Cognizant Softvision

Inversion of Cont...ainers

From VM to Containers and back to VM

Mircea-Tiberiu MATEI | September 13, 2021

PROGRAMMERS' WEEK 2021
Cognizant Softvision

AGENDA

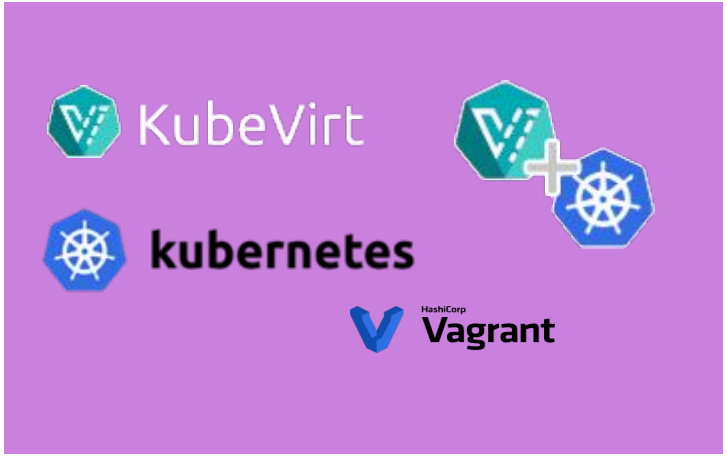
01 Introduction

02 KubeVirt

03 Vagrant

04 Demo

05 Q&A



Introduction

- Virtualization options
- The context
- Two questions
- An extra constraint

Sounds familiar?

Developer: It works on my machine

Manager:

If it works on your machine then we will ship your machine

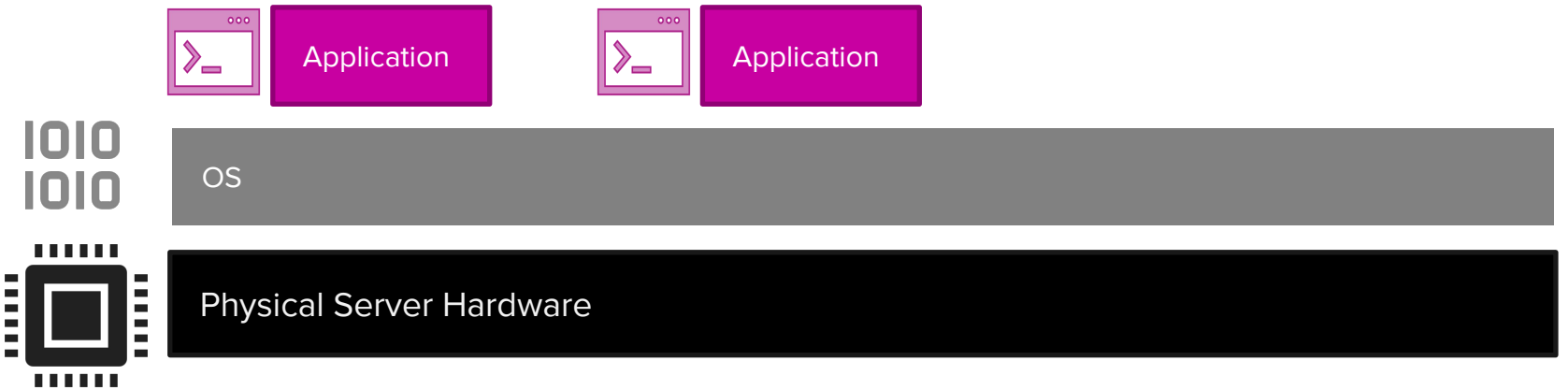
Overview of Virtualization options

Run a process

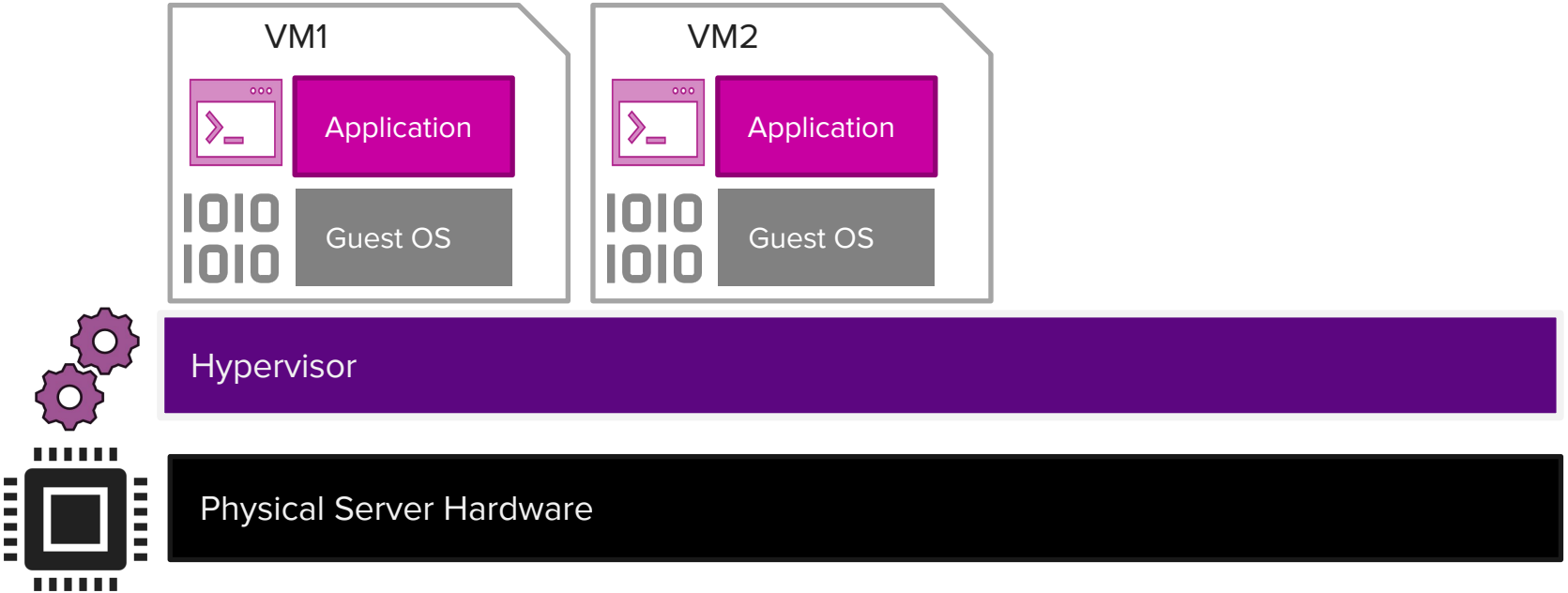
vs

Run a virtual machine

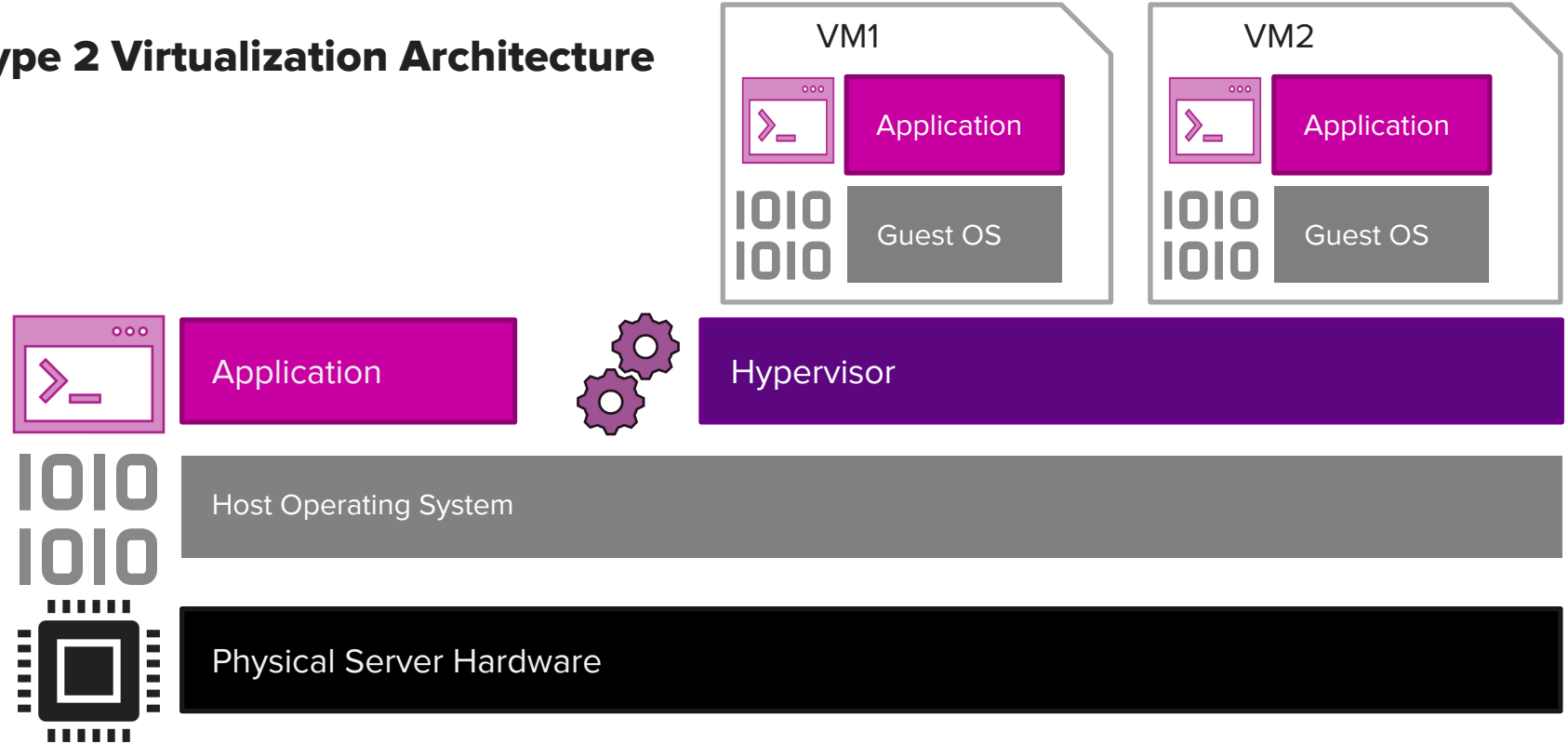
Traditional deployment



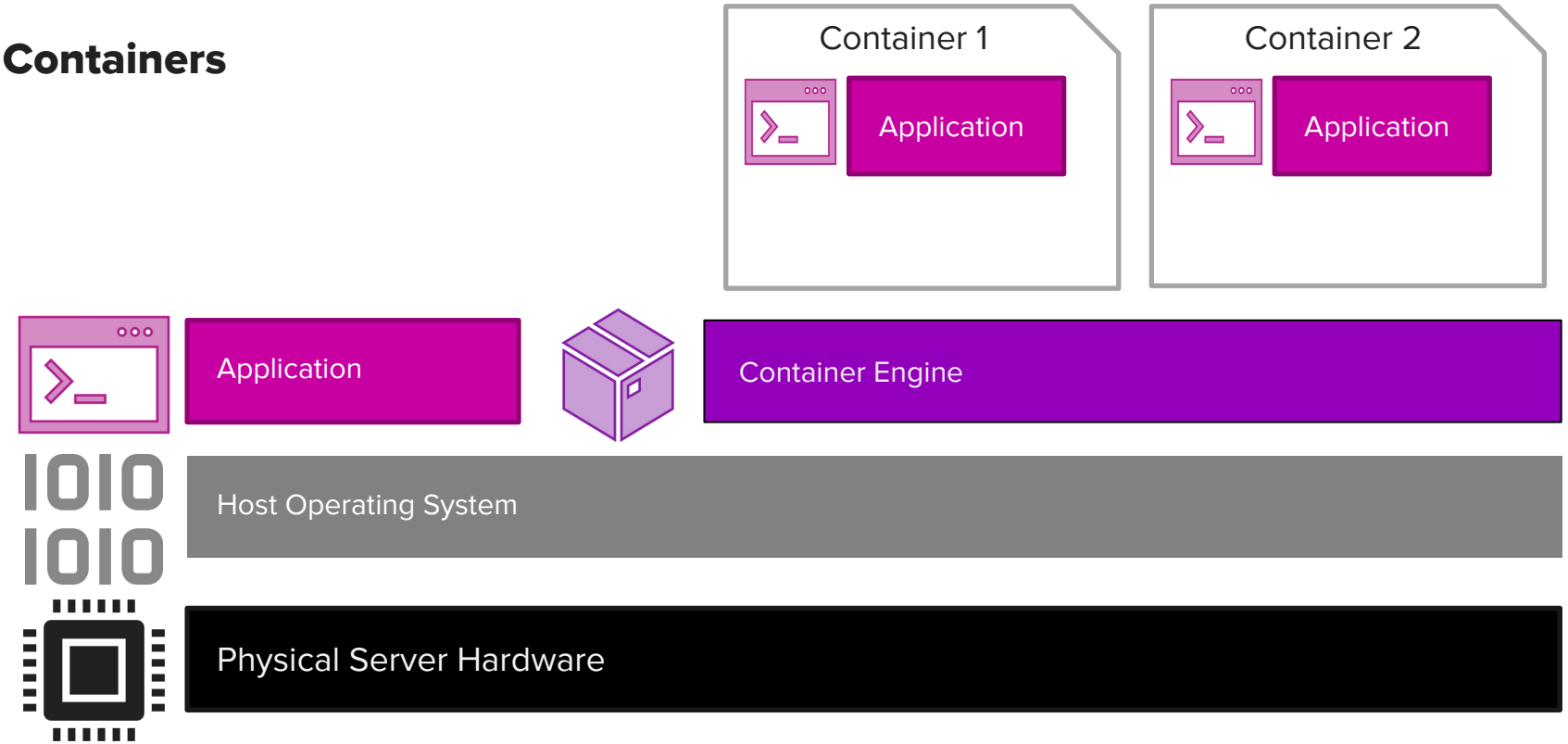
Type 1 Virtualization Architecture



Type 2 Virtualization Architecture

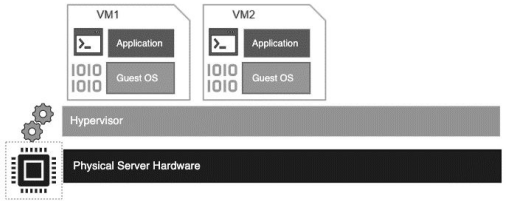


Containers

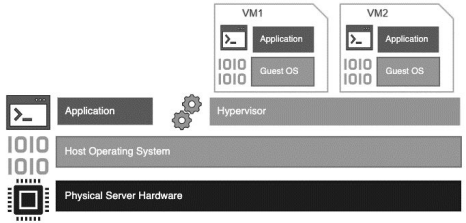

Container 1
Application
Container 2
Application
Application
Container Engine
Host Operating System
Physical Server Hardware

Introduction

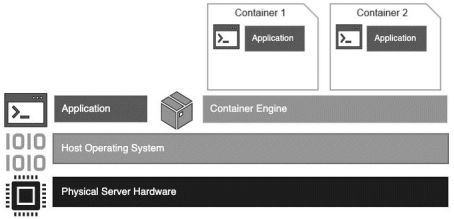
VM Type 1



VM Type 2

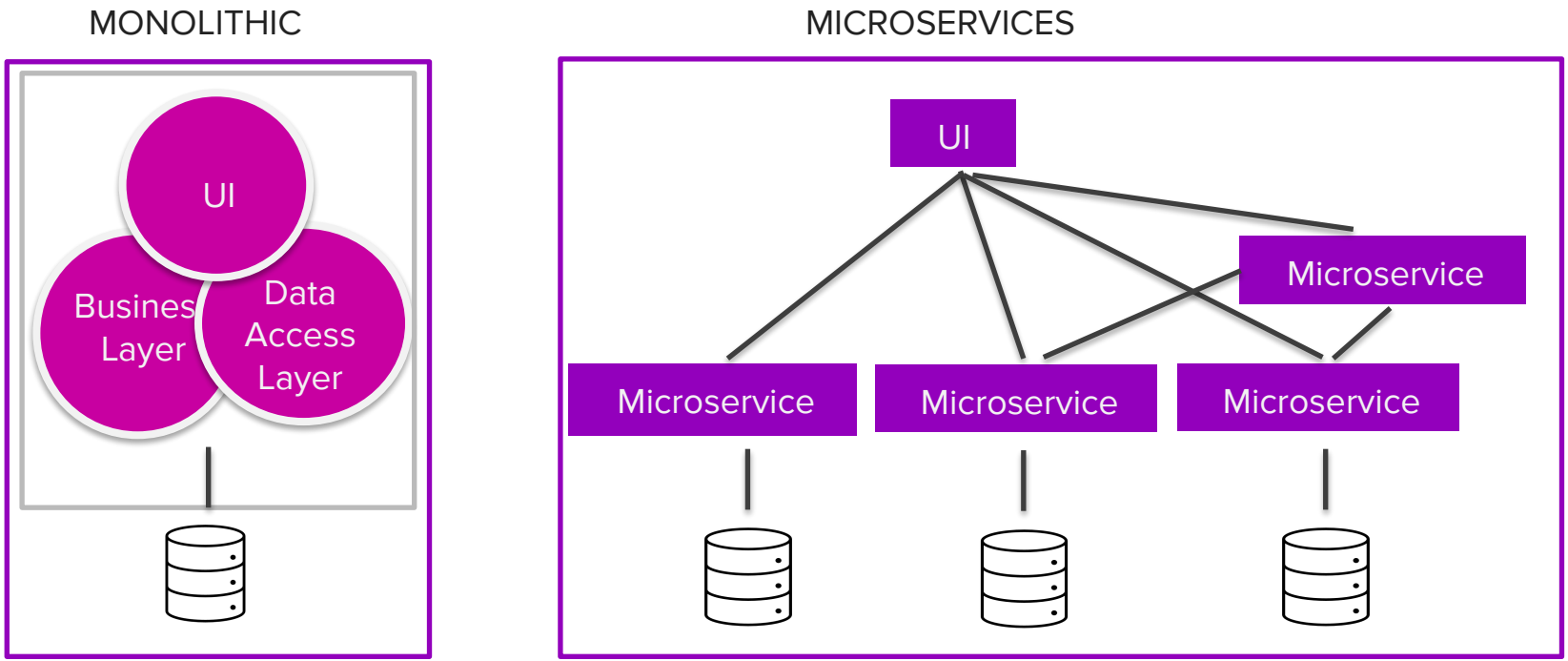


Containers



	VMs Type 1	VMs Type 2	Containers
Storage	Heavy +	Heavy ++	Light
Performance	Limited performance -	Limited performance --	Native performance
OS	own OS	own OS	All containers share the host OS
Virtualization level	Bare metal hypervisor	Hosted hypervisor	ContainerBased/OS virtualization
Startup time	Startup time in minutes	Startup time in minutes ++	Startup time in milliseconds
Memory	Allocates required memory	Allocates required memory	Requires less memory space
Isolation	Fully isolated	Isolated but vulnerable to the host OS	Process-level, possibly less secure
Scope	Datacenter	End-user	All
Horizontal Scalability	Low	Low --	High

Migrating a Monolithic architecture to Microservices

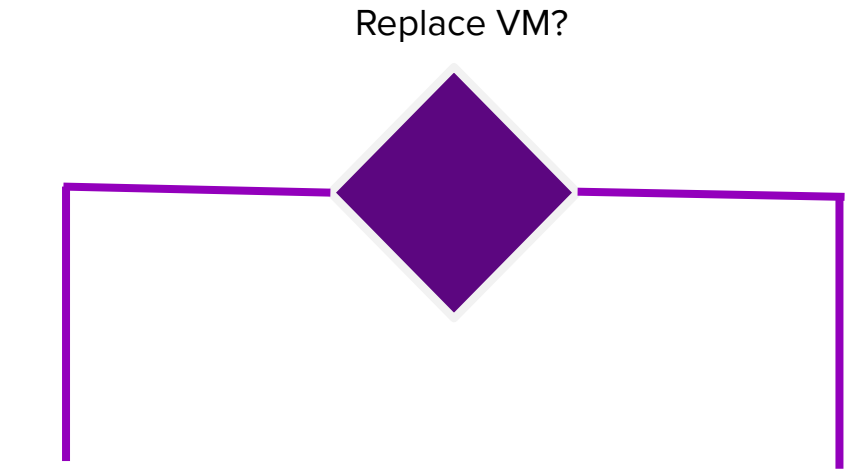


An extra constraint

Input:
You have at least one VM image with an application workload that needs to be integrated.

Required output:
The virtualized application is needed in a unified workflow, side by side with other microservice applications

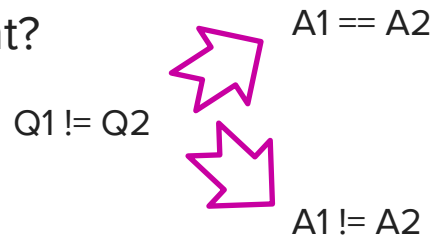
Migration vs Convergence?



Doubled infrastructure?

Two questions added to the context

Question 1: I am a software engineer.
How would I replicate the production environment for an isolated development environment?



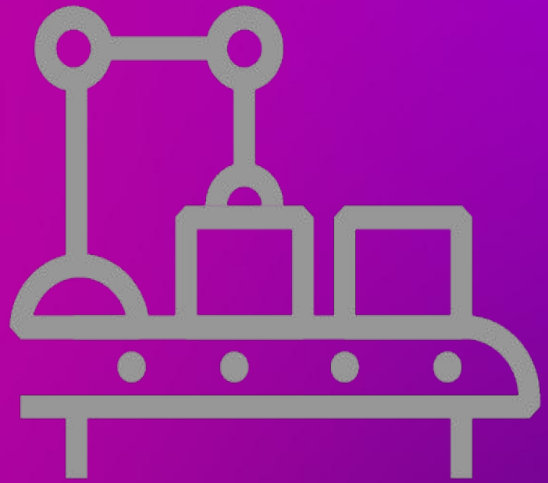
Question2: I am a devops engineer.
How would I setup/secure/provision/manage the production environment in order to be aligned with the development outcome?

Posible answers

- Kubevirt
- Terraform Operator for Kubernetes
- Azure Service Operator

- Virtlet
- Kata Containers

KubeVirt



- From Docker to KubeVirt
- Kubernetes
- Kubernetes: Developers Use Cases
- Why KubeVirt?
- What can I do with KubeVirt?
- Examples

From Docker to KubeVirt

Why containers for Microservices?	• Unit of deployment • Simplified testing • Unit of versioning • Simplified scaling
Container engine?	• Docker
Orchestration?	• Kubernetes
Non-containerizable workload?	• Kubernetes + KubeVirt

Kubernetes, also known as K8s, is an open-source system for automating deployment, scaling, and management of containerized applications.

Key features (from dev perspective)

Service Discovery/Load Balancing
Storage Orchestration
Automate Rollouts/Rollbacks
Self-healing
Secret and Configuration Management
Horizontal Scaling

Kubernetes: Developers Use Cases

- Emulate production locally
- Move from Docker Compose to Kubernetes
- Create end-to-end testing environment
- Ensure application scales properly
- Ensure secrets/configuration are working properly
- Performance testing scenarios
- Workload scenarios (CI/CD and more)
- Learn how to leverage deployment options
- Help DevOps to create resources and solve problems

What can I do with KubeVirt?

- Leverage KubeVirt and Kubernetes to manage virtual machines for impractical-to-containerize apps.
- Combine existing virtualized workloads with new container workloads on the one platform.
- Support development of new microservice applications in containers that interact with existing virtualized applications.

How it solves the problem?

- With a CRD for VM

```
apiVersion: kubevirt.io/v1alpha3
kind: VirtualMachine
metadata:
  creationTimestamp: 2021-07-04T15:03:08Z
  generation: 1
  labels:
    kubevirt.io/os: freebsd
    special: key
  name: testvm2
spec:
  running: true
  template:
    metadata:
      creationTimestamp: null
      labels:
        kubevirt.io/domain: testvm2
        special: key
    spec:
      domain:
        cpu:
          cores: 1
        devices:
          blockMultiQueue: true
          disks:
          - disk:
              bus: sata
            name: disk0
            cache: none
        machine:
          type: q35
        resources:
          requests:
            memory: 2048M
      volumes:
      - name: disk0
        persistentVolumeClaim:
          claimName: upload-datavolume
```

VM Disks?

- mapped to and stored on PersistentVolumes
- disk images can be imported using CDI

```
apiVersion: cdi.kubevirt.io/v1beta1
kind: DataVolume
metadata:
  name: upload-datavolume
spec:
  source:
      upload: {}
  pvc:
    accessModes:
      - ReadWriteOnce
    resources:
      requests:
        storage: 10Gi
```

```
        devices:
          blockMultiQueue: true
          disks:
          - disk:
              bus: sata
            name: disk0
            cache: none
        machine:
          type: q35
        resources:
          requests:
            memory: 2048M
      volumes:
      - name: disk0
        persistentVolumeClaim:
          claimName: upload-datavolume
```

Networking?

- VMs are connected to the regular pod network
- no difference between a VM and pod
- the VM ports are exposed using Services and Routes selection using labels and selector

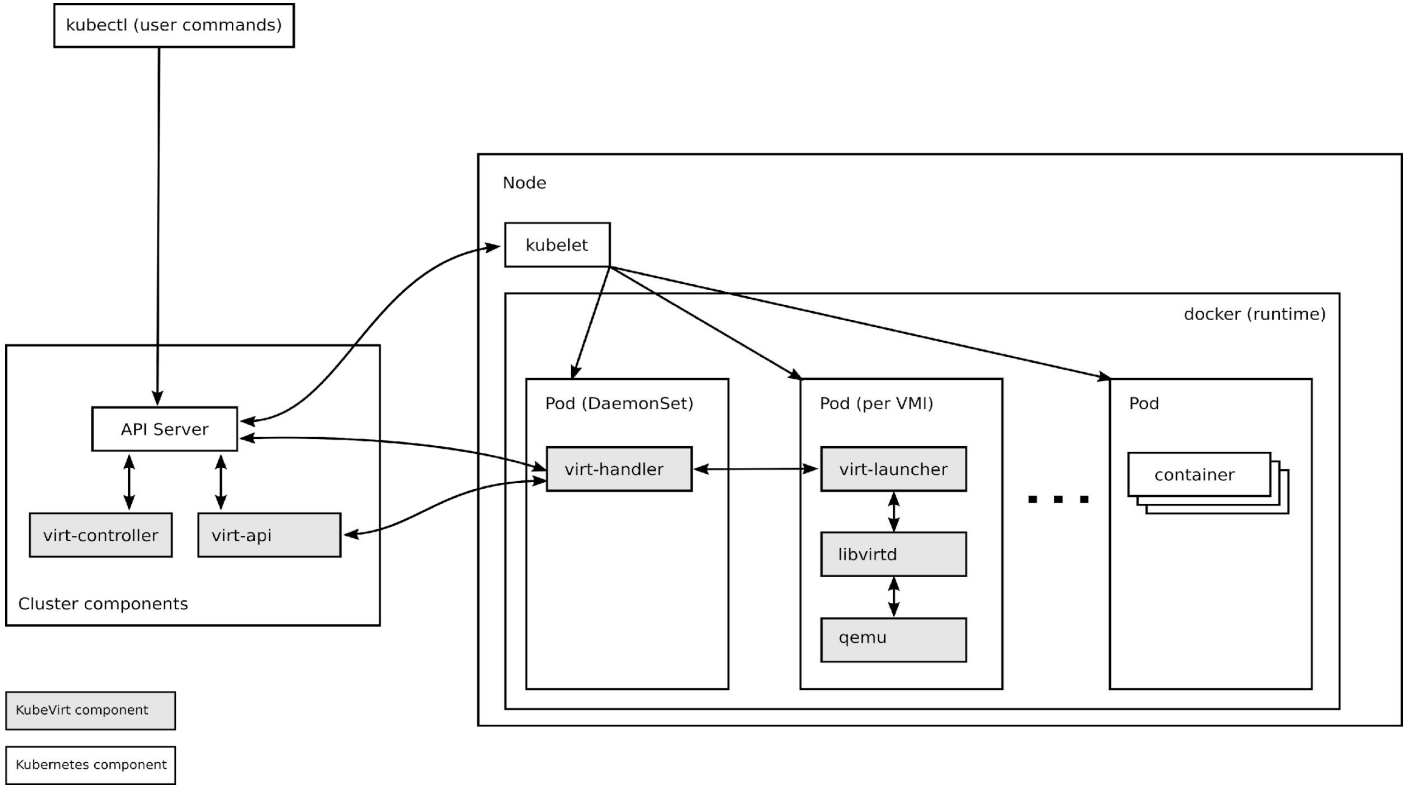
```
apiVersion: v1
kind: Service
metadata:
  name: vm2http
spec:
  externalTrafficPolicy: Cluster
  ports:
  - port: 38883
    name: rdp
    protocol: TCP
    targetPort: 80
  selector:
    kubevirt.io/domain: testvm2
  type: LoadBalancer
```

- macvtap bind mechanism for more advanced needs (v0.35.0)

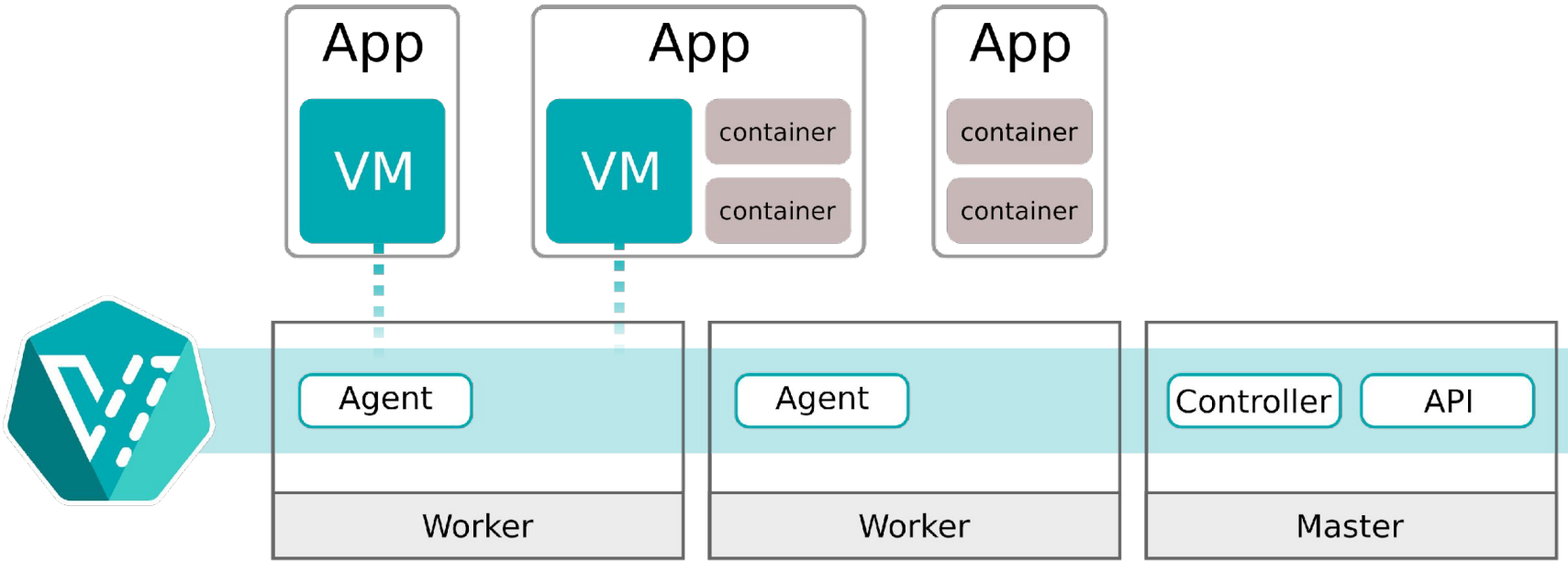
Tools?

- client tool
  - start/stop
  - vnc
  - uploading virtual machine disk images
  - live migration
- can be used as
  - kubectl plugin (kubectl virt vnc testvm2)
  - standalone (virtctl vnc testvm2)

Kubevirt: Architecture



Kubevirt: Simplified architecture



Vagrant

- Bonus tool

Vagrant

HashiCorp
Vagrant

Development Environments Made Easy

What it is?
Relation to KubeVirt:
- as KubeVirt VM controlling tool
- as KubeVirt image builder

https://www.vagrantup.com/

HashiCorp
Vagrant

Development Environments Made Easy

UNIFIED WORKFLOW

Simple and Powerful

Provides the same, easy workflow regardless of your role as a developer, operator, or designer. It leverages a declarative configuration file which describes all your software requirements, packages, operating system configuration, users, and more.

https://www.vagrantup.com/

HashiCorp
Vagrant

Development Environments Made Easy

ENFORCE CONSISTENCY

Production Parity

- It aims to mirror production environments by providing the same operating system, packages, users, and configurations, all while giving users the flexibility to use their favorite editor, IDE, and browser.
- integrates with your existing configuration management tooling like Ansible, Chef, Docker, Puppet or Salt, so you can use the same scripts to configure Vagrant as production.

Vagrant

https://www.vagrantup.com/

HashiCorp
Vagrant

Development Environments Made Easy

CROSS-PLATFORM

Works where you work

- Mac
- Linux
- Windows

Demo

- KubeVirt using AKS infrastructure

https://github.com/matei-tm/inversion-of-containers

Conclusion?

The VM is dead, long live the VM!

Q&A

Documentation Sources

- https://kubernetes.io/docs/tutorials/kubernetes-basics/
- https://www.vagrantup.com/
- https://kubevirt.io/
- https://kubevirt.io/2019/Kubevirt-vagrant-provider.html
- https://en.wikipedia.org/wiki/Microservices
- https://www.ibm.com/cloud/blog/soa-vs-microservices
- https://microservices.io/
- Kubernetes for Developers: Core Concepts (Pluralsight course by Dan Wahlin)
- Deploying ASP.NET Core Microservices Using Kubernetes and AKS (Pluralsight course by Marcel de Vries)
- Alessandro Vozza https://medium.com/cooking-with-azure/using-kubevirt-in-azure-kubernetes-service-part-3-windows-vm-363d6b653d7

Cognizant Softvision

Thank you!

Mircea-Tiberiu MATEI | September 13, 2021

PROGRAMMERS'
WEEK 2021
Cognizant Softvision