



Run TigerGraph in Containers - TigerGraph in Amazon Elastic Kubernetes Service and Google Kubernetes Engine

Emma Liu, Senior Product Manager, TigerGraph

Wenbing Sun, Senior Quality Engineer, TigerGraph

April 2021



Speakers



Benyue (Emma) Liu

Senior Product Manager, TigerGraph

- BS in Engineering from Harvey Mudd College, MS in Engineering Systems from MIT
- Prior work experience at Oracle and MarkLogic
- Focus - Cloud, Containers, Enterprise Infra, Monitoring, Management, Connectors, Developer Tools, Applications



Wenbing Sun

Senior Quality Engineer, TigerGraph

- Performance and Reliability Engineering of Tigergraph
- Certified AWS Professional Devops Engineer
- 10+ years practices in support and architecture of hybrid cloud infrastructure and applications in the industry

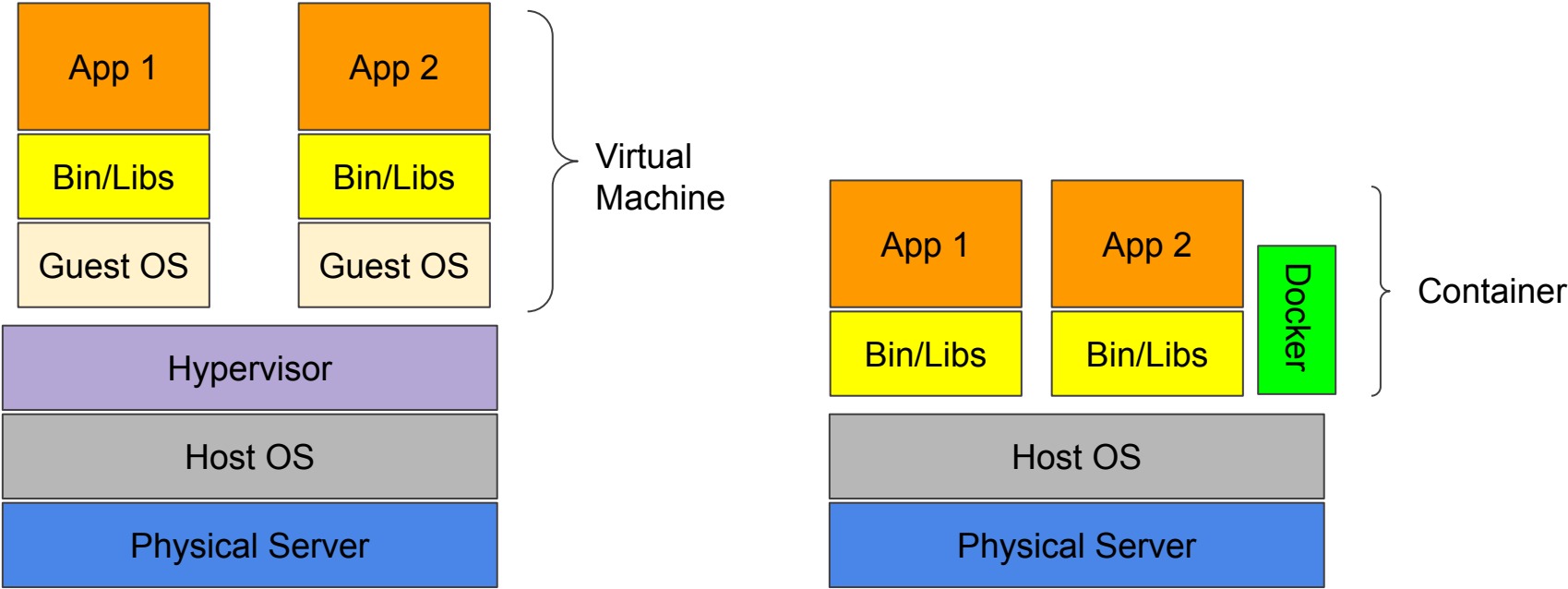
Agenda

1. Containerization Overview
 - a. Benefits of TigerGraph in Kubernetes (K8s)
2. TigerGraph in K8s Architecture
3. TigerGraph Demo in EKS and GKE

Containerization Overview (TigerGraph in K8s)



Virtual Machines vs. Containers



Containerization Overview

- Docker: one type of container, grown to be the leading container format
- Kubernetes: the winning container orchestration framework for production workload
- Docker and Kubernetes work together

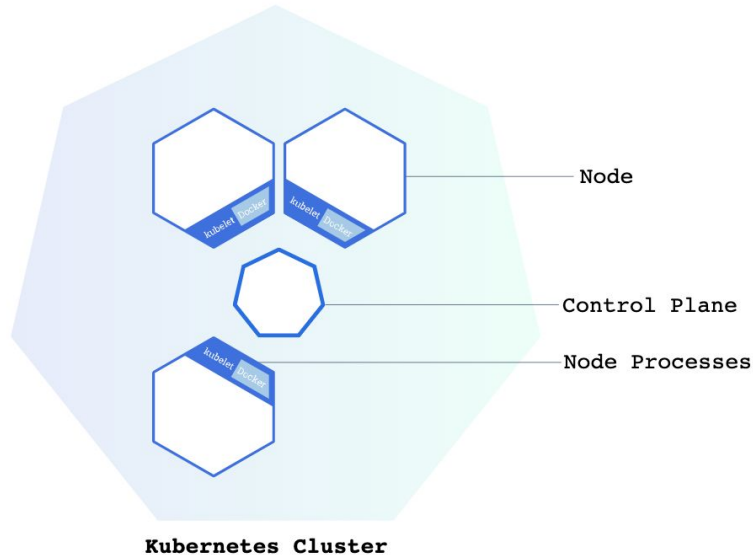
Benefits of Containers:

- Portable: Run Anywhere
- Consistent Environment
- Effective Isolation
- Lightweight and Speed
- Developer Productivity
- Resource Efficiency

Benefits of Kubernetes:

- Leading Orchestration Tool
- Portable and Open Source
- Support from All Three Major Cloud Platforms
- Workload Scalability
- Speed for Deployment
- Meet Production Operational Needs

What is Kubernetes?



- Leading containerization orchestration framework
- Automated rollouts and rollbacks
- Storage orchestration
- Automatic bin packing
- Self-healing
- Service discovery and load balancing
- Secret and configuration management
- Horizontal scaling

Figure 1. Kubernetes Cluster Overview Diagram

Retrieved from: <https://kubernetes.io/docs/tutorials/kubernetes-basics/create-cluster/cluster-intro/>

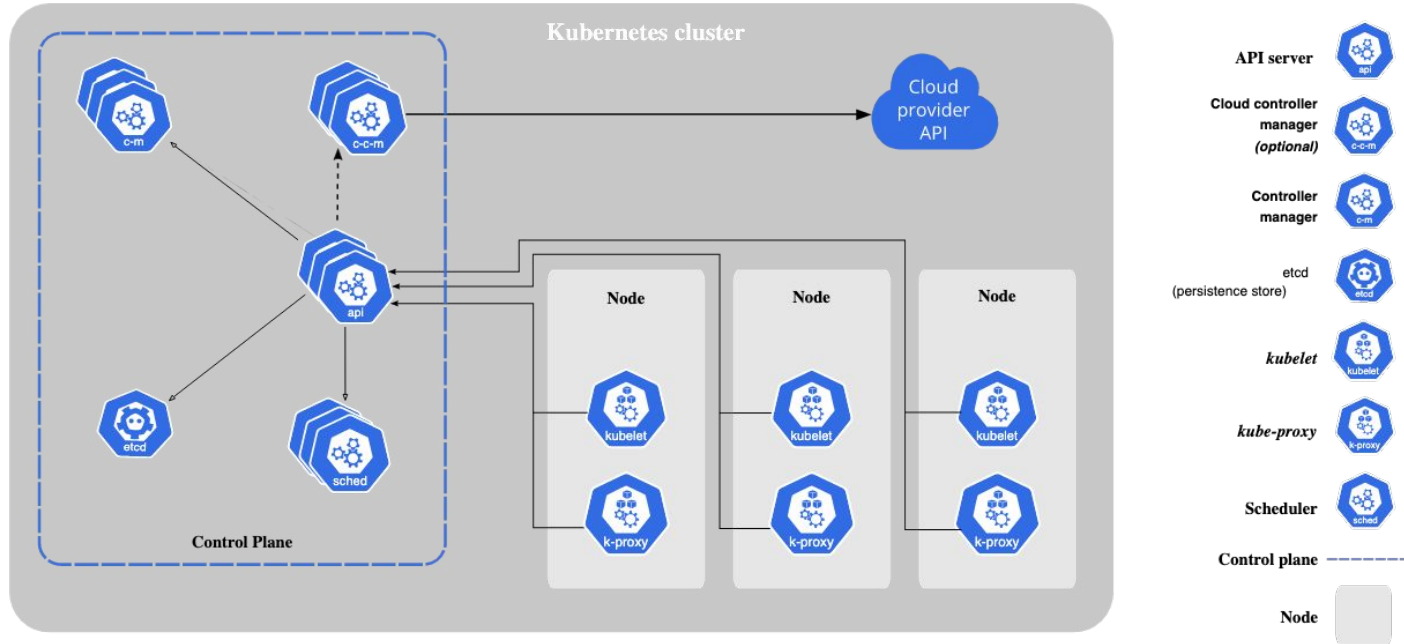


Figure 2. Kubernetes Cluster with All Components
 Retrieved from: <https://kubernetes.io/docs/concepts/overview/components/>

Containerize Database in Kubernetes

- Kubernetes is more matured now for databases -> more adoption in production for various databases
 - Key Kubernetes features for databases: StatefulSet & Persistent Volumes



- TigerGraph architecture fits well in Kubernetes framework
 - Distributed Architecture, Scalability, Continuous Availability in TG3.x

Why Using TigerGraph in Kubernetes

- Complete your containerization stack
 - “Better together”: TigerGraph database container and your application containers working together
- Run anywhere: multi cloud and hybrid cloud
- Manage full lifecycle of your TigerGraph clusters with Kubernetes’ builtin features and benefits

EKS

- Amazon Elastic Kubernetes Service
- Managed Kubernetes Services on AWS



Amazon EKS

GKE

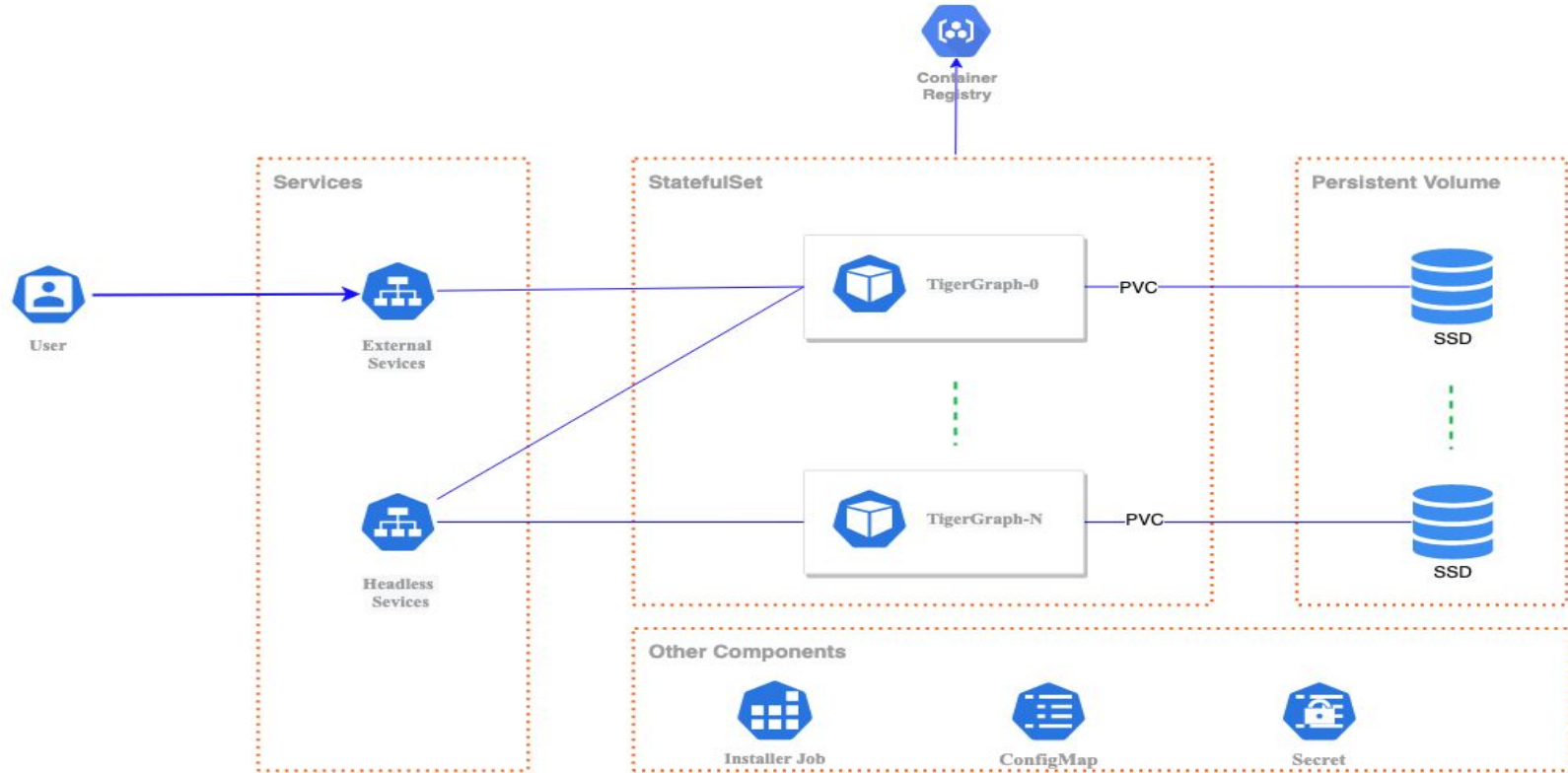
- Google Kubernetes Engine
- Managed Kubernetes Services on GCP



TigerGraph in Kubernetes (EKS and GKE)



TigerGraph in K8s Architecture



DEMO: TigerGraph in EKS and GKE

Objectives:

1. Deploy Single Node TigerGraph Cluster in EKS
2. Deploy Three Nodes TigerGraph Cluster in GKE
3. Nodes Failover Test

Demo Setup

Prerequisites:

1. Running EKS/GKE cluster with enough resources
2. The kubectl command-line tool (v1.18.0+)
3. AWS/GCP roles or permissions to manage kubernetes resource

DEMO

Summary



Summary

1. Containerization Overview
 - a. Benefits of TigerGraph in Kubernetes (K8s)
2. TigerGraph in K8s Architecture
3. TigerGraph Demo in EKS (AWS) and GKE (GCP)

*Support TigerGraph Single Node Deployment (V3.1.1+) in K8s: **Coming Soon!***

*Support For TigerGraph Cluster Deployment (V3.2.0+) in K8s: **Summer 2021***

Questions? emma.liu@tigergraph.com