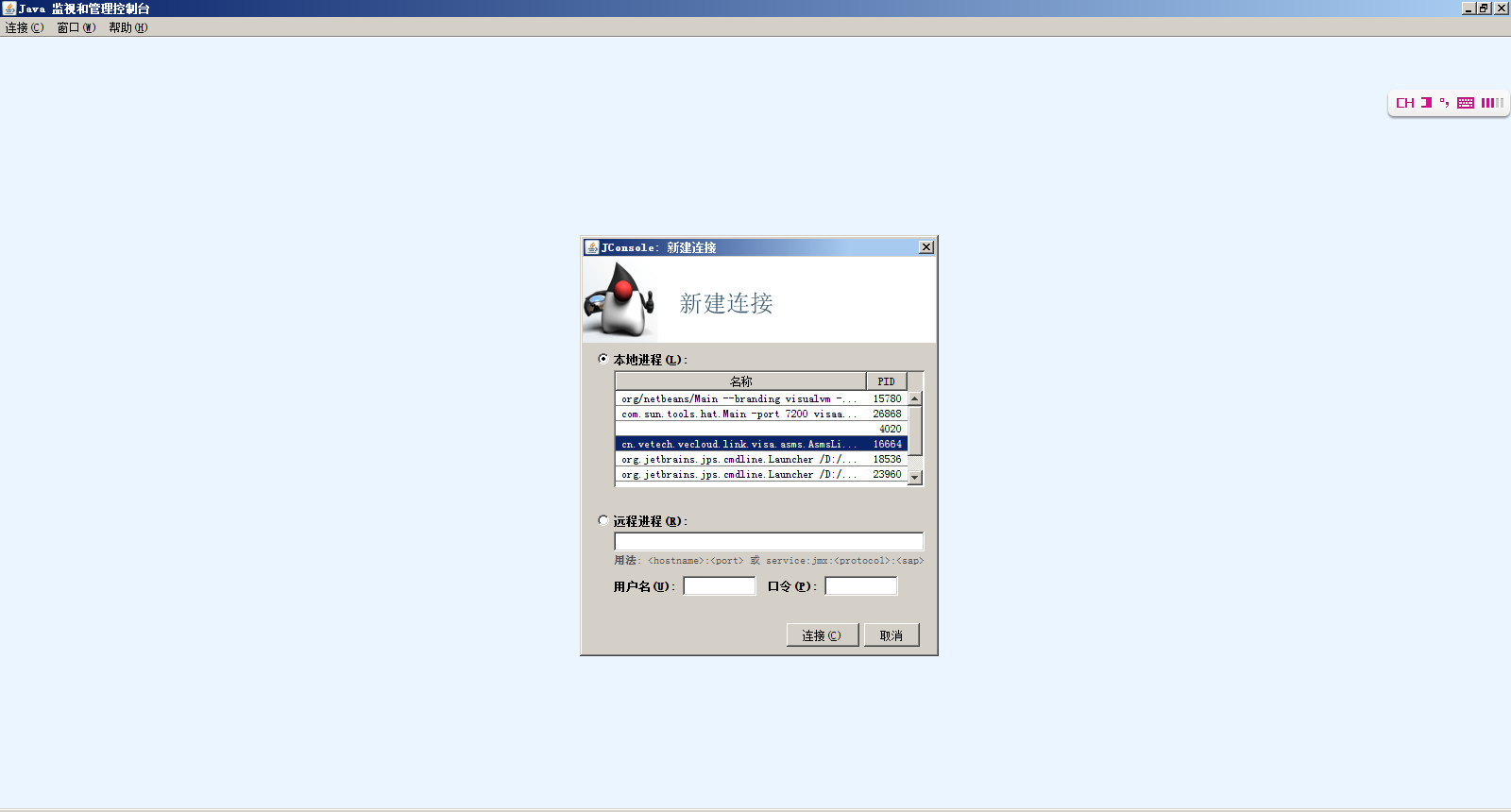
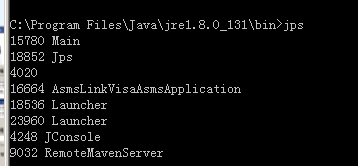
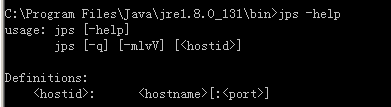
JVM内存溢出、泄露检查工具学习笔记

1. Cmd控制台： jconsole 回车，可以直接打开 java 监视管理控制台



1. Cmd控制台 jsp ，查看运行中的java进程pid





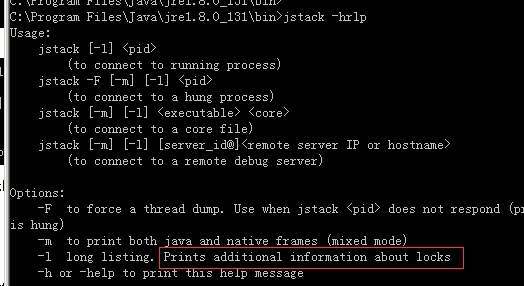
-q 不输出类名、Jar名和传入main方法的参数

-m 输出传入main方法的参数

-l 输出main类或Jar的全限名

-v 输出传入JVM的参数

1. Cmd控制台：jstack pid ，查看进程堆栈信息



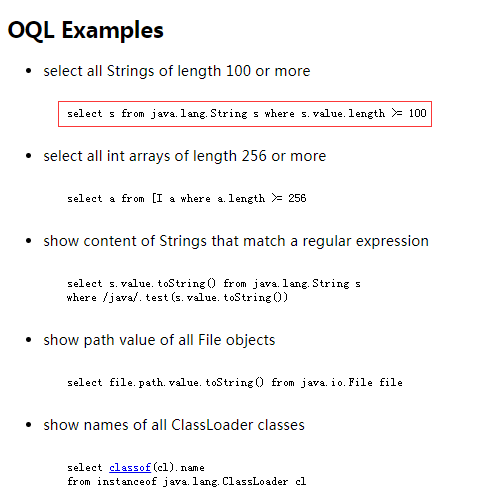
1. Cmd控制台: jmap**-dump:live,file=xxx.map [正在运行java的进程PID] ，可以导出进程JAVA相关信息，**例如：jmap -dump:live,file=visa-asms-link.map 16664

Cmd控制台: jhat -J-Xmx512m -port 7000 visa-asms-link.map ，可以启动简单预览的WEB服务

[http://localhost:7000/](http://localhost:7000/" \t "http://maclab.iteye.com/blog/_blank)  这里看到所有的类

进入oql查询界面   [http://localhost:7000/oql/](http://localhost:7000/oql/ " \t "http://maclab.iteye.com/blog/_blank)

具体OQL用法可见：[http://localhost:7000/oqlhelp/](http://localhost:7000/oqlhelp/" \t "http://maclab.iteye.com/blog/_blank)



B byte

C char

D double

F float

I int

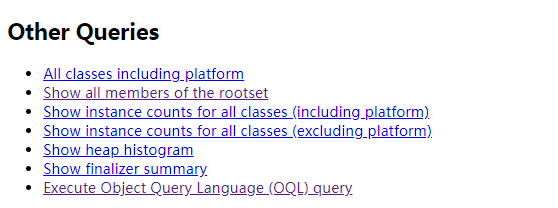
J long

Z boolean

[ 数组，如[I表示int[]

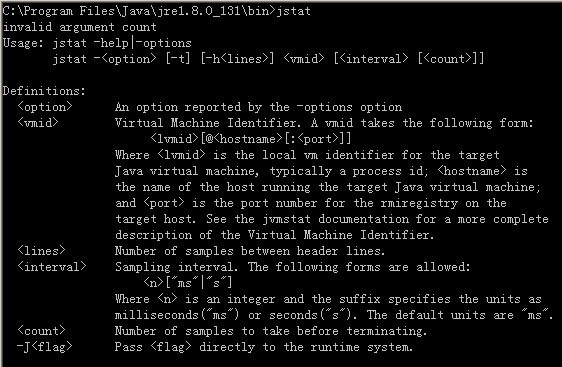
[L+类名 其他对象

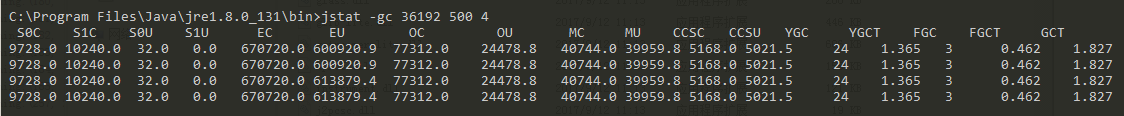
最下面有其他信息链接：



下载软件：Java VisualVM工具，连接某个线程后，点击head dump也可以查看和jmap&jhat实现的功能。

1. Cmd命令: jstat（JVM统计监测工具）





S0C、S1C、S0U、S1U：Survivor 0/1区容量（Capacity）和使用量（Used）

EC、EU：Eden区容量和使用量

OC、OU：年老代容量和使用量

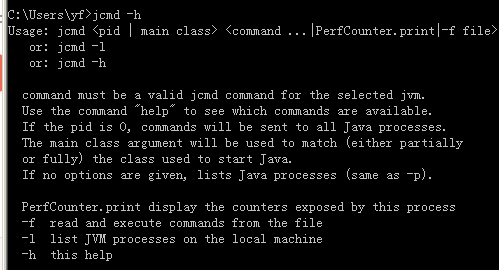
PC、PU：永久代容量和使用量

YGC、YGT：年轻代GC次数和GC耗时

FGC、FGCT：Full GC次数和Full GC耗时

GCT：GC总耗时

1. Cmd命令：jcmd



jcmd 31275 Thread.print -l # 打印线程栈

jcmd 31275 VM.command\_line # 打印启动命令及参数

jcmd 31275 GC.heap\_dump /data/31275.dump # dump heap

jcmd 31275 GC.class\_histogram #查看类的统计信息

jcmd 31275 VM.system\_properties #查看系统属性内容

jcmd 31275 VM.uptime #查看虚拟机启动时间

jcmd 31275 PerfCounter.print #查看性能统计