

jmap查看内存使用情况与生成heapdump--转 - 一天不进步，就是退步 - 博客园

如果想分析自己的JAVA Application时，可以使用jmap程序来生成heapdump文件。

例：

```
jmap -heap 1234
```

(1234为进程号)

jmap是JDK自带的一个工具，非常小巧方便，其支持参数如下：

-heap

打印heap空间的概要，这里可以粗略的检验heap空间的使用情况。

例：

```
jmap -heap 12345
```

输出：

```
Attaching to process ID 2657, please wait...
```

```
Debugger attached successfully.
```

```
Client compiler detected.
```

```
JVM version is 1.5.0_16-b02
```

```
using thread-local object allocation.
```

```
Mark Sweep Compact GC
```

Heap Configuration:

```
MinHeapFreeRatio = 40
MaxHeapFreeRatio = 70
MaxHeapSize      = 67108864 (64.0MB)
NewSize          = 655360 (0.625MB)
MaxNewSize       = 4294901760 (4095.9375MB)
OldSize          = 1441792 (1.375MB)
NewRatio         = 12
SurvivorRatio    = 8
PermSize         = 8388608 (8.0MB)
MaxPermSize      = 67108864 (64.0MB)
```

Heap Usage:

New Generation (Eden + 1 Survivor Space):

```
capacity = 4521984 (4.3125MB)
used      = 1510200 (1.4402389526367188MB)
free      = 3011784 (2.8722610473632812MB)
33.39684527853261% used
```

Eden Space:

```
capacity = 4063232 (3.875MB)
used      = 1495992 (1.4266891479492188MB)
free      = 2567240 (2.4483108520507812MB)
36.81778446320565% used
```

From Space:

```
capacity = 458752 (0.4375MB)
used      = 14208 (0.0135498046875MB)
```

```

    free      = 444544 (0.4239501953125MB)
3.0970982142857144% used
To Space:
    capacity = 458752 (0.4375MB)
    used     = 0 (0.0MB)
    free     = 458752 (0.4375MB)
0.0% used
tenured generation:
    capacity = 59342848 (56.59375MB)
    used     = 36321192 (34.638587951660156MB)
    free     = 23021656 (21.955162048339844MB)
61.20567721994064% used
Perm Generation:
    capacity = 11796480 (11.25MB)
    used     = 11712040 (11.169471740722656MB)
    free     = 84440 (0.08052825927734375MB)
99.28419325086806% used

```

以上的输出很简单，第四行起开始输出此进程我们的JAVA使用的环境。

Heap Configuration，指在我们启动时设置的一些JVM参数。像最大使用内存大小，年老代，年青代，持久代大小等。有这个可以很简单的查看本进程的内存使用情况。也许进程占用的总内存比较多，但我们在这里可以看到真正用到的并没有多少，很多都是“Free”。内存使用的堆积大多在老年代，内存泄露始于此，所以要格外关心“tenured generation”。

-heap:format=b

产生一个HeapDump文件，此为生成heapdump文件的重要参数。

例：jmap -heap:format=b 2657

会产生一个heap.bin的heapdump文件。

需要注意的是，此生成heapdump的参数为JDK1.5，在1.6中的格式为：

jmap -dump:live,format=b,file=xxx 2657

这里更加强大一些，可以指定是存活的对象，还有生成heapdump的文件名。

-histo

这里会生成一个类的统计报表，此表简单无比，如显示什么类有多少个实例，共占了多少字节等，如下：

Size	Count	Class description
8394352	105	long[]
8293192	57202	char[]
7834776	14157	byte[]
6713592	53743	* ConstMethodKlass
4194320	1	com.xjava.cms5server.Kontent[]
4055072	12319	int[]
3291104	85082	* SymbolKlass
3016040	53743	* MethodKlass
2774936	4253	* ConstantPoolKlass
1871480	4253	* InstanceKlassKlass
1811808	3990	* ConstantPoolCacheKlass
1488672	62028	java.lang.String
1203280	13258	java.lang.Object[]

-permstat

打印一些持久代上的内存使用状态，有“活”的，有“死”的。本人用之不多。

