Cisco Knowledge Network
Cisco HyperFlex Storage Integration for Kubernetes

Cisco Knowledge Network
Service Provider Data Center
April 9, 2019
What is Cisco HyperFlex?
Shared data platform

CPU
MEM
HCI
Hyper

DISK
DISK
...
DISK

CPU
MEM
HCI
Hyper

DISK
DISK
...
DISK

CPU
MEM
HCI
Hyper

DISK
DISK
...
DISK

CPU
MEM
HCI
Hyper

DISK
DISK
...
DISK

© 2018 Cisco and/or its affiliates. All rights reserved.
Bringing the cloud experience on-premises

Cisco HyperFlex Multicloud Platform

Any app. Any cloud. Any scale

- Pathway to microservices
- Full stack monitoring and security
- Agile resource provisioning
- Packaged workload management
- Easy consumption model
What is a Multicloud world?
Why Multicloud?

- Avoid Lock-in
- Innovation
- Speed
- Resiliency/Availability
- Best of breed
What is Kubernetes?
Containers Are...

- A way to package up our applications and dependencies.
- A way to guarantee execution consistency and portability.
- A way to keep your applications isolated.
- A way to use your compute resources without the overhead of VM’s.
Containers are not...

- **Microservices**
  - Microservices benefit from a lightweight packaging, distribution and deployment solution.
  - However, you can package *anything* into a container, including a badly written legacy app in some cases, therefore using containers doesn’t magically make bad code better.

- **VM’s**
  - VM’s virtualise the hardware (plus have additional overhead with the Guest Operating System), whereas Containers virtualise the Linux Kernel.
  - Containers are purely user-space, if you need kernel extensions/modules or a custom kernel, containers probably aren’t what you’re looking for.

- **Magic**
  - They have their own characteristics and require proper deployment considerations just like any other toolchain.
Traditional Infrastructure vs Containers

- **Bare Metal**
  - App
  - Bins/Libs
  - Host OS
  - Server

- **Virtual Machine**
  - App
  - Bins/Libs
  - Guest OS
  - Hypervisor
  - Server

- **Container**
  - App
  - Bins/Libs
  - Docker Engine
  - Host OS
  - Server

Virtualise the H/W

Virtualise the Kernel
Why Customers are Transitioning to Containers

**Containerise Traditional Applications**
Agility + Portability + Security + Cost Savings = Efficiency

**Transform Monolithic to Microservices**
Look for shared services to transform

**Accelerate New Applications**
Agile cloud native app development
Shipping container system for your Applications

- Automation Script
- Website
- NetDevOps App
- Chat Bot
- Network Monitoring Tool

2 editions exist:
- Docker Community (open source)
- Docker Enterprise (commercial support)

Docker Public Registry – container distribution
- Docker Hub (community content)
- Docker Store (qualified partners)

~# docker build my_app
~# docker push my_app
~#

Development Laptop
QA server
Cloud
Production Cluster
Contributor’s laptop

~# docker pull my_app
~# docker run my_app
~#
Container Orchestrators

- Group multiple nodes that can run containers into logical units (clusters)
- Schedule the placement of containers on cluster resources
- Provide service discovery tools for containers (who runs what, where?)
- Provide API services to other tiers and app developers
Containers Orchestrators Reference

- **Docker Swarm** is a Container Orchestrator provided by **Docker, Inc**. It is part of **Docker Engine**.

- **Kubernetes** started by Google, now part of the **Cloud Native Computing Foundation** project.

- Mesos **Marathon** is one of the frameworks to run containers at scale on **Apache Mesos**.

- Hashicorp **Nomad** is the Container Orchestrator provided by **HashiCorp**.

- **Amazon EKS, Google GKE, Microsoft AKS** are various container services for Kubernetes available by the leading Cloud Providers
The name **Kubernetes** originates from Greek, meaning "helmsman" or "pilot"
What is Kubernetes (K8s)?

- Kubernetes is an open source Container Orchestration system for automating deployment, scaling and management of containerised applications.
- First released June 2014, contributed to Cloud Native Computing Foundation (CNCF) in July 2015
- Its development and design are heavily inspired and influenced by Google's Borg system
What is Kubernetes (K8s)?

- Container orchestrator
  - Runs and manages containers
  - Manage applications, not machines
- 100% Open source, written in Go (aka Golang)
- Portable (public, private, hybrid)
  - Supports multiple cloud and bare-metal environments
- Extensible (modular, pluggable)
  - Rich ecosystem of plug-ins for scheduling, storage, networking, etc
What Kubernetes (K8s) provides

- Self healing, Automatic restarting, Scheduling, Load Balancing, Rolling updates
- Use the **SAME** API across bare metal and **EVERY** cloud provider
- Write once – run anywhere
- Avoid coupling app to infrastructure
- Avoid vendor lock-in
From containers and Kubernetes...
...to Kubernetes as your engine of innovation
Why HyperFlex + Kubernetes?
Storage Challenges with Containers

- Containers are stateless by design, this is one factor that makes them so portable
- Containers natively have no writable space that will persist (remain intact) across a container failure, redeploy, etc..
- So how do container applications store things like...
  - Log data, user transactions, configuration files, etc..
Persistent Storage for Pods/Containers

- Persistent storage is storage for containers that persists (remains intact) should a container fail, be redeployed, moved, updated, etc.
- Supported across both container-runtime and Kubernetes, however they are implemented slightly different
Kubernetes Persistent Volume

- API object in Kubernetes that represents a usable unit of storage
  - i.e. mount point, local disk, LUN, etc.

```
kubectl create -f <yaml_file>
```

Tell Kubernetes about the characteristics and location of the persistent volume
HyperFlex Storage – Controller VM

- Virtual machine running on HyperFlex node which assumes control of local disks through PCI pass-through
HyperFlex Storage – Controller VM

- Aggregates available storage into cluster-wide datastores configurable by the HyperFlex admin
HyperFlex Storage - IOVisor

- Runs as software VIB inside ESXi and works in conjunction with HyperFlex Controller VMs across entire cluster through high-speed data network to provide NFS target for ESXi hosts
HyperFlex Storage for Kubernetes Node VMs

- HyperFlex provides NFS datastores to vSphere for storing Kubernetes Node VM “vmdk” files
HyperFlex Storage for Kubernetes Node VMs

- On the back-end, the “vmdk” files are segmented into blocks and distributed as evenly as possible across all HyperFlex nodes in the cluster.
HyperFlex Storage for Kubernetes Node VMs

- The “vmdk” blocks are synchronously replicated within the cluster based on the HyperFlex “Replication Factor”

Based on cluster-wide Replication Factor
RF3 = three copies of data (recommended)
HyperFlex Storage for Kubernetes Node VMs

- If a node VM were to move via vMotion, it retains access to its “vmdk” blocks
HyperFlex Storage for Kubernetes Node VMs

- If a host were to fail, node VMs can be restarted and “vmdk” block copies can be leveraged.
HyperFlex FlexVolume Storage Integration for K8s
HyperFlex + Kubernetes Solutions
Cisco Container Platform
Kubernetes-as-a-Service

1. Deploy Tenant Clusters
2. Automatically Deploy & Configure HyperFlex Integration

Control Plane Cluster

Upstream Kubernetes Clusters

Cisco HyperFlex
Redhat OpenShift Container Platform
Platform-as-a-Service

1. Deploy OpenShift Clusters
2. Automatically Deploy & Configure HyperFlex Integration

Ansible Playbooks

Cisco HyperFlex
TOMORROW starts here.
cisco.com/go/hyperflex