Convergence of your virtualization and container infrastructures with KubeVirt

Fabian Deutsch, Red Hat, Virtualization & IaaS, FOSDEM 2018
Hello.

Fedora user and former package maintainer
oVirt and KubeVirt Contributor
Working at Red Hat

fabiand@redhat.com
@dummdida
fabian
https://dummdida.tumblr.com
Just one?
Need/Want Everything
A Stack?
OpenStack Installation Guide

UPDATED: 2018-01-30 01:02

- Conventions
  - Notices
  - Command prompts
- Preface
  - Abstract
  - Operating Systems
    - openSUSE and SUSE Linux Enterprise Server
    - Red Hat Enterprise Linux and CentOS
    - Ubuntu
- Get started with OpenStack
  - The OpenStack services
  - The OpenStack architecture
    - Conceptual architecture
    - Logical architecture
- Overview
Quick Start Guide

1. **Introduction**
   a. Prerequisites
      i. oVirt Engine
      ii. For each Host
      iii. Storage and Networking
      iv. Virtual Machines

2. **Install oVirt**
   a. Install oVirt Engine
   b. Install Hosts
      i. Install oVirt Node
      ii. Install Enterprise Linux Host

3. **Configure oVirt**
   a. Configure Data Center
   b. Configure Clusters
   c. Configure Network
      i. Attach oVirt Node to Enterprise Linux Host
Obviously.
Obviously not.
Storage

Network

Compute
A stack, heavier than it looks
Need/Want

Everything
Virtualization and containers
N = 2
Maybe.
Kubernetes virtualization API and runtime add-on

# If optimistic hero curious and cluster owner, then

$ kubecuddle create -f run.kubevirt.io/v/0.2.0.yaml

$ kubecuddle create -f run.kubevirt.io/v/0.2.0-on-minikube.yaml
Single API entrypoint

$ kubecuddle get pods
$ kubecuddle get vms
… inheriting authentication & authorization
... aligned logging & metrics
... aligned metric exposure
So cloud-native that it hurts thrills

Independent life-cycle, just like any other cloud-native app. No need to bother the host, who needs it anyway?! Everybody.
The famous 80% and unlucky 20.
VirtualMachine

kind: VirtualMachine
metadata:
  name: awesome2
spec:
  domain:
    devices:
      ...
  nodeSelector:
    cpu: fast
status:
  phase: Running
We get scheduled like other pods
We have display and console access!
Our network works like a pod's network

~ SOON ~
Persistent Volumes are our disks

~ SOON ~
We've got live migration!

~ SOON AGAIN ~
There is a price.
OBEY & PAY
to Kubernetes (and if you wear one of those hats)

Not that bad actually. It's a chance to think.
"Kubernetes first, virtualization second."
-- The Razor

Well, we'll not forget you, virtualization.
Obey: Network
Price: No pipes

SOON: Pipes
Price?
Obey: Scheduling
Price: No rescheduling

SOON: Descheduler
Price?
Obey: NUMA Awareness
Price: Less optimized

SOON: CPU Manager
Price?
How far will it go?

Not endless, there are differences.
Try (with minikube)

$ kubectl create \
  -f run.kubevirt.io/v/0.2.0-on-minikube.yaml \
  -f manifests/demo-pv.yaml
Going forward

Get it to you!

Stabilize

Contribute to Kubernetes

~WIP~
Summary

Single, unified, and consistent API

Converged infrastructure

80%
Thank you.

https://github.com/kubevirt/

User Guide (link)

@kubevirt

Me: @fabian