CSCI/ECEN 5673: Distributed Systems
Spring 2026

Programming Assignment One

Team: Anirudh Ragam, Ishita Badole

Github: https://github.com/anirudhragam/distributed-systems-programming-assignments

Experiments Setup

We created a performance_tests.py file to run our experiments

We define the run_buyer_operations function to run the 1000 buyer operations and the run_seller_operations to run 1000 buyer operations. These functions record a list of response times for each operation and return the list of 1000 response times to the caller.

The run_scenario function creates one thread for each seller and buyer (1, 10, or 100) in the experiment. Each thread calls its corresponding run operations function to run 1000 operations. The results of the threads are collected as the threads complete. These results are the list of response times for the 1000 operations. The run_scenario function creates a list of all seller response times and a list of all buyer response times.

The compute_metrics function takes a list of response times as input and computes the average response time and the throughput. Once all the threads have completed, run_scenario function calls computer_metrics for seller response times and buyer response times to get the average response time and throughput for one run for all sellers and buyers for a scenario

The run_experiments function runs the experiment for 10 iterations. Finally it computes the overall average of the average response times and throughputs for buyers and sellers for the 10 iterations and reports them as the final results of the experiment.

run_experiment is called for 1, 10 and 100 buyers and sellers.

Experiment Results

Scenario 1: Run 1 seller and 1 buyer.

	Sellers		Buyers	
	Average Response Time	Throughput (ops/sec)	Average Response Time	Throughput (ops/sec)

	(ms)		(ms)	
Iteration 1	0.97 ms	1032.53 ops/sec	1.11 ms	902.23 ops/sec
Iteration 2	0.82 ms	1215.72 ops/sec	1.06 ms	940.44 ops/sec
Iteration 3	0.81 ms	1231.44 ops/sec	1.17 ms	855.90 ops/sec
Iteration 4	0.86 ms	1165.14 ops/sec	1.29 ms	772.89 ops/sec
Iteration 5	0.88 ms	1133.03 ops/sec	1.43 ms	697.98 ops/sec
Iteration 6	0.86 ms	1161.74 ops/sec	1.50 ms	666.97 ops/sec
Iteration 7	1.05 ms	949.85 ops/sec	1.82 ms	549.70 ops/sec
Iteration 8	0.95 ms	1049.68 ops/sec	1.91 ms	523.90 ops/sec
Iteration 9	0.94 ms	1060.73 ops/sec	2.06 ms	485.47 ops/sec
Iteration 10	0.96 ms	1037.44 ops/sec	2.12 ms	472.32 ops/sec
Average	0.91 ms	103.73 ops/sec	1.55 ms	686.78 ops/sec

Scenario 2: Run 10 concurrent sellers and 10 concurrent buyers.

	Sellers		Buyers	
	Average Response Time (ms)	Throughput (ops/sec)	Average Response Time (ms)	Throughput (ops/sec)
Iteration 1	4.09 ms	244.77 ops/sec	19.41 ms	51.51 ops/sec
Iteration 2	7.62 ms	131.16 ops/sec	34.98 ms	28.59 ops/sec
Iteration 3	4.89 ms	204.37 ops/sec	39.69 ms	25.20 ops/sec
Iteration 4	9.02 ms	110.81 ops/sec	49.61 ms	20.16 ops/sec
Iteration 5	4.64 ms	215.70 ops/sec	50.47 ms	19.81 ops/sec
Iteration 6	5.90 ms	169.59 ops/sec	62.68 ms	15.96 ops/sec
Iteration 7	4.97 ms	201.40 ops/sec	72.65 ms	13.76 ops/sec
Iteration 8	4.86 ms	205.79 ops/sec	76.11 ms	13.14 ops/sec

Iteration 9	4.89 ms	204.60 ops/sec	83.95 ms	11.91 ops/sec
Iteration 10	4.70 ms	212.94 ops/sec	89.04 ms	11.23 ops/sec
Average	5.56 ms	190.11 ops/sec	57.86 ms	21.13 ops/sec

Scenario 3: Run 100 concurrent sellers and 100 concurrent buyers.

	Sellers		Buyers	
	Average Response Time (ms)	Throughput (ops/sec)	Average Response Time (ms)	Throughput (ops/sec)
Iteration 1	136.24 ms,	7.34 ops/sec	2049.25 ms	0.49 ops/sec
Iteration 2	133.65 ms	7.48 ops/sec	2946.54 ms,	0.34 ops/sec
Iteration 3	195.17 ms,	5.12 ops/sec	3408.77 ms,	0.29 ops/sec
Iteration 4	135.66 ms,	7.37 ops/sec	4536.50 ms,	0.22 ops/sec
Iteration 5	160.82 ms,	6.22 ops/sec	4714.69 ms,	0.21 ops/sec
Iteration 6	147.43 ms	6.78 ops/sec	5085.47 ms,	0.20 ops/sec
Iteration 7	155.24 ms	6.44 ops/sec	6263.55 ms,	0.16 ops/sec
Iteration 8	151.49 ms	6.60 ops/sec	8838.51 ms,	0.11 ops/sec
Iteration 9	DB Pool Exhausted	DB Pool Exhausted	DB Pool Exhausted	DB Pool Exhausted
Iteration 10	DB Pool Exhausted	DB Pool Exhausted	DB Pool Exhausted	DB Pool Exhausted
Average	151.96 ms	6.67 ops/sec	4730.41 ms	0.25 ops/sec

Results Explanation

Scenario 1 with one current buyer and seller has the lowest average response time, followed by Scenario 2 and Scenario 3 has the highest average response time.

The opposite trend is observed for throughput. Scenario 1 has the highest throughput, followed by Scenario 2, and Scenario 3 has the lowest throughput.

The explanation for these observations is that increasing the number of clients increases the load on the server. As concurrency rises, the server has to handle more concurrent requests, leading to higher simultaneous need for database connections and operations. So the server takes more time to complete per operation and the throughput decreases.

Every operation first checks that the session is valid. When there are more concurrent sellers and buyers, there are more database checks for session validation and that leads to a lot of database overhead. This explains the increase in the average response times for scenarios. One way to optimize this would be to keep the session table in memory for faster lookup.

For each scenario, average response time remains relatively consistent across runs but the throughput value decreases with each iteration. This could be because the size of the database tables increases over time. The performance test is designed in a way where items are continuously added to the database, so the database operations take longer leading to lower throughput. But each request does relatively the same steps, so the response time variance remains low.