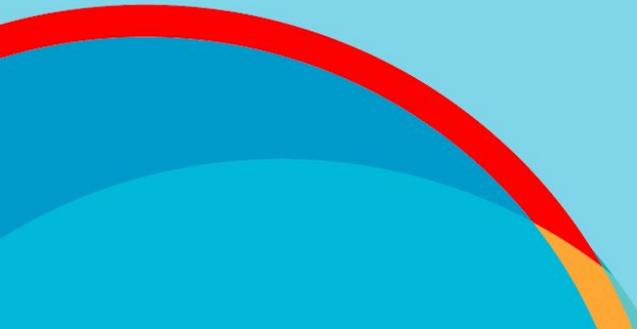
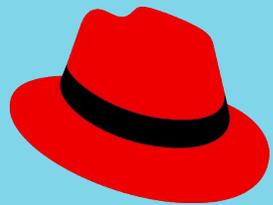




Red Hat
Summit

Connect





Red Hat

Breaking down monolithic applications with OpenShift Virtualisation

Matt Kimberley

Specialist Solution Architect

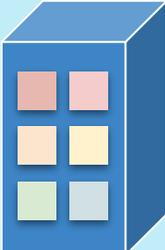
Red Hat

Lets recap...

Migrate and then Modernise

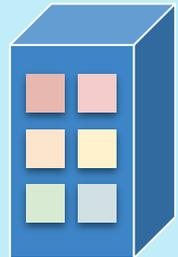
1 Deployed on Server

Existing Monolith / Application



2 Migration of service to  Red Hat OpenShift

OpenShift

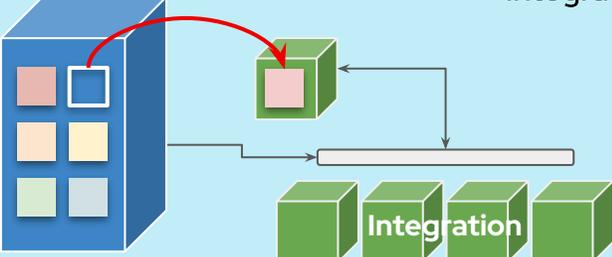


 Red Hat OpenShift Container Platform

Automated delivery & better Ops

3 Breaking down the monolith  Red Hat OpenShift

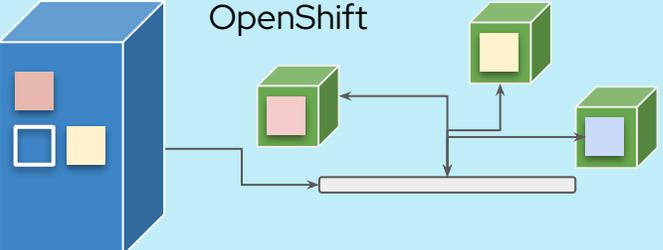
OpenShift  Red Hat Integration



Now easier to modernise

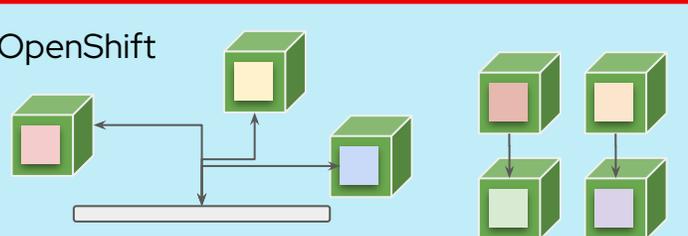
4 Continued incremental improvement  Red Hat OpenShift

OpenShift

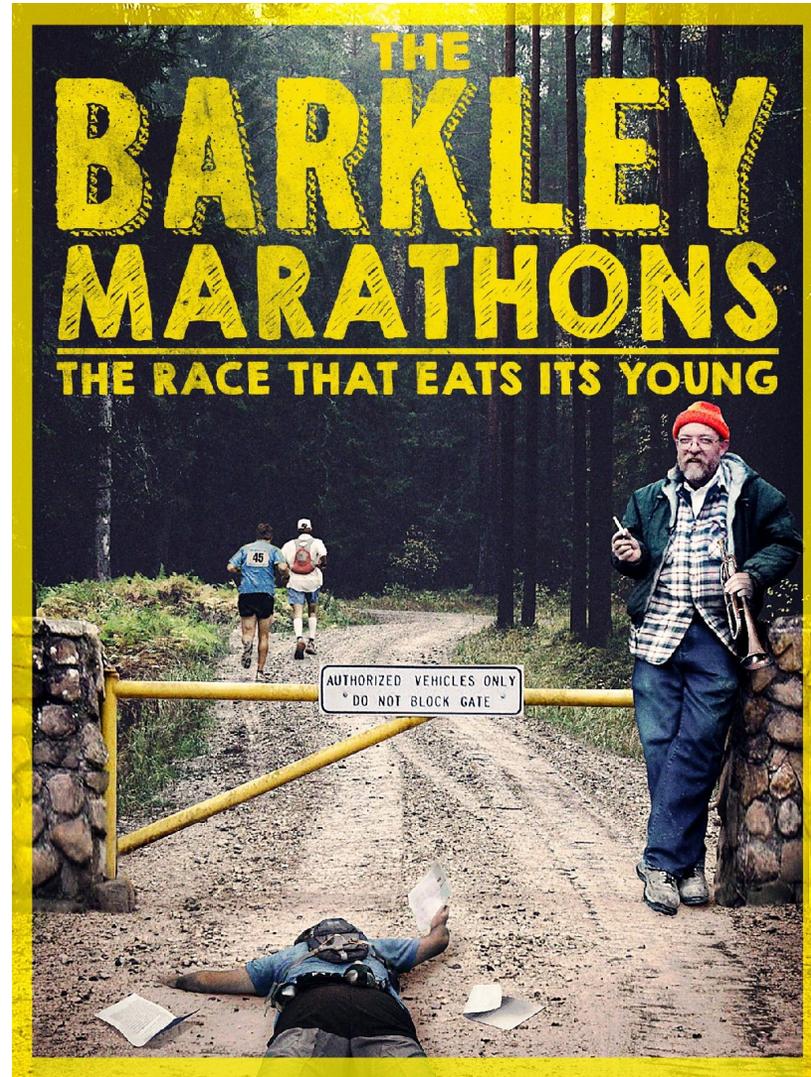


5 Legacy monolith retired  Red Hat OpenShift

OpenShift



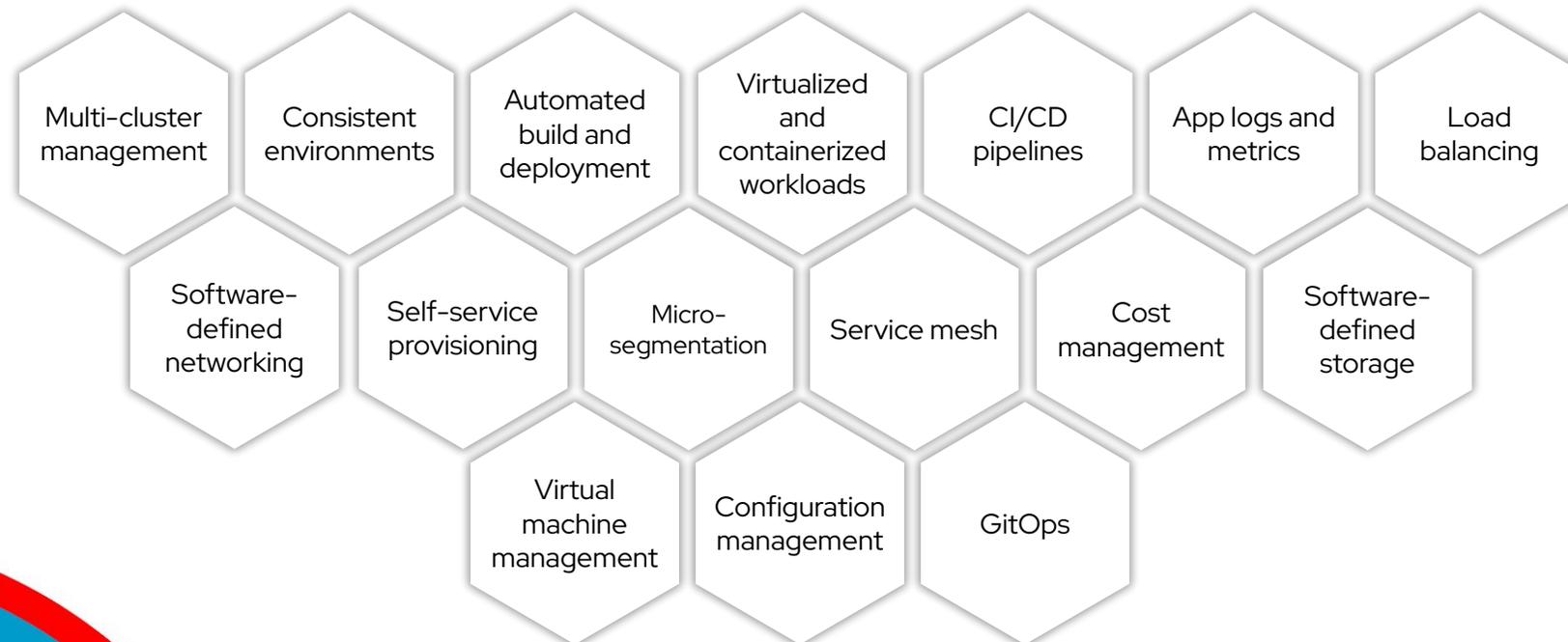
Getting there can feel like this...



- More than 40 people started the Barkley Marathon in 2023
- Only 3 Finished.

OpenShift

A Modern application platform with comprehensive lifecycle and infrastructure management



OpenShift Virtualisation

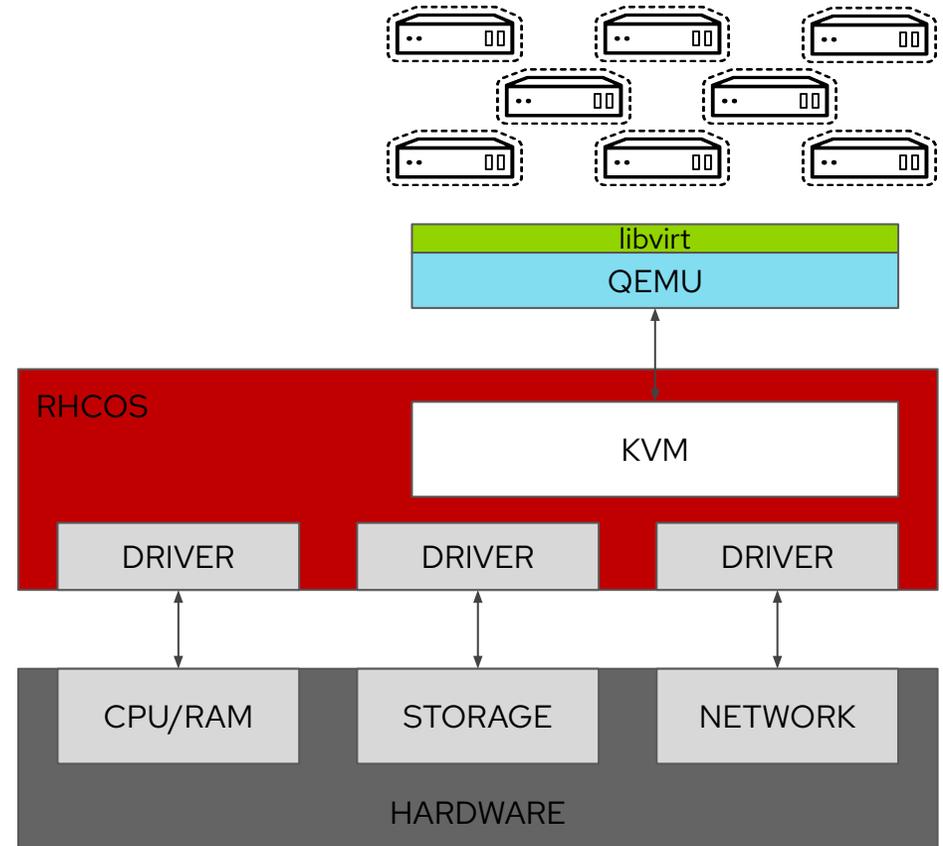
OpenShift Virtualisation

- Virtual machines
 - Run on the KVM Hypervisor on an OpenShift worker node
 - Managed by orchestration pod
- Scheduled, deployed, and managed by Kubernetes
 - Provides high availability in the event of OCP node outage
- Integrated with OpenShift resources and services
 - Traditional Pod-like SDN connectivity
 - Connectivity to external VLAN and other networks via multus
 - Persistent storage delivers storage to virtual machines



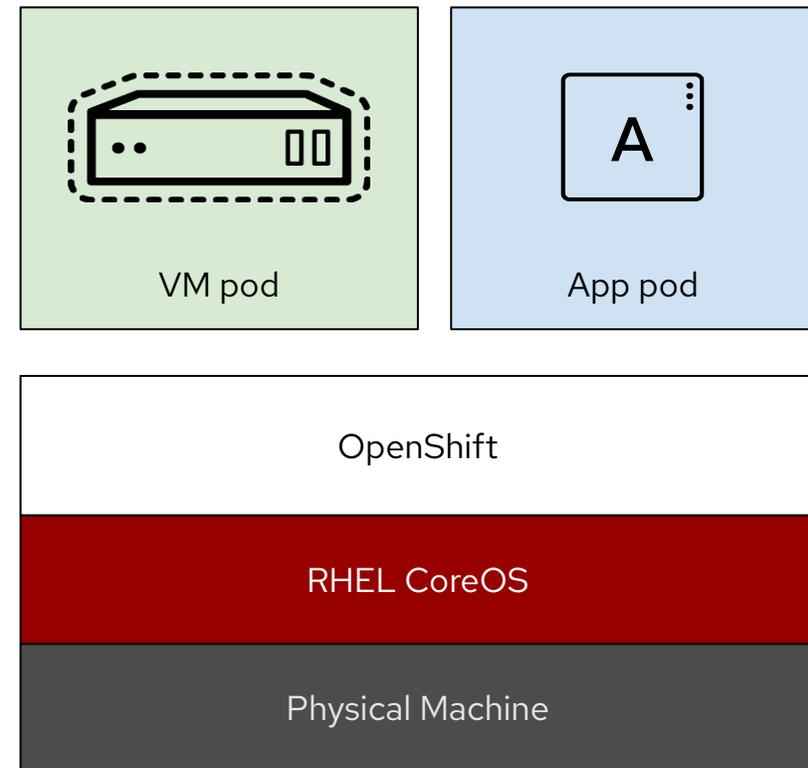
KVM

- KVM is well established long serving hypervisor
- KVM provides hardware Virtualisation
 - Used by Red Hat Virtualisation, Red Hat OpenStack Platform, and RHEL and others.
 - Operates on the OpenShift worker nodes
- QEMU provides hardware emulation
- libvirt provides a management abstraction layer and API for interaction with the virtual machines



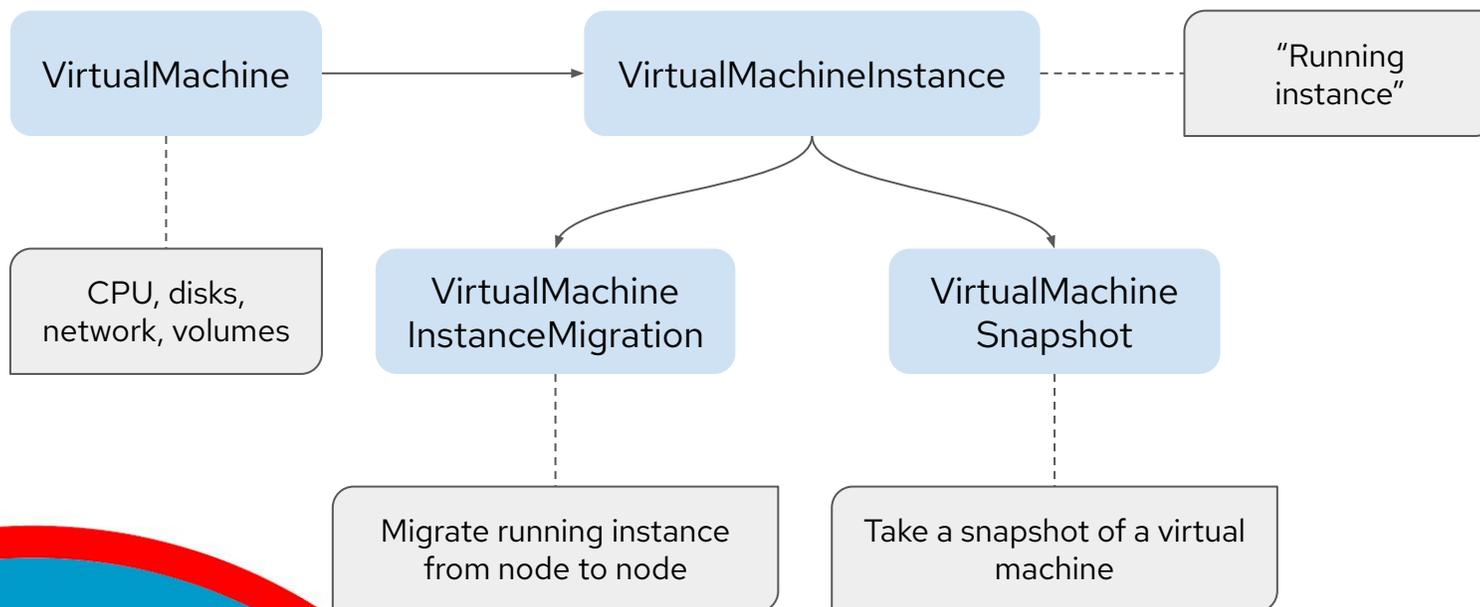
Virtual machines in a container world

- Transition application components (which can't be directly containerized) into an OpenShift environment
 - Integrates directly into OpenShift
 - Follows Kubernetes paradigms:
 - Container Networking Interface (CNI)
 - Container Storage Interface (CSI)
 - Custom Resource Definitions (CRD, CR)
- Schedule, connect, and consume VM resources as container-native



Virtualisation native to Kubernetes

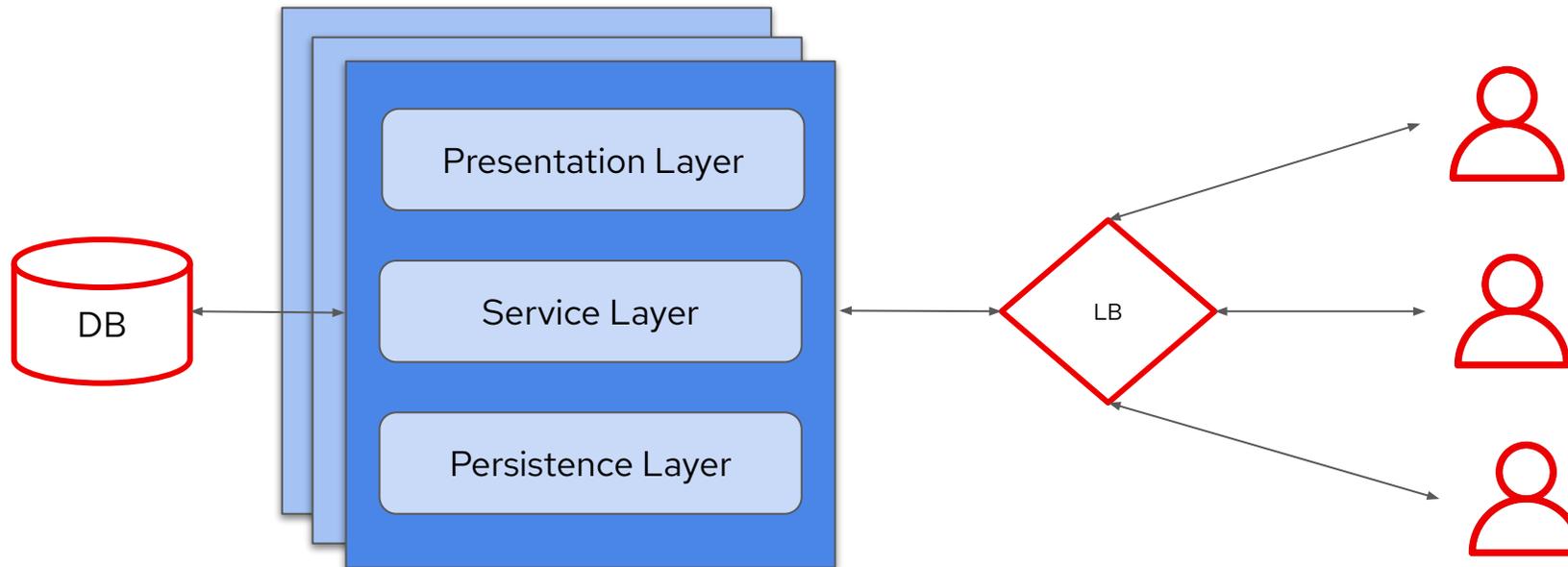
New CustomResourceDefinitions (CRDs) for native VM integration



```
apiVersion: kubevirt.io/v1alpha3
kind: VirtualMachine
metadata:
  labels:
    app: demo
    flavor.template.kubevirt.io/small: "true"
  name: rhel
spec:
  dataVolumeTemplates:
  - apiVersion: cdi.kubevirt.io/v1alpha1
    kind: DataVolume
    metadata:
      creationTimestamp: null
      name: rhel-rootdisk
    spec:
      pvc:
        accessModes:
        - ReadWriteMany
        resources:
          requests:
            storage: 20Gi
        storageClassName: managed-nfs-storage
        volumeMode: Filesystem
```

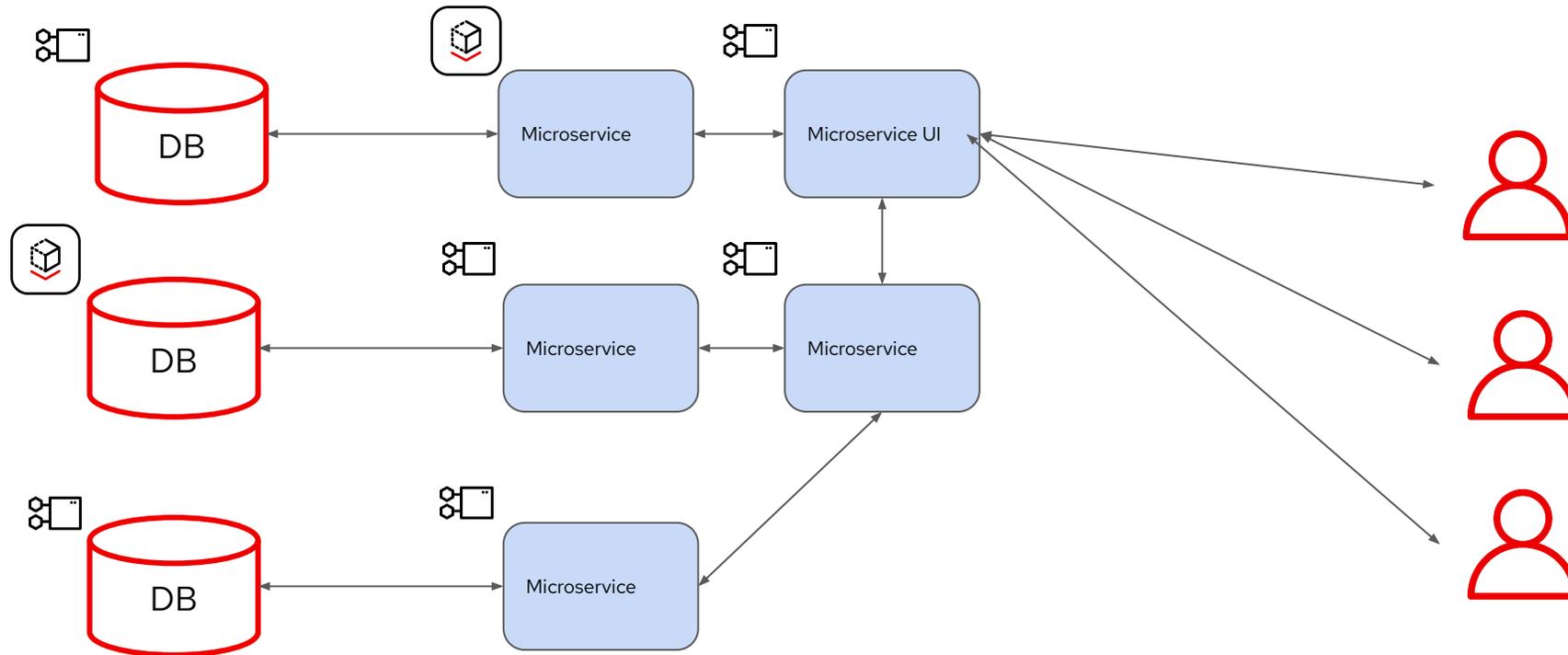
Changing our mindset

Monolithic Applications



- High Availability typically provided at VM level
- Routing and LB usually provided by an appliance
- Relatively long build and release cycles
- Complex execution workflows

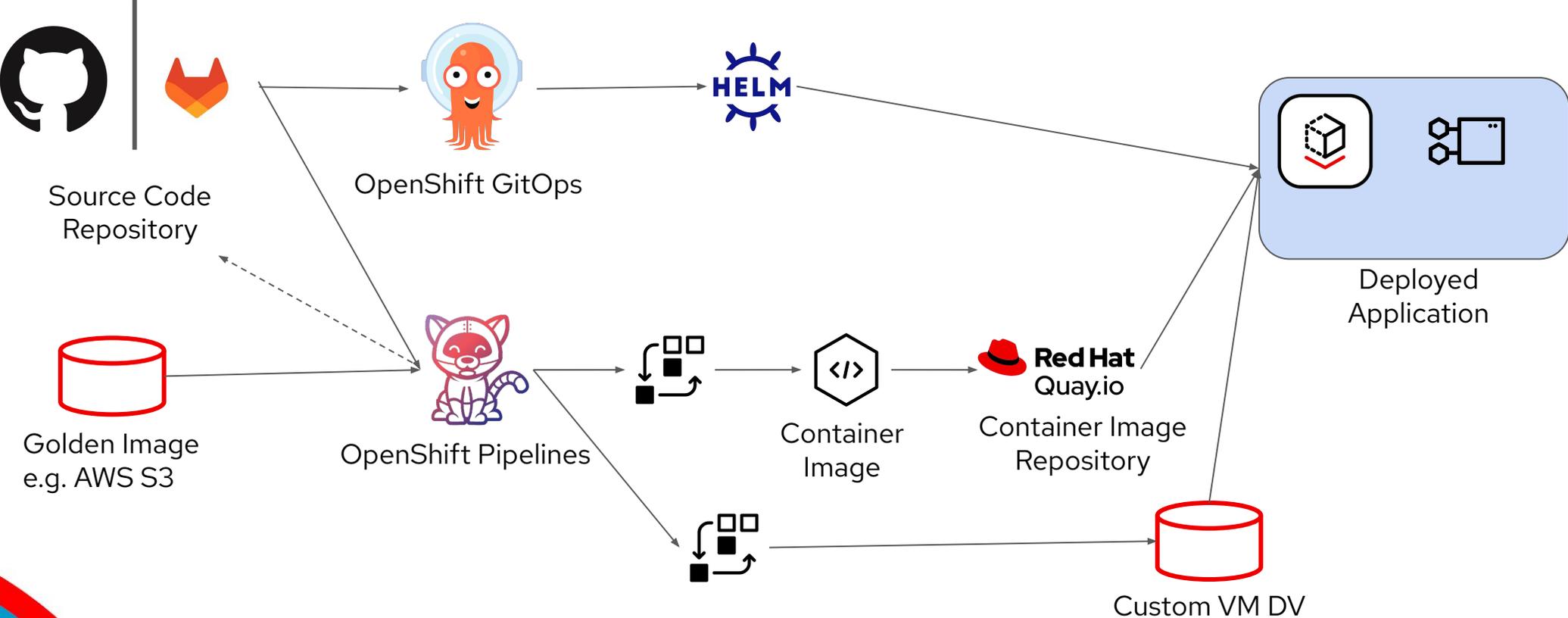
Microservices on OpenShift with VMs



- Availability at the Pod level, and node level
- Routing and LB controlled by routes and services
- Pod and VM deployment is quick and self serving in nature

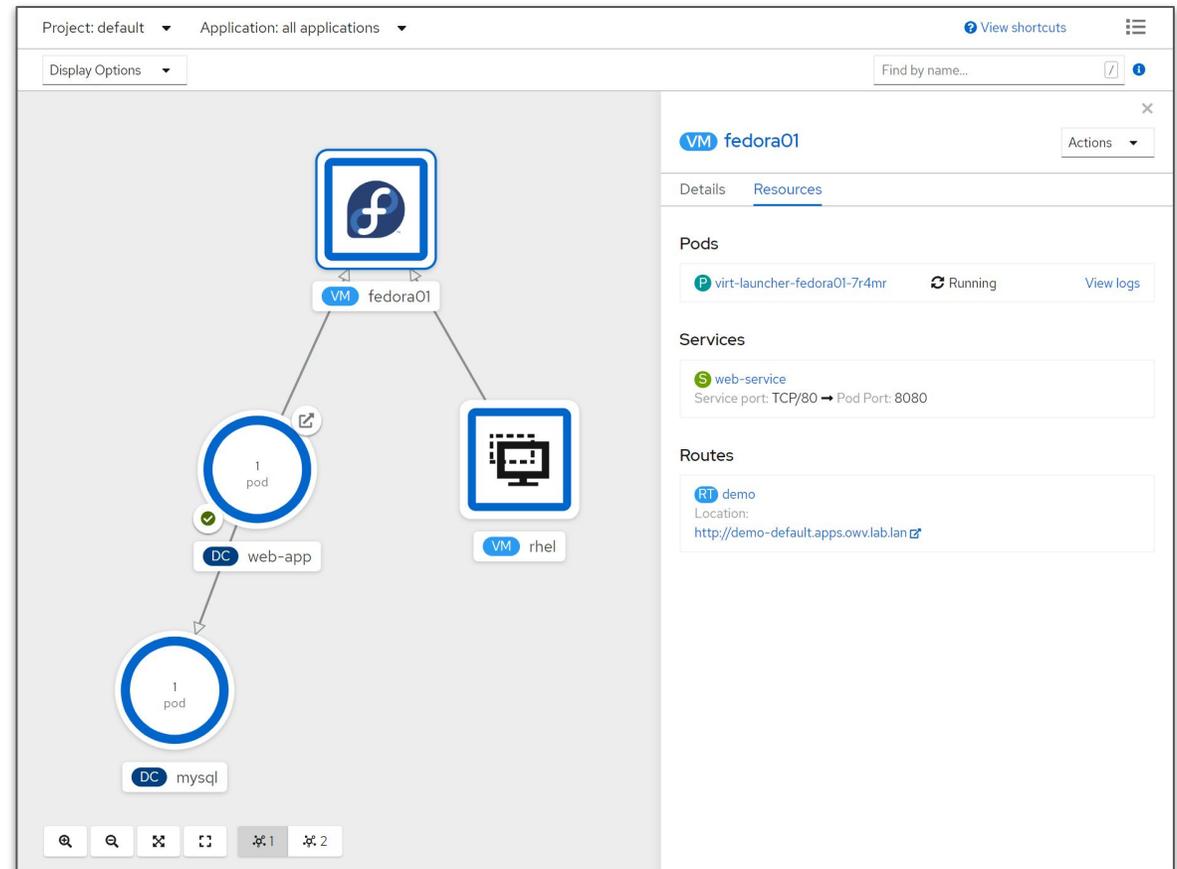
Cloud Native VM Builds

- Application Configuration
- Application Source Code
- Container Image files
- Virtual Machine Templates



Using VMs and containers together

- Virtual machines connected to pod networks are accessible using standard Kubernetes methods:
 - Service
 - Route
 - Ingress
- Network policies apply to VMs (via pods) the same as application pods
- VM-to-pod, and vice-versa, communication happens over SDN or ingress depending on network connectivity



From the core to
the edge

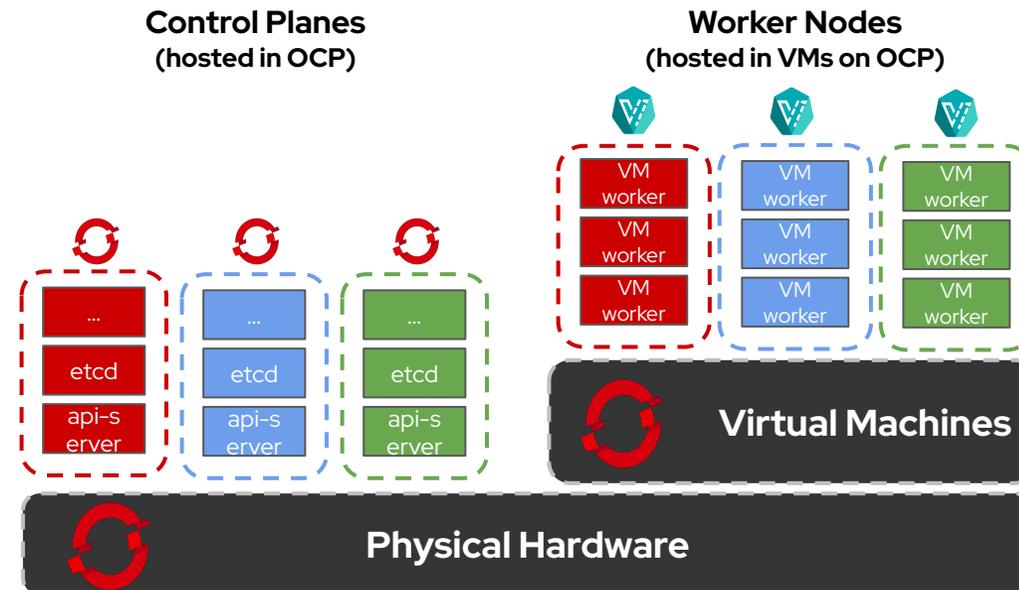
Consolidation of OpenShift Clusters with Hosted Control Planes

Increase Utilization of Infrastructure

- Consolidate multiple control planes to reduce unused and underutilized infrastructure
- Increase bare metal node utilization by hosting virtual worker nodes for multiple clusters

Reduce Dependency on Legacy Virtualization

- Eliminate the need to have legacy hypervisor layer to host your containerized infrastructure
- Underlying virtualization layer is included with hosted OpenShift cluster entitlements (no separate licensing needed)



Demo

Our Environment

Search [Option+S]

Instances (4) [Info](#)

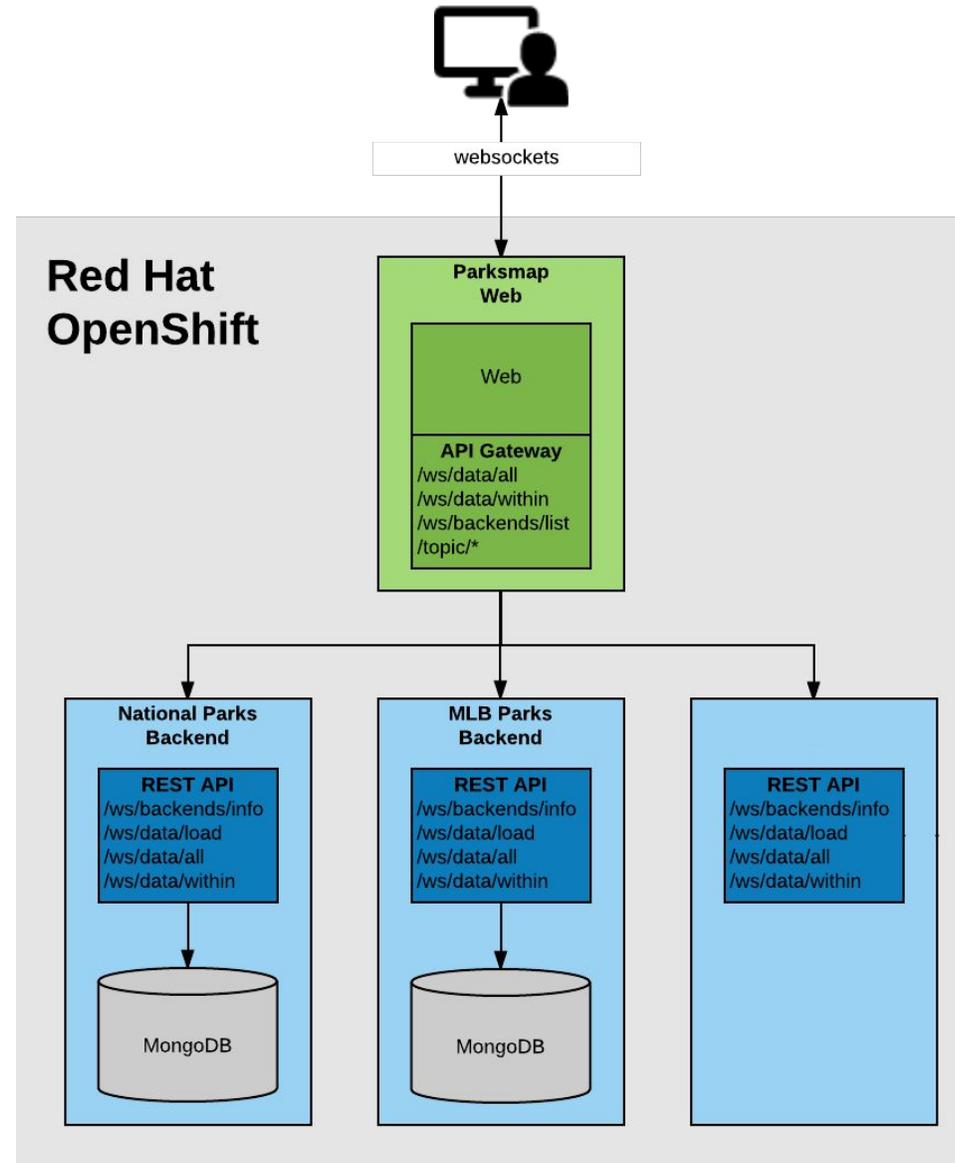
Find Instance by attribute or tag (case-sensitive)

Instance state = running X

Clear filters

<input type="checkbox"/>	Name ✎	Instance ID	Instance state ▼	Instance type ▼	Status check	Alarm status	Availability Zone ▼
<input type="checkbox"/>	sc23-j8wx8-master-0	i-0b0c0292dc95a14ed	Running + -	m6i.xlarge	2/2 checks passed	No alarms +	eu-west-2a
<input type="checkbox"/>	metal-cc72c	i-0d17ef16ca196f1d9	Running + -	m5.metal	2/2 checks passed	No alarms +	eu-west-2a
<input type="checkbox"/>	sc23-j8wx8-master-1	i-0aa9c14ad89a15ac9	Running + -	m6i.xlarge	2/2 checks passed	No alarms +	eu-west-2b
<input type="checkbox"/>	sc23-j8wx8-master-2	i-093894a2f716ee0f2	Running + -	m6i.xlarge	2/2 checks passed	No alarms +	eu-west-2c

Our example application



Let explore

- The OpenShift Virtualisation Operator
- Storage
- Networking
- Creating VMs from Templates
- How Pods Interact with VMs

Q&A

Further Information:

<https://www.redhat.com/en/technologies/cloud-computing/openshift/virtualization>



matt.kimberley@redhat.com



Red Hat
Summit

Thank you



linkedin.com/company/red-hat



facebook.com/redhatinc



youtube.com/user/RedHatVideos



twitter.com/RedHat