

jstat 监视Java虚拟机 (JVM) 统计信息。

用法 `jstat [generalOption | outputOptions vmid [interval [s | ms] [count]]`

- statOption

确定jstat命令显示的统计信息。以下列出了可用选项。使用-options常规选项显示特定平台安装的选项列表。请参见统计选项和输出。

class: 显示有关类加载器行为的统计信息。

compiler: 显示有关Java HotSpot VM实时编译器行为的统计信息。

gc: 显示有关垃圾回收堆行为的统计信息。

gccapacity: 显示有关代的容量及其相应空间的统计信息。

gccause: 显示有关GC统计信息 (-gcutil与之相同) 的摘要, 以及最后一个和当前GC事件的原因。

gcnew: 显示新生代行为的统计信息。

gcnewcapacity: 显示有关新生代及其相应空间大小的统计信息。

gcold: 显示有关老年代和Metaspace统计信息行为的统计信息。

gcoldcapacity: 显示有关老年代大小的统计信息。

gcmetacapacity: 显示有关Metaspace大小的统计信息。

gcutil: 显示有关垃圾回收统计信息的摘要。

printcompilation: 显示Java HotSpot VM编译方法统计信息。

-h n 每n条数据后显示一个列标题。

-t 将时间戳列显示为输出的第一列。时间戳是从JVM启动时间。

-J javaOption

传递javaOption给Java应用程序启动器。例如, -J-Xms48m将启动内存设置为48 MB。

-class 选项

类加载器统计信息

Loaded: 加载的类数。

Bytes: 加载的kB数。

Unloaded: 卸载的类数。

Bytes: 卸载的Kbytes数。

Time: 执行类加载和卸载操作所花费的时间。

- 编译器选项

Java HotSpot VM即时编译器统计信息。

Compiled: 执行的编译任务数。

Failed: 编译任务失败的次数。

Invalid: 已失效的编译任务数。

Time: 执行编译任务所花费的时间。

FailedType: 编译上次失败的编译的类型。

FailedMethod: 上次失败编译的类名和方法。

-gc 选项

GC堆统计信息。

类指针压缩空间, 每个对象都有一个指向它自身类的指针, _klass: 指向类的4字节指针, 64位平台上_klass: 指向类的8字节的指针, 为了节约这些空间, 引入了类指针压缩空间

S0C: 当前S0容量 (kB) 。

S1C: 当前S1容量 (kB) 。

S0U: S0利用率 (kB) 。

S1U: S1利用率 (kB) 。

EC: Eden容量 (kB) 。

EU: Eden利用率 (kB) 。

OC: 老年代容量 (kB) 。

OU: 老年代利用率 (kB) 。

MC: Metaspace容量 (kB) 。

MU: Metaspace利用率 (kB) 。

CCSC: 类指针压缩空间容量 (kB) 。

CCSU: 使用的类指针压缩空间 (kB) 。

YGC: 新生代GC活动的数量。

YGCT: 新生代GC时间。

FGC: Full GC的数量。

FGCT: Full GC时间。

GCT: GC总时间。

-gccapacity 选项

内存池生成和空间容量。

NGCMN: 最小新生容量 (kB) 。

NGCMX: 最大新生容量 (kB) 。

NGC: 当前的新生代容量 (kB) 。

S0C: 当前S0容量 (kB) 。

S1C: 当前S1容量 (kB) 。

EC: 当前的Eden空间容量 (kB) 。

OGCMN: 最小老年代容量 (kB) 。

OGCMX: 最大老年代容量 (kB) 。

OGC: 当前的老年代容量 (kB) 。

OC: 当前老年代容量 (kB) 。

MCMN: 最小Metaspace容量 (kB) 。

MCMX: 最大Metaspace容量 (kB) 。

MC: Metaspace容量 (kB) 。

CCSMN: 类指针压缩空间最小容量 (kB) 。

CCSMX: 类指针压缩空间最大容量 (kB) 。

CCSC: 类指针压缩空间容量 (kB) 。

YGC: 新生代GC事件的数量。

FGC: FullGC事件的数量。

-gcause 选项

上次GC事件的原因以及当前GC事件的原因。

LGCC: 上次垃圾回收的原因

GCC: 当前垃圾回收的原因

-gcnew 选项

新生代统计数据。

S0C: 当前S0容量 (kB) 。

S1C: 当前S1容量 (kB) 。

S0U: S0利用率 (kB) 。

S1U: S1利用率 (kB) 。

TT: JVM动态调整的移动到老年代的阈值。

MTT: 移动到老年代的最大阈值, 超过这个YGC次数, 一定会进入老年代。

DSS: 期望的survivor大小 (kB) 。

EC: 当前的Eden空间容量 (kB) 。

EU: Eden空间利用率 (kB) 。

YGC: 新生代GC事件的数量。

YGCT: 新生代GC时间。

-gcnewcapacity 选项

新生代空间大小统计。

NGCMN: 最小新生容量 (kB) 。

NGCMX: 最大新生容量 (kB) 。

NGC: 当前的新生代容量 (kB) 。

S0CMX: 最大S0容量 (kB) 。

S0C: 当前S0容量 (kB) 。

S1CMX: 最大S1容量 (kB) 。

S1C: 当前S1容量 (kB) 。

ECMX: 最大Eden空间容量 (kB) 。

EC: 当前的Eden空间容量 (kB) 。

YGC: 新生代GC事件的数量。

FGC: FullGC事件的数量。

-gcold 选项

老年代和Metaspace行为统计。

MC: Metaspace容量 (kB) 。

MU: Metaspace利用率 (kB) 。

CCSC: 类指针压缩空间容量 (kB) 。

CCSU: 使用的类指针压缩空间 (kB) 。

OC: 当前老年代容量 (kB) 。

OU: 老年代利用率 (kB) 。

YGC: 新生代GC事件的数量。

FGC: FullGC事件的数量。

FGCT: FullGC时间。

GCT: GC总时间。

-gcoldcapacity 选项

老年代大小统计。

OGCMN: 最小老年代容量 (kB) 。

OGCMX: 最大老年代容量 (kB) 。

OGC: 当前的老年代容量 (kB) 。

OC: 当前老年代容量 (kB) 。

YGC: 新生代GC事件的数量。

FGC: FullGC事件的数量。

FGCT: FullGC时间。

GCT: GC总时间。

-gcmetacapacity 选项

Metaspace大小统计。

MCMN: 最小Metaspace容量 (kB) 。

MCMX: 最大Metaspace容量 (kB) 。

MC: Metaspace容量 (kB) 。

CCSMN: 类指针压缩空间最小容量 (kB) 。

CCSMX: 类指针压缩空间最大容量 (kB) 。

YGC: 新生代GC事件的数量。

FGC: FullGC事件的数量。

FGCT: FullGC时间。

GCT: GC总时间。

-gcutil 选项

GC统计摘要。

S0: S0利用率占空间当前容量的百分比。

S1: S1占空间当前容量的百分比。

E: Eden空间利用率占空间当前容量的百分比。

O: 老年代利用率占空间当前容量的百分比。

M: Metaspace利用率占空间当前容量的百分比。

CCS: 压缩的类空间利用率百分比。

YGC: 新生代GC事件的数量。

YGCT: 新生代GC时间。

FGC: FullGC事件的数量。

FGCT: FullGC时间。

GCT: GC总时间。

-printcompilation 选项

Java HotSpot VM编译方法统计。

Compiled: 最近编译的方法执行的编译任务数。

Size: 最近编译的方法的字节代码的字节数。

Type: 最近编译的方法的编译类型。

Method: 标识最近编译的方法的类名和方法名。类名使用斜杠 (/) 而不是点 (。) 作为名称空间分隔符。方法名称是指定类中的方法。这两个字段的格式与HotSpot -XX:+PrintCompilation选项一致。