



OpenShift 4 Roadmap Update

Chris Blum
Senior Architect

William Markito Oliveira
Product Management

OpenShift 4 Platform

CLUSTER SERVICES

Metrics, Chargeback, Registry, Logging

APPLICATION SERVICES

Middleware, Service Mesh, Functions, ISV

DEVELOPER SERVICES

Dev Tools, Automated Builds, CI/CD, IDE

AUTOMATED OPERATIONS

KUBERNETES

Red Hat Enterprise Linux or RHEL CoreOS

Best IT Ops Experience

CaaS \longleftrightarrow PaaS \longleftrightarrow FaaS

Best Developer Experience

OpenShift 4.2



Disconnected, air-gapped installation

Full-stack Automation for Azure, GCP, OpenStack

Migration tooling



Operators for Red Hat Integration, Business Automation, Runtimes

Special resources operator For GPU



Developer console with application topology

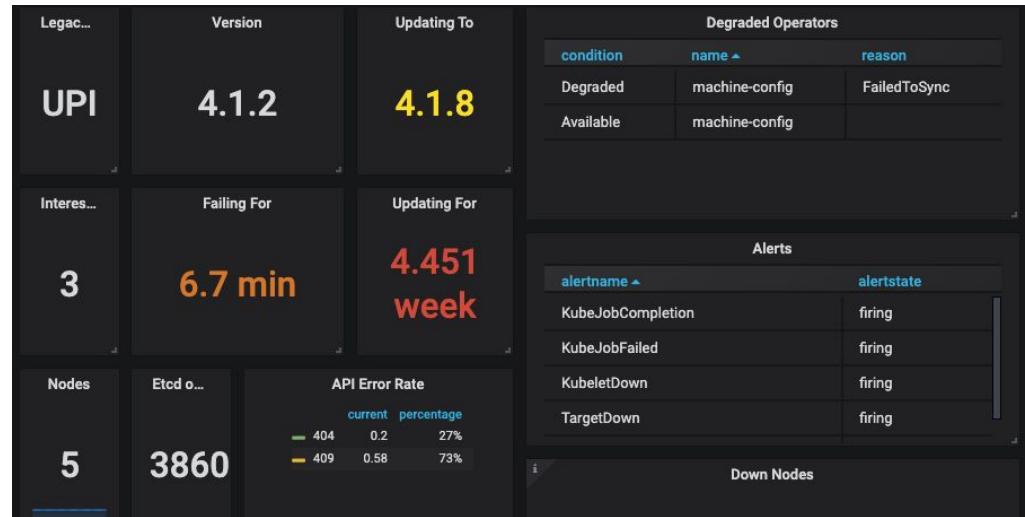
CodeReady Containers

Service Mesh fully supported

Connected Customer

- 4 months since GA
- 11 z-stream releases
- 186 bugs fixed
- 207 CVEs across RHEL+OCP

20% of bugs fixed to date have been from information obtained through telemeter



OPENSHIFT PLATFORM

4.2 Supported Providers

Full Stack Automation (IPI)



Microsoft Azure *



RED HAT[®]
OPENSTACK[®]
PLATFORM

Pre-existing Infrastructure (UPI)



Bare Metal



* Support for full stack automated installs to pre-existing VPC & subnets and deploying as private/internal clusters is planned for 4.3.

Generally Available

Installation Experiences

OPENSHIFT CONTAINER PLATFORM

Full Stack Automated

Simplified opinionated “Best Practices” for cluster provisioning

Fully automated installation and updates including host container OS.



Red Hat
Enterprise Linux
CoreOS

Pre-existing Infrastructure

Customer managed resources & infrastructure provisioning

Plug into existing DNS and security boundaries



Red Hat
Enterprise Linux
CoreOS



Red Hat
Enterprise Linux

HOSTED OPENSHIFT

Azure Red Hat OpenShift

Deploy directly from the Azure console. Jointly managed by Red Hat and Microsoft Azure engineers.

OpenShift Dedicated

Get a powerful cluster, fully Managed by Red Hat engineers and support.

Generally Available



Comparison between deployments methods

	Full Stack Automation	Pre-existing Infrastructure
Build Network	Installer	User
Setup Load Balancers	Installer	User
Configure DNS	Installer	User
Hardware/VM Provisioning	Installer	User
OS Installation	Installer	User
Generate Ignition Configs	Installer	Installer
OS Support	Installer: RHEL CoreOS	User: RHEL CoreOS + RHEL 7
Node Provisioning / Autoscaling	Yes	Only for providers with OpenShift Machine API support
Customization & Provider Support	Best Practices: AWS	Yes: AWS, Bare Metal, VMware, GCP(4.2), Azure(4.2)

OPENSHIFT PLATFORM

Full stack automated deployments of AWS, Azure, GCP & OSP!

```
$ ./openshift-install --dir ./demo create cluster
? SSH Public Key /Users/demo/.ssh/id_rsa.pub
? Platform azure
? azure subscription id xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
? azure tenant id xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
? azure service principal client id xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
? azure service principal client secret ****
INFO Saving user credentials to "/Users/demo/.azure/osServicePrincipal.json"
? Region centralus
? Base Domain example.com
? Cluster Name demo
? Pull Secret [? for help] ****
INFO Creating infrastructure resources...
INFO Waiting up to 30m0s for the Kubernetes API at https://api.demo.example.com:6443...
INFO API v1.14.0+4788f50 up
INFO Waiting up to 30m0s for bootstrapping to complete...
INFO Destroying the bootstrap resources...
INFO Waiting up to 30m0s for the cluster at https://api.demo.example.com:6443 to initialize...
INFO Waiting up to 10m0s for the openshift-console route to be created...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export KUBECONFIG=/Users/demo/openshift-install/demo/auth/kubeconfig'
INFO Access the OpenShift web-console here:
https://console-openshift-console.apps.demo.example.com
INFO Login to the console with user: kubeadmin, password: <password>
```



```
$ ./openshift-install --dir ./demo create cluster
? SSH Public Key /Users/demo/.ssh/id_rsa.pub
? Platform gcp
? Service Account (absolute path to file or JSON content)
/Users/demo/.secrets/ServiceAccount.json
INFO Saving the credentials to "/Users/demo/.gcp/osServiceAccount.json"
? Project ID openshift-gce-devel
? Region centralus
? Base Domain example.com
? Cluster Name demo
? Pull Secret [? for help] ****
INFO Creating infrastructure resources...
INFO Waiting up to 30m0s for the Kubernetes API at https://api.demo.example.com:6443...
INFO API v1.14.0+4788f50 up
INFO Waiting up to 30m0s for bootstrapping to complete...
INFO Destroying the bootstrap resources...
INFO Waiting up to 30m0s for the cluster at https://api.demo.example.com:6443 to initialize...
INFO Waiting up to 10m0s for the openshift-console route to be created...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export KUBECONFIG=/Users/demo/openshift-install/demo/auth/kubeconfig'
INFO Access the OpenShift web-console here:
https://console-openshift-console.apps.demo.example.com
INFO Login to the console with user: kubeadmin, password: <password>
```



Simplified Cluster Creation

Easily provision a “best practices” OpenShift cluster on Microsoft Azure

- CLI-based installer with interactive guided workflow
- Installer takes care of provisioning the underlying Infrastructure significantly reducing deployment complexity

Faster Install

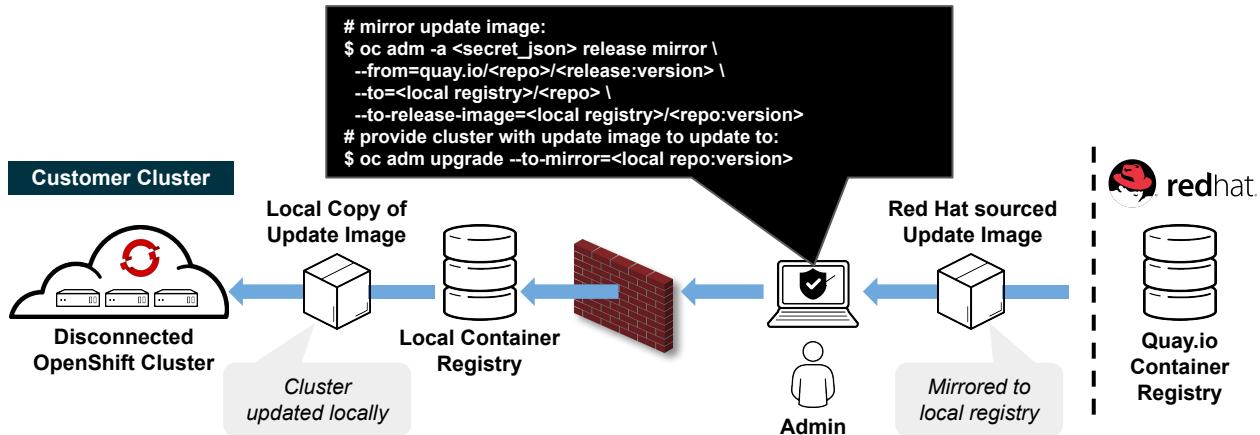
The installer typically finishes within 30 minutes

- Only minimal user input needed with all non-essential install config options now handled by component operator CRD’s
- Leverages RHEL CoreOS for all node types enabling full stack automation of installation and updates of both platform and host OS content

Generally Available



Disconnected “Air-gapped” Installation & Upgrading



Overview

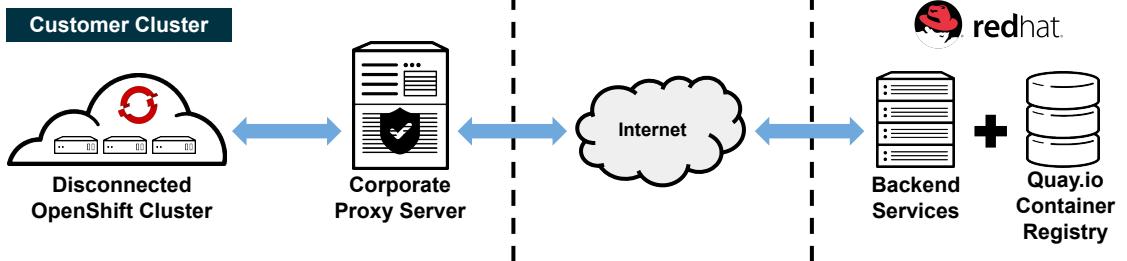
- 4.2 introduces support for installing and updating OpenShift clusters in disconnected environments
- Requires local Docker 2.2 spec compliant container registry to host OpenShift content
- Designed to work with the user provisioned infrastructure deployment method
 - Note: Will not work with Installer provisioned infrastructure deployments*

Installation Procedure

- Mirror OpenShift content to local container registry in the disconnected environment
- Generate install-config.yaml: \$./openshift-install create install-config --dir <dir>
 - Edit and add pull secret (PullSecret), CA certificate (AdditionalTrustBundle), and image content sources (ImageContentSources) to install-config.yaml
- Set the OPENSHIFT_INSTALL_RELEASE_IMAGE_OVERRIDE environment variable during the creation of the ignition configs
- Generate the ignition configuration: \$./openshift-install create ignition-configs --dir <dir>
- Use the resulting ignition files to bootstrap the cluster deployment

Generally Available

Cluster-wide Egress Proxy



Overview

- 4.2 introduces support for installing and updating OpenShift clusters through a corporate proxy server
- Leverages new proxy controller within the cluster-network-operator, which is responsible for:
 - Reconciling a proxy object and writing spec > status upon successful validation.
 - Reconciling user-provided trust bundles referenced by trustedCA, validating the trust bundle certificates, merging the certificates with the system trust bundle and publishing the merged bundle to the openshift-config-managed/trusted-ca-bundle configmap.

Installation Procedure

- Installer will use PROXY* environment variables from the shell it's invoked from
- Generate install-config.yaml: \$./openshift-install create install-config --dir <dir>
 - Edit proxy information (httpProxy, httpsProxy, & noProxy) and CA certificate (AdditionalTrustBundle) to install-config.yaml!
- Installer validates the provided install-config.yaml parameters, renders the necessary assets to create the cluster, and initiates the installation process based on the install method used:
\$./openshift-install create cluster --dir <dir>

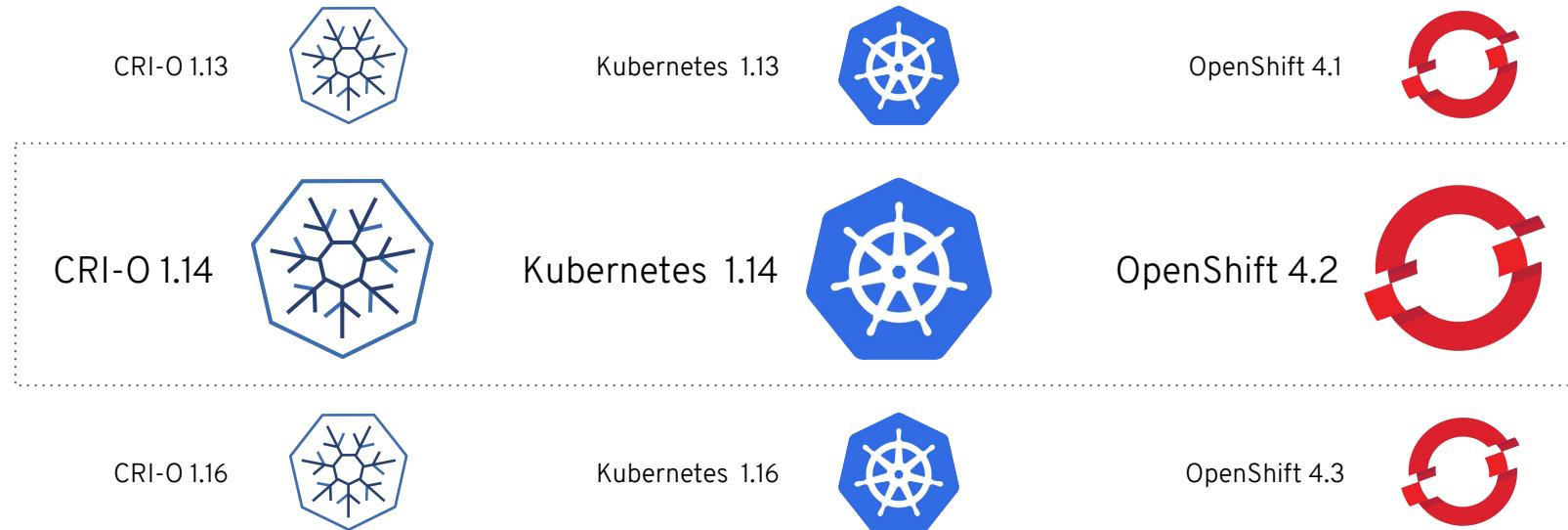
An admin with privileges can interact with the proxy object using 'oc' commands (use the 'oc edit' command to modify the proxy information.) Here is an example proxy config:

```

$ oc get proxy/cluster -o yaml
apiVersion: config.openshift.io/v1
kind: Proxy
metadata:
  creationTimestamp: "2019-08-21T22:36:49Z"
  generation: 2
  name: cluster
  resourceVersion: "24913"
  selfLink: /apis/config.openshift.io/v1/proxies/cluster
  uid: 2a344b01-d267-11f9-a4f3-025de4b59c38
spec:
  httpProxy: http://<username>:<pswd>@<ip>:<port>
  httpsProxy: https://<username>:<pswd>@<ip>:<port>
  noProxy: example.com
  readinessEndpoints:
    - http://www.google.com
    - https://www.google.com
  trustedCA:
    name: user-ca-bundle
status:
  httpProxy: http://<username>:<pswd>@<ip>:<port>
  httpsProxy: https://<username>:<pswd>@<ip>:<port>
  noProxy:
    10.0.0.0/16,10.128.0.0/14,127.0.0.1,169.254.169.254,172.30.0.0/16,api-int.demo.example.com,api.demo.example.openshift.com,etcd-0.demo.example.com,etcd-1.demo.example.com,etcd-2.demo.example.com,example.com,localhost
  
```

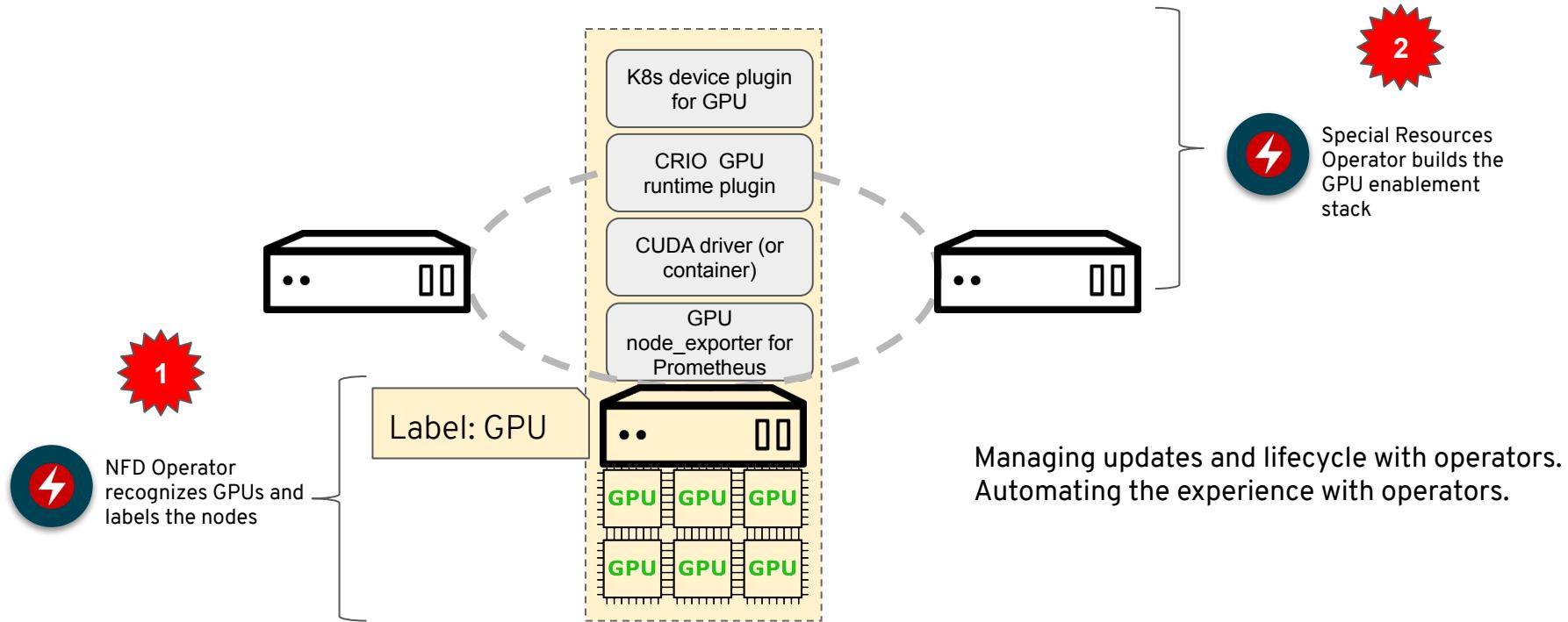
CRI-O Support in OpenShift

CRI-O tracks and versions identical to Kubernetes, simplifying support permutations



Generally Available

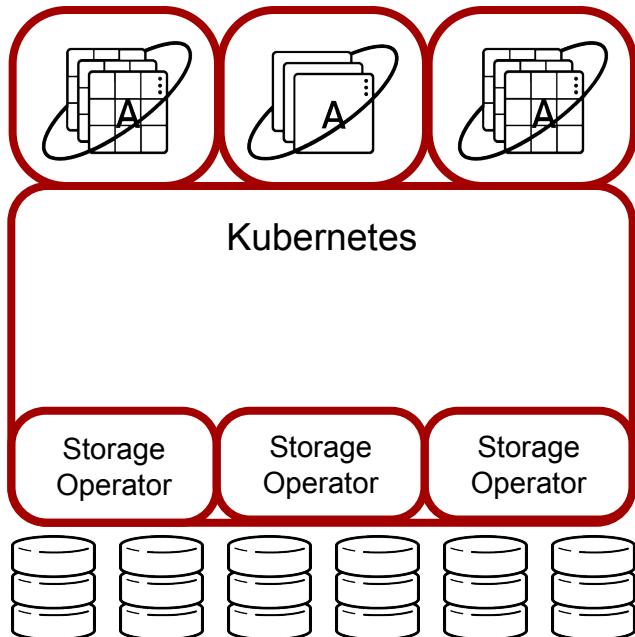
Enablement of GPUs in an OpenShift Cluster



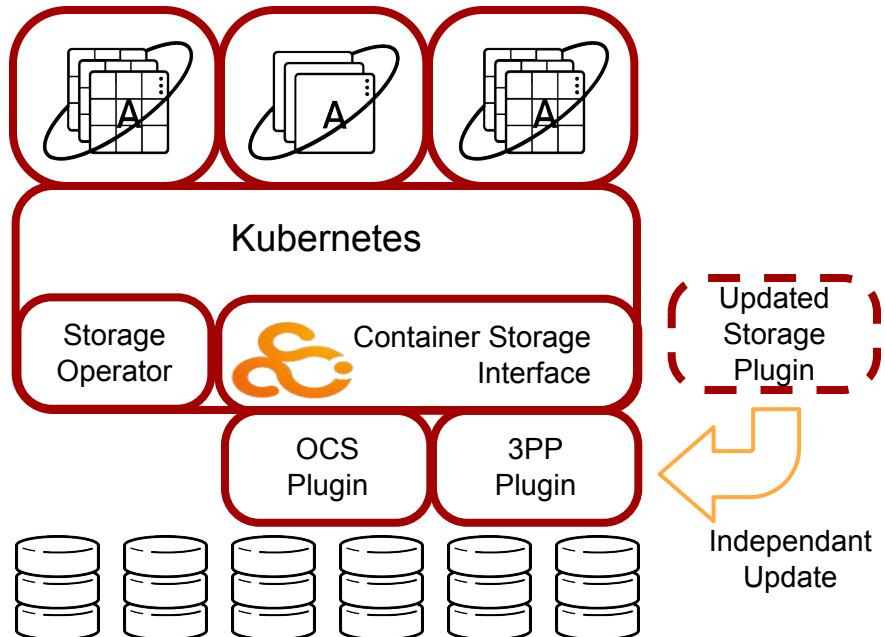
OPENSHIFT STORAGE

OPENSHIFT CSI

4.1



4.2



CSI: ENABLING OCS AND
PARTNERS

OPENSHIFT STORAGE

STORAGE DEVICES

Storage Focus

- Cluster Storage Operator
 - Sets up the default storage class
 - Looks through cloud provider and sets up the correct storage class
- Drivers themselves remain in-tree for now, CSI versions to follow later
- New GA storage in 4.2
 - Local Volume
 - Raw Block
 - Cloud providers (AWS, GCP, Azure, vSphere)
 - Local Volume

Supported	
AWS EBS	iSCSI
Azure File & Disk	Fibre Channel
GCE PD	HostPath
VMware vSphere Disk	Local Volume NEW
NFS	Raw Block NEW

OCS for Persistent Storage

- New Storage backend for OCP 4 based on Ceph and Nooba
 - Adds enterprise S3 storage
- Install from Operator Hub
- Offers Ceph via Rook and Nooba as Multi-Cloud-Gateway
- Manages storage at scale
- Handles upgrades and health state
- Identifies availability zones and configures data replication accordingly
- Integrated with Openshift monitoring

OCS for Persistent Storage

The screenshot shows the Red Hat OpenShift web interface with the 'Storage' tab selected in the top navigation bar. The left sidebar contains links for Home, Dashboards, Projects, Status, Search, Events, Catalog, Workloads, Networking, Storage, Builds, Monitoring, Compute, and Administration. The main dashboard displays various metrics and status indicators for OCS storage.

Details:

- Name: cluster-name
- Provider: Bare Metal
- OCS version: v1.0

Inventory:

- 3 Nodes
- 24 Disks
- 20 Pods
- 12 PVs
- 18 PVCs

Health: OCS is healthy.

Capacity: Total Capacity: 120 available of 279 Gi. 59% Used.

Data Resiliency: Your Data is Resilient.

Performance: 6 Hours

- IOPS: 151,215 IOPS
- Latency: 3.35 ms
- Throughput: 2.32 Gi/s
- Recovery rate: 7.5 Gi/s

Top Consumers: Projects, Top 5, By used capacity.

Used capacity: A line chart showing usage over time for Project 5, Project 2, Project 10, Project 4, and Project 9.

Events:

- A few seconds ago - 1 time in the last 24 hours: rook OSD-10-328949 crashed. From kubelet ip-10-0-147-109.ec2.internal. Rebuild initiated as Disk 5 failed.
- 2 minutes ago - 2 times in the last 24 hours: host-name-1. From kubelet ip-1-0-120-109.ec2.internal. CPU utilization over 50%. Migrated 2 pods to other hosts.
- 10 minutes ago - 1 time in the last 24 hours: host-name-1.

Red Hat Logo:

OCS for Persistent Storage

The screenshot shows the Red Hat OpenShift web interface with the 'Storage' dashboard selected. The left sidebar includes links for Home, Dashboards, Projects, Status, Search, Events, Catalog, Workloads, Networking, Storage, Builds, Monitoring, Compute, and Administration. The main content area displays the following sections:

- Details:** Name: cluster-name, Provider: Bare Metal, OCS version: v1.0.
- Inventory:** 3 Nodes (1 online), 24 Disks (8 offline), 20 Pods (1 alert), 12 PVs, 18 PVCs.
- Health:** OCS is health is degraded. Alerts include: Node 2 is offline, 8 Disks are offline, Your cluster is running out of disk space, Alert 4, and Warning 2.
- Capacity:** Total capacity 25 available of 168 Gi (85% Used).
- Data Resiliency:** Rebuilding data resiliency (Rebuilding in progress, 85.2%).
- Performance:** IOPS (151,215 IOPS), Latency (3.35 ms), Throughput (2.32 Gi/s), Recovery rate (7.5 Gi/s).
- Events:** A few seconds ago - 1 time in the last 24 hours: rook-osd-10-328949 crashed. From kubelet ip-10-0-147-109.ec2.internal. Rebuild initiated as Disk 5 failed. 2 minutes ago - 2 times in the last 24 hours: host-name-1. From kubelet ip-1-0-120-109.ec2.internal. CPU utilization over 50%. Migrated 2 pods to other hosts. 10 minutes ago - 1 time in the last 24 hours: host-name-1.
- Top Consumers:** A line chart showing used capacity over time for various consumers.

App migration experience

Using open source tooling based on Velero

Velero is an upstream project previously known as Ark. Check out [this video](#) if you are curious and want to get a sneak peek at our capabilities.

What's moved during a migration

- Namespaces
- Persistent Volumes (move or copy)
- All important resource objects (Deployments, StatefulSets, etc)

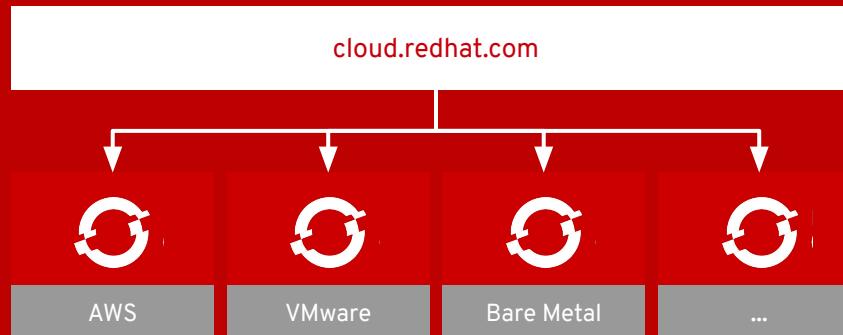
Available in OpenShift 4.2

The screenshot shows the 'Migration Plan Wizard' interface. At the top, it says 'Create a migration plan'. On the left, a vertical navigation bar lists steps: 1 General, 2 Migration Source (which is selected and highlighted in blue), 3 Persistent Volumes, 4 Migration Targets, and 5 Results. To the right of the navigation bar, there is a section titled 'Source Cluster' with a dropdown menu set to 'Summit Demo Source Cluster'. Below this, a table titled 'Select projects to be migrated:' lists two projects: 'robot-shop' and 'sandbox', with a checkbox next to each. Further down, a section titled 'Migration Plans' lists two entries:

Name	Migrations	Source	Target	Repository	Persistent Volumes	Last Status
demo plan	⌚ 2	Summit Demo Source Cluster	Target cluster	mydemobucket	2	Migrated Success
demo2	⌚ 2	Summit Demo Source Cluster	Target cluster	mydemobucket	2	Migrated Success

Cloud-like Simplicity, Everywhere

Full-stack automated operations across any on-premises,
cloud, or hybrid infrastructure



CLOUD-LIKE SIMPLICITY, EVERYWHERE

OpenShift Cluster Manager cloud.redhat.com/openshift

Enhanced OpenShift Web Console Integration

Bi-directional navigation to and from the OpenShift web console for cluster administrators. Deep linking from OCM to the console where relevant.

OpenShift Dedicated cluster management

Self-service cluster provisioning, scaling, and basic management for OpenShift Dedicated customers (4.1+).

Cluster Monitoring

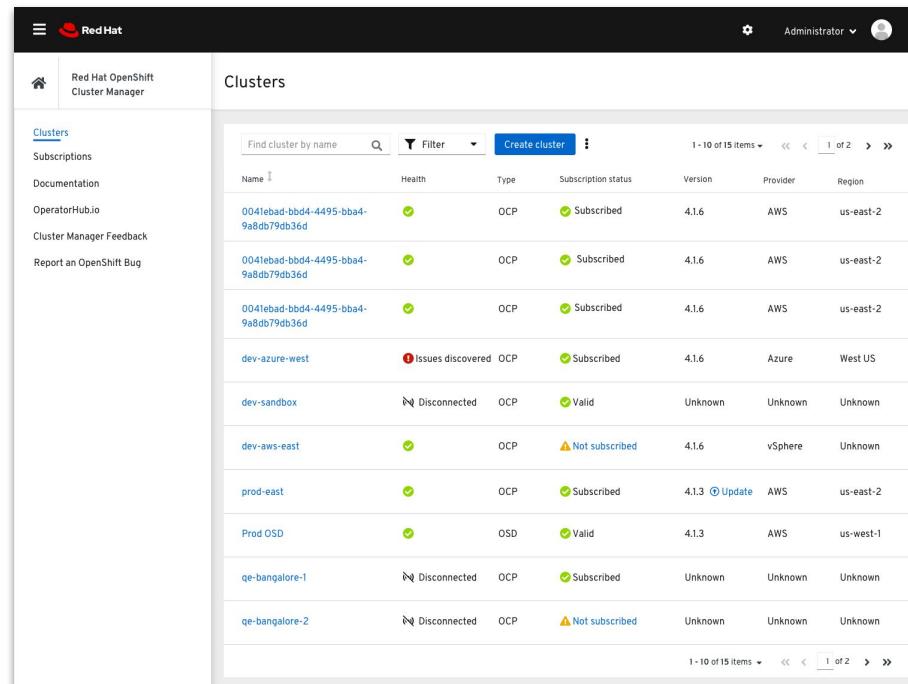
New tab available on all cluster detail pages helps cluster administrators discover critical issues impacting their clusters.

Cluster Updates & New Metrics

OCM provides a link to your cluster's settings page when updates are available for your cluster. Infrastructure provider and region are now captured and displayed for all clusters, where available.

More OpenShift Install Options

New infrastructure providers, including CodeReady Containers, are now listed as install options.



The screenshot shows the Red Hat OpenShift Cluster Manager web interface. The top navigation bar includes the Red Hat logo and the title "Red Hat OpenShift Cluster Manager". On the left, there is a sidebar with links for "Clusters", "Subscriptions", "Documentation", "OperatorHub.io", "Cluster Manager Feedback", and "Report an OpenShift Bug". The main content area is titled "Clusters" and displays a table of 15 items. The columns are: Name, Health, Type, Subscription status, Version, Provider, and Region. The data in the table is as follows:

Name	Health	Type	Subscription status	Version	Provider	Region
0041ebad-bbd4-4495-bba4-9a8db79db36d	✓	OCP	✓ Subscribed	4.1.6	AWS	us-east-2
0041ebad-bbd4-4495-bba4-9a8db79db36d	✓	OCP	✓ Subscribed	4.1.6	AWS	us-east-2
0041ebad-bbd4-4495-bba4-9a8db79db36d	✓	OCP	✓ Subscribed	4.1.6	AWS	us-east-2
dev-azure-west	! Issues discovered	OCP	✓ Subscribed	4.1.6	Azure	West US
dev-sandbox	! Disconnected	OCP	✓ Valid	Unknown	Unknown	Unknown
dev-aws-east	✓	OCP	⚠ Not subscribed	4.1.6	vSphere	Unknown
prod-east	✓	OCP	✓ Subscribed	4.1.3 (Update)	AWS	us-east-2
Prod OSD	✓	OSD	✓ Valid	4.1.3	AWS	us-west-1
qe-bangalore-1	! Disconnected	OCP	✓ Subscribed	Unknown	Unknown	Unknown
qe-bangalore-2	! Disconnected	OCP	⚠ Not subscribed	Unknown	Unknown	Unknown

Generally Available



IMPROVE OBSERVABILITY

Metering

ShowBack/ChargeBack Reports available from OperatorHub

- Base functionality on all providers
- Tie into cloud providers for \$\$
- Included reports for 80% use-case
- Customers can write custom reports and time periods
- Offer basic UI reporting but main use is to plug into customer's BI tool of choice

Name	Namespace	Labels	Created At
cluster-cpu-capacity	openshift-metering	operator-metering=true	7 minutes ago
cluster-cpu-capacity-raw	openshift-metering	operator-metering=true	7 minutes ago
cluster-cpu-usage	openshift-metering	operator-metering=true	7 minutes ago
cluster-cpu-usage-raw	openshift-metering	operator-metering=true	7 minutes ago
cluster-cpu-utilization	openshift-metering	operator-metering=true	7 minutes ago
cluster-memory-capacity	openshift-metering	operator-metering=true	7 minutes ago
cluster-memory-capacity-raw	openshift-metering	operator-metering=true	7 minutes ago
cluster-memory-usage	openshift-metering	operator-metering=true	7 minutes ago
cluster-memory-usage-raw	openshift-metering	operator-metering=true	7 minutes ago
cluster-memory-utilization	openshift-metering	operator-metering=true	7 minutes ago



Generally Available

IMPROVE OBSERVABILITY

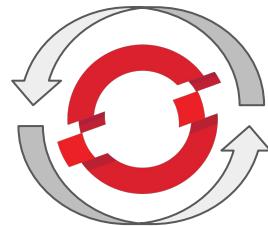
Cluster Logging

Overall performance improvements in collecting logs in 4.x w/ fluentd

- Tripled the amount of logs we collect and at the same time reduced resource consumption by ~50%

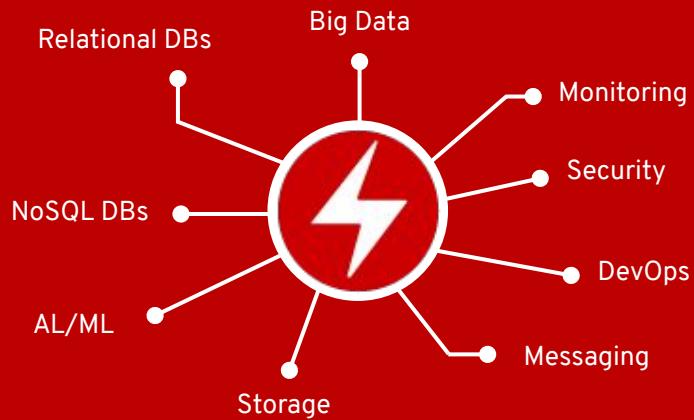
3.11	4.1
750 - 1000 1Kb messages/second/fluentd over 10 namespaces at ~600Mb RSS memory used and 75-85% utilization of a single core.	2250 - 2500 1Kb messages/second/fluentd over 10 namespaces at ~325Mb RSS memory used and 35% utilization of a single core.

Be notified when something is wrong with Fluentd with new alerting rules in OCP 4.2.



A broad ecosystem of workloads

Operator-backed services allow for a
SaaS experience on your own infrastructure

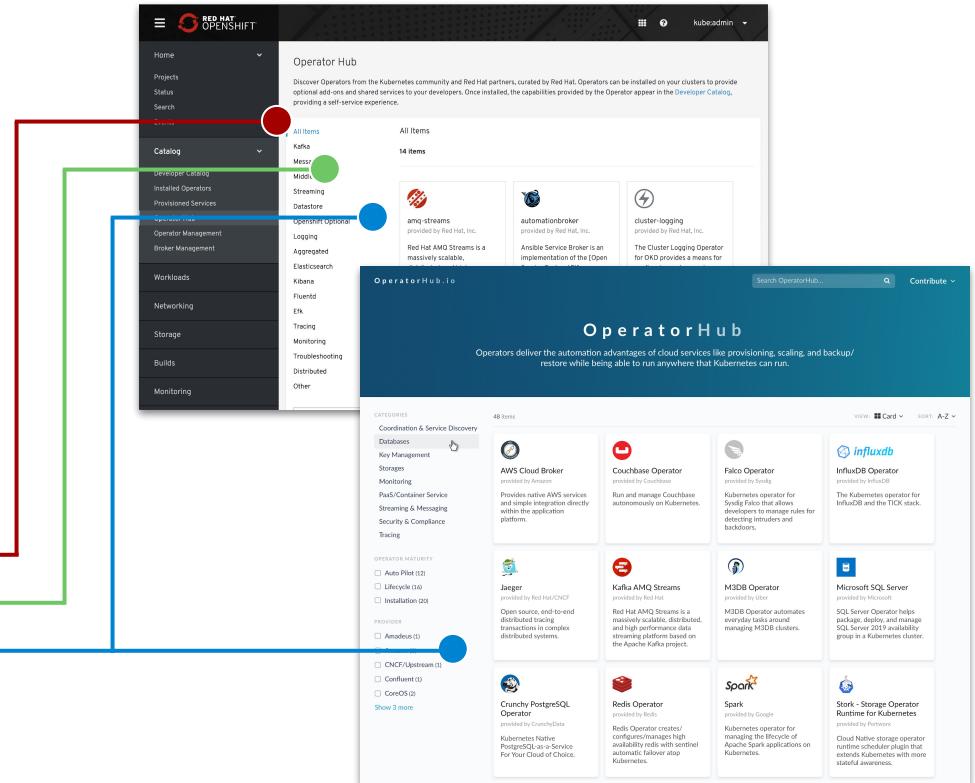


OPERATORHUB

- Accessible to admins only
- Discovery/install of all optional components and apps
- Upstream and downstream content
- ISV partners will support their Operators

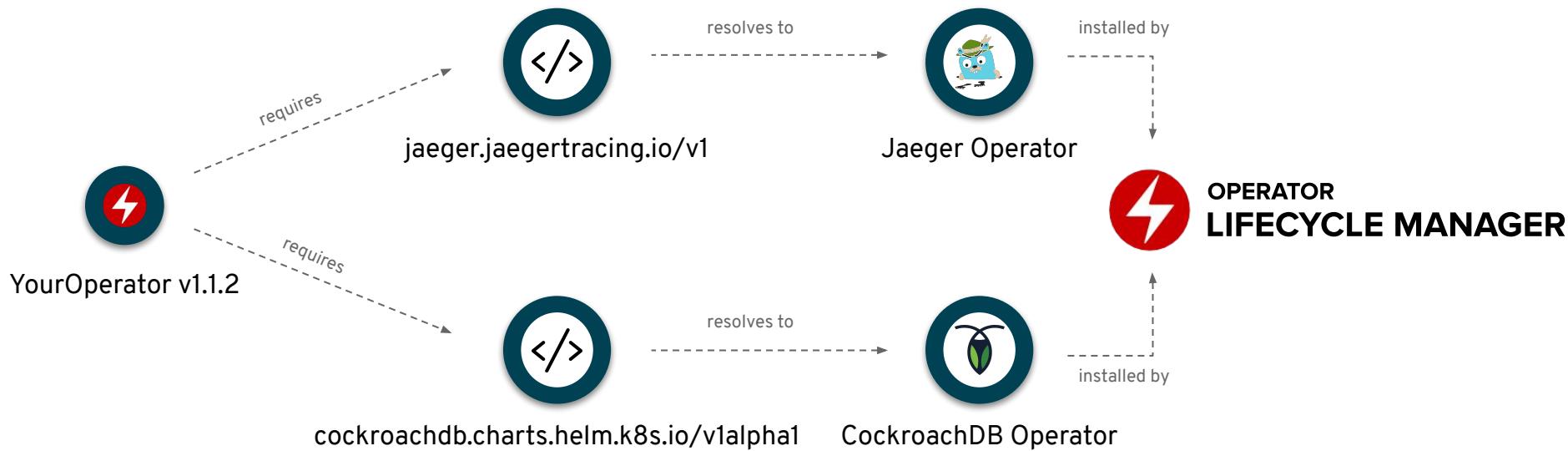
TYPES OF OPERATORS

Red Hat Products
ISV Partners
Community



4.2 Automated Dependency Resolution

Operator Framework Dependency Graphs



OpenShift Console

The future is now.

Extending the
Console

Improve
Observability

Administration
made easy

Scaling your
Cluster

EXTENDING THE CONSOLE

Expose Third Party App Console for Operator-backed Services

This is an example notification message with an optional link. [Optional link text](#)

Red Hat Applications

OpenShift Cluster Manager

Third Party Applications

Couchbase Server Web Console

Name	Namespace	Deployment	Status	Provided APIs
AMQ Streams	NS tony	amq-streams-cluster-operator	InstallSucceeded Up to date	Kafka Kafka Connect Kafka Connect S2I Kafka MirrorMaker View 2 more...
Couchbase Operator	NS tony	couchbase-operator	InstallSucceeded Up to date	Couchbase Cluster
etcd	NS tony	etcd-operator	InstallSucceeded Up to date	etcd Cluster etcd Backup etcd Restore
Knative Serving Operator	NS tony	knative-serving-operator	InstallSucceeded Up to date	Knative Serving

Expose Operator-backed service Console through console CRD

Easily integrate/onboard third-party user interfaces in order to develop, administer, and configure Operator-backed services.

IMPROVE OBSERVABILITY

Enhanced Visibility with the New Dashboard

Cluster-wide Dashboard gives Admins Clear Insights

Drill down in context from the new dashboard widgets:

- Cluster Details
- Cluster Health
- Cluster Inventory
- Cluster Capacity
- Cluster Utilization
- Cluster Utilization
- Top Consumers

The screenshot shows the OKD Cluster-wide Dashboard interface. The left sidebar has a dark theme with a navigation menu including Home, Dashboards, Projects, Search, Events, Operators, Workloads, Networking, Storage, Builds, Monitoring, Compute, and Administration. The main content area is titled "Dashboards" and includes tabs for Overview, Persistent Storage, and Object Service. The Overview tab is active, displaying sections for Cluster ID (47cf4c68-9022-482b-8d31-6cb0faa3e4c2), Provider (AWS), OpenShift Version (4.2.0-0.ci-2019-07-24-105737), Cluster Inventory (6 Nodes, 203 Pods, 3 PVCs, VMs, Bare Metal Hosts), Cluster Health (warning: Multiple errors, Cluster health is degraded), Alerts (multiple alerts about throttling and StatefulSets), Cluster Capacity (CPU, Memory, Storage, Network usage), and Events (recent events for nodes and master). A bottom footer shows "Pods by CPU time" with a dropdown for "By CPU".

Configuring Authentication for your desired Identity Providers

Customize and determine how users log into the cluster

- Basic Authentication
- GitHub
- GitLab
- Google
- HTPasswd
- Keystone
- LDAP
- OpenID Connect
- Request Header

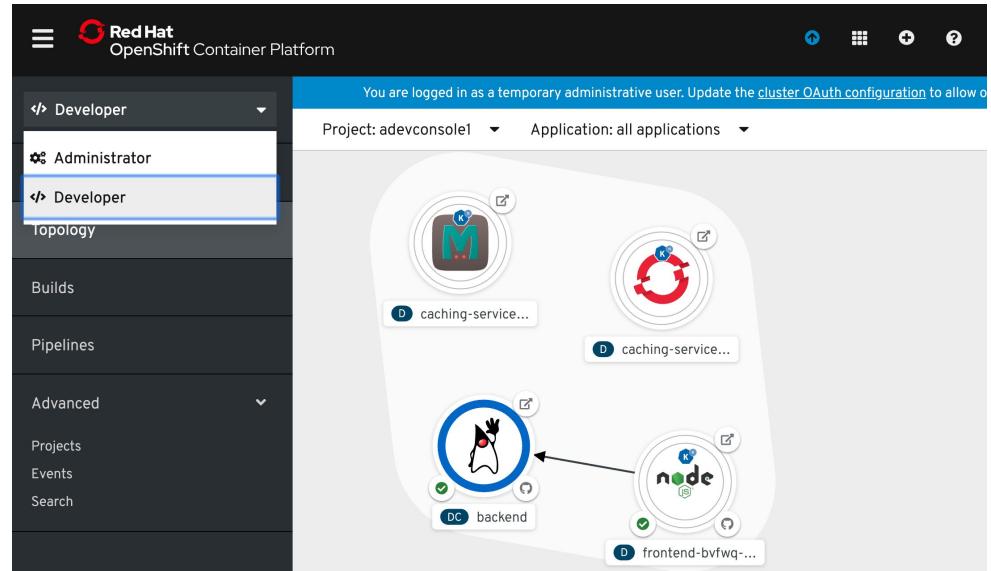
The screenshot shows the Red Hat OpenShift Container Platform web interface. On the left, a sidebar menu is open under the 'Administration' section, showing options like Cluster Status, Cluster Settings, Namespaces, Service Accounts, Roles, Role Bindings, Resource Quotas, Limit Ranges, Chargeback, API Explorer, and Custom Resource Definitions. A dropdown menu is overlaid on this sidebar, listing 'Add' and several identity provider options: Basic Authentication, GitHub, GitLab, Google, HTPasswd, Keystone, LDAP, OpenID Connect, and Request Header. To the right of the sidebar, the main content area displays the 'Identity Providers' configuration page. It includes sections for Labels (No labels), Annotations (1 Annotation), and Created At (Jul 10, 7:08 am). Below this, there's a sub-section titled 'Identity Providers' with the sub-instruction: 'Identity providers determine how users log into the cluster.' A second dropdown menu is open here, listing the same set of identity providers: Basic Authentication, GitHub, GitLab, Google, HTPasswd, Keystone, LDAP, OpenID Connect, and Request Header. The right side of the interface shows a detailed configuration form for adding a 'Keystone Authentication' provider, with fields for Name (keystone), Domain Name, URL, CA File, Certificate, and Key.

Web console - Developer Perspective

Key Features

An alternative perspective in the OpenShift UI that will sit beside the admin console and focus on developer use cases.

All OpenShift developer tool UIs will be surfaced here...though some (like CodeReady Workspaces) will be links out to unique UIs.



Easily Create an Application

Key Features

- Source in Git
- Existing container image
- YAML definition
- Build from Dockerfile
- Explore catalog
- Databases from catalog

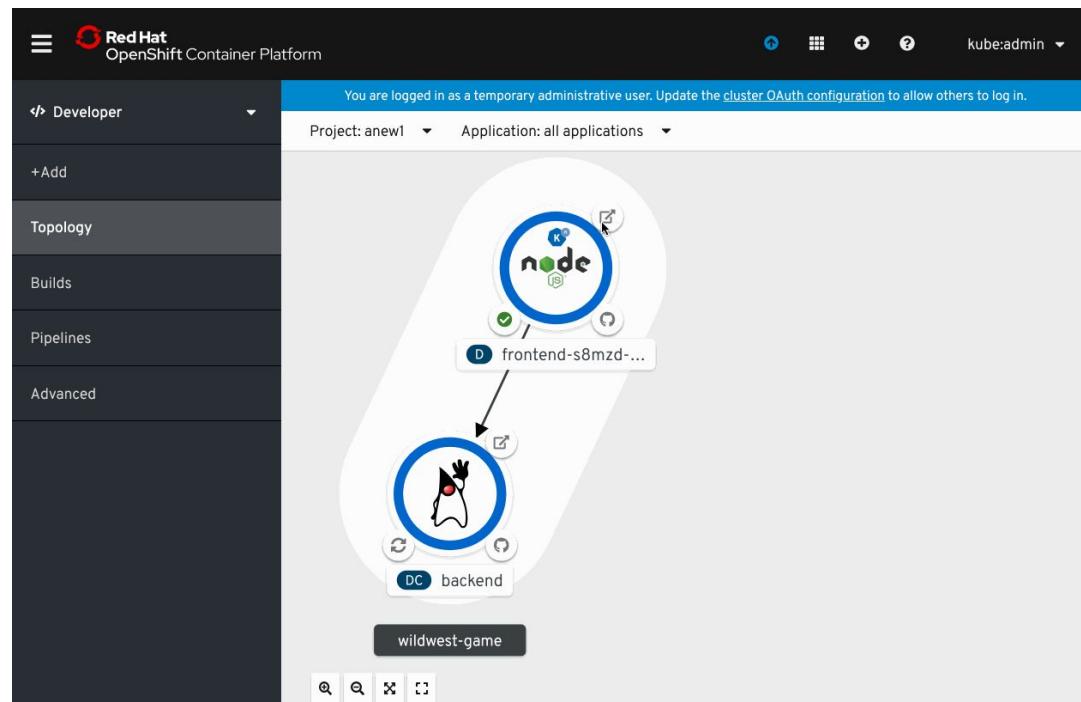
The screenshot shows the Red Hat OpenShift Container Platform developer interface. The top navigation bar includes the Red Hat logo and the text "OpenShift Container Platform". The left sidebar has a "Developer" section with options like "+Add", "Topology", "Builds", "Pipelines", and "Advanced". The main content area displays a message: "You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow other..." followed by "Project: adevconsole1" and "Application: all applications". A central heading says "Select a way to create an application, component or service from one of the options." Below this are six cards arranged in a 2x3 grid:

- From Git**: Import code from your git repository to be built and deployed.
- Container Image**: Deploy an existing image from an image registry or image stream tag.
- From Catalog**: Browse the catalog to discover, deploy and connect to services.
- From Dockerfile**: Import your Dockerfile from your git repo to be built & deployed.
- YAML**: Create or replace resources from their YAML or JSON definitions.
- Database**: Browse the catalog to discover database services to add to your application.

Application Topology

Key Features

- View structure and status of app components
- Drill into specific workloads
- Quickly navigate to pod logs
- Manually scale
- Pod donut!
- Access route/URL
- Linked build and source



Next wave of developer tools

OpenShift has all of the latest tools to make
your devs more productive

Code
Serverless

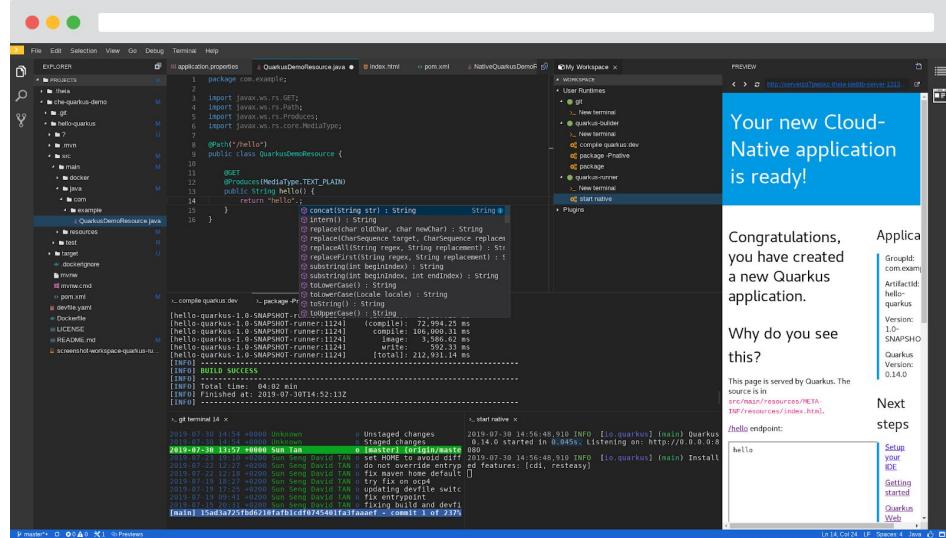
Containers
Service Mesh

NEXT WAVE OF DEVELOPER TOOLS

CodeReady Workspaces 2.0

Based on Eclipse Che 7

- Kubernetes-based developer workspaces:** Fully containerized developer workspaces allowing to bring your K8S application runtime easily in your dev environment.
- New Editor:** New default web-based editor provides a VSCode like experience in the browser.
- Devfile:** Configure a devfile for your project and get reproducible and portable developer environments.
- VSCode plug-ins compatibility**
- Swappable Editor**
- OpenShift VSCode Plug-in**
- Easier to Monitor and Administrate:** Prometheus and Grafana dashboards.



Shipped independently from 4.2, shortly after

CodeReady Containers: OpenShift on your Laptop

Provides a pre-built development environment based on **Red Hat Enterprise Linux** and **OpenShift** for quick container-based application development. Use with OpenShift on-premises or cloud.

```
$ crc setup  
Prepare your machine for running OpenShift  
  
$ crc start -b  
crc-hyperkit-4.2.0.crcbundle  
Start with the Hyperkit 4.2 bundle  
  
$ crc status  
Get the status of the cluster
```

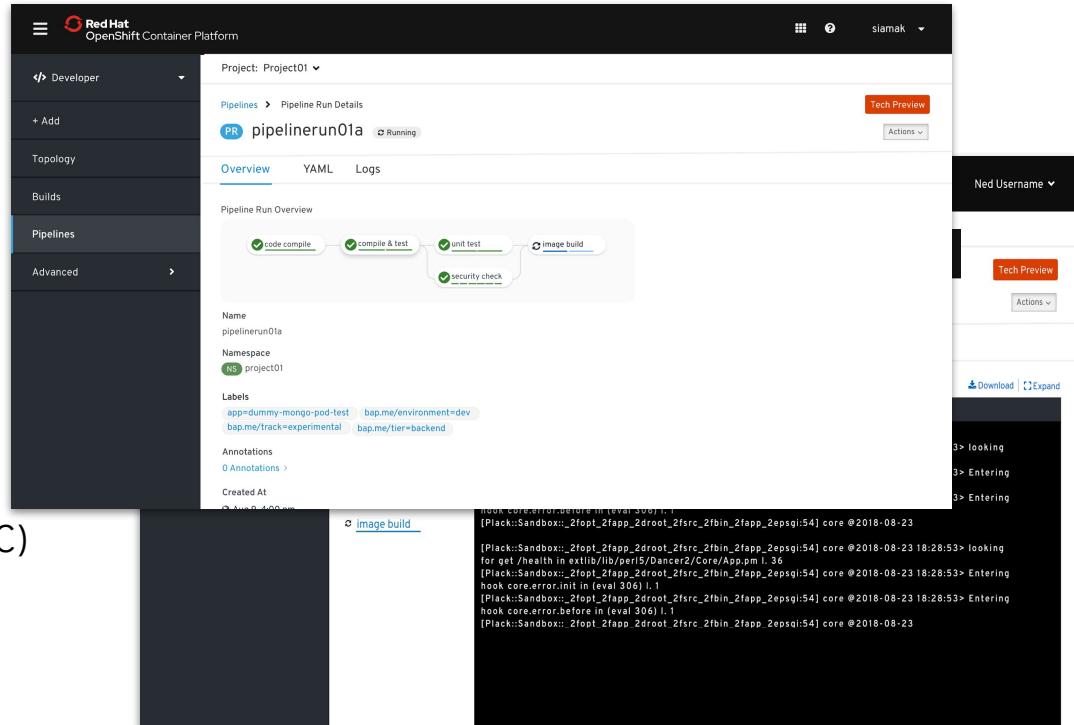
- Based on OpenShift 4.x
- Linux (libvirt)
- Windows (Hyper-V)
- MacOS (Virtualbox)
- External beta available
- Replaces the 3.x experiences around:
 - Minishift
 - CDK
 - oc cluster up

Use It To: Simplify direct-to-OpenShift development on laptops.

NEXT WAVE OF DEVELOPER TOOLS

Cloud-native CI/CD with OpenShift Pipelines

- Based on Tekton Pipelines
- Runs serverless (no babysitting!)
- Containers as building blocks
- Deploy to multiple platforms
- Standard CRDs
- Build images with Kubernetes tools
(s2i, buildah, kaniko, jib, buildpack, etc)
- Pipelines portable to any Kubernetes
- Available in OperatorHub



NEXT WAVE OF DEVELOPER TOOLS



OpenShift Serverless

Key Features

- Familiar to Kubernetes users. Native.
 - Scale to 0 and autoscale to N based on demand
 - Applications and functions. Any container workload.
 - Powerful eventing model with multiple event sources.
 - Operator available via OperatorHub
 - Knative v0.7.1 (v1beta1 APIs)
 - No vendor lock in

[Learn more](#)

<https://openshift.com/learn/topics/serverless>

<https://redhat-developer-demos.github.io/knative-tutorial>

The screenshot shows the Red Hat OpenShift Container Platform interface. The left sidebar includes navigation links for Home, Dashboards, Projects, Search, Explore, Events, Operators (selected), Workloads, Serverless, Services, Revisions, Routes, and Networking. The main content area has two tabs: 'Operator Details' and 'Application Overview'. The 'Operator Details' tab for 'Serverless Operator' shows it is version 1.0.0 provided by Red Hat. The 'Application Overview' tab for 'spring-petclinic-bchpw-deployment' shows a scaling status of 4/10, with a large blue circle indicating progress. It also lists deployment details: Name: spring-petclinic-bchpw-deployment, Namespace: markito-rhte, Update Strategy: RollingUpdate, MaxUnavailable: 25% of 10 pods, and Labels: app=spring-petclinic-bchpw, app.kubernetes.io/name=spring-petclinic-bchpw, app.kubernetes.io/namespace=markito-rhte, serving.knative.dev/configurable=true, serving.knative.dev/icon=spring-petclinic-bchpw, serving.knative.dev/iconUrl=https://serving.knative.dev/_static/spring-petclinic-bchpw/icon.png, serving.knative.dev/revisionId=8b3c3d49f5dcb1-1, serving.knative.dev/revisionName=spring-petclinic-bchpw-1, serving.knative.dev/revisionVersion=v1, serving.knative.dev/revisionWeight=1.0, serving.knative.dev/revisionWeightPercentage=100, and serving.knative.dev/scale=true.

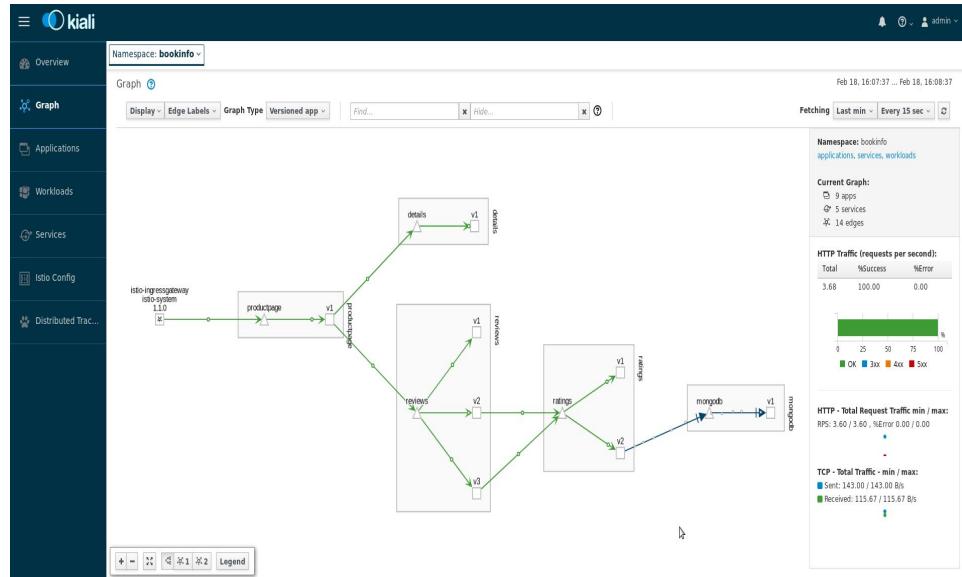


NEXT WAVE OF DEVELOPER TOOLS

OpenShift Service Mesh

Key Features

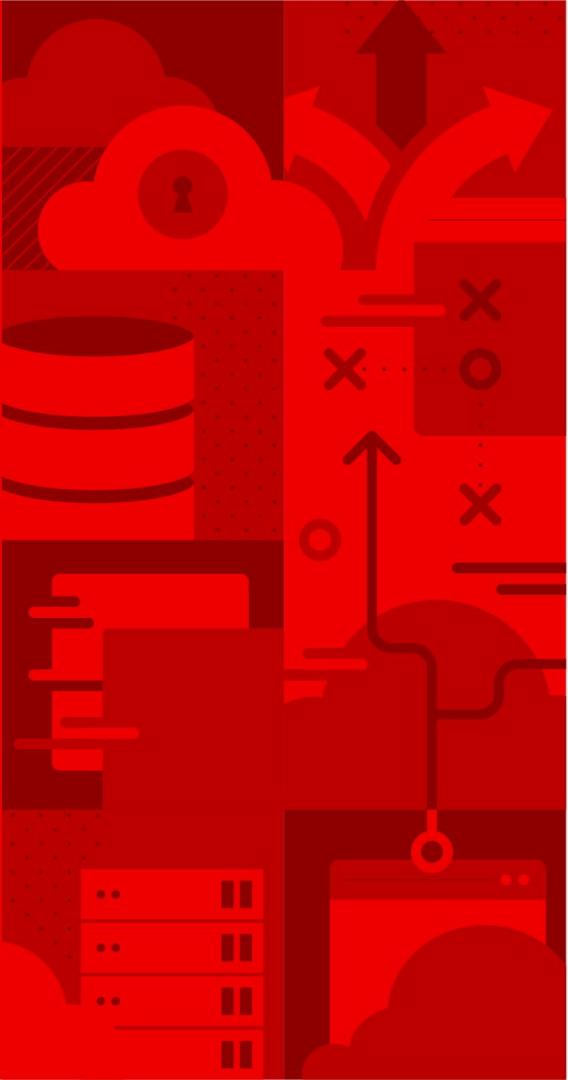
- A dedicated network for service to service communications
- Observability and distributed tracing
- Policy-driven security
- Routing rules & chaos engineering
- Powerful visualization & monitoring
- Available via OperatorHub



Roadmap

2019 Roadmap

Q2 CY2019 OpenShift 4.1		Q3 CY2019 OpenShift 4.2		Q4 CY19/Q1 CY20 OpenShift 4.3	
HOSTED	PLATFORM	HOSTED	PLATFORM	HOSTED	PLATFORM
HOSTED	PLATFORM	HOSTED	PLATFORM	HOSTED	PLATFORM
HOSTED	PLATFORM	HOSTED	PLATFORM	HOSTED	PLATFORM
● OpenShift Serverless (Knative) - DP	● Developer Console GA	● OpenShift Serverless (Knative) - GA	● Kubernetes 1.13 with CRI-O runtime	● Kubernetes 1.16 w/ CRI-O runtime	● cloud.redhat.com - Subscription Mgmt Improvements
● OpenShift Pipelines (Tekton) DP2	● OpenShift Pipelines (Tekton) DP3	● OpenShift Pipelines (Tekton) - TP	● RHEL CoreOS, RHEL7	● Automated Installer for RHV	● Insights Operator
● CodeReady Workspaces	● CodeReady Containers Alpha	● CodeReady Containers GA	● Automated Installer for AWS	● Private/Internal Clusters support from the installer	
● CodeReady Containers Alpha	● Developer CLI (odo) Beta	● Developer CLI (odo) GA	● Pre-existing Infra Installer for Bare Metal, VMware, AWS	● Deploy to pre-existing VPC & Subnets	
● Developer CLI (odo) Beta			● Kubernetes 1.14 w/ CRI-O runtime	● OVN GA w/ Windows Networking Integration (Planned)	
● OperatorHub	● OperatorHub Enhancements	● Disconnected Install and Update	● Cluster-wide Egress Proxy	● FIPS	
● Operator Lifecycle Manager	● Operator Deployment Field Forms	● Automated Installer for Azure, OSP, GCP	● OVN Tech Preview	● Pre-existing Infra Installer for OSP	
● Service Mesh (~4 month after)		● Pre-existing Infra Installer for GCP	● OpenShift Container Storage 4.2 (1 month after)	● OpenShift Container Storage 4.3	
			● Insights Operator		
● Kubernetes 1.13 with CRI-O runtime	● Kubernetes 1.14 w/ CRI-O runtime	● Kubernetes 1.16 w/ CRI-O runtime			
● RHEL CoreOS, RHEL7	● Disconnected Install and Update	● Automated Installer for Azure, OSP, GCP			
● Automated Installer for AWS	● Automated Installer for AWS	● Pre-existing Infra Installer for GCP			
● Pre-existing Infra Installer for Bare Metal, VMware, AWS	● Pre-existing Infra Installer for Bare Metal, VMware, AWS	● Cluster-wide Egress Proxy			
● Automated, one-click updates	● Multus (Kubernetes multi-network)	● OVN Tech Preview			
● Multus (Kubernetes multi-network)	● Quay v3	● OpenShift Container Storage 4.2 (1 month after)			
● Quay v3					
● cloud.redhat.com - Multi-Cluster Mgmt					
● OCP Cluster Subscription Management					
● Azure Red Hat OpenShift					
● OpenShift Dedicated consumption pricing					



Questions?

try.openshift.com

 linkedin.com/company/red-hat

 facebook.com/redhatinc

 youtube.com/user/RedHatVideos

 twitter.com/RedHat