



Serverless Data Processing (CSCI-5410)

Part – A: Summary

Evaluation of container orchestration systems for
deploying and managing NoSQL data base clusters

Submitted by:

Shivam Gupta (B00810723)

What the authors have presented in the paper:

Docker containers and orchestration systems like Docker Swarm, Kubernetes, Mesos and OpenShift 3.0 are getting popular in the industry for easy, efficient, and automated software deployment, scaling and management. Using NoSQL databases like MongoDB can improve the consistency, availability, and network partition of data tier of a distributed service over an IaaS cloud platform as Docker image is portable and light weighted. These systems are very important in large-scale deployments. In the paper, the performance overhead of container orchestration system is quantified and evaluated with NoSQL databases deployment on top of IaaS cloud platform. Performance overhead of Docker Swarm and Kubernetes with MongoDB is evaluated with an OpenStack IaaS cloud in a closed lab. This is compared with VM-based deployment of these databases.

If any specific issue is addressed:

There are some disadvantages with Cloud Provisioned endpoints such as automated support for TCP/IP location transparency still is not successful and needs to be implemented, there are some additional operational management cost includes the avoidance of host port conflicts and other services, and some other management cost with container access control policies.

If any experiments or studies performed:

Firstly, the goals and motivations are described. Then, the research questions are stated, which includes questions like “What is the overhead CO database deployment with cloud end points and their comparison to VM ware based or Docker only deployment?”, “What is the overhead of CO based deployment with cluster provisioned endpoints in comparison to cloud provisioned end points?” and “Can the performance overhead of CO based deployment of cluster provisioned endpoints be reduced by choosing different network plugin?”. Lastly, the experimental design is explained, YCSB benchmarks are used for evaluation of different workload types, which has Read/Insert/Update option. This is tested in private openStack cloud platform. Each database deployment was setup with 3 different instances and placement constraint are specified on failure to evaluate auto recovery. The database deployment is configured to acknowledge write operations and read operation which are processed by invoke database instance. The mean, median, and 95th quantile of response latency is measured.

Analysis or findings made by the authors:

- The DockerOnly deployment used the network stack of node's OS, K8 deployment configured with weave net network plugin and in Swarm, no virtual network needed for exposing containers via hostport, but the traffic is redirected by virtual network docker_gwbridge.
- DockerOnly performs better than VM based deployment, while Swarm and K8 had performance overhead of 27% and 20% respectively.
- Swarm service IP deployment has scalability issues and very large performance overhead. K8 HostNameNodePort deployment works the best. No significant performance overhead with cluster provisioned endpoints.
- Swarm Service overlay works twice as faster than Swarm Service weave deployment.