



//Exploring OpenShift 4 Event

- OCP 4 -- What's New - Phil
- OCP 4 -- On Prem - A Briefing - Aaren
- OCP 4 -- Deploy to GCP & Service Mesh - Paul
- OCP 4 -- Migrations - Phil & Paul

# What's New (and exciting)

# //OCP 4 Where we at?

IPI or Full Stack Automation

UPI or Pre-existing Infrastructure

 **OPENSIFT<sup>®</sup>**  
by Red Hat<sup>®</sup>  
4.1<sup>\*</sup>



 **OPENSIFT<sup>®</sup>**  
by Red Hat<sup>®</sup>  
4.2



 **OPENSIFT<sup>®</sup>**  
by Red Hat<sup>®</sup>  
4.3 (tentative)



*\* Requires Internet connectivity; support for cluster proxy & disconnected installation/updating not planned until 4.2*

*\*\* On qualified hardware stack*

# //OCP 4 Where we at?

## Q2 CY2019 OpenShift 4.1

DEV

- Serverless w/ Knative Dev Preview
- OpenShift Pipelines (Tekton) Dev Preview
- CodeReady Workspaces
- CodeReady Containers Alpha
- Developer CLI (odo) Beta

APP

- OperatorHub
- Operator Lifecycle Manager
- Service Mesh (~2 month after)

PLATFORM

- Kubernetes 1.12 with CRI-O runtime
- RHEL CoreOS, RHEL7
- Automated Installer for AWS
- Pre-existing Infra Installer for Bare Metal, VMware, AWS
- Automated, one-click updates
- Multus (Kubernetes multi-network)
- Quay v3

HOSTED

- Universal Hybrid Cloud (UHC)
- OCP Cluster Subscription Management
- OpenShift on Azure by MSFT and RHT
- OpenShift Dedicated consumption pricing

## Q3 CY2019 OpenShift 4.2

DEV

- Developer Console GA
- Serverless w/ Knative Tech Preview
- OpenShift Pipelines (Tekton) Tech Preview
- CodeReady Containers GA
- Developer CLI (odo) GA

APP

- GPU metering
- OperatorHub Enhancements
- Operator Deployment Field Forms
- Application Binding with Operators
- Application Migration Console

PLATFORM

- Kubernetes 1.14 w/ CRI-O runtime
- Disconnected Install and Update
- Automated Installer for Azure, OSP, GCP
- OVN Tech Preview
- FIPS
- Federation Workload API
- Automated App cert rotation
- OpenShift Container Storage 4.2

HOSTED

- UHC Multi-Cluster deployment
- Proactive Support Operator

## Q4 CY2019 OpenShift 4.3

DEV

- Serverless w/ Knative GA
- OpenShift Pipelines (Tekton) GA

APP

- Metering for Services
- Windows Containers

PLATFORM

- Kubernetes 1.15 w/ CRI-O runtime
- Automated Installer for IBM Cloud, Alibaba, RHV, Bare Metal Hardware Appliance
- Pre-existing Infra Installer for Azure, OSP, GCP
- OVN GA w/ Windows Networking Integration

HOSTED

- UHC Subscription Mgmt Consumption Improvements

- Red Hat Enterprise Linux CoreOS (RHCOS) for the control plane (masters)
- Operators, Operators, Operators! - Using kubernetes to manage kubernetes
- CRI-O
- <https://cloud.redhat.com>
- ~~Minishift~~ Code Ready Containers

- MachineSet - Like a daemonset for Machines
- Chargeback - showback?
- Code Ready Workspace
- Cluster Updates
- API Explorer
- odo



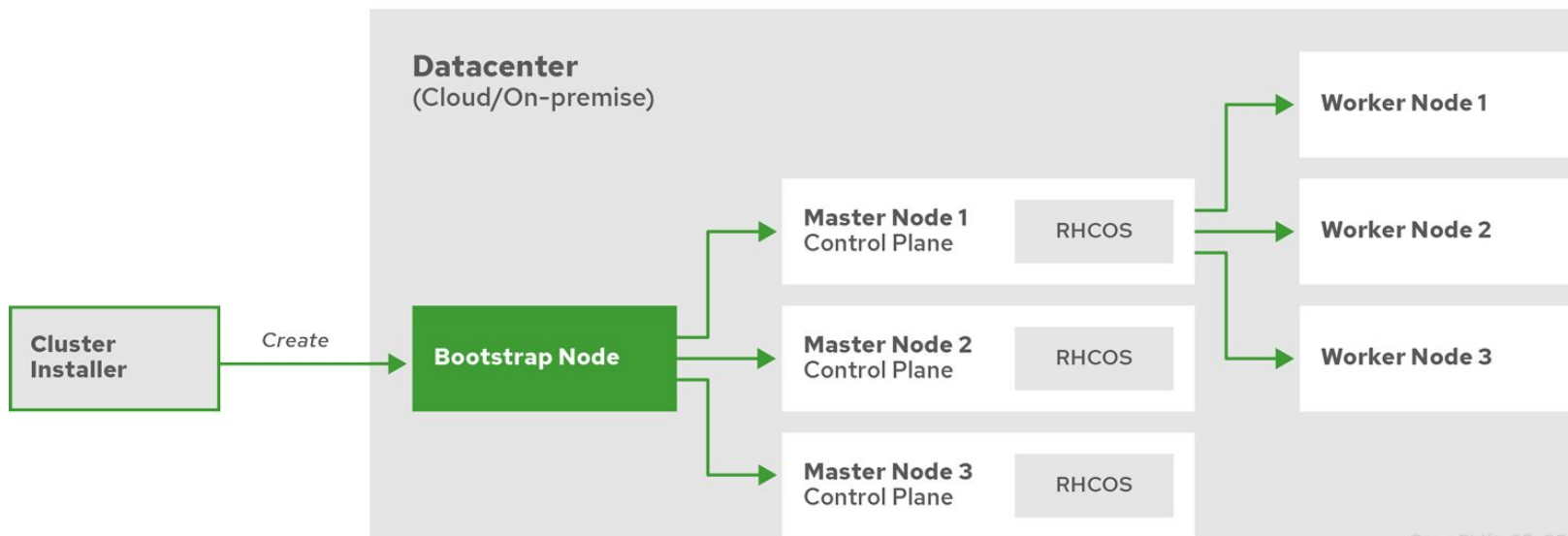
# //OCP 4 What's New - Resources?

## Resources

- [docs.openshift.com/container-platform/4.2/release\\_notes/ocp-4-2-release-notes.html](https://docs.openshift.com/container-platform/4.2/release_notes/ocp-4-2-release-notes.html)
- <https://try.openshift.com>
- <https://code-ready.github.io/crc/>
- <https://blog.openshift.com/enabling-openshift-4-clusters-to-stop-and-resume-cluster-vms/>
- <https://github.com/openshift/odo>



- What do we need?
  - Azure Account   DNS   Pull Secret   ServiceAccount (SP)   oc & kubectl
- Generate the install config
- Go!



- Can we customize the deploy?
- Can I shut it down with 24hrs of initial deploy?

Parameter	Description	Values
<code>machines.platform.azure.type</code>	The Azure VM instance type.	VMs that use Windows or Linux as the operating system. See the <a href="#">Guest operating systems supported on Azure Stack</a> in the Azure documentation.  <code>machines.platform.azure.osDisk.diskSizeGB</code>
The Azure disk size for the VM.	Integer that represents the size of the disk in GB, for example <b>512</b> . The minimum supported disk size is <b>120</b> .	<code>platform.azure.baseDomainResourceGroupName</code>
The name of the resource group that contains the DNS zone for your base domain.	String, for example <code>production_cluster</code> .	<code>platform.azure.region</code>

# //OCP 4 on Azure - Resources

- <https://github.com/openshift/installer/tree/master/docs/user/azure>
- <https://github.com/openshift/installer/blob/master/docs/user/customization.md#examples>
- <https://github.com/openshift/installer/blob/master/docs/user/azure/customization.md>
- <https://blog.openshift.com/enabling-openshift-4-clusters-to-stop-and-resume-cluster-vms/>

- Tech Preview in 4.2 & GA 4.3
- Its an Operator! – interact with pipeline using oc/kubectl
- Portable across different K8s platforms
- Containers are the building blocks
- Decoupled - Tasks can be run in isolation



# TEKTON

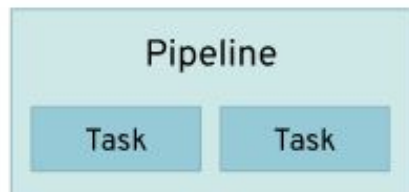
**A Kubernetes-native pipeline resource**

# //OpenShift Pipelines

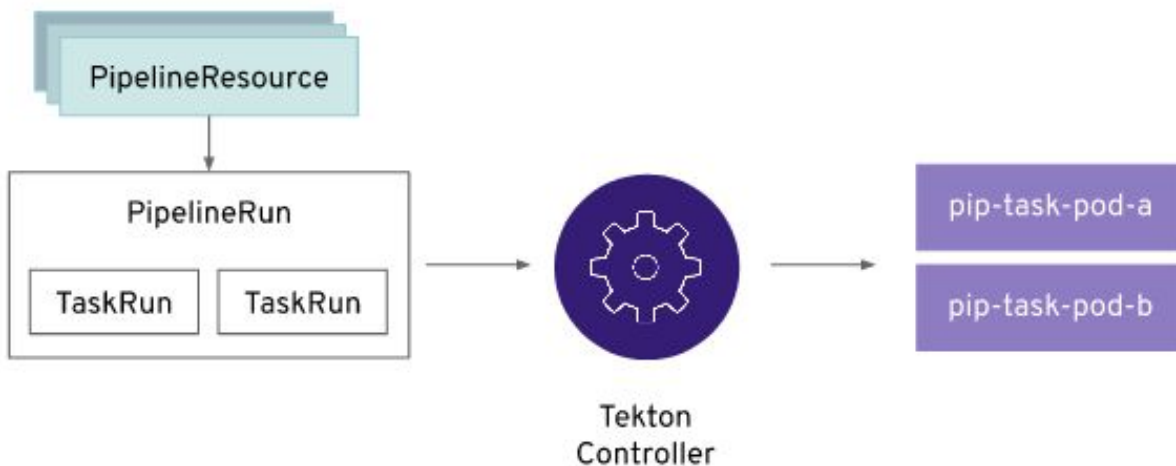
- Custom Resources:

- Tasks    Pipelines    TaskRuns    PipelineRuns    PipelineResources

Define pipeline



Invoke pipelines





Demo!

## Resources

- <https://github.com/tektoncd/catalog>
- <https://github.com/tektoncd/pipeline>
- <https://github.com/openshift/pipelines-catalog>
- <https://github.com/openshift/pipelines-tutorial>
- <https://github.com/openshift/pipelines-docs>



The background of the slide is a close-up photograph of a brick wall. The bricks are reddish-brown and arranged in a standard running bond pattern. On the right side of the image, there is a small, arched window. The window is covered with a dark, metal grate that has a grid-like pattern. The overall lighting is somewhat dim, giving the image a slightly moody or industrial feel.

# On-Prem - A Briefing



## Current State

- UPI is currently best-case











- RHV, KVM, or VMWare

- 4.1 for support/ stability

- 4.2 are features & nightlies  
(\*risk)

- 4.3+ .. ?

## UPI or Pre-existing Infrastructure

	   Bare Metal
	
	  

## Current State

- UPI is what we get for now
- RHV, KVM, or VMWare
- 4.1 for support/ stability
- 4.2 are features & nightlies (\*risk)

## Experience

Where	What	How	Details / Why
oVirt / RHV	OCP 4.1	UPI, Bare Metal	Automate your own infra; <b>ISO install CoreOS for ignition</b> or pxe-boot ignition for CoreOS
KVM			advise : automate your infra + extra requirements (* next slide)
VMware			

Remember, UPI stands for

~~U~~ Provision Indefinitely

~~U~~ nidentified Probable Interface

~~U~~ ndiscovered Personal Identity

User Provisioned Infrastructure



Remember, UPI stands for

~~U Provision Indefinitely~~

~~Unidentified Probable Interface~~

~~Undiscovered Personal Identity~~

~~User Provisioned Infrastructure~~

Automate that stuff!



## Details you need

Essentially → know your hypervisor and RTF(OCP4)M..

1. Key minimal **requirements** on VMs
  - a. Bootstrap node: 4vcpu, 8GB RAM, 120GB disk
  - b. Masters/workers: 8vcpu, 12-16GB RAM, 120GB disk
2. Pre-downloaded, binaries, packages, tgz, where possible (speed)
3. Ignition needs internet!
4. **Cert** and SSH key for the installer to use, from [try.openshift.com](https://try.openshift.com)

## Details you need (cont'd)

Essentially → gather all the ducks in a row and RTF(OCP4)M..

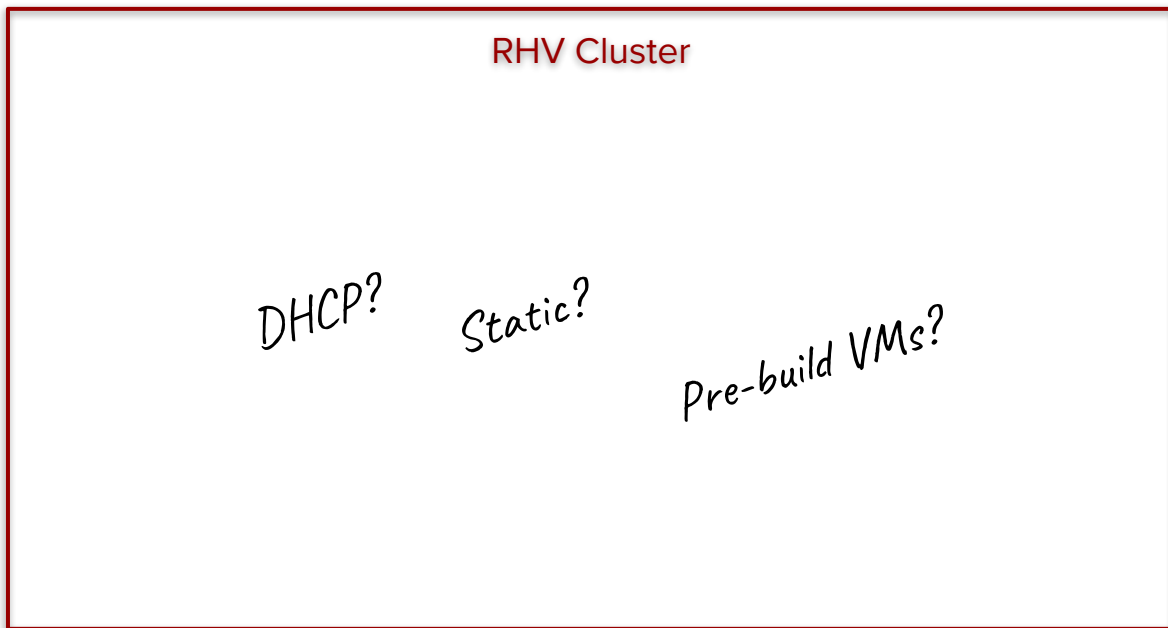
5. Bare-metal openshift installer files
6. Openshift 4 client package
7. Bare-metal BIOS/UEFI file
8. A triple-check of all the above
9. Some DHCP and DNS control

And all on-prem deploys have “subtle nuances” per hypervisor chosen

## **Sandbox**

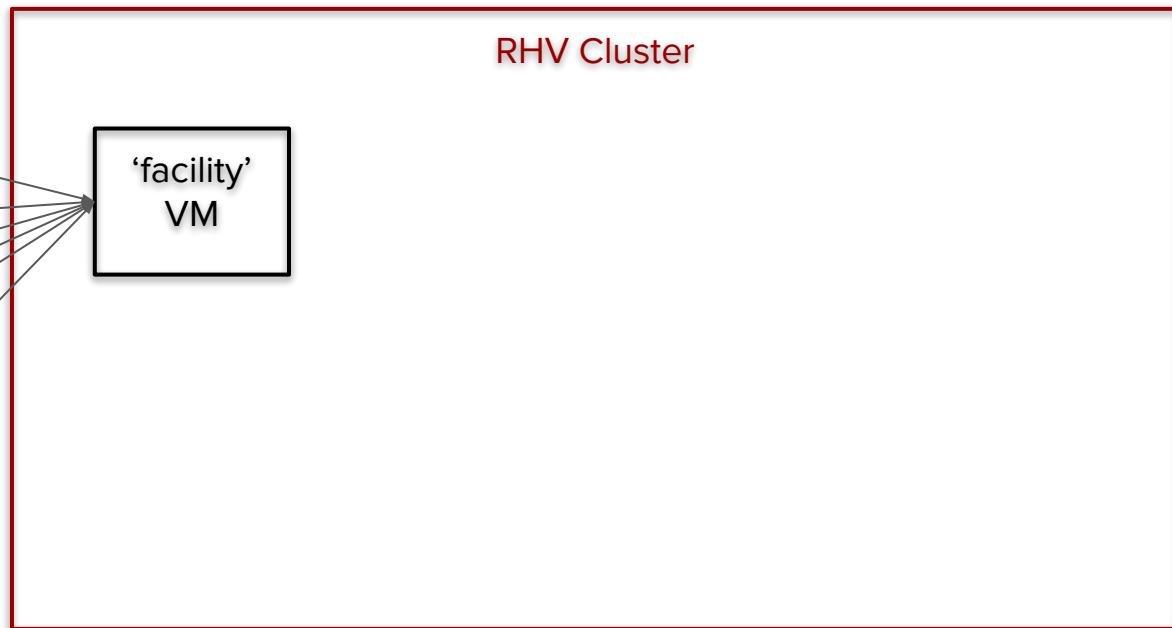
Preparation?

*Remember, we don't  
have on-prem, what  
we take for granted in  
the cloud!*



## Sandbox Preparation !!

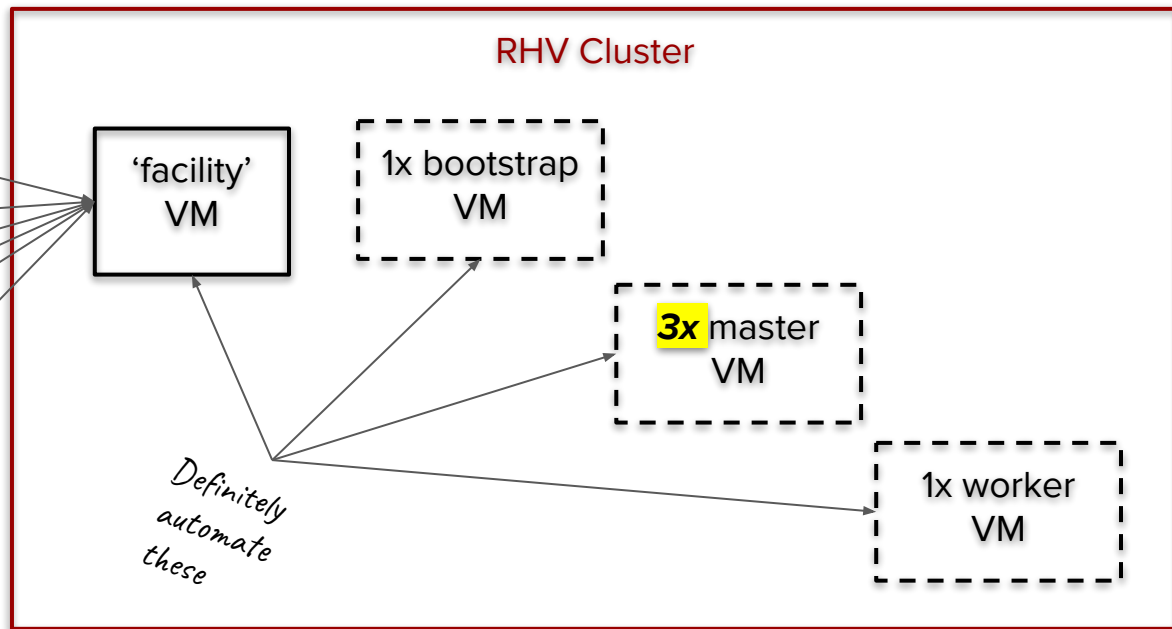
- Load Balancers
- apache (httpd)
- pxe/ tftp
- DHCP
- DNS
- NFS (opt'l)





## Components of Note

- Load Balancers
- apache (httpd)
- pxe/ tftp
- DHCP
- DNS
- NFS (opt'l)





Compute » Virtual Machines

Vms: ocp4



New

Edit

Remove

Run



Suspend

Shutdown



Reboot

Console



Migrate

Create Snapshot



1 - 7



		Name	Memory	CPU	Network	Graphics	Status
		bootstrap.ocp4.cdcug.local	0%	5%	0%	SPICE	Up
		master0.ocp4.cdcug.local	0%	2%	0%	SPICE	Up
		master1.ocp4.cdcug.local	0%	1%	0%	SPICE	Up
		master2.ocp4.cdcug.local	0%	2%	0%	SPICE	Up
		ocp4-facility	28%	3%	0%	SPICE	Up
		ocp4-net-test	--	--	--	None	Down
		worker0.ocp4.cdcug.local	0%	0%	0%	SPICE	Up

```
Red Hat Enterprise Linux CoreOS 410.8.20190920.2
WARNING: Direct SSH access to machines is not recommended.

---
[core@master2 ~]$
```

# CoreOS Installer

Writing image to disk

72351744%

```
Red Hat Enterprise Linux CoreOS 410.8.20190520.0 (Ootpa) 4.1
SSH host key: SHA256:dh/8szvXys+cZICz/maIXONMEdTfGMY5sKDSrPOnvDA (ED25519)
SSH host key: SHA256:/bq7ivL2o2EBrv46zUmhWaJ3CE+BbsT6OMNtxS5sbRg (ECDSA)
SSH host key: SHA256:E30dBpiGRc5eHSzIvL50CnKiveyJNP0NPbHAXiFB5Wk (RSA)
ens3: 10.110.111.230 fe80::9d12:6f63:cdd3:d53a
```

```
bootstrap login: _
```

## In Short

1. Download
2. Prepare infra
3. Prepare configs
4. Run installer
5. ?????
6. Profit!



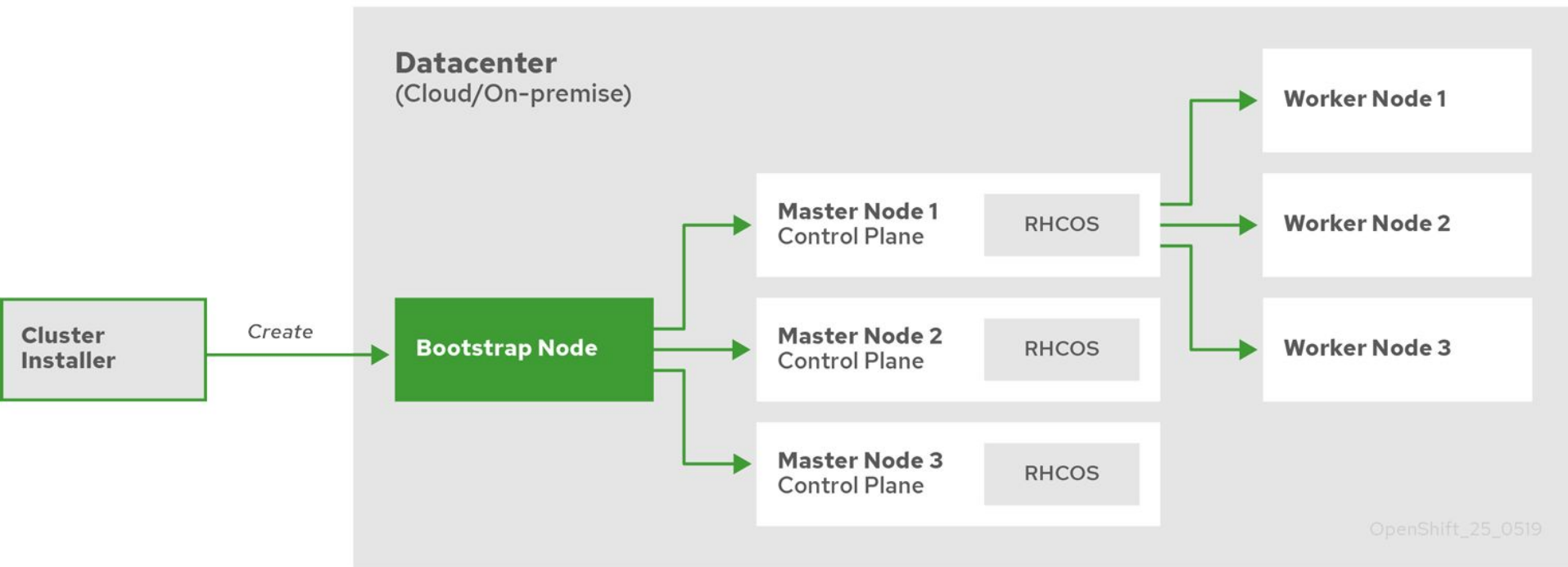
# Deploy to GCP

## **Openshift 4.2 can be installed on Google Cloud Platform**

- Currently in Developer Preview
- Single binary installer
- Based on ignition and terraform
- RHEL CoreOS the base operating system



# Installer Architecture



## Requirements for the GCP environment:

- DNS - Publicly accessible and registered with the Google DNS service in GCP
- Quotas - 500GB SSD drives minimum ( excluded some DCs by default )
- Service Accounts Permissions
- API/Service enablement

## What does the default installer deploy?

- Six Virtual Machines ( n1-standard-4 (4 vCPUs, 15 GB memory) )
- Storage buckets ( Image registry storage bucket)
- Network Objects ( master/worker subnets, public ips, firewall rules and routes )
- DNS zone ( api endpoint and internal DNS zone )

## Download the single openshift binary and.....

```
[~]$ openshift-install create cluster
```

```
? SSH Public Key /home/user_id/.ssh/id_rsa.pub
```

```
? Platform gcp
```

```
? Project ID my_gcp_project ( it should pick this up automatically if the key is installed correctly )
```

```
? Region us-east1
```

```
? Base Domain example.com ( it should pick this up automatically if the key is installed correctly )
```

```
? Cluster Name mycluster
```

```
? Pull Secret [? for help] < paste the pull secret here. You can find it on the install website >
```

# //Installer Recommendations

- Use **debug** command!! It is pretty helpful ( openshift-install create cluster --log-level debug )
- Make sure your **DNS** works properly and can resolve from the internet ( the installer will fail if the kubernetes API is not installed)
- Once the installer is complete, keys and credentials are saved to an auth directory where you ran the installer

# GCP Object Walkthrough

# Red Hat Service Mesh

## What is it and Why use it?

A service mesh is the network of microservices that make up applications in a distributed microservice architecture and the interactions between those microservices.

It is an abstraction placed on top of network services in a Openshift cluster that can control and manipulate flow and policy

We can use a service mesh to control and set policy on network traffic between every microservice involved in the mesh.



# Istio

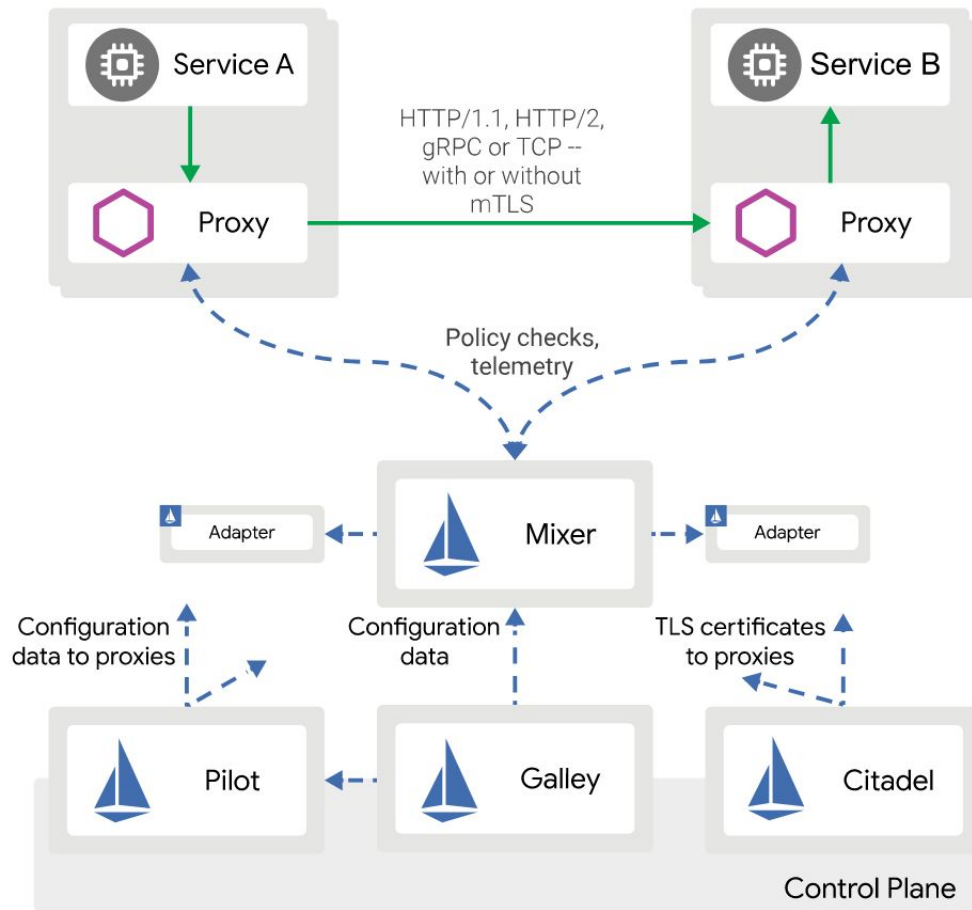
An opens source service mesh abstraction for the network that Red Hat Service Mesh is based on.

Core to its architecture is the separation of the control plane and the data plane

Example Features:

- Traffic routing/shaping (ingress/egress)
- Canary A/B testing
- Circuit breaking and fault injection
- Certificate generation for Mutual TLS
- Real time telemetry and tracing

# //Istio Architecture



## Differences with Service Mesh and Istio

- Is Multi-tenancy by default
- Kiali and Jaeger are prerequisites and build in
- OpenSSL and the CNI plugin framework are used

The Red Hat Service Mesh is controlled by the Service Mesh Operator. It request both the Kiali and Jaeger operators to be installed.

**Everything is in an Operator**



# Red Hat Service Mesh Demo

# OCP3 to 4 Migration

- Maybe just re-deploy?
- Master-config.yaml?
- Namespace at a time
- Cluster admin
- Is there a tool for this?

The OpenShift Migration Controller assists with installation of application migration tooling on OpenShift 3.x and 4.x clusters:

Supports migrations of Openshift **3.7.x** to Openshift **4.1.x**



## Migration Components

- Migration Operator
- Migration Controller (mig-controller)
- Migration UI (mig-ui)
- Velero
- An S3 compatible Bucket is used as scratch space for the backup ( AWS S3, NooBaa, MinIO )

## High Level Installation Overview

- Install the Mig Operator to both the 3x and 4x Openshift clusters
- Install the Migration Controller to the 3x and 4x Openshift Clusters
- Install the CAMS UI to one of the clusters
- Enable CORS ( Cross-Origin resource sharing ) on both clusters
- Install a compatible S3 bucket (NooBaa\* in this case)

# Migration Tool Demo

# //OCP 4 Migrations - Resources

<https://www.youtube.com/channel/UCBDU5UK5Okq3mIIMygpkbNA>

<https://github.com/fusor/mig-operator>

<https://github.com/fusor/mig-agnosticd>

<https://github.com/vmware-tanzu/velero>

