Performance Analysis

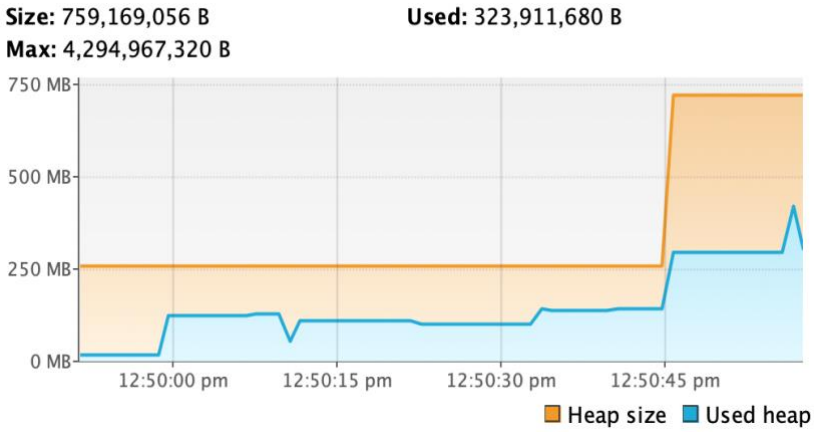
This report compares the time and memory consumption using MemAppender with list types LinkedList and ArrayList, FileAppender and ConsoleAppender. Each appender is compared using VelocityLayout and PatternLayout.

Results:

MemAppender using ArrayList and VelocityLayout:

Total Time Taken:
1 minute 9 seconds

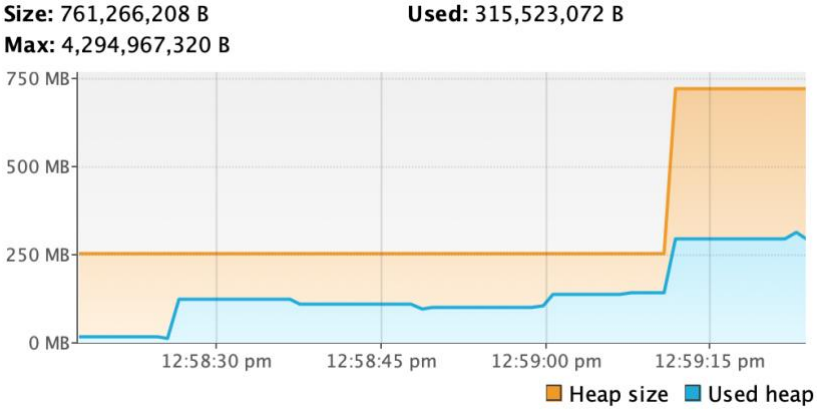
maxSize	Time Taken
1	11s 551ms
10	11s 458ms
1,000	11s 457ms
10,000	11s 548ms
100,000	11s 490ms
1,000,000	11s 507ms



MemAppender using ArrayList and PatternLayout:

Total Time Taken:
1 minute 9 seconds

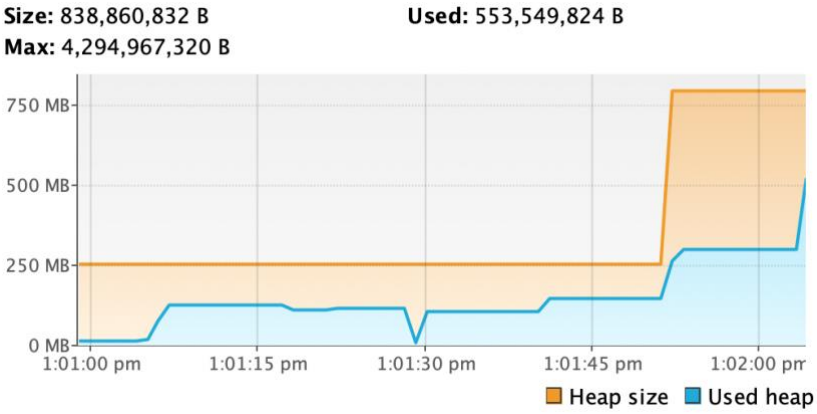
maxSize	Time Taken
1	11s 563ms
10	11s 453ms
1,000	11s 453ms
10,000	11s 543ms
100,000	11s 494ms
1,000,000	11s 505ms



MemAppender using Linked List and VelocityLayout:

Total Time Taken:
1 minute 9 seconds

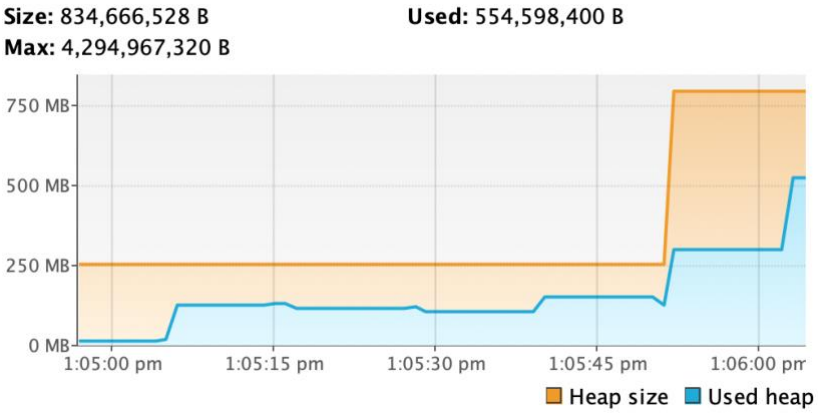
maxSize	Time Taken
1	11s 575ms
10	11s 448ms
1,000	11s 450ms
10,000	11s 474ms
100,000	11s 544ms
1,000,000	11s 562ms



MemAppender using Linked List and PatternLayout:

Total Time Taken:
1 Minute 9 seconds

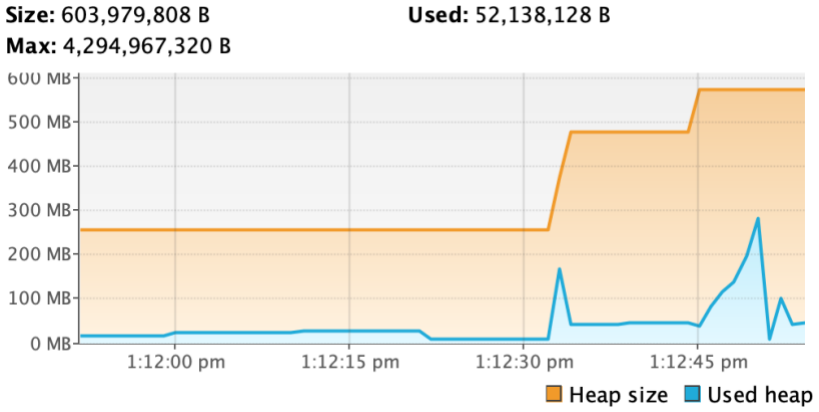
maxSize	Time Taken
1	11s 596ms
10	11s 456ms
1,000	11s 445ms
10,000	11s 477ms
100,000	11s 551ms
1,000,000	11s 560ms



FileAppender using VelocityLayout:

Total Time Taken:
1 minute 5 seconds

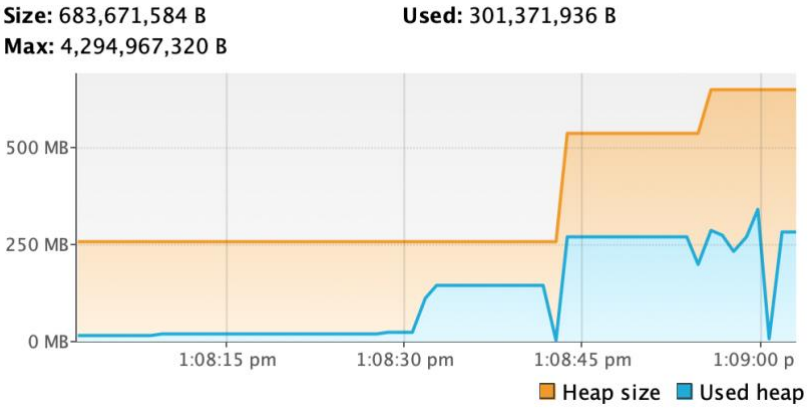
maxSize	Time Taken
1	11s 141ms
10	11s 30ms
1,000	11s 210ms
10,000	12s 8ms
100,000	19s 418ms



FileAppender using PatternLayout:

Total Time Taken:
1 minute 3 seconds

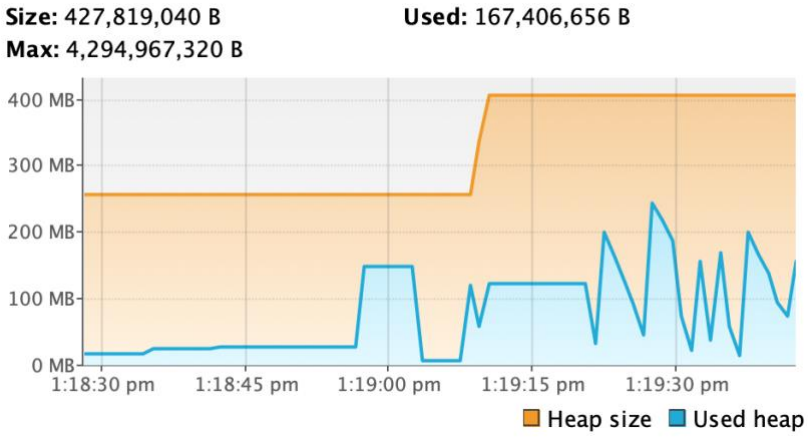
maxSize	Time Taken
1	11s 63ms
10	11s 39ms
1,000	11s 171ms
10,000	11s 823ms
100,000	18s 113ms



ConsoleAppender using VelocityLayout:

Total Time Taken:
1 minute 18 seconds

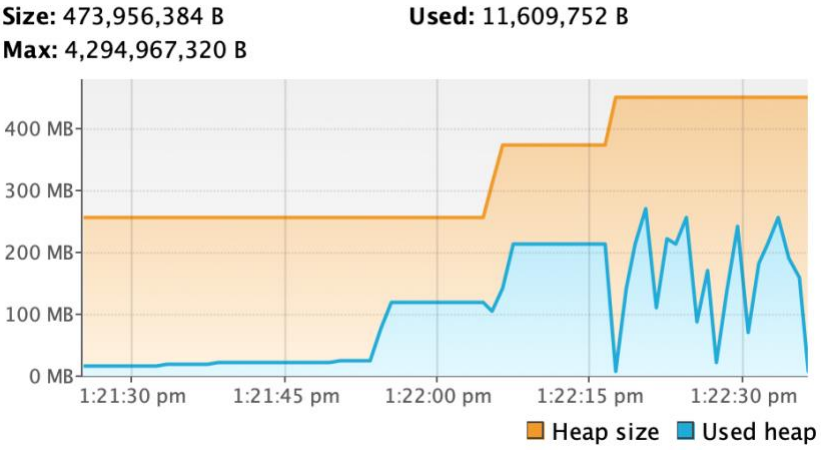
maxSize	Time Taken
1	11s 138ms
10	11s 39ms
1,000	11s 293ms
10,000	13s 128ms
100,000	31s 318ms



ConsoleAppender using PatternLayout:

Total Time Taken:
1 minute 12 seconds

maxSize	Time Taken
1	10s 59ms
10	10s 30ms
1,000	10s 261ms
10,000	12s 77ms
100,000	29s 645ms



Summary:

Each test was run as a parametrized test. The MemAppender related tests, took in a variable for maxSize from 1 to 1,000,000, and the number of logs being generated remained constant at 100,000. Hence the largest value of maxSize was greater than the number of logs being generated. The FileAppender and ConsoleAppender tests took in a variable for the number of logs to be generated.

Memory and Time Consumption:
During both MemAppender tests using an ArrayList, the max heap and used heap sizes increased when maxValue was at its largest. Comparatively, during both MemAppender tests using a LinkedList, we again saw that the max heap and used heap sizes increased when maxValue was at its largest. We can see that the MemAppender when used with a LinkedList used more memory at larger values of maxSize, compared with the MemAppender used with the ArrayList as would be expected. Each of the MemAppender tests took a total time of one minute and nine seconds.

During both File Appender tests, we can see that as the number of logs generated increased, so did the max heap and used heap sizes. During both Console Appender tests, when the number

of logs generated was greater than or equal to 1000, the used heap size and max heap size increased significantly. The total time taken varied for each of the tests based on the layout used.

For each of the File Appender and Console Appender tests, we can see that they don't use as much memory when compared with the MemAppender tests. The MemAppender used up to 500mb of heap space compared with 300mb maximum in the File Appender tests and 250mb maximum in the Console Appender tests.

Velocity Layout vs Pattern Layout:

Within the MemAppender tests, there isn't much variation with regards to time taken based on the layout. Each of the tests took 1 minute and 9 seconds. In the MemAppender test using an ArrayList, there was a slight peak towards the end in the used heap when using the Velocity Layout compared with the Pattern Layout, otherwise the graphs heap usage was very similar. In the MemAppender using a LinkedList, the graphs of the heap used was also very similar.

In the File Appender tests, the Velocity Layout tests look longer than the Pattern Layout Tests. This was most significant at the larger values of the generated logs. The file appender using the Velocity Layout also used overall less heap space than that of the Pattern Layout.

In the Console Appender tests, the Velocity Layout tests also took longer than the Patter Layout Tests. This was also most significant at the larger values of the generated logs. It is hard to tell from the used heap graphs which Layout used more memory as the trend is quite inconsistent as the number of logs generated increased.