Kubernetes: Essential for Cloud Transformation

John Jeyasekaran Lawrence, Edmond Prakash and Chaminda Hewage
Cardiff School of Technologies, Cardiff Metropolitan University
J.Lawrence11@outlook.cardiffmet.ac.uk {EPrakash | CHewage} @cardiffmet.ac.uk

I. Overview

The cloud computing model has brought in a bevy of digital innovations and disruptions. With the faster proliferation of containers, which are being positioned as the most optimal runtime for microservices-centric applications, the usage of container orchestration platform solutions (alternatively container life-cycle management platforms) has recently gone up significantly. Therefore, establishing, sustaining and managing containerized cloud environments is getting a lot of interest among worldwide cloud experts and data center pioneers.

Containers are lightweight virtualization solution gaining a huge mind and market share these days. Physical machines/bare metal servers are being partitioned into hundreds of containers. Therefore, Due to the heightened container density in a typical cloud center, the operational and management complexities of containerized clouds are bound to rise drastically. This has propelled and pointed out the need for automated solutions to simplify and streamline container cloud management aspects. Having realized this predicament, Cloud professionals and pundits have insisted for container orchestration platform solutions. In this regard, Kubernetes platform has gained the much-needed prominence and dominance to facilitate the movement towards Kubernetes-managed containerized clouds, which host microservices-centric applications.

This development, deployment and management model is being termed as cloud-native computing. Kubernetes is being positioned as the key container orchestration system framework for managing containerized applications for the cloud-native era. Kubernetes is a powerful orchestration platform for containerized applications and services and can be applied into important future technologies including cloud/edge computing and IoT gateways. Its feature HPA provides dynamic and effective scaling for applications without the necessity of human intervention.

Kubernetes (K8s) is being established as the platform for container life-cycle management. Containerized applications are being exposed as pods. Now pods are being managed intelligently through K8s. We can safely expect the futuristic cloud environments will be containerized and Kubernetes-managed. As cloud-native environments are typically dense, the interaction and collaboration complexities are bound to be high. So, researchers are exploring different aspects of containerized clouds. We have decided to dig deeper in order to clearly understand the performance bottlenecks in any Kubernetes environment and to bring forth a series of steps to proactively eliminate performance issues with the sole intention of establishing and sustaining high-performance K8s environments, which host all types of applications. All kinds of enterprise-scale software (enterprise resource planning (ERP), supply chain management (SCM), customer relationship management (CRM), knowledge management (KM), e-commerce, etc.) are being taken to K8s environments. Similarly, mobile, wearable, IoT, block-chain, and telecom applications are also accordingly modernized and migrated to Kubernetes environments. Precisely speaking, all kinds of transaction, analytical, and operational applications are hosted and run on K8s environments. For such environments, high performance is definitely essential. In this paper, we are to articulate and accentuate the performance challenges and concerns and how they can be surmounted through a variety of technologies.

II. Technical Innovations

Kubernetes simplifies the deployment and management of applications on a wide range of infrastructure solutions. It helps IT teams manage distributed applications in containers, but it also introduces new challenges. Performance is critical for any application deployed in a Kubernetes cluster to ensure that the cluster scales to meet changes in request volumes. And the use of containers for large-scale systems opens many challenges in the area of resource management. Kubernetes has definitely brought in a number of noteworthy advancements in administering, operating and managing containerized clouds in an automated manner. Self-healing and auto scaling are the most important contributions and optimized the cloud operating costs and Quality of Service. Service resiliency, IT reliability, and insensibility are being accomplished with additional tools on Kubernetes. Newer workloads are being hosted, run, governed, orchestrated and enhanced through the steadily growing Kubernetes tools ecosystem which allows containerized applications and services to run resiliently without the need of human intervention. And the focus is to reduces the response time and to meet the service level objectives.

1) Exploring the Performance Bottlenecks of Kubernetes Environments at Infrastructure, Platform, Container, and Microservice levels.
2) Articulating and Accentuating the Best Practices for surmounting Performance Limitations and Lacunae.
3) Leveraging Machine and Deep Learning Algorithms and Approaches for detecting Performance problems and their resolutions.
4) To Build an enabling Artificial Intelligence (AI)-inspired framework for Performance Engineering and Enhancement for Kubernetes Environments.

III. Early Stage Ideas with Proof of Concept

The goal of the performance analysis is to build highly available, scaleable and stable autoscaling algorithms with the help of machine learning and deep learning methods. This research entails the classification of data analytics and artificial intelligence (AI) algorithms. The other enabling technologies and tools play a very vital role in shaping up a viable mechanism towards the seamless and spontaneous transition of data to knowledge. We focus on the following

1) Performing Performance Analysis and Assessment of Microservices, Containers, Service Composition, Container Orchestration, etc. and Pinpointing Performance Challenges of Kubernetes Clusters. Performance Challenges of Kubernetes Clusters.
2) Explaining the Tools and Approaches for accelerated Deployment of Artificial Intelligence (AI) and Data Science Models on Kubernetes Systems.
3) Performing Performance Analytics using Machine Learning (ML) Algorithms and arriving Performance Engineering and Enhancement (PE2) Methods.

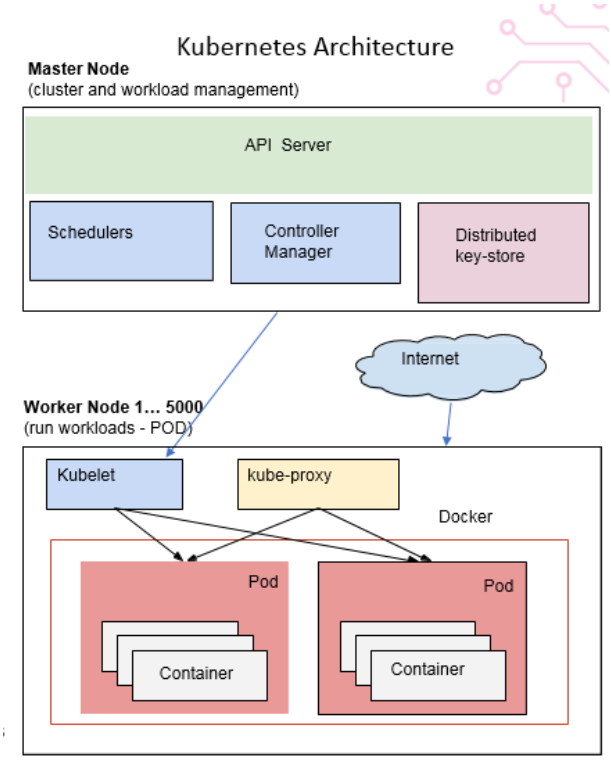

Fig. 1. Kubernetes Architecture

IV. Framework, Modeling & Experimentation

This research entails the following steps:

A. Data collection and cleansing

The aim is for the cluster to auto scale when incoming requests exceed normal usage patterns. And also understand the impact, by comparing the performance of Kubernetes

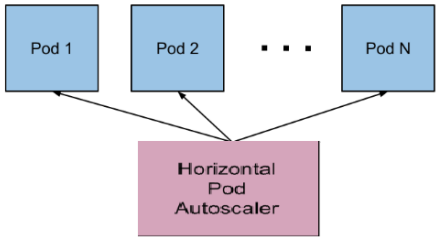

Fig. 2. Kubernetes Horizontal Pod Auto-scalar

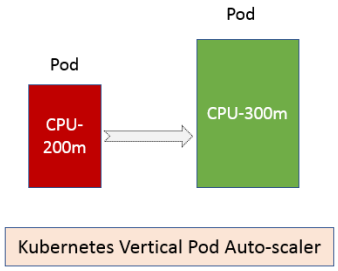

Fig. 3. Kubernetes Vertical Pod Autoscaler

cluster and evaluating the impact of decisions on cluster performance. This is a very crucial step towards performance prediction and prescription for performance enhancement. We need Kubernetes clusters' operational, performance, scalability, log, and security data from multiple sources. When there is a big data, the decisions being taken out of the data are also accurate. Further on, wrong data leads to wrong conclusion and hence data has to be clean, error-free, and complete. There are enabling tools to repair all kinds of data. As we are to leverage the increasing power of artificial intelligence (AI) algorithms (machine and deep learning), data volumes play a very vital role in bringing forth the useful and usable knowledge from them.

B. Data Pre-processing and Storage

Data has to be subjected to a variety of deeper investigations in order to do desired translations of data according to the target requirements. As data has to be mined, analyzed and processed accordingly, data has to be prepared for that actions. Data, coming from different and distributed sources, are in disparate format and structure. All the deviations have to be closed down to facilitate data analytics. Secondly, for big data analytics through batch processing, data storage is mandated. There are file systems, SQL and NoSQL databases, in-memory databases, etc.

C. Data Analytics

Now data is in prime position to be analyzed. There are big, fast and streaming data analytics platforms and methods.

These are primarily used for deterministic and diagnostic analytics activities. Now with the surging popularity of artificial intelligence (AI) algorithms for bringing forth predictive and prescriptive insights, the data analytics scenario is bound to go through drastic changes in the days to unfold. Thus, extracting actionable insights out of data heaps is getting simplified and streamlined through the smart leverage of machine and deep learning algorithms.

D. Knowledge visualization

Data analytics helps in discovering knowledge, which gets disseminated to actuation systems, business executives and other automation systems to ponder about timely counter measures. Thus, analytics technologies and tools play a very vital role in shaping up a viable mechanism towards the seamless and spontaneous transition of data to knowledge.

V. Conclusions

The prime problems include the performance engineering and enhancement of Kubernetes clusters and to visualize and realize intelligent container clouds. The study includes the system responds to a sudden increase in requests, responds under a heavy load and the survives survival of system under a constant, moderate load for longer duration of times. And with the better prediction accuracy and the recent mature of the artificial intelligence (AI) contributed by the adaptation of using deep learning and Machine Learning (ML) Algorithm in the various applications domains such as speech recognition and Facebook's Deep Text and Google's Deep Dream etc. can be used to solve sharply the auto scaling and the performance problems so that it can enhance the confidence of Kubernetes operators and cloud users.

References

[1] A. Abdel Khaleq and I. Ra. Agnostic approach for microservices autoscaling in cloud applications. In *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 1411–1415, 2019.

[2] Y. Al-Dhuraibi, F. Paraiso, N. Djarallah, and P. Merle. Elasticity in cloud computing: State of the art and research challenges. *IEEE Transactions on Services Computing*, 11(2):430–447, 2018.

[3] O. Anisfeld, E. Biton, R. Milshtein, M. Shifrin, and O. Gurewitz. Scaling of cloud resources-principal component analysis and random forest approach. In *2018 IEEE International Conference on the Science of Electrical Engineering in Israel (ICSEE)*, pages 1–5, 2018.

[4] E. Casalicchio and V. Perciballi. Auto-scaling of containers: The impact of relative and absolute metrics. In *2017 IEEE 2nd International Workshops on Foundations and Applications of Self* Systems (FAS*W)*, pages 207–214, 2017.

[5] Z. Chen, J. Hu, G. Min, A. Y. Zomaya, and T. El-Ghazawi. Towards accurate prediction for high-dimensional and highly-variable cloud workloads with deep learning. *IEEE Transactions on Parallel and Distributed Systems*, 31(4):923–934, 2020.

[6] H. Fathoni, C.-T. Yang, C.-H. Chang, and C.-Y. Huang. Performance comparison of lightweight kubernetes in edge devices. In C. Esposito, J. Hong, and K.-K. R. Choo, editors, *Pervasive Systems, Algorithms and Networks*, pages 304–309, Cham, 2019. Springer Intl. Publishing.

[7] Z. He. Novel container cloud elastic scaling strategy based on kubernetes. In *2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)*, pages 1400–1404, 2020.

[8] S. Horovitz and Y. Arian. Efficient cloud auto-scaling with sla objective using q-learning. In *2018 IEEE 6th International Conference on Future Internet of Things and Cloud (FiCloud)*, pages 85–92, 2018.

[9] M. Imdoukh, I. Ahmad, and M. G. Alfailakawi. Machine learning-based auto-scaling for containerized applications. In *Neural Computing and Applications*, volume 32, page 9745–9760, 2020.

[10] Y. Jin-Gang, Z. Ya-Rong, Y. Bo, and L. Shu. Research and application of auto-scaling unified communication server based on docker. In *2017 10th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, pages 152–156, 2017.

[11] D. F. Kirchoff, M. Xavier, J. Mastella, and C. A. F De Rose. A preliminary study of machine learning workload prediction techniques for cloud applications. In *2019 27th Euromicro Intl. Conf. on Parallel, Dist. and Network-Based Processing (PDP)*, pages 222–227, 2019.

[12] kubernetes.io. Horizontal pod autoscaler. https://kubernetes.io.

[13] J. Kumar, A. K. Singh, and R. Buyya. Self directed learning based workload forecasting model for cloud resource management. *Information Sciences*, 543:345 – 366, 2021.

[14] C. Lin., H. Pai., and J. Chou. Comparison between bare-metal, container and vm using tensorflow image classification benchmarks for deep learning cloud platform. In *Proceedings of the 8th International Conference on Cloud Computing and Services Science - Volume 1: CLOSER,*, pages 376–383. INSTICC, SciTePress, 2018.

[15] T.-T. Nguyen, Y.-J. Yeom, T. Kim, D.-H. Park, and S. Kim. Horizontal pod autoscaling in kubernetes for elastic container orchestration. *Sensors*, 20(16):4621, July 2020.

[16] M. Orzechowski, B. Balis, R. G. Słota, and J. Kitowski. Reproducibility of computational experiments on kubernetes-managed container clouds with hyperflow. In V. V. Krzhizhanovskaya, G. Závodszky, M. H. Lees, J. J. Dongarra, P. M. A. Sloot, S. Brissos, and J. Teixeira, editors, *Computational Science – ICCS 2020*, pages 220–233, Cham, 2020. Springer International Publishing.

[17] ProphetStor. How prophetstor uses reinforcement learning for optimizing resource management in kubernetes, Sept. 2018. https://medium.com/@prophetstor/how-prophetstor-uses-reinforcement-learning-for-optimizing-resource-management-in-kubernetes-ed5273331c96.

[18] ProphetStor. Performance prediction and anomaly detection using deep learning, Nov. 2018. https://medium.com/prophetstor-data-science-blog/performance-prediction-and-anomaly-detection-using-deep-learning-2c16bf08e782.

[19] G. Rattihalli, M. Govindaraju, H. Lu, and D. Tiwari. Exploring potential for non-disruptive vertical auto scaling and resource estimation in kubernetes. In *2019 IEEE 12th International Conference on Cloud Computing (CLOUD)*, pages 33–40, 2019.

[20] M. Rodriguez and R. Buyya. Container orchestration with cost-efficient autoscaling in cloud computing environments. In *In Handbook of Research on Multimedia Cyber Security. edited by Brij B. Gupta , and Deepak Gupta*, pages 190–213. Hershey, PA: IGI Global, 2020.

[21] P. P. Shinde and S. Shah. A review of machine learning and deep learning applications. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–6, 2018.

[22] M. Song, C. Zhang, and E. Haihong. An auto scaling system for api gateway based on kubernetes. In *2018 IEEE 9th Intl. Conf. on Software Engineering and Service Science (ICSESS)*, pages 109–112, 2018.

[23] B. Thurgood and R. G. Lennon. Cloud computing with kubernetes cluster elastic scaling. In *Proceedings of the 3rd International Conference on Future Networks and Distributed Systems*, ICFNDS '19, New York, NY, USA, 2019. Association for Computing Machinery.

[24] L. Toka, G. Dobreff, B. Fodor, and B. Sonkoly. Adaptive ai-based auto-scaling for kubernetes. In *2020 20th IEEE/ACM Intl. Symp. on Cluster, Cloud and Internet Comp. (CCGRID)*, pages 599–608, 2020.

[25] S. Verreydt, E. H. Beni, E. Truyen, B. Lagaisse, and W. Joosen. Leveraging kubernetes for adaptive and cost-efficient resource management. In *Proc. of the 5th Intl. Workshop on Container Technologies and Container Clouds*, WOC '19, page 37–42. Assoc. for Comp. Machinery, 2019.

[26] vp autoscaler. Vertical pod autoscaler. https://github.com/kubernetes/autoscaler/tree/master/vertical-pod-autoscaler.

[27] M. Wang, D. Zhang, and B. Wu. A cluster autoscaler based on multiple node types in kubernetes. In *2020 IEEE 4th Inf. Tech. Networking, Electronic and Automation Control Conf. (ITNEC)*, volume 1, pages 575–579, 2020.

[28] H. Zhao, H. Lim, M. Hanif, and C. Lee. Predictive container auto-scaling for cloud-native applications. In *Intl. Conf. on Information and Communication Tech. Convergence (ICTC)*, pages 1280–1282, 2019.