

STRING INTERNING AND PERFORMANCE TUNING IN JAVA

String interning is a method of storing only one copy of each distinct string value. Strings in Java are immutable and hence this sharing is perfectly safe and give you better performance. The distinct values are stored in a fixed-size hashtable usually referred to as string intern pool or string pool. The single copy of each string is called its 'intern'. You can read more about the basics of String interning with examples

The table of interned strings is held in native memory (not heap memory) as a fixed-size hashtable. In releases prior to Java 7u40, the default size of the table is 1,009 buckets. In 64-bit versions of Java 7u40 and later, the default size is 60,013. This table should be sized according to the expected data for better performance. Starting in Java 7, the size of this table can be set when the JVM starts by using the flag:

```
-XX:StringTableSize=<value>
```

If an application will intern a lot of strings, this number should be increased. And when you set it, make sure that the value is a prime number for best efficiency.

Ideal bucket size

What is a **bucket**? A hashtable data structure contains an array that can hold some number of entries and each element in the array is called a bucket. In simple terms, the bucket location is calculated using a hashing algorithm. Two elements can have the same value for hashing algorithm, and this situation is called collision. A collision is usually handled by creating a linked list at each array location (bucket) and adding every element that hashes to that bucket to that linked list. Read more about hashing for more clarity.

Ideally, the average bucket size should be 0 or 1 which means there are no collisions, and hence better performance. If the averages are larger than 1, we can increase the string table size accordingly. Each bucket takes only 4 or 8 bytes (depending on whether you have a 32-

or 64-bit JVM). So there won't be any notable penalty for setting the size of the string table too high.

String interning statistics

To see the details of your string table, run your application with the flag

```
-XX:+PrintStringTableStatistics
```

This flag requires JDK 7u6 or later and default value is false. When the JVM exits, it will print out a table with details such as number of buckets, average bucket size, variance of bucket size, std. dev. of bucket size and maximum bucket size. This command also displays information about the symbol table. The symbol table is used to hold some class information. This flag is not generally tunable, but JDK 8 has an experimental flag to adjust the size of that table.

The number of interned strings an application and their total size can also be obtained using the jmap command as:

```
jmap -heap <process_id>
```

Within the output, look out for 'interned Strings occupying' phrase as:

```
3000 interned Strings occupying 300000 bytes
```

Source : <http://javajee.com/string-interning-and-performance-tuning-in-java>