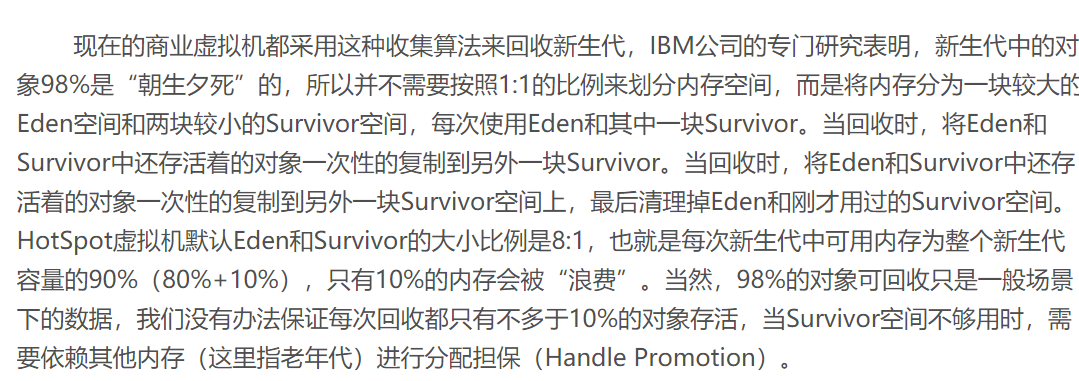
概念：被标记的被清除

缺点：效率太差，以后GC的时候在选址的时候会不连续，可能会再触发一次GC，还有空间问题。

* + - 1. 复制算法

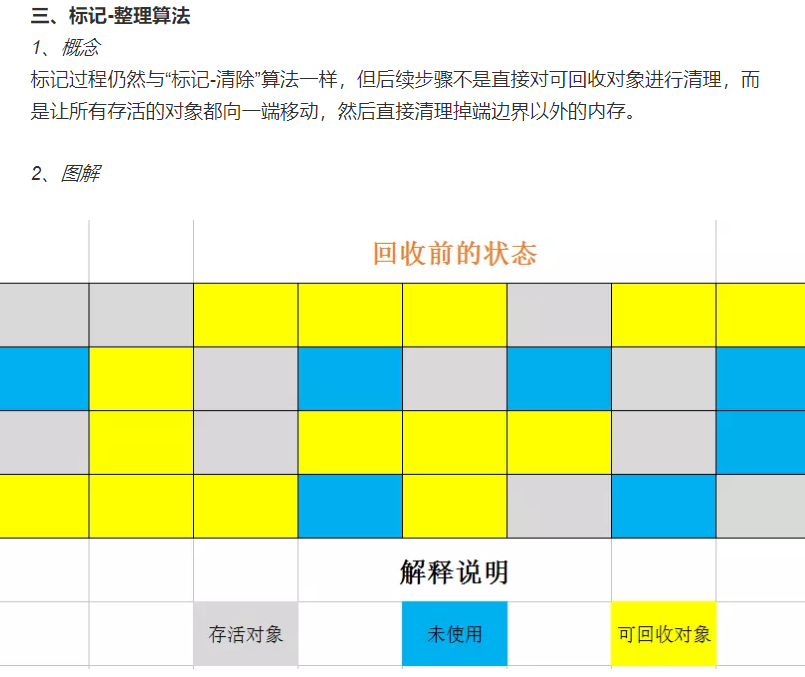


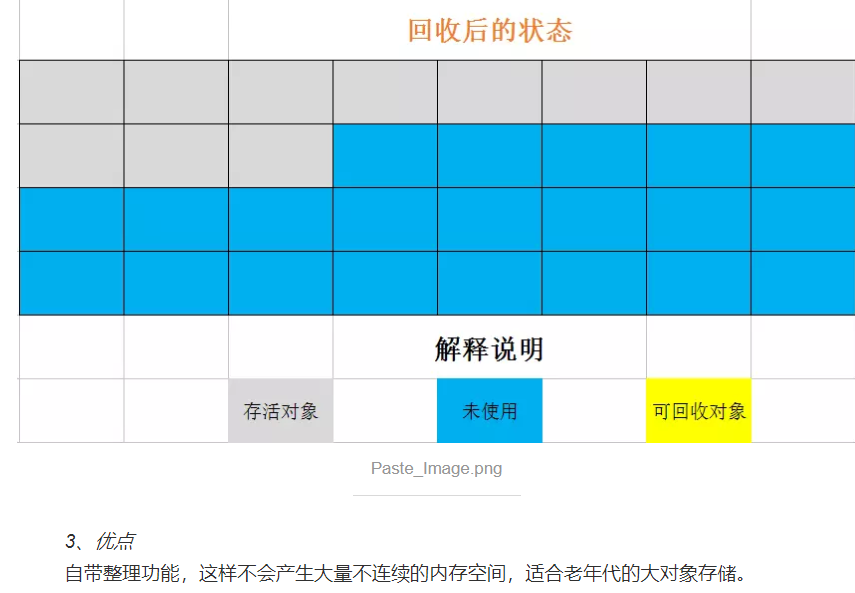




* + - 1. 标记-整理-清除算法

如果对于老年代来说，老年代有可能90%都是存活的，所以不适用于复制算法，浪费太多空间。对于垃圾回收率比较低的老年代，采用标记整理算法





* + - 1. 分代数据算法

是指根据新生代还是老年代选择不同的垃圾回收算法。

* + 1. 常见的垃圾回收器
       1. serial

最基本，发展最悠久，单线程垃圾收集器。应用于桌面应用。

* + - 1. parnew

多线程，新生代收集器，了解即可。

* + - 1. parellel

多线程，复制算法，新生代收集器

* + - 1. cms（concurrent mark sweep）

并发标记清除收集器。

* + - 1. G1（目前最好的）
         1. 优势

并行与并发

分代收集

空间整合

可预测的停顿

* + - * 1. 步骤

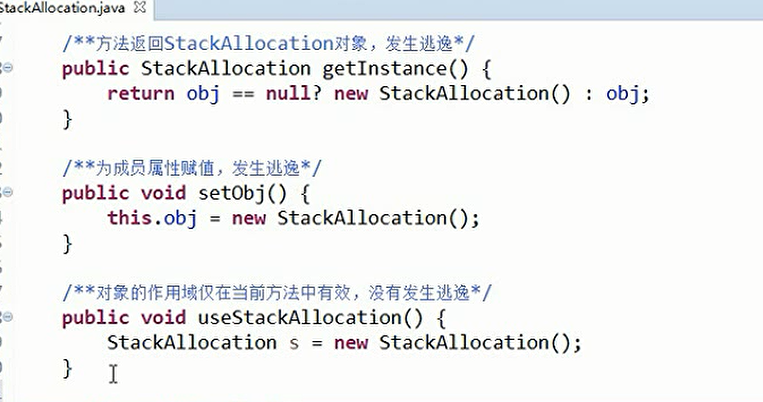
初始标记

并发标记

最终标记

筛选回收

1. 内存分配
   1. 概述
      1. 优先分配到eden
      2. 大对象直接分配到老年代
      3. 长期存活对象分配到老年代
      4. 空间分配担保
      5. 动态对象年龄判断
      6. 逃逸分析：如果这个对象只在这个方法体内部有效，那么就可以认为这个对象没有发生逃逸，就可以把这个对象分配到栈上去。



* 1. 打印测试
     1. 启用searial垃圾回收器-XX+UseSerialGC
     2. -Xms20M -Xmx20M 限制堆内存的区域
     3. -Xmn10M 指定新生代内存
     4. 62246800 0106401629