Sequential Request Experiments:

End-to-end Response times

End to End Response Times , 1000 Sequential Requests			
	Buy	Lookup	Search
Item 1	0.036129	0.013081	
Item 2	0.033040	0.013377	
Item 3	0.034209	0.013217	
Item 4	0.034005	0.012911	
Topic 1			0.012885
Topic 2			0.012939

The above output is from running ./SequentialRequestExperiments.sh in the tests folder. Note that the initial stock for each item was varied, so that we could see if the average response time would change with more failed buy requests. Furthermore, these scripts regularly clear the /orders/ list so that there is not increasing overhead from managing a large order databse.

Observations - the average response time for buy requests is almost 3x higher. This could be attributed to the fact that buy requests go through more tiers (Frontend - Order - Catalog - Order - Frontend), whereas lookup are search go through less (Frontend - Catalog - Frontend). Average response time for lookup and search are extremely close, which makes sense since both methods are just routed from frontend to catalog directly.

Per-tier Response Times

	Order - Catalog Query	Frontend - Order Buy	Frontend - Catalog Query
buy item 1	0.008143213272	0.01327	
lookup item 1			0.01110
search topic 1			0.01114

The above output is from running ./PerTierExperiments.sh int the tests folder. Note we only experimented on 1 item or topic, since the previous experiment seemed to indicate that performance does not vary across items. The average Order-Catalog tier response time is saved to experiment_results/buy_query_times, and the average Frontend - Order tier response time is computed by subtracting the previous component from the end-to-end response time.

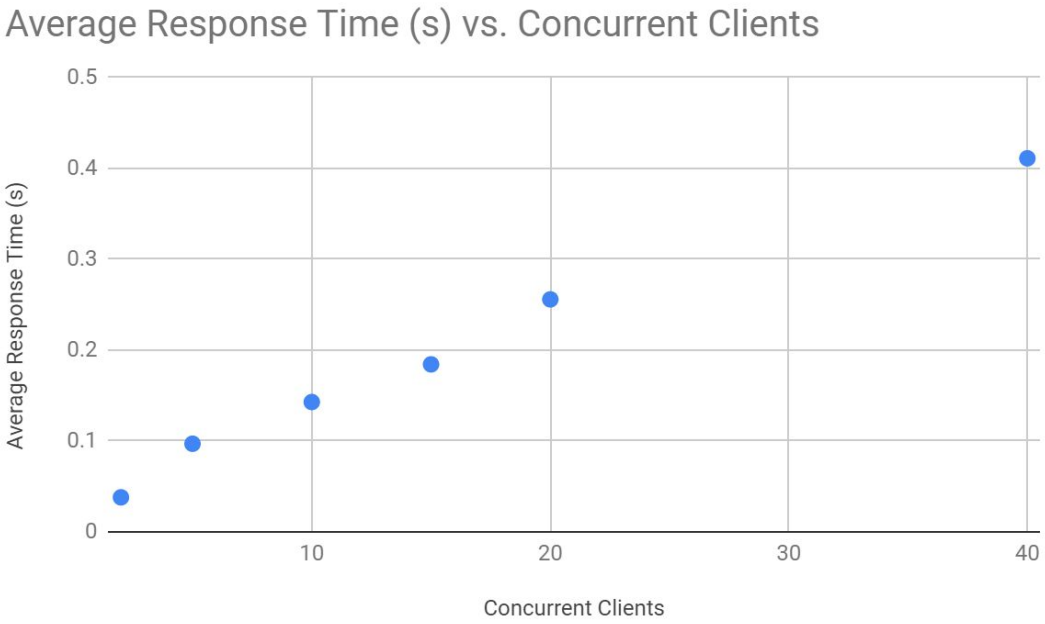
Furthermore, we set the stock to 0 so that update operations are never called from the order server. This is to isolate the response time of query operations.

Observations:

The the single-tier response time for Frontend-Catalog query is essentially the same as the end-to-end response time, since the end-to-end response time goes through the exact same tier. The Frontend-Order response time is close to the Frontend-Catalog response times, while the Order-Catalog response time is slightly lower. These results seem to indicate that the time spent in each tier is close to 0.01 seconds.

Concurrent Request Experiments:
This experiment can be replicated by going into /tests/concurrency_experiments and running all scripts in the folder.

The setup for this experiment is to deploy the system on 3 machines, then run shell scripts from a 4th machine that concurrently spawn multiple clients making continuous requests. All clients made 100 requests each.



Concurrent Clients	Average Response Time (s)
2	0.03797
5	0.09694
10	0.14286
15	0.18429
20	0.25583
40	0.41103

Observations: the end-to-end response time seems to increase linearly with the number of concurrent clients.