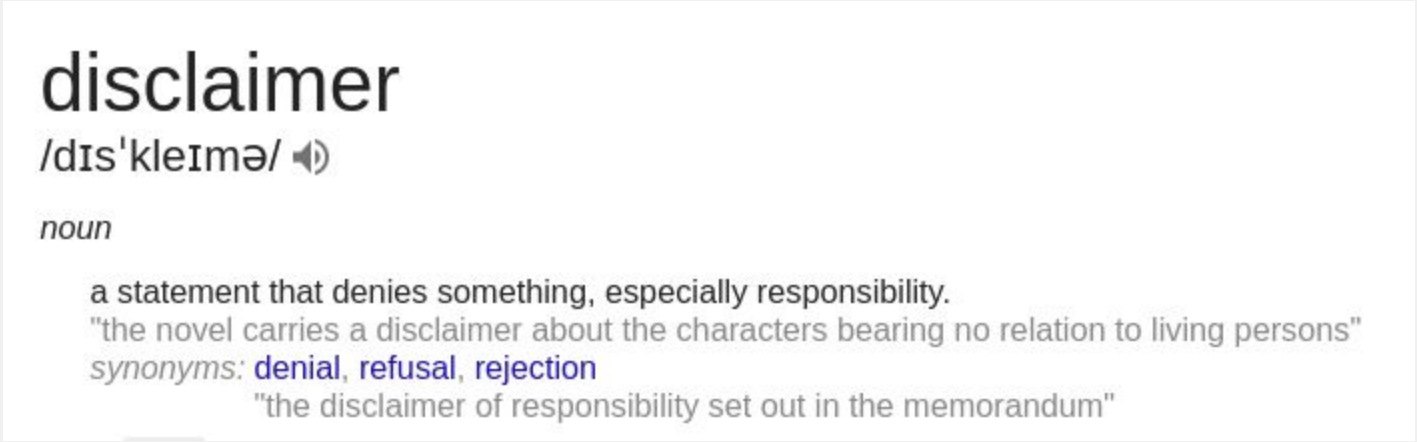
OpenShift Futures

CoreOS integration

Chris Milsted
Principal Solution architect, FSI
cmilsted@redhat.com



The content set forth herein does not constitute in any way a binding or legal agreement or impose any legal obligation or duty on Red Hat. This information is provided for discussion purposes only and is subject to change for any or no reason.

Introduction

My Goals

- Give you a view of “the future” of OpenShift
- Get you thinking about how you might consume this over the next 12-24 months
- Start a dialogue

Photo by ÁLVARO MENDOZA on Unsplash

What this is not

- A detailed roadmap
- Fully up to date - this is changing a lot
- Scary

Photo by David von Diemar on Unsplash

Questions

- Who is using OpenShift today?
- Who is using OpenShift today in Production?
- Who has more than 1 cluster?
- Who has more than 10 clusters?
- Who has more than 100 Pods?
- Who has more than 10,000 Pods?

Middle

Kubernetes is Done.... The next phase is all about the owners experience...

Or we built a platform to handle disposable microservices - but we forgot to do that for the platform as well.

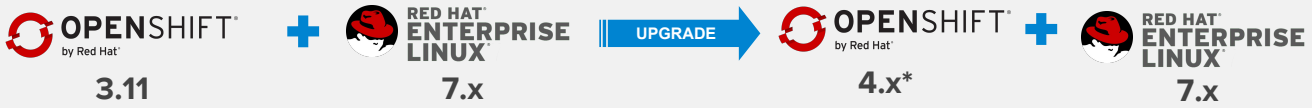
First point

- Don’t panic

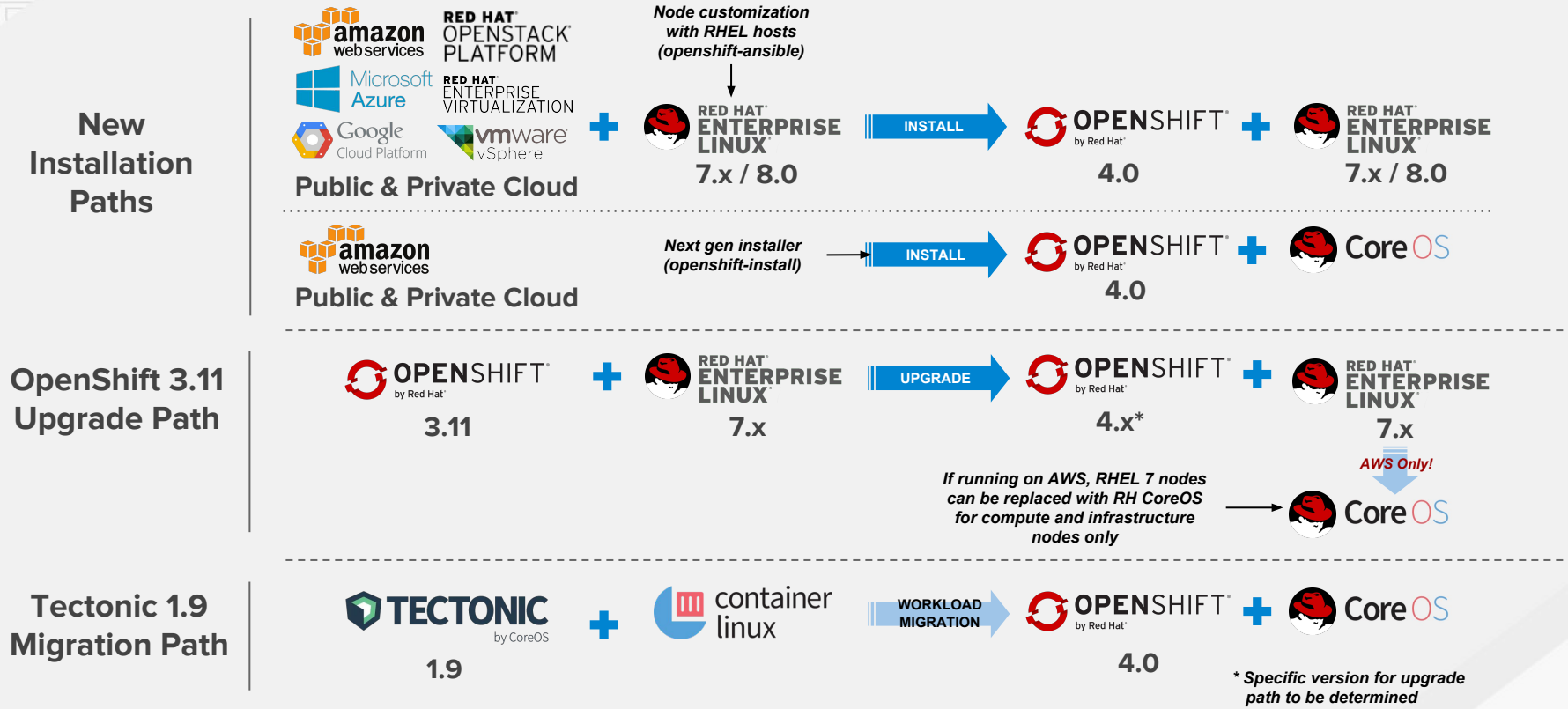
http://images-assets.nasa.gov/image/as11-44-6551/as11-44-6551~orig.jpg

Install, Upgrade, & Migration Paths to OpenShift 4

OpenShift 3.11 Upgrade Path



Install, Upgrade, & Migration Paths to OpenShift 4



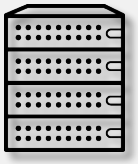
DEV OPS

Next Gen Installer

Opinionated “best practices” single cluster provisioning (openshift-install)

- CLI-based installer designed to easily provision a “best practices” OpenShift cluster on RH CoreOS infrastructure
  - 4.0: AWS
  - 4.1: RHOSP, Azure
  - 4.2: Bare Metal
- Guided workflow allowing users to walk through each step and customize as needed:
  - `init → render → prepare → launch`
- Only supports deployments on immutable infrastructure (RH CoreOS)
  - Host OS updates are fully automated and pushed alongside OpenShift updates
- Quickly download installation client (with embedded token) from cloud.openshift.com and run from anywhere (Linux, Windows, and Mac) to deploy OpenShift

```
# Generate initial installation configuration
$ openshift-install init
    Select a supported provider [aws/azure/openstack]: aws
    Enter the base-domain: foo.example.com

Generated ./install-config.yaml

# Modify config settings (e.g EC2 master machine sizes)
$ vi ./install-config.yaml

# Render assets from install-config parameters
$ openshift-install render

# Edit rendered assets (e.g modify kube-dns manifest)
$ vi ./manifests/kube-dns-daemonset.yaml

# Generate final assets for use in bootstrap tool (ignition, tfvars)
$ openshift-install prepare
$ git add . && commit -m ‘my cluster’

# Launch from generated assets:
$ openshift-install launch
```



Node Customization

RHEL host based single cluster provisioning (openshift-ansible)

- Traditional method for installing OpenShift clusters, but with significant refactoring in 4.0
  - openshift-ansible will no longer be used to update OpenShift
  - Cluster upgrades will now be performed from the cluster console with the operator framework
  - Minimizes openshift-ansible's role to just the initial configuration needed for provisioning a control plane, top level operator, and compute nodes
- Easily provision OCP 4.0 on RHEL nodes running on-premise or in the public cloud
  - No support for provisioning on immutable RH CoreOS nodes
- Allows existing node provisioning tooling to be leveraged by customers
  - Customers can customize RHEL based on their environment requirements
  - RHEL OS updates are the responsibility of the administrator, but to assist with this a playbook will be provided with hooks to facilitate a rolling restart of the nodes after upgrading the OS
- Support for in-place upgrades of OCP 3 (RHEL-based) clusters to OCP 4
  - Specific OCP versions for upgrade path to be determined
  - Won't support moving off RHEL nodes to immutable RH CoreOS nodes
  - Won't support OCP 3 + RHEL Atomic Host upgrades; customers must first move to RHEL before upgrading

An Operator is a Kubernetes pattern that is extending the Kubernetes control plane with a custom Controller and Custom Resource Definitions that add additional operational knowledge of an application.

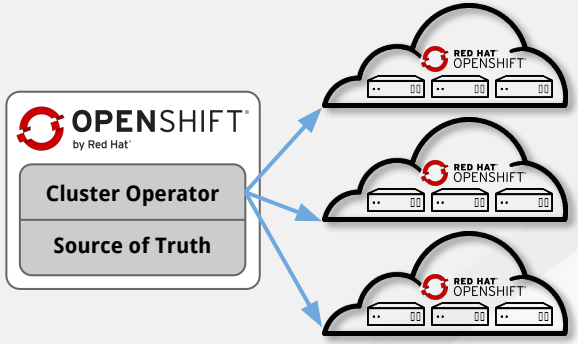
https://itnext.io/wth-is-a-operator-lifecycle-manager-873cf1661b04

Future Deliverable

Cluster Operator

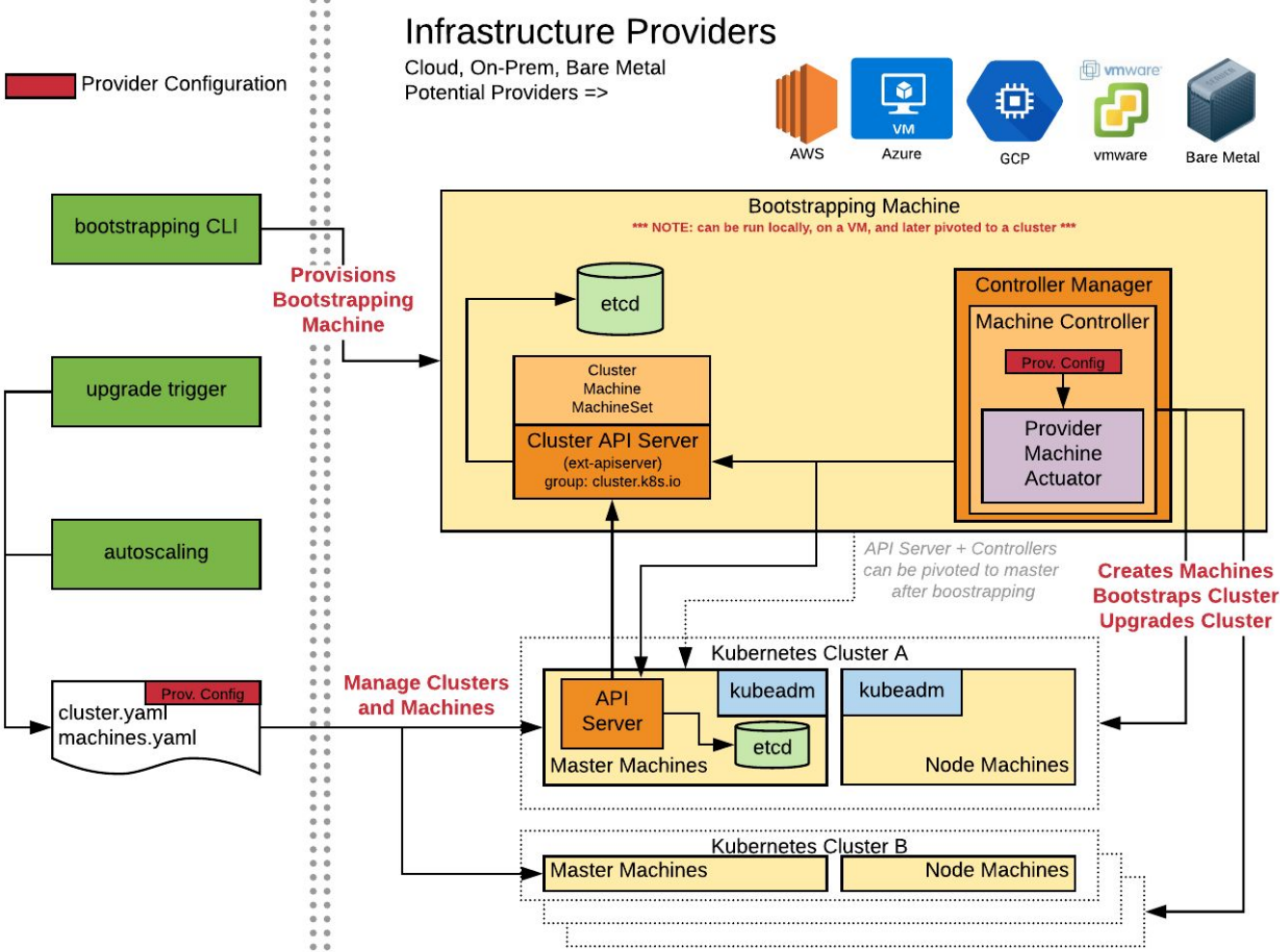
Multi-cluster Provisioning & Orchestration

- Reliably provision/deprovision, upgrade, & configure OpenShift (& RH CoreOS) clusters
  - 4.0: RH Internal Developer focused release
    - Support for installing/uninstalling on AWS only; upgrades won't be supported in initial release
    - Enables developers to easily stand up real-world clusters for development and testing of various OpenShift components (operators, core kube, etc)
    - Same system can be used to drive automated CI/CD testing of PRs
- Installed on OpenShift cluster via an operator
  - Becomes central source of truth for all clusters it manages
- Leverages work from:
  - ***Next gen installer*** - Uses CLI to launch clusters in the public cloud
  - ***Unified Cluster API*** - Declarative, Kubernetes-style API for cluster creation, configuration, and management

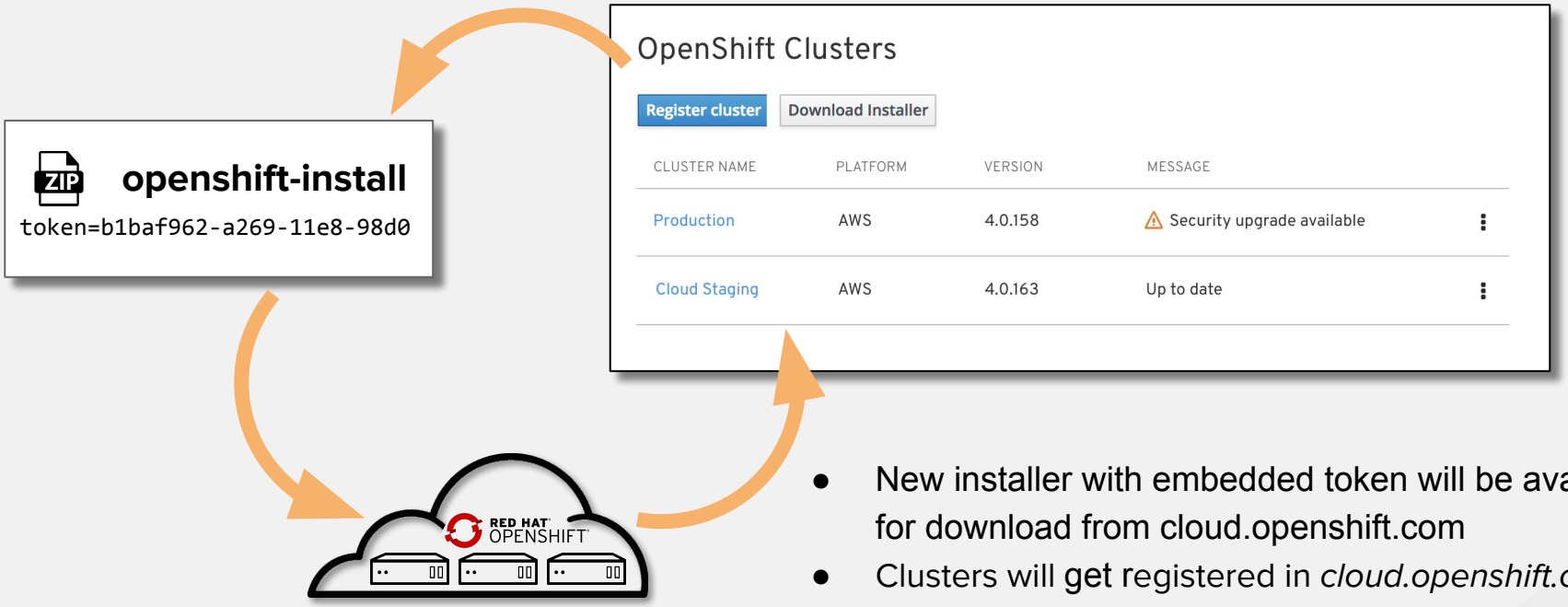


Cluster API

https://github.com/kubernetes-sigs/cluster-api

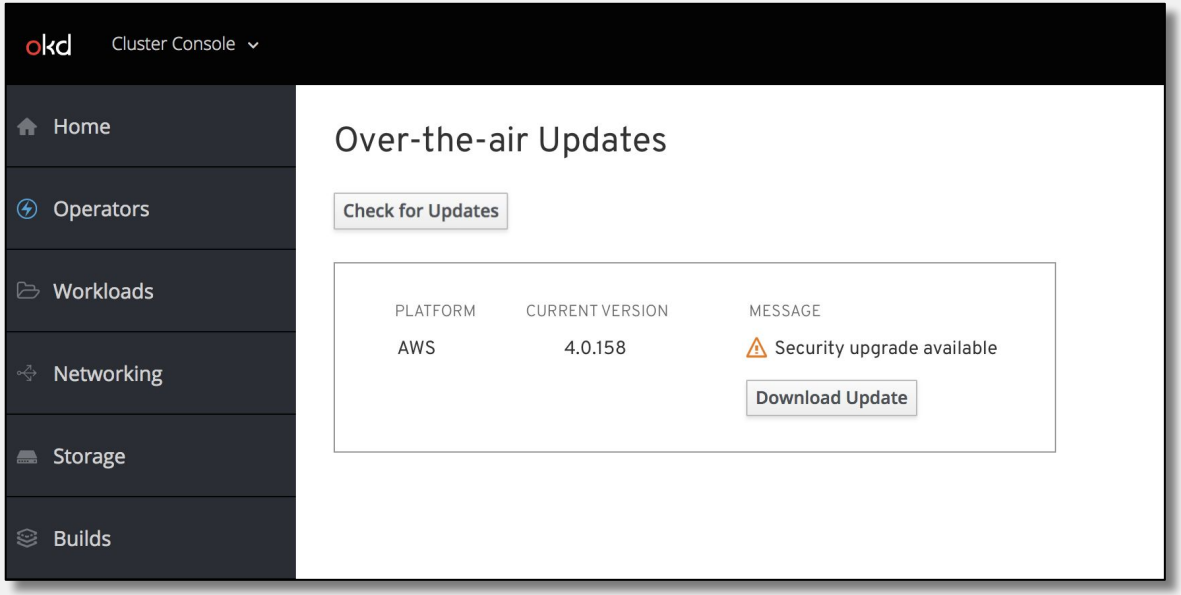


Unified Hybrid Cloud & Over-The-Air Updates



- New installer with embedded token will be available for download from cloud.openshift.com
- Clusters will get registered in *cloud.openshift.com*
- View/control over-the-air updates across all clusters

On-Premises Updates

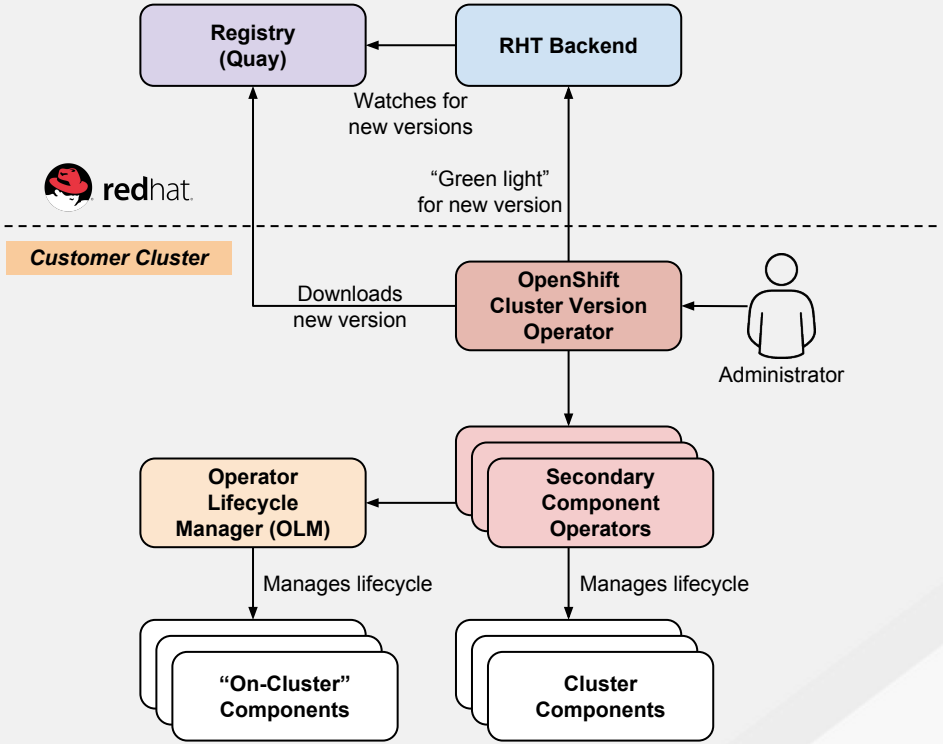


- “Over-The-Air” updates can be driven from either *cloud.openshift.com* or the Cluster Console right on the cluster
- Manual updates will be supported for offline (disconnected) environments
  - Tooling to automate updates will be added in later release

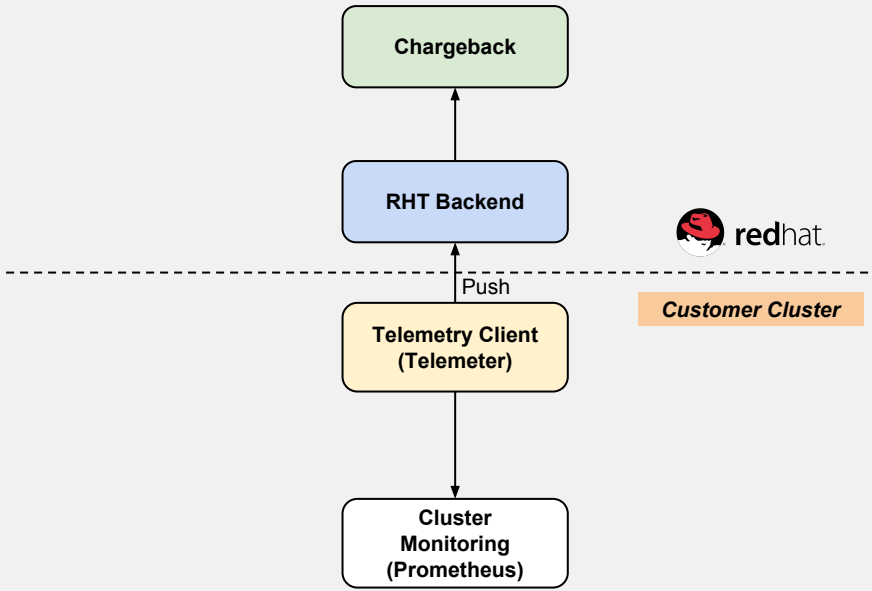
Over-The-Air Updates

How Over-The-Air Updates Work

- RHT backend builds a graph of upgrade possibilities from release images in registry
- OpenShift clusters tell RHT backend who they are and what version it's running
- Policy engine combines information from customer entitlement and upgrade graph to tell clusters what they can upgrade to
- Either an administrator or automatic update controller will edit the Cluster Version Operator's CR with the update version
- Cluster Version Operator will get release image from registry and apply the changes



Connected Customer Telemetry

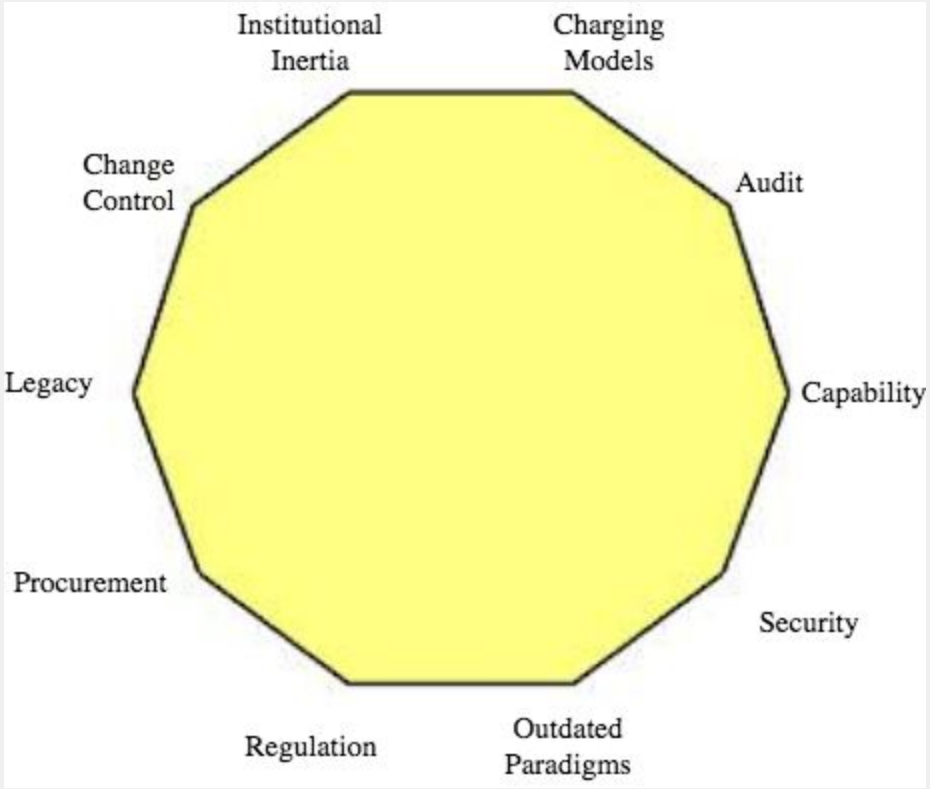


Insight into Updates & Customer Success

- Customer clusters send metrics back to Red Hat, which is scraped with Prometheus and pushed by Telemeter
- Centralized metrics will be used for:
  - **OTA Updates** - Exceeding failure thresholds will stop the rollout of a given release
  - **SRE / CEE** - Is cluster healthy?
  - **Billing** - Chargeback for hourly usage
  - **App Partners** - Operator metering for the Marketplace

End

What you are all thinking now… the decagon of despair



https://zwischenzugs.com/2018/10/02/why-are-enterprises-so-slow/

If you can work with public cloud, openshift 4.+ inherits this thinking...

This is the “fourth cloud”

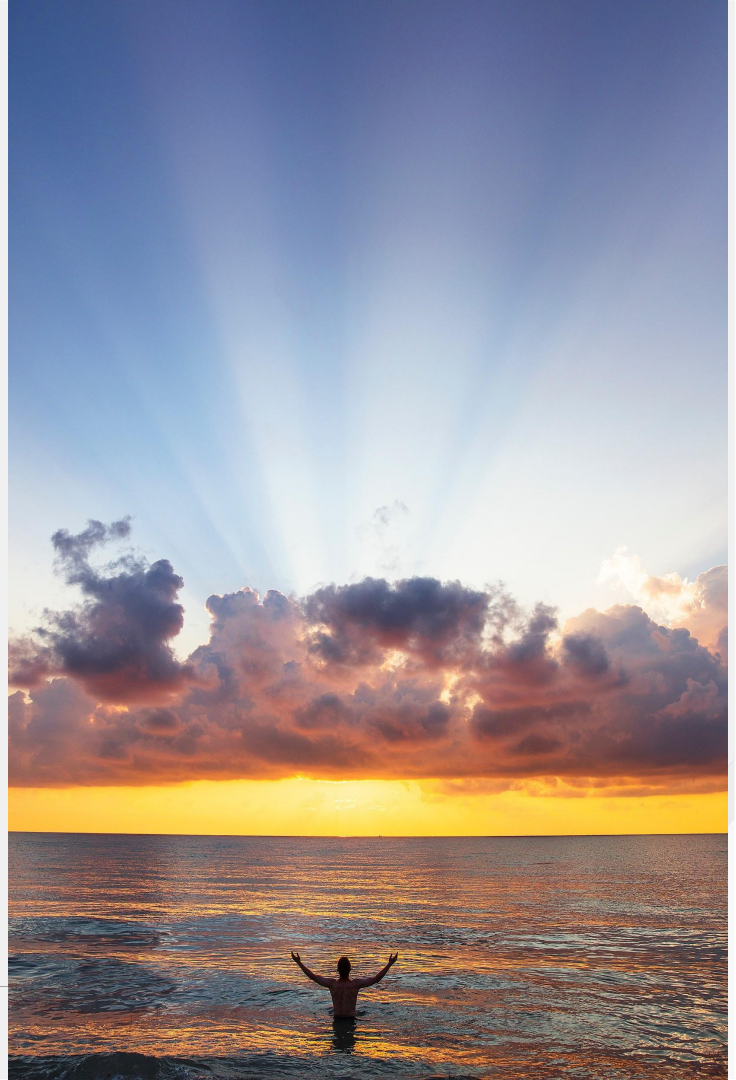


Photo by Joshua Earle on Unsplash

So what next

- The 2018 CIO Survey plainly reveals that the job of the CIO is changing. As digitalization and innovation put more emphasis on the information rather than the technology in “IT,” the CIO’s role is transforming from delivery executive to business executive — from controlling costs and re-engineering processes to driving revenue and exploiting data.

https://www.gartner.com/imagesrv/cio-trends/pdf/cio_agenda_2018.pdfm

- Start planning for OpenShift 4.+!


Thank you
Q&A

https://itnext.io/wth-is-a-operator-lifecycle-manager-873cf1661b04