

Performance Engineering

Out of memory

Monitor the growth of the heap.

If heap gets filled up in short time, increase the heap size.

Enable GC log. Check the effect of full GC.

If heap not getting freed, memory leak case.

Take heap dump. Investigate heap and find out what's sticking around for long.

Use JvisualVM / Eclipse MAT to determine the objects with large retained heap.

If heap gets freed, increase the heap size.

Slower program

Find CPU IO wait time and CPU idle time using top. CPU idle time - anything >25% considered good enough. CPU IO wait - > 10% is high.

CPU IO wait time is low and the CPU idle time is low.

Check CPU user time

If user time is high, server CPU has been utilized. Use top command to check the list of process consuming CPU

CPU IO wait time is low and the CPU idle time is high.

Application not using CPU

Analyze application

Threads blocked on monitor? Dead lock?

Use profiler & thread dump to spot the issue

Parallelize the program.

CPU IO wait time is high

Look at the swap usage

High swap usage

Less RAM. Upgrade RAM.

Less swap usage

Real IO problem.

Make sure disk usage below 80% to avoid excessive fragmentation

Have free physical memory so the OS can cache disk blocks in memory. Higher IOwait could be because of disk swapping as well.

Choose good buffer sizes when performing IO operations

Find out the type of disk being used. Look the average IOPS for the disk

Single process?

Restart that process. CPU consumption still abnormally high?

Upgrade. Add more cores?

Waiting on network?

Check thread dump

Look at garbage collection

Use gc log. Look at the time GC takes. If full gc happens with less bytes reclaimed, potentially a memory leak.

Look at loops

Look at hotspot methods.

Code execution taking much time.

Look at loops

Look at hotspot methods.

CPU max out

Look at running threads

Infinte loops?

Waiting on network?

Heap full and full GC running with less JVM throughput?

Mem leak

Analyse heap

Multiple processes?

Upgrade your sever. Add cpu cores or scale horizontally