Hakeem Jones
Nate Newsome
12/06/16
Extra Credit Extension

Introduction:
In our main project we implemented a simple publish/subscribe system. We tested this system over two different networks with seven subscription cases. On both network configurations we found average latency to increase linearly with the number of subscribers. We also saw a correlated increase of dropped subscribers with the number of subscriptions. In our main submission, our broker used a single linked list to store all of its connected subscribers.

Upon reviewing the results we hypothesized that the one list implementation was adding to our system latency. Our reasoning behind this was that when receiving a new message our broker has to look at every subscriber in the system to see should it be forwarded the incoming message. Thus in addition to network latency, our broker originally added a latency by a factor of O(n) where n is the total number of subscribers. To remove this latency we decided to make each topic maintain its list of subscribers. We believed this would reduce the broker delay on average by a factor of n/nT, where n is the number of subscribers and nT is the number of topic channels. With many topics we figured this would be much faster than the O(n) delay we had originally.

Implementation:
To implement this we changed the topic array structure in the message header struct to an integer representing the topic channel. Next we added a linked list for each topic in the broker. Upon receiving a new subscribe message our new code looks for the appropriate list to insert the socket. Now when receiving a new publish message our system examines the message header for the topic and sends to the subscribers on that topic list. All of our functions had to be updated to comply with this new structure.

Test and Results:
To test we used the same experimental set up as our main project. To understand how location of the clients affects the latency we run our simulation for two different test cases. In both cases the broker is run on a department linux machine (imp4) located on campus. In the first test case, clients are run on a linux virtual machine located off campus, and in the second they are running on another department linux machine (imp5) located on campus. For each we gather the overall average latency of a variable number of subscribers (ranging from 10 to 1000).

Table 1 shows the results for seven tests under the vm to department linux machine condition. Table 2 describes the same tests for a department machine to department machine condition.

Num. Subscribers	10	50	100	250	500	750	1000
Average Latency (µs)	16558.8	19227.9	18957.71	29337.176	70856.194	105561.788	135536.6412
Dropped Subs.	0	0	0	0	0	0	150
Standard Deviation	779.0971698	782.7540665	1270.478915	6064.335196	14353.85403	21333.45408	26507.04269

Table 1. VM to imp4

Num. Subscribers	10	50	100	250	500	750	1000	10000
Average Latency(µs)	786	843.32	9991.47	36815.612	70309.902	100300.1813	137854.281	1369857.456
Dropped Subs.	0	0	0	0	0	0	0	0
Standard Deviation	9.626352719	90.70684784	9282.607802	11332.26664	15195.01342	20611.96548	28827.73649	270454.4989

Table 2. Imp4 to imp5

Both tables 1 and 2 show a similar linear trajectory with the latency increasing with the number of subscribers. This follows the same trend we seen in the main project (see figures 1-4).

Number of Subs.	Average Latency (Microseconds)	Dropped Subs.
10	18781.2	0
50	29093.6	0
100	30335.5	0
250	33824.2	1
500	79533.7	1
750	101864.0	17
1000	152711.1	199

Figure 1.

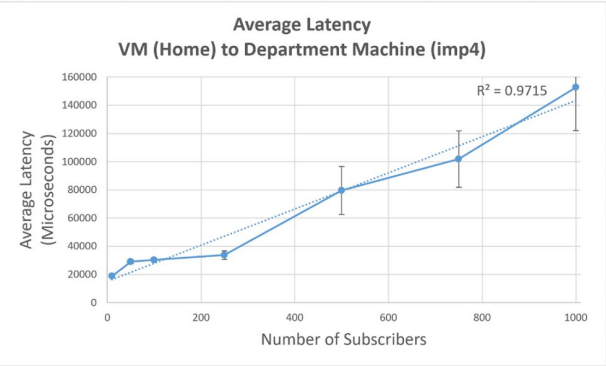

Figure 2.

Number of Subs	Average Latency (Microseconds)	Dropped Subs.
10	765	0
50	832.66	0
100	8978.44	0
250	34848.4	0
500	64265.8	0
750	100843	0
1000	128913	0

Figure3.

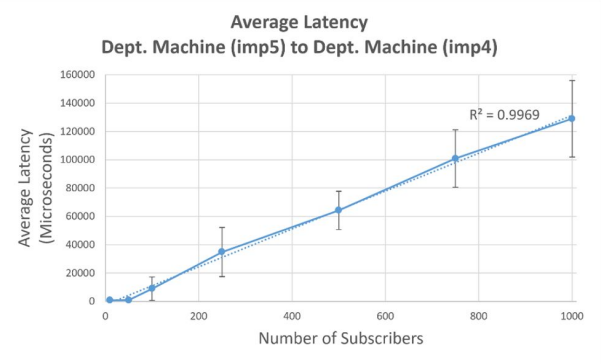

Figure 4.

The results show the latency for the vm to department machine configuration to have diminished at all but one level of subscribers. On the department machine to department machine tests the latencies actually increased (very slightly) for all levels of subscribers. An unexpected result we noticed was a significant reduction in the number of dropped subscriptions in the new system (50 less on the vm to imp4 test with 1000 subscribers). Also, though we had no measure, the time for a client to subscribe was greatly reduced with the updated implementation. This was immediately noticeable with over 500 subscribers. We believe the inconsistent reduction in latency across testing is due to the fact that network delays are a significant and uncontrollable contributor to the latency we measured.