

使用OptimizeIT进行Java内存泄露的检测 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/271/2021_2022__E4_BD_BF_E7_94_A8Opti_c104_271652.htm java程序也会引起内存泄露已经

是公开的秘密了，工作中需要对该问题引起充分的重视。

市场上能够进行java内存检测的工具也有不少，我在工作中使用JBuilder自带的Optimize工具，现就最近一次使用Optimize的工程及经验进行以下梳理。前一段时间根据需要，对工作中维护的某个模块进行了java内存泄露的测试，结果发现了很严重的问题：该模块一共有6大基本功能，每个功能每操作一次，大约要新消耗9-10k的内存空间，生成110个左右的新对象。作为商用软件产品是绝不允许这样的问题的出现的。

OptimizeIT有整合进JBuilder版本，也有单独的版本，使用起来基本上都是一样的，它有4大功能，能够进行内存profile，cpu的profile，代码的覆盖率的检查，以及threads debug等，能够对普通应用程序，jsp/servlet，以及服务器请求等进行剖析。其它的没有用过，这次主要使用内存profile，发现这个这个咚咚比较好用。（其实，borland出品的工具比较很好用的，但是borland IDE目前的处境真是让人揪心啊）下面介绍如何使用OptimizeIT工具进行内存泄露的检测和解决。先说检测的思路：程序运行从一种状态进经过运行返回到初始状态，中间过程会产生很多的对象，而这些中间产物最终是应该被清除掉，如果没有被完全清除，就说明程序中有了内存泄露。换句话说，通过比较程序运行前后两种状态下堆内存中的对象的种类和数量的多少，可以判断出程序中有没有产生内存泄露。当终了时刻程序堆内存里面对象的种类和数量，

比初始时刻多，基本就可以确定程序中存在内存泄露了。接着说这次使用OptimizeIT检测的过程。

- 1.首先使用JBuilder建立一个可运行的工程，再进行工程运行属性的配置的时候可以同时进行OptimizeIT的配置，主要是选择内存泄露选项（以及其它的可选项，不过很少用）。
- 2.点击工具栏上的Optimize按钮开始检测，该按钮位于程序运行按钮的右边，带有一个秒表的样式。
- 3.Optimize启动之后会展示几个Tab页视图，我们主要关心内存视图。
- 4.默认的在内存视图中显示的是堆内存视图，通过在该视图下方输入过滤条件，在该视图的中部，可以得到我们关心的类的对象实例情况。
- 5.运行程序到某一个初始状态。
- 6.回到Optimize，运行垃圾回收器，为了尽可能调动GC，建议连续运行GC三次以上(简单的说就是多点击记下按钮而已)。
- 7.对当天的状态进行标记，也就是点击一下视图右上角的带铅笔样式的图标。
- 8.行程序的功能，然后关闭该功能，使程序的退回到原来的状态，基本上就是在Optimize上作标记时刻的界面样子。
- 9.Optimize堆内存视图，多次进行GC。
- 10.Optimize报告出从最近一次做标记以来内存对象（实例）的变化情况。由于我们已经进行过滤，所以Optimize不会显示我们关心的包（或者类）以外的对象的情况。

通过这种比较，我们可以发现程序中是否有内存泄露。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com