DISTRIBUTED SYSTEMS – LAB 3 REPORT

Distributed Systems – Lab 3 Message Queues

Submitted: October 27, 2014
By: Devan Shah 100428864
Submitted to: Weina Ma

Lab Report Questions:

Task #1:

Use your system specific CPU monitoring tool. What is the maximal CPU utilization over all your CPUs? For Ubuntu Linux, you can install htop command to monitor multiple CPU usages.

A. The task 1 application was run using the following conditions: 50 repeat value, with an input file which had 6 lines (total message queue 50*6) and query word was hello with an edit distance threshold of 3.

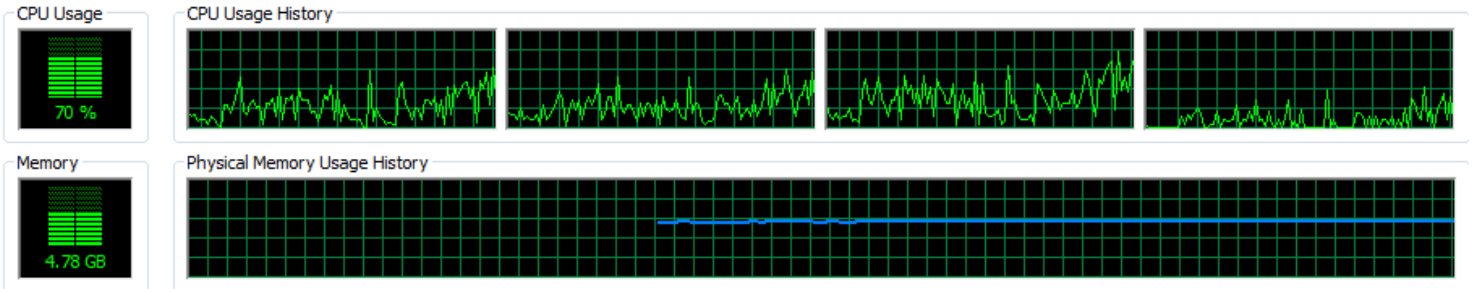

Figure 1 Shows the CPU utilization for using 50 repeat iterations with 6 lines in the inputfiles

If multiple consumers are created, what would be the potential problem if they all write to the same file without synchronization?

A. When there are multiple consumers created there would be a potential problem of the results not being places in the same order as it was read from the inputFile. My implementation is using BlockingQueue as the messaging queue therefore this should not be an issues because the BlockingQueue is built in a way that it is synchronized already. The take functions will be synced where it would block and wait for consume so that they are consumed in order of input.

Task #2:

Repeat the CPU utilization measurement for various numWorkers values (for example: 2, 4, 6, 8, 10)

A. Shows the utilization for numWorkers

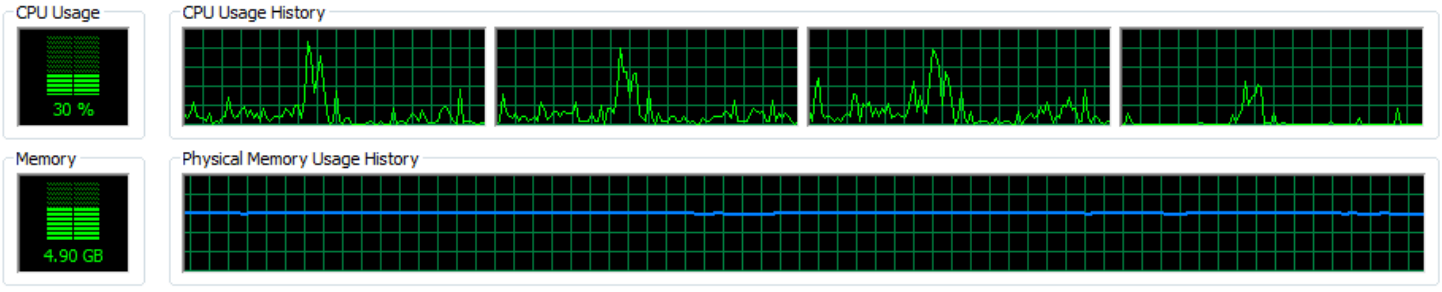

Figure 2 Shows the CPU utilization using 50 repeat iterations with 6 files in the inputfile and numWorkers of 2.

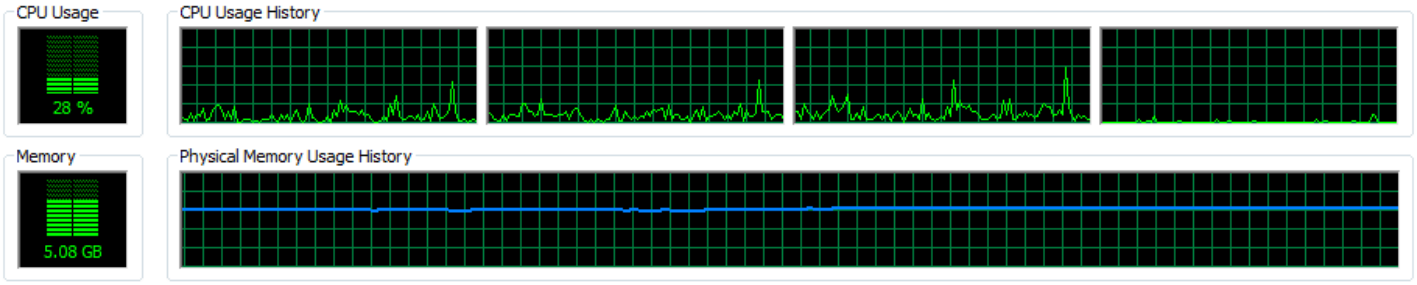

Figure 3 Shows the CPU utilization using 50 repeat iterations with 6 files in the inputfile and numWorkers of 4.

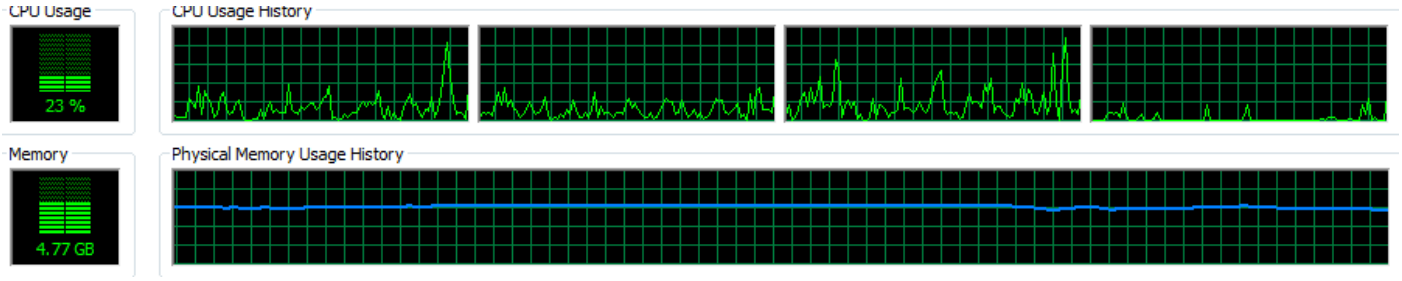

Figure 4 Shows the CPU utilization using 50 repeat iterations with 6 files in the inputfile and numWorkers of 6.

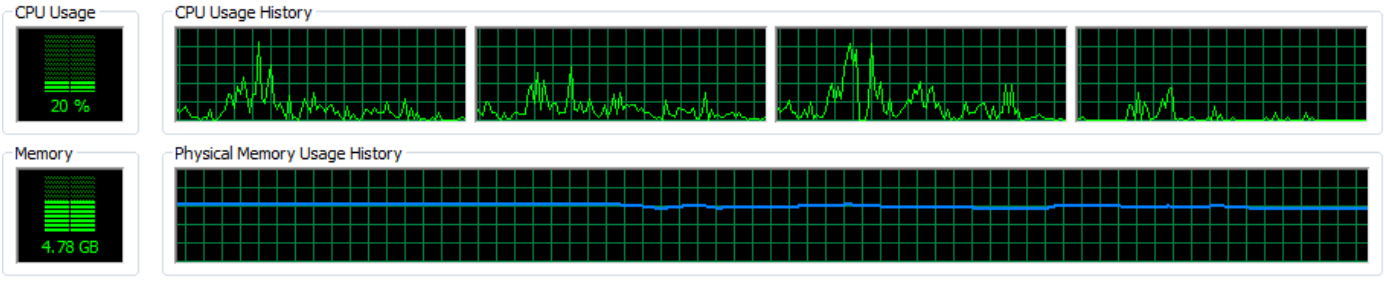

Figure 5 Shows the CPU utilization using 50 repeat iterations with 6 files in the inputfile and numWorkers of 8.

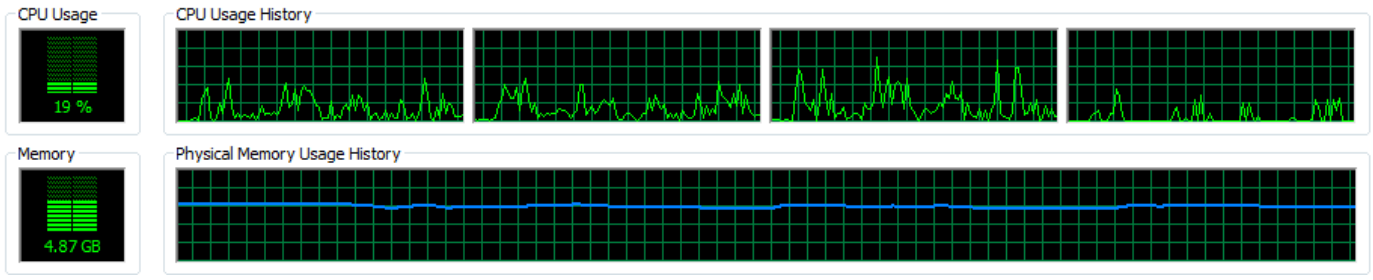

Figure 6 Shows the CPU utilization using 50 repeat iterations with 6 files in the inputfile and numWorkers of 10.

Measure the total processing time for various numWorkers values (for example: 2, 4, 6, 8, 10). What is the optimal setting for numWorkers? Explain

numWorkers	Total Processing Time
2	44 ms
4	41 ms
6	36 ms
8	33 ms
10	35 ms

This shows that the optimal setting for numWorkers is around 8 because the processing time increased after the 8th numWorker, whereas before this the processing time was decreasing. This is the case because the more threads take up more CPU therefore would cause it to be slower.

General Question:

How would you suggest to implement message queues that can connect **two** Java programs running on different machines, connected by the Internet?

A. I would suggest to use Java RMI client server communication to perform this, as that will allow to communicate over the internet. The Clients can be the LineConsumer, and the server would perform LineProcuder and also the ResultProcuder actions. This would have to be serialized to send over client and server for the purpose of security and also would require some sync on the client and server to make sure that there is no issues in updating the BlockingQueue from client server side.