A2 Micro-Service Report

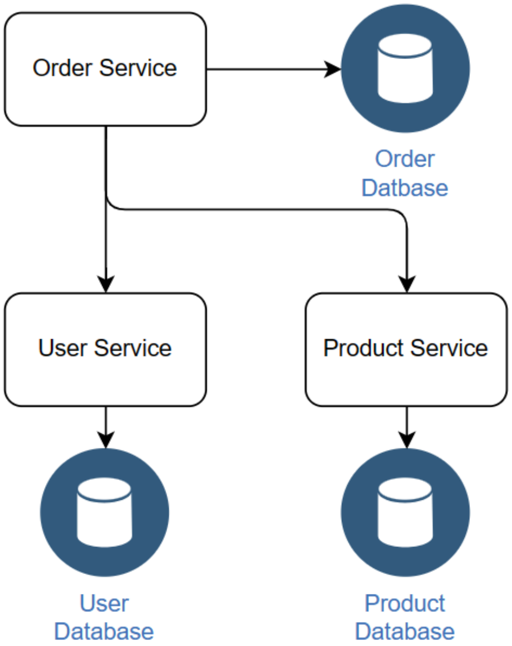
Workload

Non-optimized and optimized systems were tested with the sim_highload.load file located in src/loadgen/. This file contains the following.
sim_highload.load

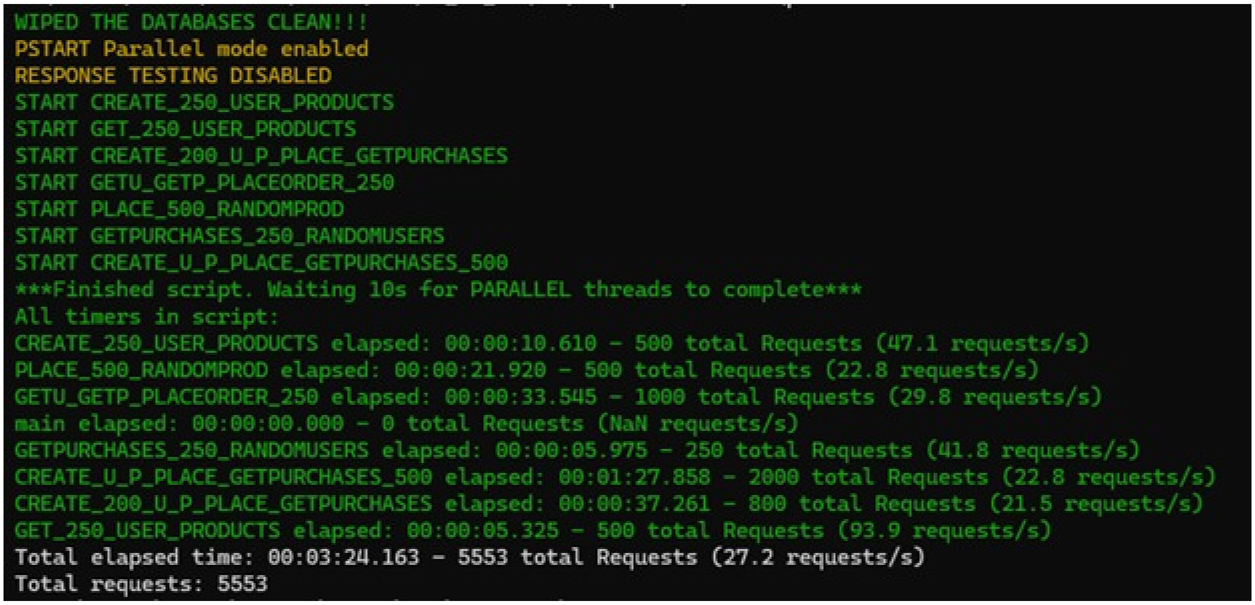
- Creating x Users and Products.
- Getting x Users and Products.
- Creating x Users and Products, waiting then placing x Orders, waiting then getting Purchased for those Users.
- Getting x User and Products, waiting then placing x Orders, waiting then getting Purchased for those Users.
- Placing x random Orders.
- Getting x Purchased for random Users.
- Creating x Users, Products, Orders, and then getting Purchased for that User.

Baseline

Architecture



Performance



```
WIPED THE DATABASES CLEAN!!!
PSTART Parallel mode enabled
RESPONSE TESTING DISABLED
START CREATE_250_USER_PRODUCTS
START GET_250_USER_PRODUCTS
START CREATE_200_U_P_PLACE_GETPURCHASES
START GETU_GETP_PLACEORDER_250
START PLACE_500_RANDOMPROD
START GETPURCHASES_250_RANDOMUSERS
START CREATE_U_P_PLACE_GETPURCHASES_500
***Finished script. Waiting 10s for PARALLEL threads to complete***
All timers in script:
CREATE_250_USER_PRODUCTS elapsed: 00:00:10.610 - 500 total Requests (47.1 requests/s)
PLACE_500_RANDOMPROD elapsed: 00:00:21.920 - 500 total Requests (22.8 requests/s)
GETU_GETP_PLACEORDER_250 elapsed: 00:00:33.545 - 1000 total Requests (29.8 requests/s)
main elapsed: 00:00:00.000 - 0 total Requests (NaN requests/s)
GETPURCHASES_250_RANDOMUSERS elapsed: 00:00:05.975 - 250 total Requests (41.8 requests/s)
CREATE_U_P_PLACE_GETPURCHASES_500 elapsed: 00:01:27.858 - 2000 total Requests (22.8 requests/s)
CREATE_200_U_P_PLACE_GETPURCHASES elapsed: 00:00:37.261 - 800 total Requests (21.5 requests/s)
GET_250_USER_PRODUCTS elapsed: 00:00:05.325 - 500 total Requests (93.9 requests/s)
Total elapsed time: 00:03:24.163 - 5553 total Requests (27.2 requests/s)
Total requests: 5553
```

Performance for the baseline system was slow overall. It completed **5553 requests** in **3.24 minutes** which is about **27.2 requests/s**.

Optimizations

Virtual Threads, Lifetime HttpClient, and Logging

Before:

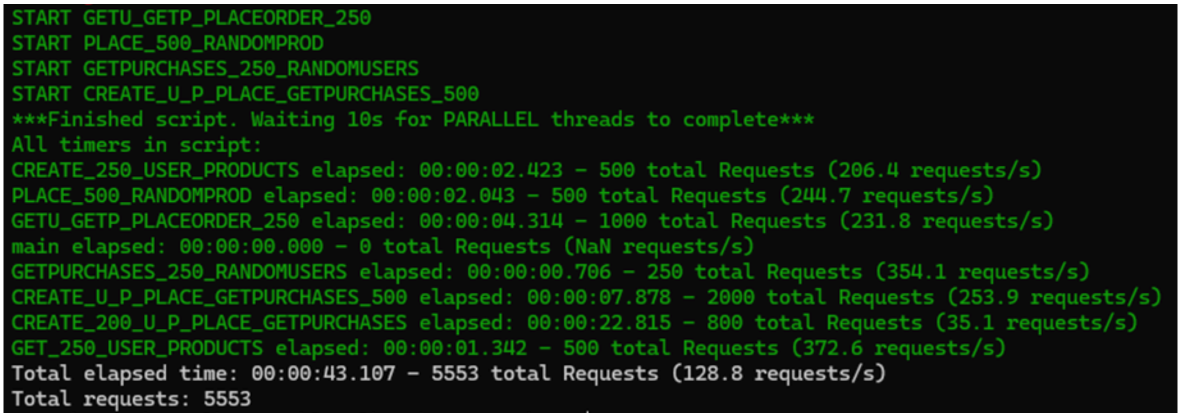
- Outbound Http client was created and terminated with each call.
- System threads were used which had a lot of overhead and was restricted in number on lab machines.
- Logging printed everything on the screen reducing performance

After:

- Outbound Http client is created on service bootup and lasts for the lifetime of the services.
- Virtual threads in Java 21+ have less overhead and are not limited on lab machines.
- Logging was reduced to increase performance.

Results:

- Total Requests: **5553**
- Avg: **128.8 requests/s**



```
START GETU_GETP_PLACEORDER_250
START PLACE_500_RANDOMPROD
START GETPURCHASES_250_RANDOMUSERS
START CREATE_U_P_PLACE_GETPURCHASES_500
***Finished script. Waiting 10s for PARALLEL threads to complete***
All timers in script:
CREATE_250_USER_PRODUCTS elapsed: 00:00:02.423 - 500 total Requests (206.4 requests/s)
PLACE_500_RANDOMPROD elapsed: 00:00:02.043 - 500 total Requests (244.7 requests/s)
GETU_GETP_PLACEORDER_250 elapsed: 00:00:04.314 - 1000 total Requests (231.8 requests/s)
main elapsed: 00:00:00.000 - 0 total Requests (NaN requests/s)
GETPURCHASES_250_RANDOMUSERS elapsed: 00:00:00.706 - 250 total Requests (354.1 requests/s)
CREATE_U_P_PLACE_GETPURCHASES_500 elapsed: 00:00:07.878 - 2000 total Requests (253.9 requests/s)
CREATE_200_U_P_PLACE_GETPURCHASES elapsed: 00:00:22.815 - 800 total Requests (35.1 requests/s)
GET_250_USER_PRODUCTS elapsed: 00:00:01.342 - 500 total Requests (372.6 requests/s)
Total elapsed time: 00:00:43.107 - 5553 total Requests (128.8 requests/s)
Total requests: 5553
```

Connection Pooling

Before:

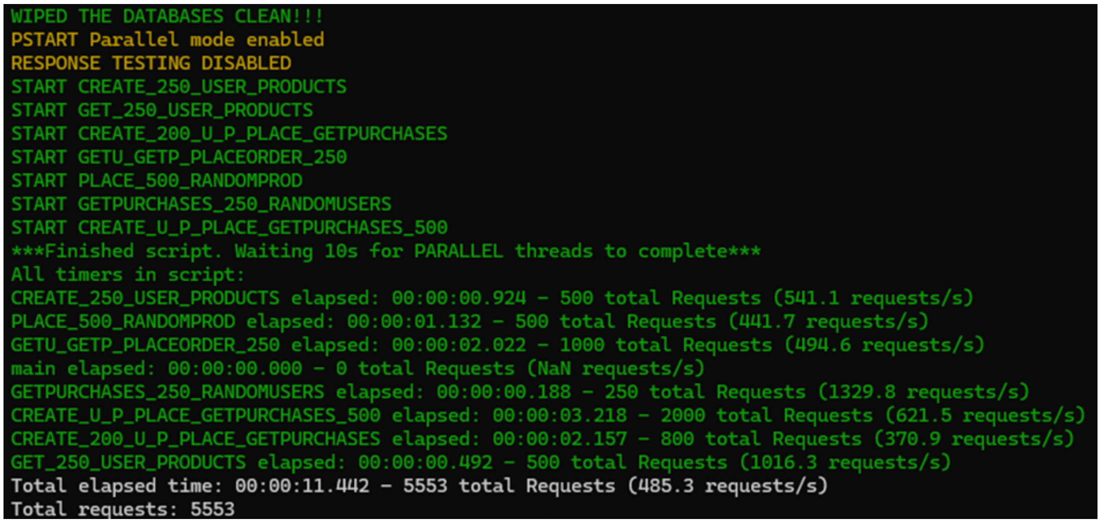
- Connections to the database were created and destroyed each database call.

After:

- Creates new connections and pools them for re-use.

Results:

- Total Requests: **5553**
- Avg: **485.8 requests/s**



```
WIPED THE DATABASES CLEAN!!!
PSTART Parallel mode enabled
RESPONSE TESTING DISABLED
START CREATE_250_USER_PRODUCTS
START GET_250_USER_PRODUCTS
START CREATE_200_U_P_PLACE_GETPURCHASES
START GETU_GETP_PLACEORDER_250
START PLACE_500_RANDOMPROD
START GETPURCHASES_250_RANDOMUSERS
START CREATE_U_P_PLACE_GETPURCHASES_500
***Finished script. Waiting 10s for PARALLEL threads to complete***
All timers in script:
CREATE_250_USER_PRODUCTS elapsed: 00:00:00.924 - 500 total Requests (541.1 requests/s)
PLACE_500_RANDOMPROD elapsed: 00:00:01.132 - 500 total Requests (441.7 requests/s)
GETU_GETP_PLACEORDER_250 elapsed: 00:00:02.022 - 1000 total Requests (494.6 requests/s)
main elapsed: 00:00:00.000 - 0 total Requests (NaN requests/s)
GETPURCHASES_250_RANDOMUSERS elapsed: 00:00:00.188 - 250 total Requests (1329.8 requests/s)
CREATE_U_P_PLACE_GETPURCHASES_500 elapsed: 00:00:03.218 - 2000 total Requests (621.5 requests/s)
CREATE_200_U_P_PLACE_GETPURCHASES elapsed: 00:00:02.157 - 800 total Requests (370.9 requests/s)
GET_250_USER_PRODUCTS elapsed: 00:00:00.492 - 500 total Requests (1016.3 requests/s)
Total elapsed time: 00:00:11.442 - 5553 total Requests (485.3 requests/s)
Total requests: 5553
```

Parallel Requests Increase

Before:

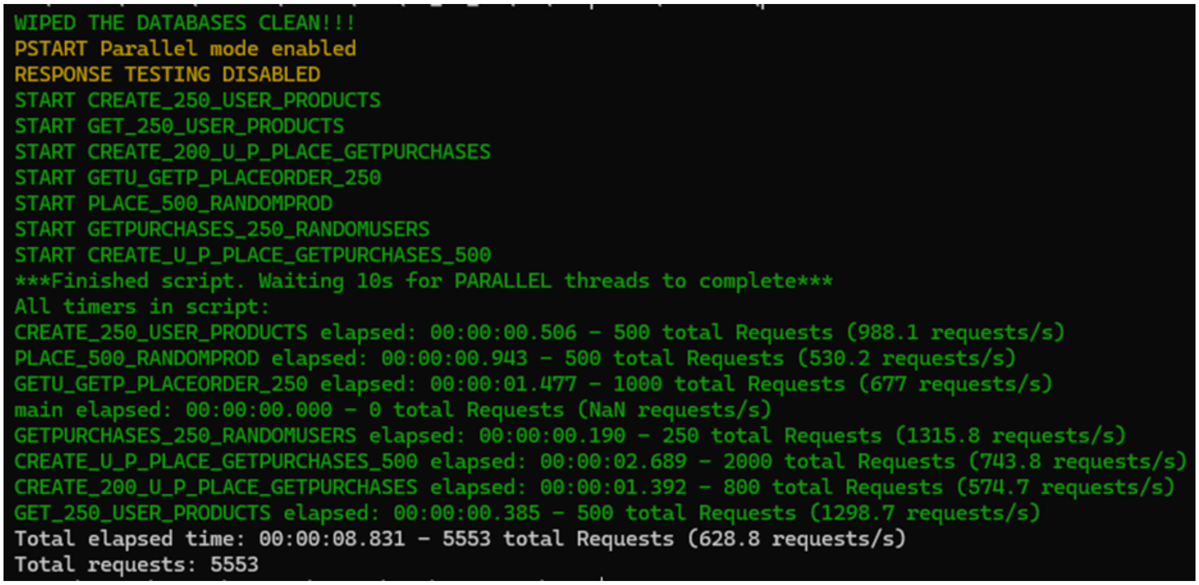
- With previous optimizations, the services were now under-utilized.

After:

- Due to previous optimizations we can send more parallel requests increasing throughput

Results:

- Total Requests: **5553**
- Avg: **628.8 requests/s**



```
WIPED THE DATABASES CLEAN!!!
PSTART Parallel mode enabled
RESPONSE TESTING DISABLED
START CREATE_250_USER_PRODUCTS
START GET_250_USER_PRODUCTS
START CREATE_200_U_P_PLACE_GETPURCHASES
START GETU_GETP_PLACEORDER_250
START PLACE_500_RANDOMPROD
START GETPURCHASES_250_RANDOMUSERS
START CREATE_U_P_PLACE_GETPURCHASES_500
***Finished script. Waiting 10s for PARALLEL threads to complete***
All timers in script:
CREATE_250_USER_PRODUCTS elapsed: 00:00:00.506 - 500 total Requests (988.1 requests/s)
PLACE_500_RANDOMPROD elapsed: 00:00:00.943 - 500 total Requests (530.2 requests/s)
GETU_GETP_PLACEORDER_250 elapsed: 00:00:01.477 - 1000 total Requests (677 requests/s)
main elapsed: 00:00:00.000 - 0 total Requests (NaN requests/s)
GETPURCHASES_250_RANDOMUSERS elapsed: 00:00:00.190 - 250 total Requests (1315.8 requests/s)
CREATE_U_P_PLACE_GETPURCHASES_500 elapsed: 00:00:02.689 - 2000 total Requests (743.8 requests/s)
CREATE_200_U_P_PLACE_GETPURCHASES elapsed: 00:00:01.392 - 800 total Requests (574.7 requests/s)
GET_250_USER_PRODUCTS elapsed: 00:00:00.385 - 500 total Requests (1298.7 requests/s)
Total elapsed time: 00:00:08.831 - 5553 total Requests (628.8 requests/s)
Total requests: 5553
```

Caching (distributed when used in a cluster)

Before:

- No caching was available, all requests required database calls.

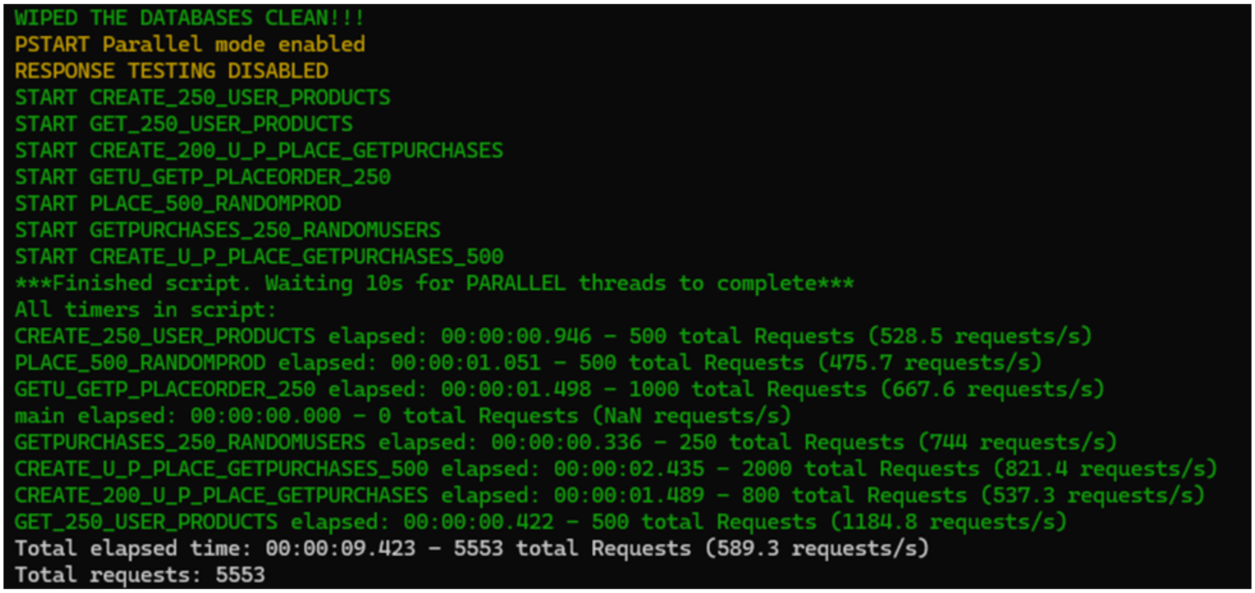
After:

- Caching was enabled across the services using Hazelcast.

Results:

- Total Requests: **5553**
- Avg: **589.3 requests/s**

NOTE: (Distributed) caching doesn't show its real benefits until you start distributing and/or make even more requests, right now there are too few requests to really highlight performance. And services are under-utilized.



```
WIPED THE DATABASES CLEAN!!!
PSTART Parallel mode enabled
RESPONSE TESTING DISABLED
START CREATE_250_USER_PRODUCTS
START GET_250_USER_PRODUCTS
START CREATE_200_U_P_PLACE_GETPURCHASES
START GETU_GETP_PLACEORDER_250
START PLACE_500_RANDOMPROD
START GETPURCHASES_250_RANDOMUSERS
START CREATE_U_P_PLACE_GETPURCHASES_500
***Finished script. Waiting 10s for PARALLEL threads to complete***
All timers in script:
CREATE_250_USER_PRODUCTS elapsed: 00:00:00.946 - 500 total Requests (528.5 requests/s)
PLACE_500_RANDOMPROD elapsed: 00:00:01.051 - 500 total Requests (475.7 requests/s)
GETU_GETP_PLACEORDER_250 elapsed: 00:00:01.498 - 1000 total Requests (667.6 requests/s)
main elapsed: 00:00:00.000 - 0 total Requests (NaN requests/s)
GETPURCHASES_250_RANDOMUSERS elapsed: 00:00:00.336 - 250 total Requests (744 requests/s)
CREATE_U_P_PLACE_GETPURCHASES_500 elapsed: 00:00:02.435 - 2000 total Requests (821.4 requests/s)
CREATE_200_U_P_PLACE_GETPURCHASES elapsed: 00:00:01.489 - 800 total Requests (537.3 requests/s)
GET_250_USER_PRODUCTS elapsed: 00:00:00.422 - 500 total Requests (1184.8 requests/s)
Total elapsed time: 00:00:09.423 - 5553 total Requests (589.3 requests/s)
Total requests: 5553
```

Parallel Requests Increase 2

Before:

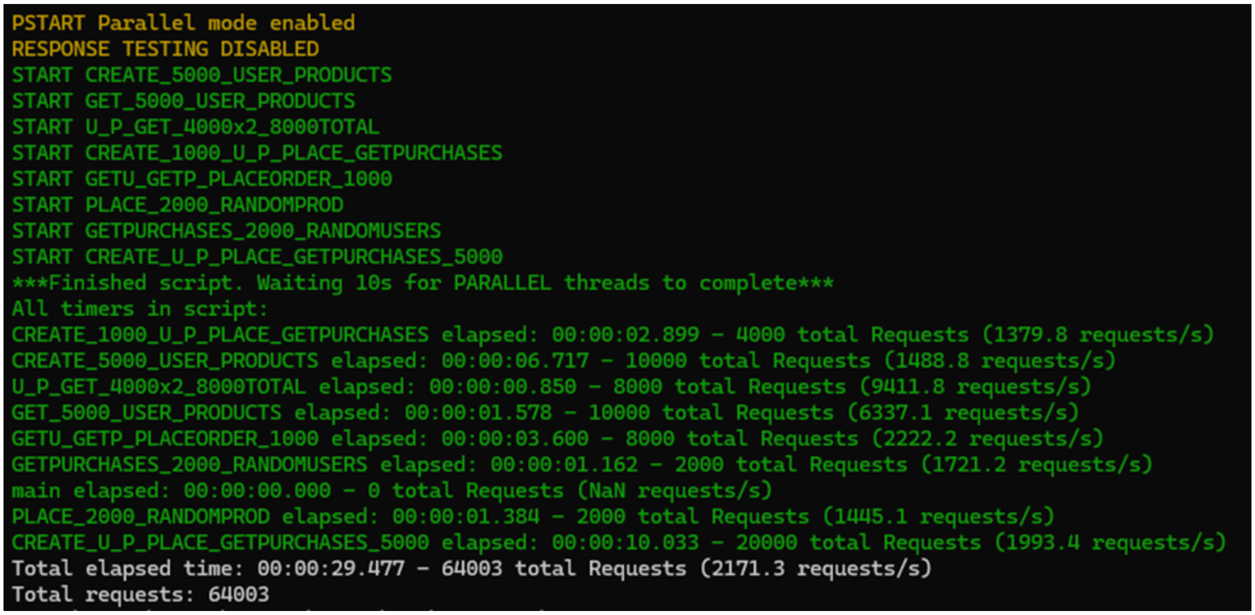
- Previous optimizations are now underutilized.

After:

- Due to caching and other optimizations, we can increase parallel requests further and total number of requests to really make use of caching.

Results:

- Total Requests: **64003**
- Avg: **2171.3 requests/s**



```
PSTART Parallel mode enabled
RESPONSE TESTING DISABLED
START CREATE_5000_USER_PRODUCTS
START GET_5000_USER_PRODUCTS
START U_P_GET_4000x2_8000TOTAL
START CREATE_1000_U_P_PLACE_GETPURCHASES
START GETU_GETP_PLACEORDER_1000
START PLACE_2000_RANDOMPROD
START GETPURCHASES_2000_RANDOMUSERS
START CREATE_U_P_PLACE_GETPURCHASES_5000
***Finished script. Waiting 10s for PARALLEL threads to complete***
All timers in script:
CREATE_1000_U_P_PLACE_GETPURCHASES elapsed: 00:00:02.899 - 4000 total Requests (1379.8 requests/s)
CREATE_5000_USER_PRODUCTS elapsed: 00:00:06.717 - 10000 total Requests (1488.8 requests/s)
U_P_GET_4000x2_8000TOTAL elapsed: 00:00:00.850 - 8000 total Requests (9411.8 requests/s)
GET_5000_USER_PRODUCTS elapsed: 00:00:01.578 - 10000 total Requests (6337.1 requests/s)
GETU_GETP_PLACEORDER_1000 elapsed: 00:00:03.600 - 8000 total Requests (2222.2 requests/s)
GETPURCHASES_2000_RANDOMUSERS elapsed: 00:00:01.162 - 2000 total Requests (1721.2 requests/s)
main elapsed: 00:00:00.000 - 0 total Requests (NaN requests/s)
PLACE_2000_RANDOMPROD elapsed: 00:00:01.384 - 2000 total Requests (1445.1 requests/s)
CREATE_U_P_PLACE_GETPURCHASES_5000 elapsed: 00:00:10.033 - 20000 total Requests (1993.4 requests/s)
Total elapsed time: 00:00:29.477 - 64003 total Requests (2171.3 requests/s)
Total requests: 64003
```

Distributing and Nginx

Before:

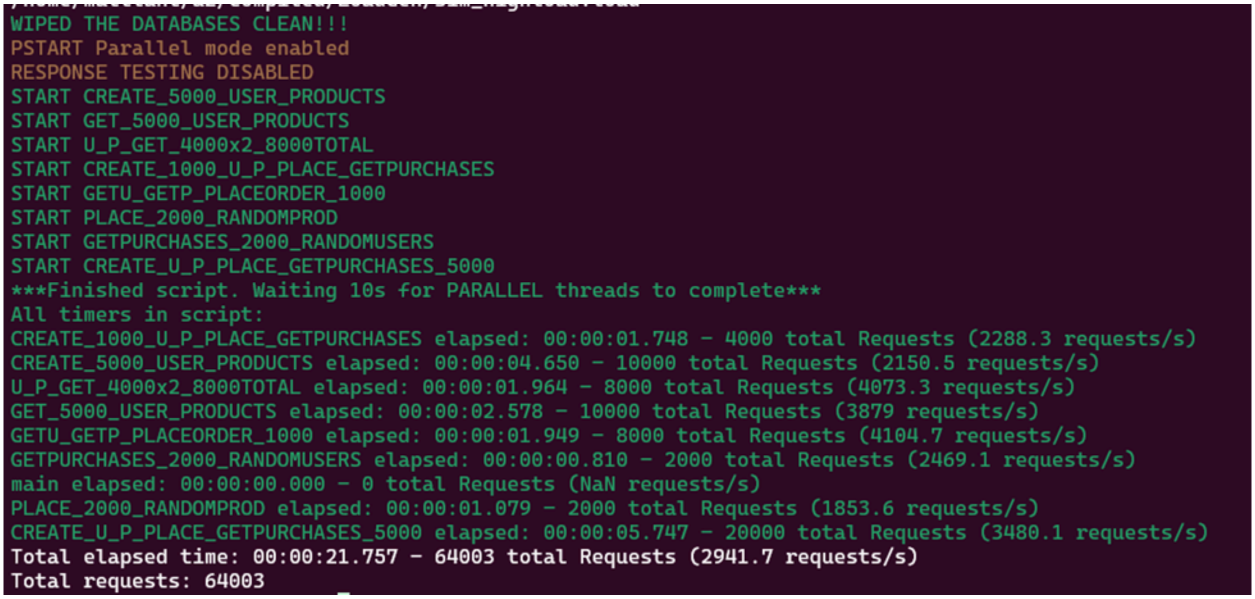
- Only had one copy of each service running on one machine.

After:

- Distributed another copy of each service on a different machine.
- Added Nginx to handle load balancing between the two machines.
- Increased parallel requests.

Results:

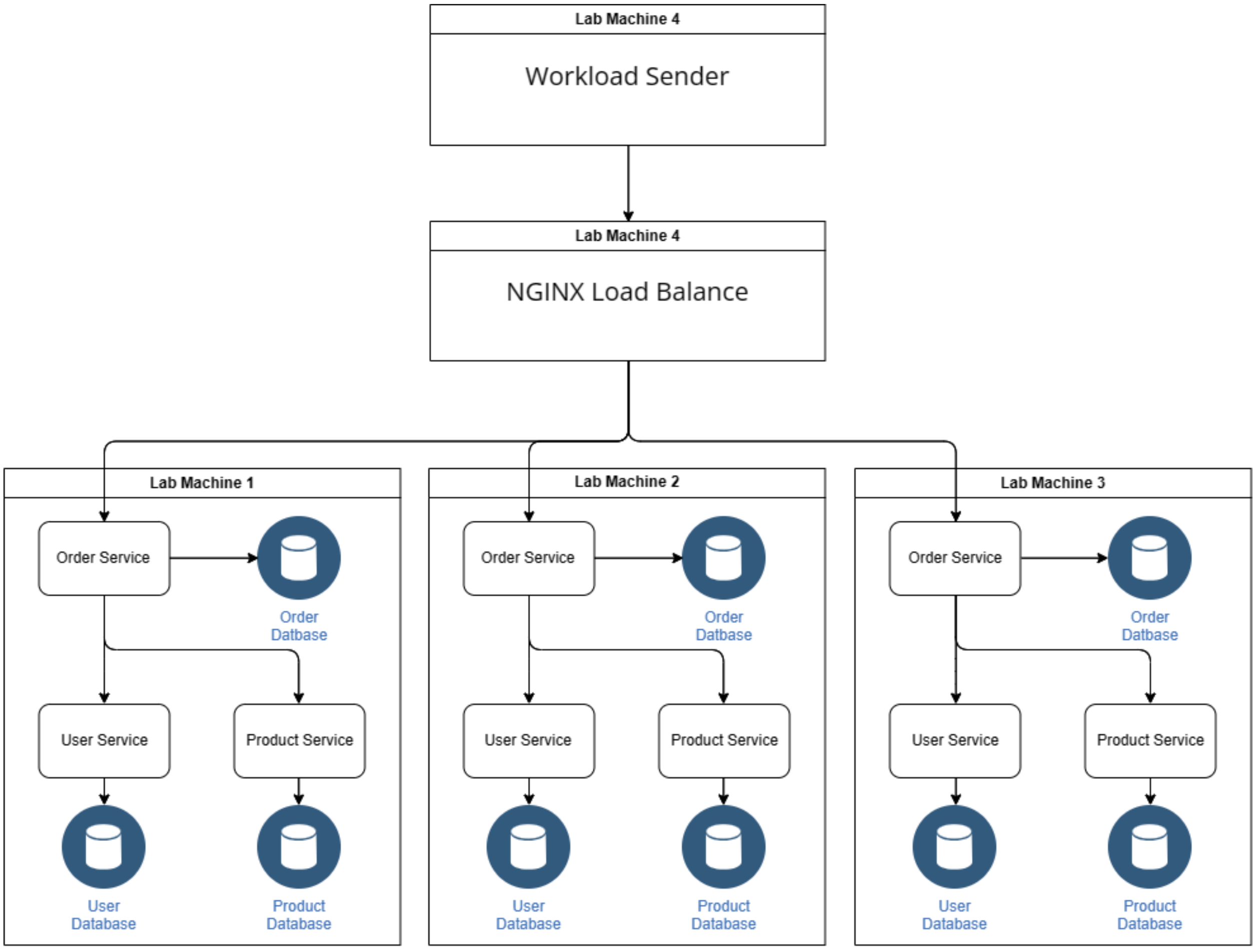
- Total Requests: **64003**
- Avg: **2941.7 requests/s**



```
WIPED THE DATABASES CLEAN!!!
PSTART Parallel mode enabled
RESPONSE TESTING DISABLED
START CREATE_5000_USER_PRODUCTS
START GET_5000_USER_PRODUCTS
START U_P_GET_4000x2_8000TOTAL
START CREATE_1000_U_P_PLACE_GETPURCHASES
START GETU_GETP_PLACEORDER_1000
START PLACE_2000_RANDOMPROD
START GETPURCHASES_2000_RANDOMUSERS
START CREATE_U_P_PLACE_GETPURCHASES_5000
***Finished script. Waiting 10s for PARALLEL threads to complete***
All timers in script:
CREATE_1000_U_P_PLACE_GETPURCHASES elapsed: 00:00:01.748 - 4000 total Requests (2288.3 requests/s)
CREATE_5000_USER_PRODUCTS elapsed: 00:00:04.650 - 10000 total Requests (2150.5 requests/s)
U_P_GET_4000x2_8000TOTAL elapsed: 00:00:01.964 - 8000 total Requests (4073.3 requests/s)
GET_5000_USER_PRODUCTS elapsed: 00:00:02.578 - 10000 total Requests (3879 requests/s)
GETU_GETP_PLACEORDER_1000 elapsed: 00:00:01.949 - 8000 total Requests (4104.7 requests/s)
GETPURCHASES_2000_RANDOMUSERS elapsed: 00:00:00.810 - 2000 total Requests (2469.1 requests/s)
main elapsed: 00:00:00.000 - 0 total Requests (NaN requests/s)
PLACE_2000_RANDOMPROD elapsed: 00:00:01.079 - 2000 total Requests (1853.6 requests/s)
CREATE_U_P_PLACE_GETPURCHASES_5000 elapsed: 00:00:05.747 - 20000 total Requests (3480.1 requests/s)
Total elapsed time: 00:00:21.757 - 64003 total Requests (2941.7 requests/s)
Total requests: 64003
```

Final Form

Architecture



Performance

Before:

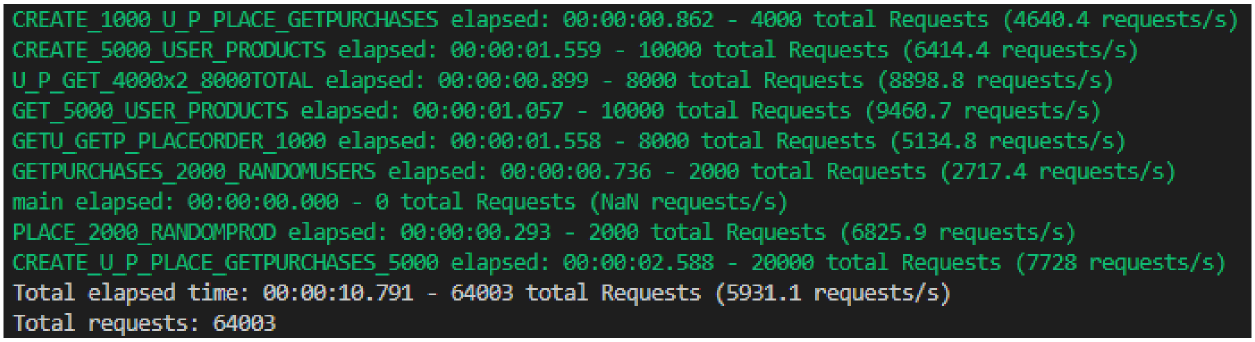
- Only 2 copies on 2 machines.
- Running on the home network which was slow.

After:

- Ran on school network.
- Used 4 machines, 3 machines running 3 copies of the services, 1 machine running Nginx.

Results:

- Total Requests: **64003**
- Avg: **5931.1 requests/s**



```
CREATE_1000_U_P_PLACE_GETPURCHASES elapsed: 00:00:00.862 - 4000 total Requests (4640.4 requests/s)
CREATE_5000_USER_PRODUCTS elapsed: 00:00:01.559 - 10000 total Requests (6414.4 requests/s)
U_P_GET_4000x2_8000TOTAL elapsed: 00:00:00.899 - 8000 total Requests (8898.8 requests/s)
GET_5000_USER_PRODUCTS elapsed: 00:00:01.057 - 10000 total Requests (9460.7 requests/s)
GETU_GETP_PLACEORDER_1000 elapsed: 00:00:01.558 - 8000 total Requests (5134.8 requests/s)
GETPURCHASES_2000_RANDOMUSERS elapsed: 00:00:00.736 - 2000 total Requests (2717.4 requests/s)
main elapsed: 00:00:00.000 - 0 total Requests (NaN requests/s)
PLACE_2000_RANDOMPROD elapsed: 00:00:00.293 - 2000 total Requests (6825.9 requests/s)
CREATE_U_P_PLACE_GETPURCHASES_5000 elapsed: 00:00:02.588 - 20000 total Requests (7728 requests/s)
Total elapsed time: 00:00:10.791 - 64003 total Requests (5931.1 requests/s)
Total requests: 64003
```