

KubeVirt

Cats and Dogs Living Together?

Stephen Gordon

@xsgordon / sgordon@redhat.com



...for Virtualization in a cloud-native world:

• I need a place to install, run, and manage my Kubernetes clusters.



...for Virtualization in a cloud-native world:

• I need a place to install, run, and manage my Kubernetes clusters.

=> OpenStack, oVirt, VMware, AWS, GCE, Azure, etc.



...for Virtualization in a cloud-native world:

- I need a place to install, run, and manage my Kubernetes clusters.
 => OpenStack, oVirt, VMware, AWS, GCE, Azure, etc.
- I need a way to provide strict isolation of my application containers.



...for Virtualization in a cloud-native world:

- I need a place to install, run, and manage my Kubernetes clusters.
 => OpenStack, oVirt, VMware, AWS, GCE, Azure, etc.
- I need a way to provide strict isolation of my application containers.

=> Katacontainers, gVisor



What about existing workloads?



VIRTUALIZED WORKLOADS

Virtualized Workloads are not going anywhere fast! Business reasons and technical reasons.



What about existing workloads?



VIRTUALIZED WORKLOADS

Virtualized Workloads are not going anywhere fast! Business reasons and technical reasons.



CONTAINER INFRASTRUCTURE AND ORCHESTRATION

Linux Containers with Kubernetes orchestration are becoming **the** standard platform for new applications.



What about existing workloads?



VIRTUALIZED WORKLOADS

Virtualized Workloads are not going anywhere fast! Business reasons and technical reasons (older/different operating system)



CONTAINER INFRASTRUCTURE AND ORCHESTRATION

Linux Containers with Kubernetes orchestration are becoming **the** standard platform for new applications.



COMPLEX APPLICATIONS

As this transition occurs, complex applications will increasingly be made up of both containers and virtual machines. How can we bring these two worlds closer together?



Linux Virtualization Stack



Hmm... processes you say?



Enter KubeVirt



What is KubeVirt?

Technology enabling developer use of Kubernetes as a unified platform for building, modifying, and deploying applications residing in both containers and virtual machines in a common, shared environment.



Add virtual machines to your Kubernetes/OpenShift projects just as easily as application containers!



What is KubeVirt?

- Drops directly into existing Kubernetes Clusters.
- Takes as K8S-native an approach as possible.
- Leverage Container Networking Interface (CNI), Container Storage Interface (CSI). and other K8S-native integrations.
- Apache License, Version 2.0





Together at last!



Resultant virtual machines are able to run side by side directly on the same Kubernetes nodes as application containers.



Example Use Case



Example Use Case: We have a VM!





Example Use Case: Import





Example Use Case: New Functionality





Example Use Case: Decomposition





Let's keep digging...



Components

KubeVirt The virtual machine operator

Containerized Data Importer (CDI) Importing disks

CSI/Ember

Leverage 80+ existing storage drivers

Virt-v2v (APB) Importing a whole virtual machine

OpenShift Web Console (Optional) With UI extensions





High Level Architecture





Custom Resource Definitions

- Build on Kubernetes, adding new API-level resources.
- Declarative when paired with a controller.

\$ kubectl get crds	
NAME	AGE
datavolumes.cdi.kubevirt.io	5m
virtualmachineinstancepresets.kubevirt.io	5m
virtualmachineinstancereplicasets.kubevirt.io	5m
virtualmachineinstances.kubevirt.io	5m
virtualmachines.kubevirt.io	5m



VirtualMachine Operator and API

- Virtual Machines have their own kind
 - Ability to express all common virtual machine parameters and actions
 - Targeted feature set is comparable to libvirt

apiVersion: kubevirt.io/v1alpha1 kind: VirtualMachine metadata: name: vm-fedora spec: domain. devices: disks: . . . resources: requests: memory: 1024M volumes: . . . status: interfaces: - ipAddress: 172.17.0.12 nodeName: localhost phase: Running



Scheduling

- Virtual Machines are scheduled as pods
 - Same set of features (affinity/anti-affinity, labels and selectors, taints and tolerations)
 - Custom scheduler as needed

- Applications within virtual machines are exported using Service and Routes
 - Selection using labels and selectors



Virtual Machines and Pods

- Virtual Machines **live** in pods
 - Transparent to higher-level management systems (monitoring, metrics, ...)
 - Technically: Not worse than today

- Virtual Machines leverage pods
 - Metadata Labels and annotations, passed through and additional
 - CPU and memory resources
 - Affinity and anti-affinity
 - $\circ \quad \text{Storage and network} \quad$

- Virtual Machines **specifics**
 - Specific events



Disks - Import and Storage

- Virtual Machine disks are mapped to and stored on **PersistentVolumes**
 - 1:1 mapping of disk to PV Alignment with Kubernetes concepts
 - Mutable and immutable
 - Benefit from the Kubernetes/OpenShift ecosystem and thus indirectly from OpenStack Cinder and it's own ecosystem



Import using CDI

- Disk images can be imported using <u>containerized-data-importer</u>
- Declarative Kubernetes utility
- Controller watches for importer specific claims
- Use cases
 - Fetch disks via HTTP
 - Copy images from read-only Kubernetes namespace



Import using virt-v2v

- Virtual Machines can be imported into KubeVirt using <u>virt-v2v</u>
 - From VMware, libvirt, and ova
- Limitations
 - \circ Single NIC
 - Single attached disk



Network Connectivity

- Virtual Machine are connected to the regular pod network
 - From the outside there is no difference between a VM and pod

- Applications within virtual machines are exported using Service and Routes
 - Selection using labels and selectors

• Advanced networking (SR-IOC, L2, infiniband, ...) is under discussion in upstream Kubernetes



Demo

Demo

Pre-requisites:

- kubectl
- minikube/minishift

http://kubevirt.io/get_kubevirt/



NB: Yes, we're running nested virt here - fine for getting started!



Future Plans

• Production Workloads

• Embrace the Platform

• Supportability!



Collaborating

- Website:
 - <u>https://kubevirt.io</u>
- GitHub:
 - <u>https://github.com/kubevirt/</u>
- Mailing List:
 - <u>https://groups.google.com/forum/#!forum/kubevirt-dev</u>
- IRC:
 - #kubevirt on irc.freenode.net
- Slack (K8S virtualization working group):
 - #virtualization on kubernetes.slack.com



OPEN SOURCE SUMMIT

