Evaluation Document

Testing and Performance Evaluation

We have increased the number of clients from 1 to 5 and measure the latency as the load goes up. Made plots showing number of clients on the X-axis and response time/latency on the Y-axis. We sent 1000 requests from clients concurrently with docker and without docker and noted the latency in response. The latency varies significantly for trade and lookup requests when we checked for client applications concurrently.
The graphs are plotted below:

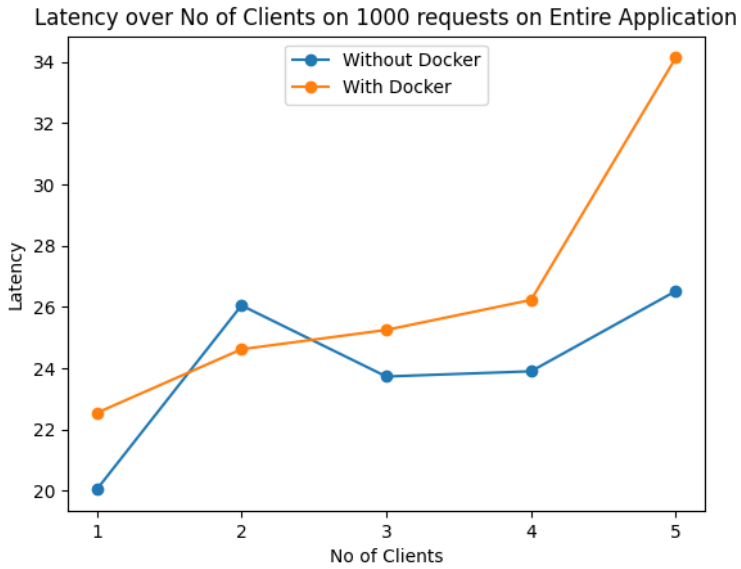

Figure 1:Latency over No of clients for **1000** requests with and without docker(In seconds)

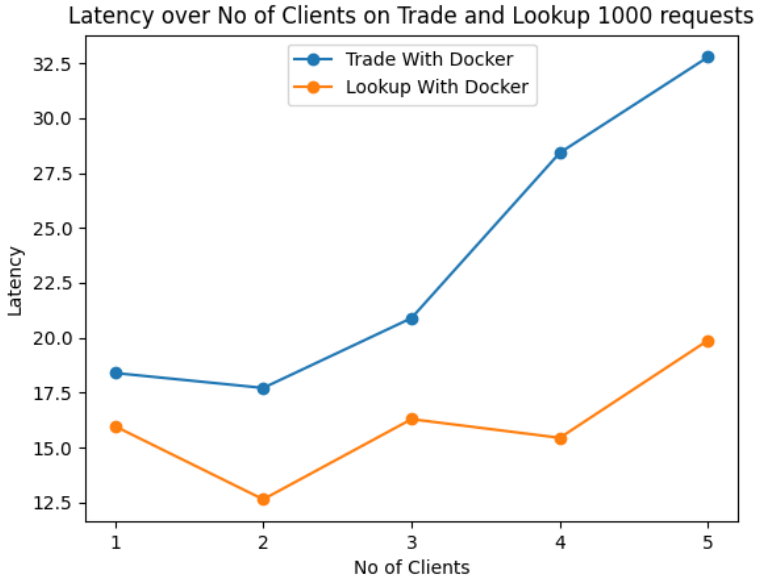

Figure 2: Latency over No of clients for **1000** requests Trade V/s Lookup without docker(In seconds)

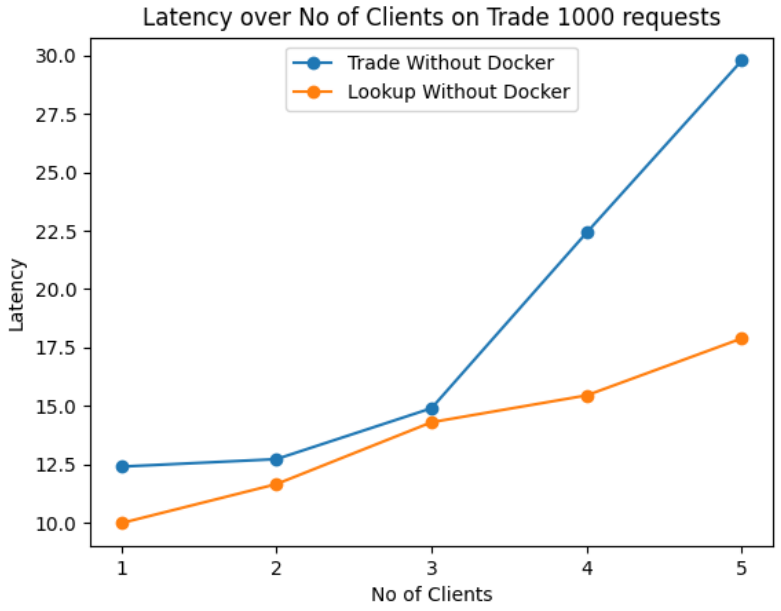

Figure 3: Latency over No of clients for **1000** requests with Docker.(In Seconds)

1. Does the latency of the application change with and without Docker containers? Did virtualization add any overheads?
   Answer:

   Yes, the virtualization overhead results in increased latency for the application running in a Docker container. In Figure 1, we can see that the Latency for the client application on docker is higher than the client application running on a different machine. At client =1 , the response time for docker application is 22 seconds for 1000 requests whereas for application on local machine it is 19.5 Seconds. It has an intersection with the latency plot of w/o docker line at client number=2, this is because of the sudden latency surge in application running on machine when client increased to 2. As we further increase the number of clients, at client=4, there is a spike in response time for docker application compared to one running on local machine.

2. How does the latency of the lookup requests compare to trade? Since trade requests involve all these microservices, while lookup requests only involve two microservices, does it impact the observed latency?

   Answer:

   Yes, The latency of lookup requests are less compared to trade requests. This is due to multiple reasons. The most important one, trade request has to pass through 3 microservices whereas lookup is only going through frontend and catalog services. Lookup is having a single GET request to frontend and catalog and its response but, in the

case of trade, there is a GET lookup request from frontend to order service and from order to catalog and catalogs response back to order service and then modifying the quantity for buying or selling and response back to client. Also order service is parallelly sending the POST request again to the catalog server for updation of catalog log in the trade. These request overhead of lookup inside the trade request causes the trade request to be slower.

The second reason is because we are using the ReadWrite Locks for reading and writing to the logs. When multiple clients are accessing the catalog database, the read lock will synchronize the lookup requests and this results in a delay in response for client. Also the trade request is updating the order log with transaction details. When concurrent requests requests come from multiple clients, this will slow down the system.

3. How does the latency change as the number of clients change? Does it change for different types of requests?

Answer:

Yes, the latency increases as number of clients increases. For lookup and trade requests in both cases, docker and without docker, the latency increases. Because when we increase the number of clients, the application experience higher latency due to resource contention and increased load on the system. Since trade requests involve all 3 microservices, while lookup requests only involve two microservices, lookups has a faster response than trade request. The latency of trade requests increases more significantly than the latency of lookup requests. This might be due to the multiple number of readwrite locks each trade request encountering while reading and writing the Order log and catalog database. When concurrent requests come from multiple clients, this will slow down the trade requests more.