java服务程序使用内存的几个制约因素  
**1、jvm使用内存限制**  
在32位windows的机器上，堆最大可以达到1.4G至1.6G。  
在32位solaris的机器上，堆最大可以达到2G  
而在64位的操作系统上，32位的JVM，堆大小可以达到4G  
  
**2、创建的线程不使用jvm内存，而是向操作系统申请内存**  
在java中每new一个线程，jvm都是向操作系统请求new一个本地线程，此时操作系统会使用剩余的内存空间来为线程分配内存，而不是使用jvm的内存。  
这样，当操作系统的可用内存越少，则jvm可用创建的新线程也就越少。  
  
**3、系统进程最大内存限制**主要是32位进程的限制  
32位Windows上，最大为3G（执行程序加编译参数，系统修改启动3G设置）  
64位Windows上，最大为4G（执行程序加编译参数）  
32位和64位Linux上，理论上最大可达到4G  
**总结**由此可以得出公式：  
**JVM内存 + 线程使用内存 = 进程使用内存**  
  
因此如果是线程很多的服务，就得根据需要调小JVM内存，以给线程留出足够的内存空间

有的时候不是-Xmx给得越大就使程序的可用内存更多，给JVM内存更多会使JVM以外内存更少，使用到NativeLibrary时尤其有影响