Summary of the work done from week 13 (16.09.2019) to week 16 (12.10.2019)

During the thirteenth week of the internship I analyzed the Flight recordings and the flame graphs that I collected to find the reason for the drop in throughput of the new Ballerina version. And there was a separate stack trace for Packerina in the new version. After further investigation, I found that Packerina is a packaging tool used in Ballerina to package Java code. This was called because I ran the Ballerina file using ballerina run command instead of java -jar command. When I changed that code and re-ran the tests the Packerina stack trace was removed. Further analyzing the jfr files, it seemed that not all contentions are displayed in the Java Mission Control. That was due to the contention threshold which will collect only up to that duration. Therefore I searched a way to change the synchronization threshold and found that it can be changed by creating a new profile for Flight recording and analyzed them as well. After that I started working on the Bayesian optimization of Gaussian Process Regressor to optimize Ballerina threadpool size. And successfully created that script and ran some initial tests.

In the next week I ran tests to check whether the auto tuning actually works. But, the default test and the auto-tuned tests gave the same 99 percentiles. Therefore I started investigated on the issue. To debug I used the remote jmx connection from jconsole. But, due to the firewall in the servers I couldn't connect to the jmx port. To work around this I used ssh tunneling to tunnel that port into the localhost of my machine and then connected to that port of my localhost to get the jmx connection in the jconsole. It seemed that unless both maxpool size and corepool size are set the threadpool size won't go down the moment they are set. After fixing the issue the auto tuning worked and there were significant improvement in the 99 percentile of the passthrough and prime services. After that I started working on modifying the script to tune multiple server threadpool sizes. And ran the tests and multiple services are also tuned with significant improvement in the performance. After that collected the prometheus metrics and jtl to create a report about the improvement.

In the fifteenth week, I was searching to find how to handle concept drift in Bayesian optimization as this is needed if we want to tune at real-time. After that I started working on Ballerina Database writing service and tried to tune that. But, there was no improvement at all. To further investigate I studied about database connection pooling and max_connections variable inside the MySQL server. Then I ran tests by changing the values. But, still I there was no improvement in the 99 percentile of the service.

During the sixteenth week, I was working on further investigating where the actual bottleneck was. So, I was searching for configuring p6spy to get the latency values in the database. After that I spent time reading research papers on how to tune database. Then I made a presentation for some members of the Research team about the papers and discussed on how to use it for our use case. After that I was also investigating on the Ballerina code base regarding the database connection pool used in Ballerina to change the connection pool size as well.

………………………………

Signature