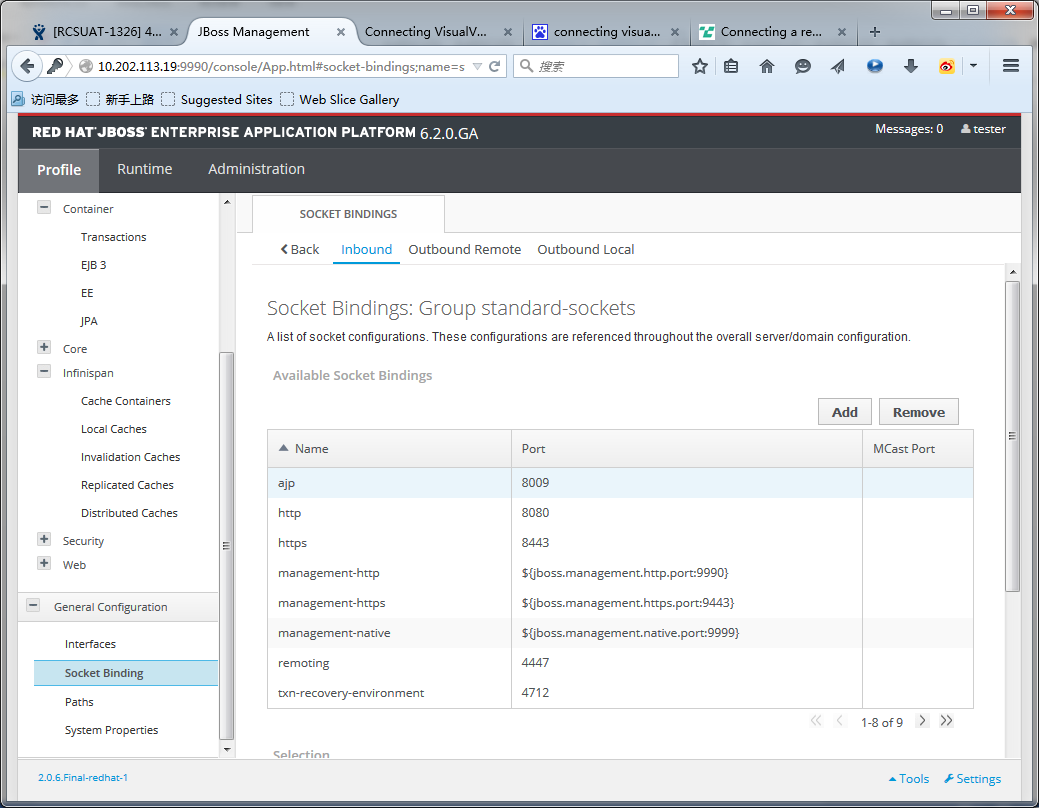
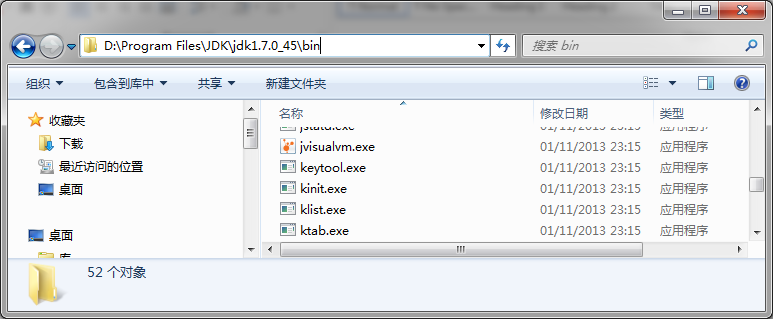
Session1 Jvisualvm setup.

1. Use {JBOSS\_HOME}/bin/add-user.sh username password” to create a user in ManagementRealm. For domain mode, please use “{JBOSS\_HOME}/bin/add-user.sh –a username password” to create a user in ApplicationRealm. This user will be used by jvisualvm to connect jboss.
2. Use parameter “-Djboss.bind.address.management=” to start jboss. For example, if the jboss is in standalone mode, use “standalone.bat -Djboss.bind.address.management={the IP of the server}” so thatjvisualvm.exe can access jboss remotely.
3. Open the admin console of jboss. And open below screen:

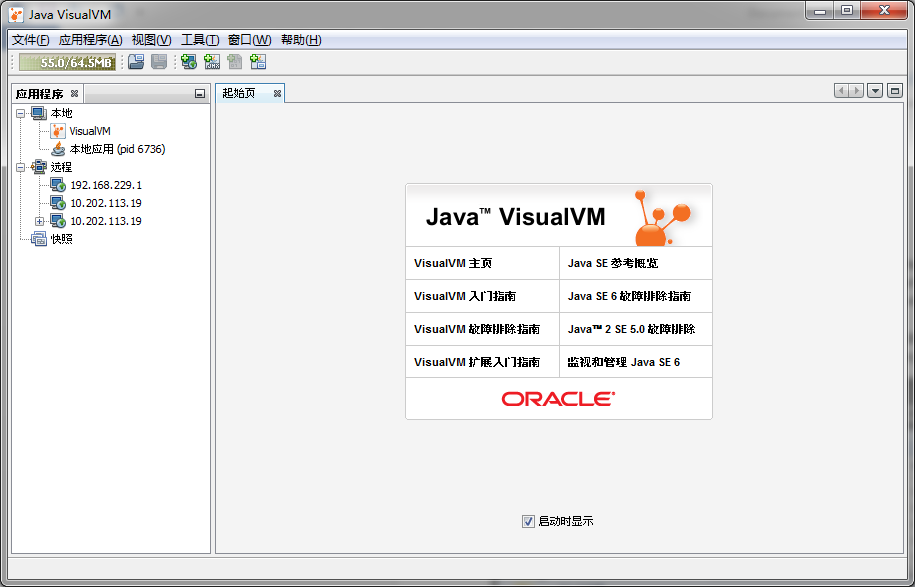


By default, the port for management-native is 9999 and remoting is 4447. In this screen, we can verify whether this default setting has been changed. In standalone mode, management-native should be used. In domain mode, remoting should be used.

1. In the web application server, please find the jboss-client.jar from “{JBOSS\_HOME}\client \jboss-client.jar”. Assume this jar is copy to the PC that is used to run jvisualvm.exe and saved in “D:\download\jboss-client.jar”
2. Got a JDK 1.7 and copy the JDK1.7 to a PC(No need to install, JDK also work even though you just copy the whole folder to other PC). Then, you can find the jvisualvm.exe in the bin folder



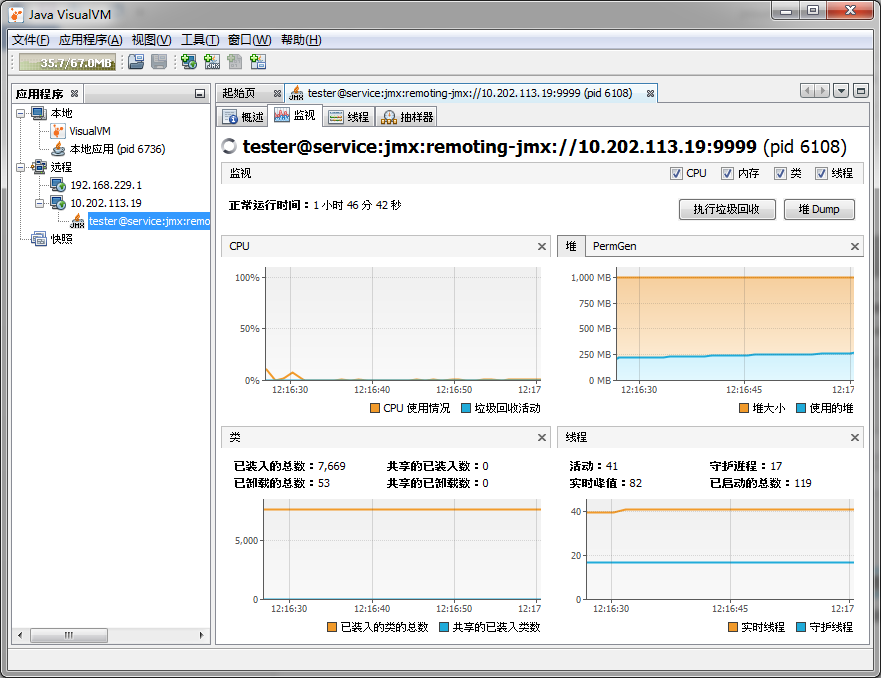
1. In the PC that is used to run jvisualvm.exe, run command “jvisualvm -cp d:\download\jboss-client.jar”, you will see:



1. In the jvisualvm, open menu “File-> Add JMX Connection”, below dialog will be shown. Key in url, user id, password. The format of url must be “service:jmx:remoting-jmx://{jboss IP address}:{port}” . For standalone mode, the port should be the “management-native”. For domain mode, the port should be the “remoting” which has been explained in point 3. For user id and password, please refer point 1.

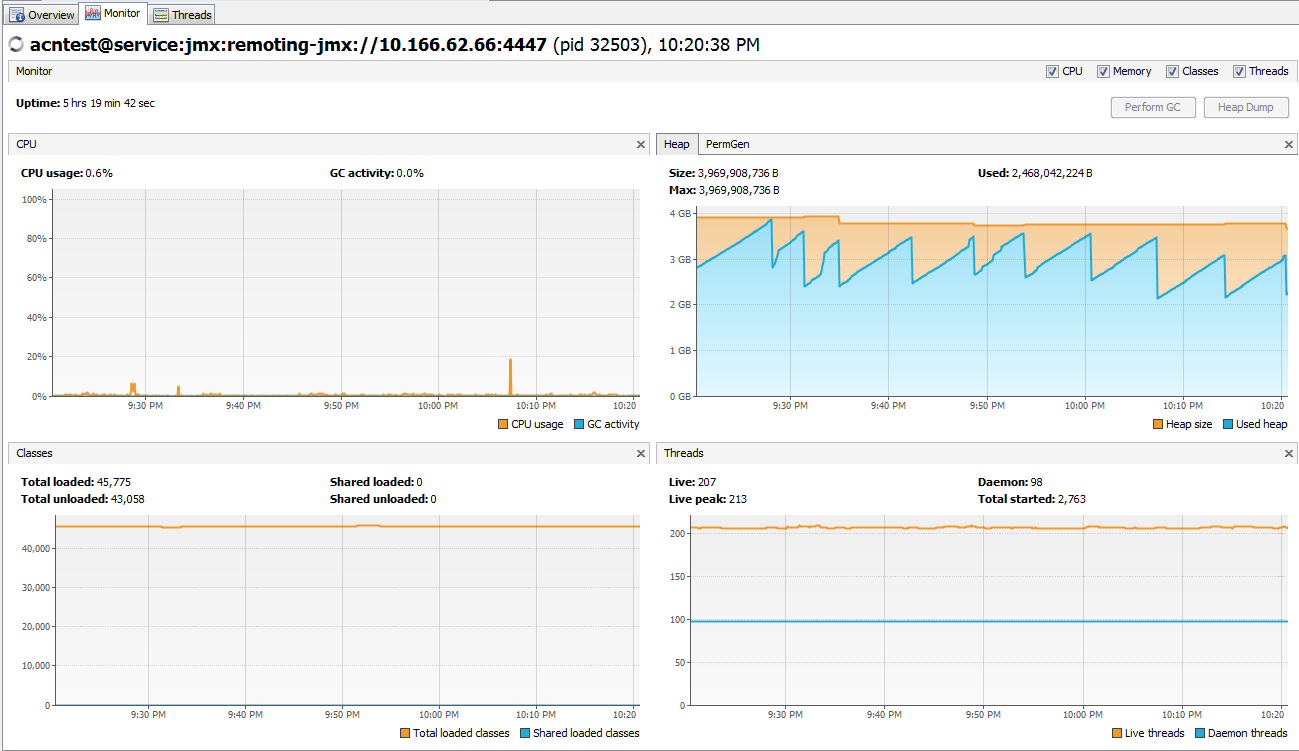


1. If jvisualvm connect jboss successfully, you will see:



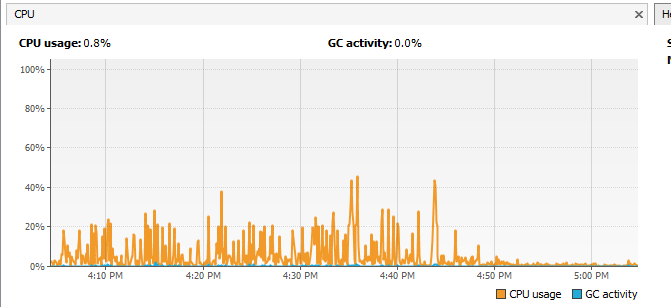
Session2 Monitor.

1. By default, there are four part in Monitor page.

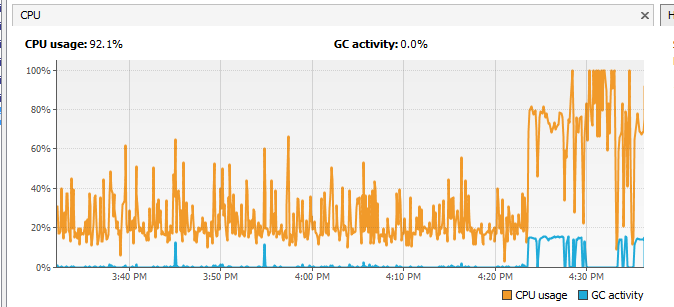


Part 1 CUP

Normally, CUP usage will stable within a low certain range.



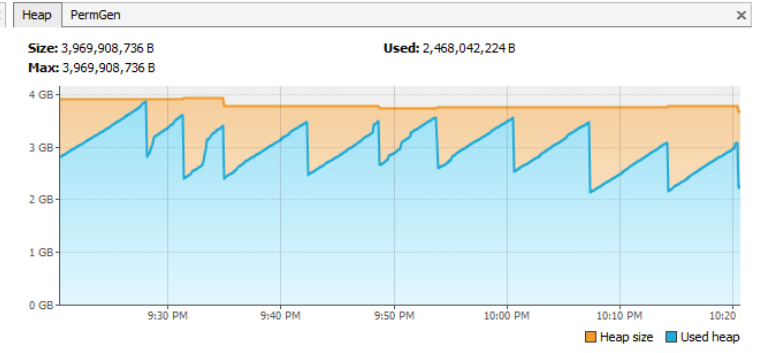
When there is a performance issue, the CUP usage may increase suddenly and remain in high level.



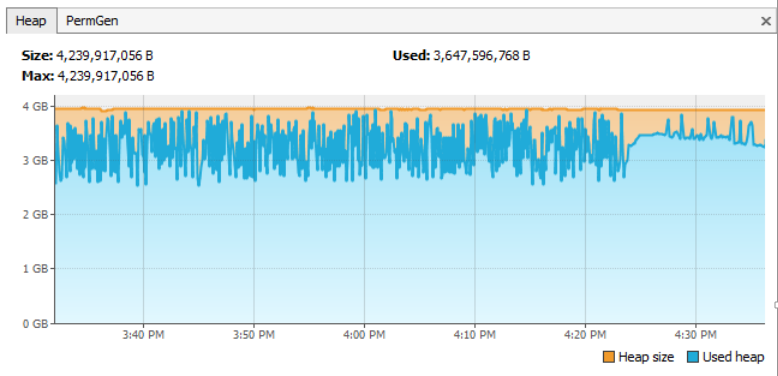
Part 2 Memory

这个区域分成 JVM Heap Space 和 PermGen Space.

Heap Space 正常情况下每次GC会释放大量内存，因此程现锯齿形。



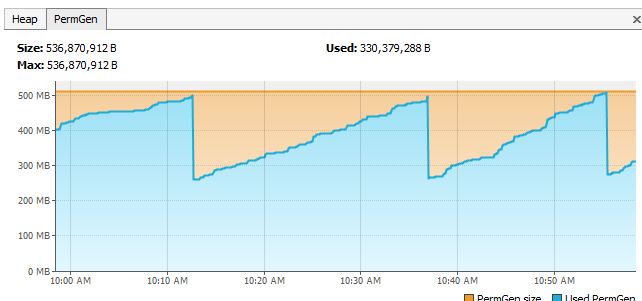
异常情况时，内存不能被释放，因此不断接近最大值并且频繁GC，



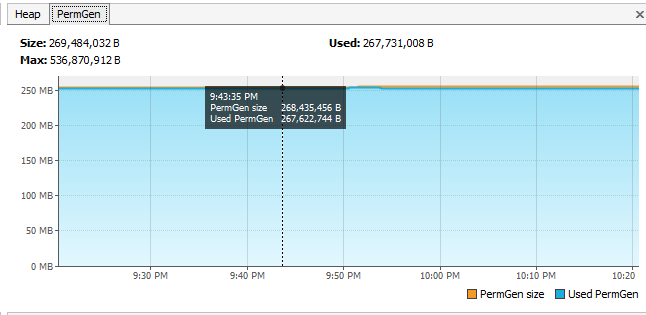
PermGen Space

现象和Heap Space一样，只是这个区域的GC频率要比Heap Space低很多。

正常时

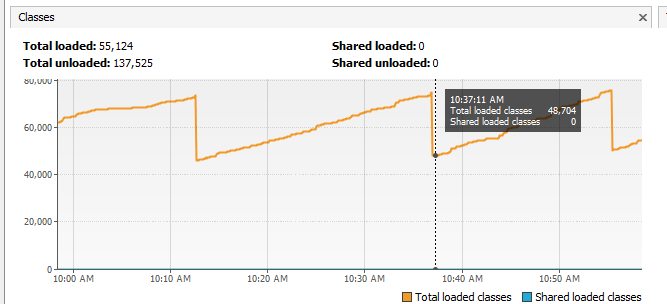


异常时，维持在最大值附近并伴随频繁FULL GC



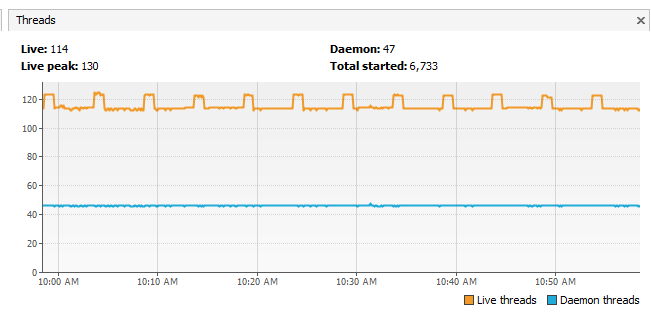
Part 3 Classes

这个区域的情况与PermGen Space相对应

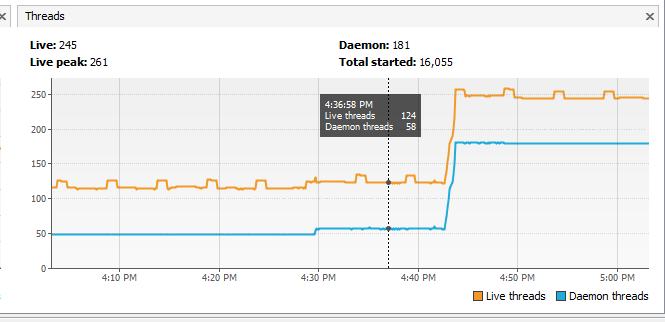


Part 4 Threah

正常情况下，线程数会保持在JBOSS初始化的线程池大小



当系统异常响应变慢时，Thread pool会自动增加线程数量来处理新进的请求。



当系统出现异常时，通常会四个区域都受到影响

