DALHOUSIE
UNIVERSITY

FACULTY OF COMPUTER SCIENCE

ASSIGNMENT 1: PART A

In
The Class of

CSCI5710: SERVERLESS DATA PROCESSING

by

Janvi Patel [B00863421]

Submitted to
Prof. Saurabh Dey
Department of Computer Science
Dalhousie university.

Date: 25th May 2021

TASK 1: SUMMARY

Title: Evaluation of container orchestration systems for deploying and managing NoSQL database clusters.

Authors aims to analyze different container orchestration systems such as Docker swarm and Kubernetes to deploy and manage NoSQL database clusters with MongoDB as database.

This research is about comparison for several automated support systems for distributed applications. The authors also have used an OpenStack IaaS cloud which is in less automated manner. Now a days, there is a strict increase in the automate container deployment, scaling and deployment using orchestration systems. Moreover, companies are not only using docker packing approach as it is a simple containerized solution but also cloud based production which handles large scale deployments not only on deployment platform but also for the cloud-based system. The benefits of NoSQL databases (1) fast auto-recovery and (2) support for location transparency of TCP/IP connections. Additionally, the offering of orchestration systems are highly customizable, automated node failure detection, integration with services of distributed persistent volumes for storing the data, virtual networks containers and a built-in service proxy.

The goal of this paper is to measure and compare the performance overhead of container orchestration systems. The issues addressed here are four kinds for deploying the same database cluster. One of them is non-containerized deployments installed in Linux OS and having TCP/IP as endpoint, second is docker only deployment container runs in HOST mode and third is exposed via a cloud provisioned endpoint. Moreover, forth issue is CO deployments having cluster provisioned endpoint.

During the experimental studies, they expect that the higher CPU time will be spend by CO-based deployments in processing soft interrupts that are redirected by the network traffic using the database container. So, basically the CPU in-memory database workload is providing the answers. Another expectations was to get a higher response latency because the virtual bridge need to redirect VM-level network input/output for the database containers two times having one time as VM-level network and second time as another of virtual bridge of the database container.

As experiment setup, Authors have used the YCSB benchmark for evaluation the database clusters. The ran 5 different workload types. The authors have Intel Xeon® CU E5-2650@2.00GHz processors and 64GB DIMM DDR3 memory with ubuntu, while the master controller is an Intel® Xeon® and the master controller having Intel® Xeon® CPU E5-2430@2.20GHz. As a statistical view, multiple ping tests between droplets and VMs' means and standard are 0.694ms and 0.306ms respectively. In the same breadth, virtual network layer is better for the write operations. Basically, there are three instances having 2vCPUs and 4 GB of RAM. At last, each experiment is setup as a scalability stress test for each deployment all 5 workloads runs for 15 times having 10s requests.

There are 3 major finding after the experiments. One is all the deployments uses Host Port as endpoint configurations. The eye-catching difference used is Node Platform and the type of cloud provisioned endpoint. Docker deployment have directly the network stack of the node's

OS and K8 deployments are configured with the weave net network. Second is performance overhead is about SwarmServiceIP is exceptionally large across the four patterns of reoccurring rankings of deployments in terms of the 95th response latency. Third is Cluster Provisioned endpoints for update heavy workloads for A and F system and read heavy workloads B, C and D. The K8ServiceIP and K8ContainerDNSName deployments perform better than the cloud provisioned endpoints.

In conclusion, the performance overhead of deploying and managing MongoDB database clusters on top of OpenStack cloud. Compare to Docker only deployment that runs in host mode the CPU bound database workload container orchestration is better in terms of performance. Some YCSM pairs, cloud provisioned IP endpoint is less performance overhead than CO-based compare to cluster provisioned endpoints. At the end, Service IP and containerDNSName endpoints perform the same update operation with read heavy workloads in which Node Ports are better.

Reference: *"Evaluation of container orchestration systems for deploying and managing NoSQL database clusters" by Eddy Truyen, Dimitri Van Landuyt, Bert Lagaisse, Wouter Joosen, Matt Bruzek*