

# Red Hat Advanced Cluster Management for Kubernetes

## Master Deck

Presenter's Name

Title

Presenter's Name

Title



# Agenda

- ▶ Market Trends and Challenges
- ▶ Key Personas
- ▶ Introducing Red Hat Advanced Cluster Management for Kubernetes
- ▶ Detailed Use Cases
- ▶ RHACM & OpenShift
- ▶ What's new in RHACM 2.5
- ▶ Architecture and Installation Overview
- ▶ Customer Success Stories
- ▶ Resources





# Market Trends and Challenges

# Hybrid multicloud management is really hard

As organizations deploy more across multiple clouds, new challenges arise.

- ▶ **Difficult and error prone** to manage at scale
- ▶ **Inconsistent security controls** across environments
- ▶ **Overwhelming to verify** components, configurations, policies, and compliance

IDC Survey of 200 US-based \$1B companies actively using two or more “infrastructure clouds” for production applications

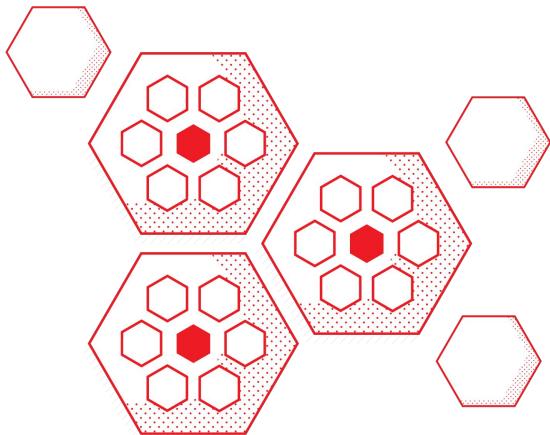
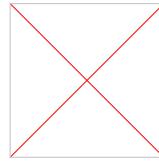


—  
Using multiple infrastructure clouds\*



—  
Using multiple public clouds and one or more private/dedicated clouds\*

## Kubernetes adoption leads to multicloud



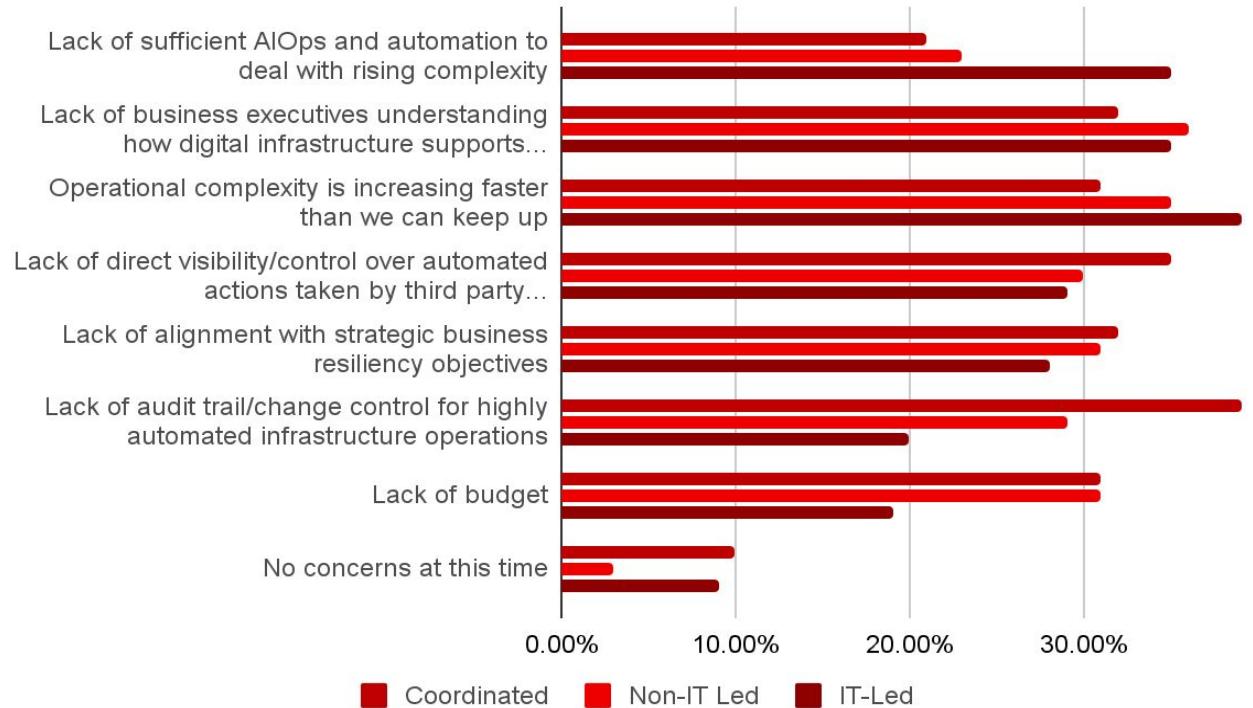
"As Kubernetes gains adoption across the industry, scenarios are arising in which I&O teams are finding **they must deploy and manage multiple clusters**, either in a single region on-premises or in the cloud, or across multiple regions....for a number of reasons, including multi-tenancy, disaster recovery, and with hybrid, multicloud, or edge deployments."



# Digital infrastructure concerns are around automation

Smaller, IT-Led organizations are concerned about lack of automation and AIOps while larger, more collaborative organizations are more concerned about lack of change control and limited business executive understanding of how digital infrastructure supports business resiliency

**What are the greatest concerns regarding organizations' overall digital infrastructure strategy being able to fully support business resiliency plans?**



# Remote container clusters are becoming modern-day LAN servers

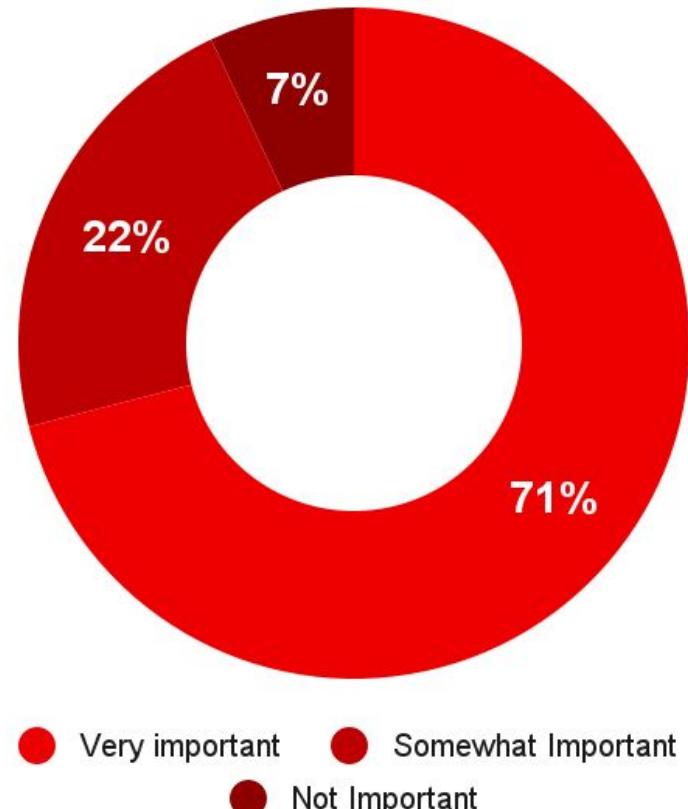
In many ways, remote container clusters are becoming modern-day LAN servers, which requires tooling for managing the distributed clusters.

Software and service providers are differentiating their offerings by providing more complete solutions, adding capabilities such as

- ▶ **Distributed management of multiple clusters**
- ▶ **Application lifecycle integration**
- ▶ **Policy management**
- ▶ **Monitoring and security to the foundational container orchestration capabilities**

# Unified, consistent, autonomous operations priorities

Importance of Unified Management Control Plane



\*Most important reasons organizations need a unified management control plane for all digital infrastructure resource is to



—  
Improve data integration and data protection



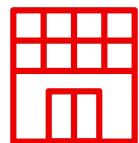
—  
Optimize infrastructure costs and usage

# Where is the growth in cluster deployments?



## Small Scale Dev Teams

- Managing and syncing across Dev/QE/Pre-Prod/Prod clusters can be difficult



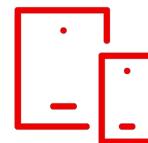
## Medium Scaled Organizations

- Retail with small clusters across 100s of locations
- Organizations with plan for growth 10-15 clusters moving to 100s



## Large scale

- Global organizations with 100s of clusters, hosting thousand of applications
- Large Retail with 1000s of stores



## Edge scale / Telco

- 100s of zones, 1000s of clusters and nodes across complex and air-gapped topologies



## Reasons for deploying clusters



Application availability



Reduced latency



Address industry standards



Geopolitical data residency guidelines



Disaster recovery



Edge deployments



CapEx cost reduction



Avoid vendor lock-in



# Multicloud management challenges

How do I normalize and centralize key functions across environments?

Management requirements

## </> Centralized operations

Build and deploy a containerized app

- ▶ Easy cluster provisioning
- ▶ Controlling cluster configuration drift
- ▶ Ensuring app deployment from development to production

## ∞ DevOps

Develop, test, and produce clusters

- ▶ Consistent cluster provisioning
- ▶ Policy enforcement and governance across development, test, and production clusters
- ▶ Finding/modifying resources across clusters

## Cloud Hybrid multicloud

Automated clusters deployed across public, private clouds, edge, in different geographies

- ▶ Single pane of glass visibility
- ▶ Deploying and distributing applications at scale
- ▶ Auditing and compliance
- ▶ Configuration with zero touch provisioning

Single cluster

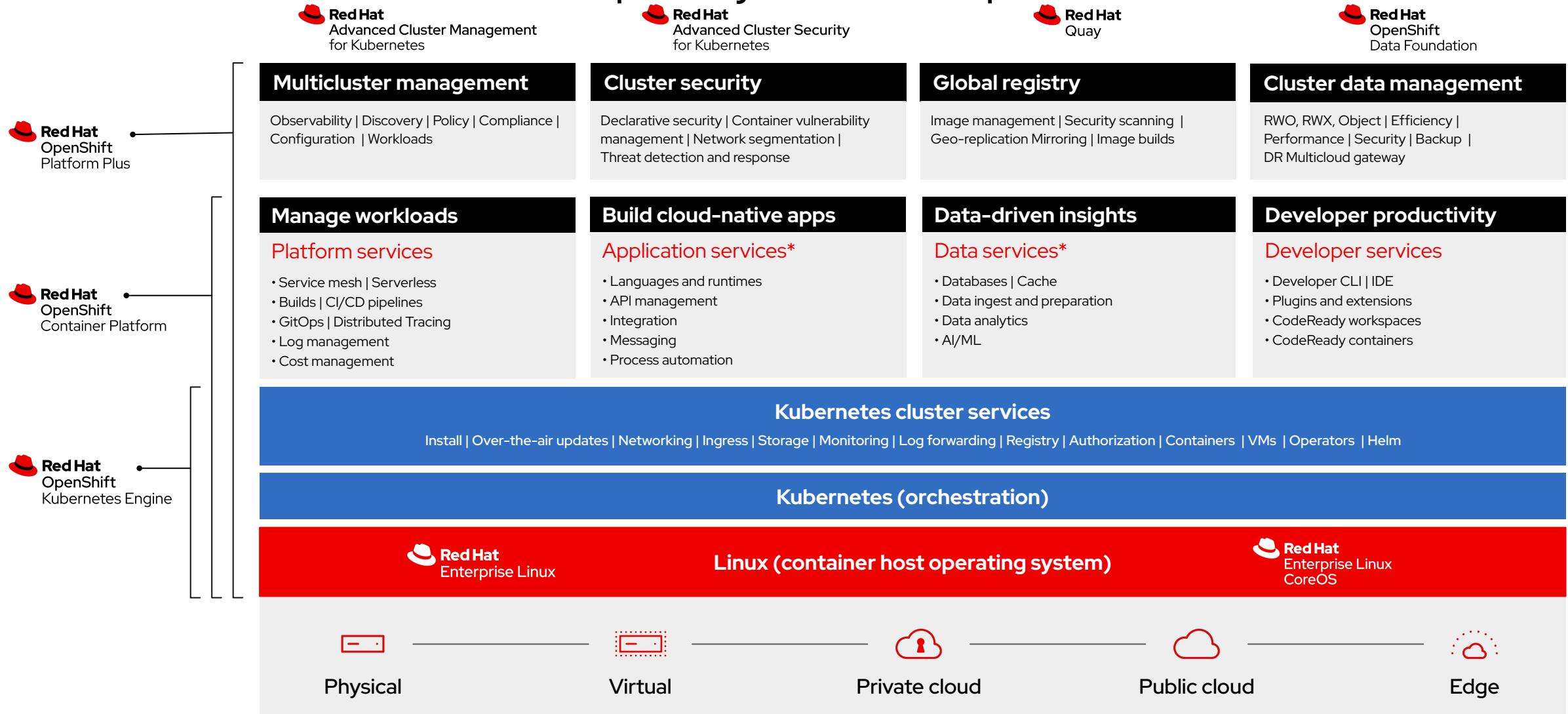
Multicloud growth

Distributed multicloud

# Where are our customers?



# Red Hat open hybrid cloud platform



\* Red Hat OpenShift® includes supported runtimes for popular languages/frameworks/databases. Additional capabilities listed are from the Red Hat Application Services and Red Hat Data Services portfolios.

\*\* Disaster recovery, volume and multicloud encryption, key management service, and support for multiple clusters and off-cluster workloads requires OpenShift Data Foundation Advanced

# Open Source Commitment

- Open Cluster Management has been accepted as a CNCF Sandbox
  - <https://www.cncf.io/projects/open-cluster-management/>
- Collaboration in Key Kubernetes Special Interest Groups
  - Sig-MultiCluster
  - Sig-Application
  - Sig-Policy
- Growing together with support from partners and contributors
  - Ant Group
  - Alibaba
  - Tencent
  - Microsoft \*\*

The screenshot shows the official website for Open Cluster Management. At the top, there's a navigation bar with links for Community, Contribute, Document, English, and other language options. The main heading is "Open Cluster Management" with a logo. Below the heading, a large call-to-action text reads: "Make working with many Kubernetes clusters super easy regardless of where they are deployed". A detailed description follows: "Open Cluster Management is a community-driven project focused on multicluster and multicloud scenarios for Kubernetes apps. Open APIs are evolving within this project for cluster registration, work distribution, dynamic placement of policies and workloads, and much more." There's a prominent "Get Started" button and a note: "If you like Open Cluster Management, give it a star on GitHub!". The background features a futuristic illustration of a person in a spacesuit working at a desk with a computer monitor, surrounded by floating hexagonal shapes and celestial bodies.

## Feature Overview

This section provides a high-level overview of the project's key features:

- Cluster inventory**: Registration of multiple clusters to a hub cluster to place them for management.
- Work distribution**: The work API that enables resources to be applied to managed clusters from a hub cluster.
- Content placement**: Dynamic placement of content and behavior across multiple clusters.
- Vendor neutral APIs**: Avoid vendor lock-in by using APIs that are not tied to any cloud providers or proprietary platforms.

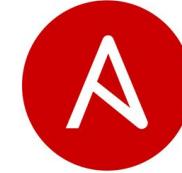
# Strong open source community & ecosystem



**Open Policy Agent**



Hive



**Kyverno**



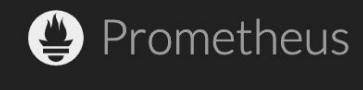
**argo**



**Red Hat**  
Advanced Cluster  
Management  
for Kubernetes



**Open Cluster  
Management**



Prometheus



# Solving real customer challenges



Enforced policies at scale across clusters using the governance and risk framework in RHACM



End-to-end multi-cloud management, visibility, and compliance enforcement for multiple clusters at scale



Faster deployment and ease of managing clusters and applications at a number of edge locations



Transformed several applications to become cloud-native and prepared for a full migration to cloud



Seamless management and operations of the complete Kubernetes environment



Increased observability and control for managing the Kubernetes environment

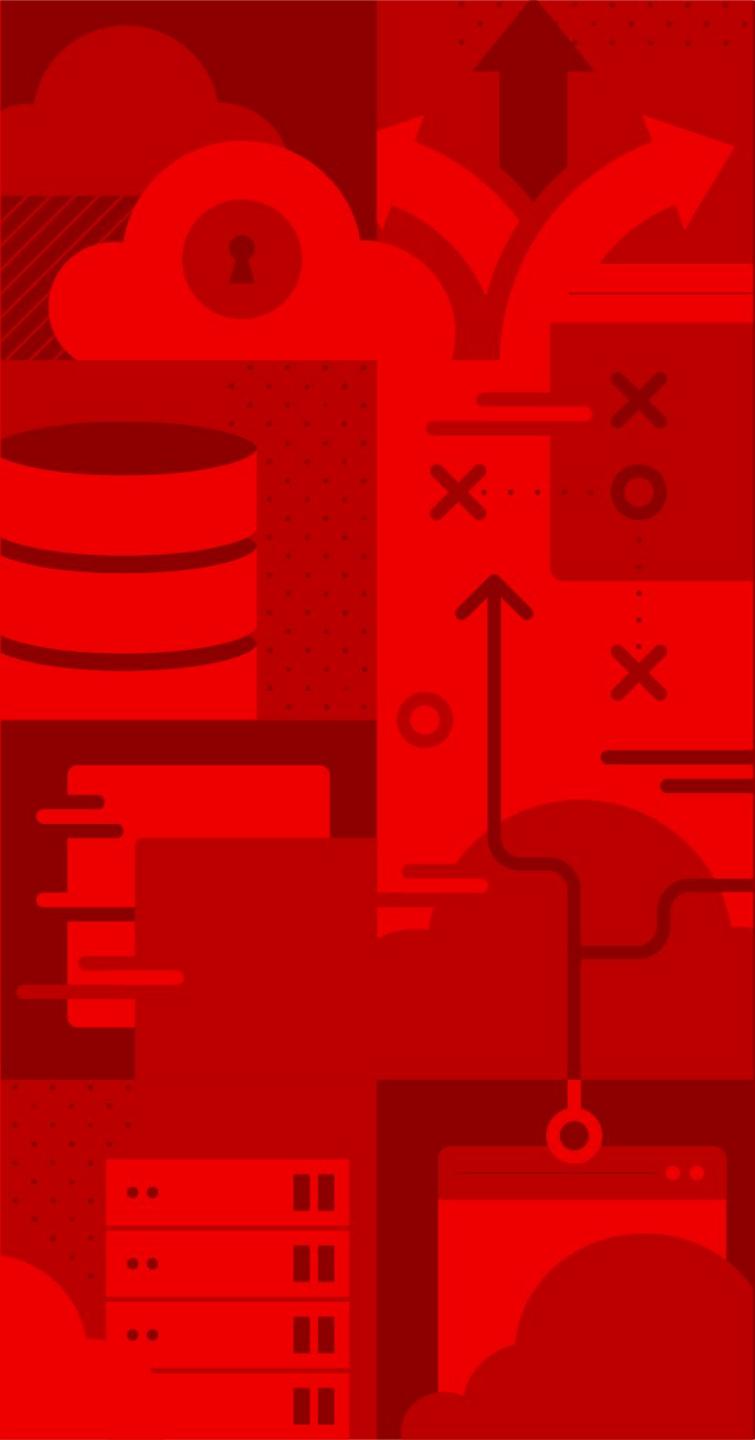


# Our open hybrid cloud portfolio accelerates digital transformation

Red Hat Advanced Cluster Management for Kubernetes enables enterprises to build highly agile, scalable architectures with enhanced cluster security that can be deployed at any location.

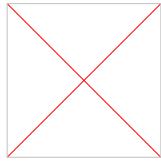
Red Hat aims to scale automation across enterprise through a three-pronged strategy comprising: business-critical workload automation, Red Hat Ansible Automation for infrastructure and Red Hat Advanced Cluster Manager for container management.

- ▶ **Innovative container management**
- ▶ **Deeper integration with Red Hat Ansible Automation Platform for container automation**



# Key Personas

# IT Operations



“How can I manage the lifecycle of multiple clusters regardless of where they reside using a single control plane?”

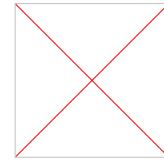
“How can I quickly get to the root cause of failed components?”

“How do I monitor usage across multiple clouds?”

—  
**Louise Mar**  
Senior IT Ops, Acme Unlimited



# SRE/DevOps



“How do I get a simplified understanding of my cluster health and the impact on my application availability?”

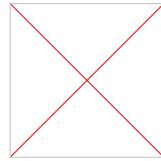
“How do I automate provisioning and destroying of my clusters, workload placement based on capacity and policies, and the pushing of application from dev to prod?”

**Clete Liedl**

Senior DevOps Engineer, Acme Unlimited



# SecOps



“How do I ensure all my clusters are compliant with my defined policies?”  
“How do I set consistent security policies across diverse environments and ensure enforcement?”  
“How do I get alerted on any configuration drift and remediate it?”

**Jay Schefter**  
Senior Security Engineer, Acme Unlimited





# Introducing Red Hat Advanced Cluster Management For Kubernetes

# Robust. Proven. Award winning.

## End-to-end automation with Red Hat Ansible Automation Platform integration



Multicloud lifecycle management



Policy driven governance, risk, and compliance



Advanced application lifecycle management



Multicloud observability for health and optimization



Multicloud networking for interconnecting

The screenshot displays three main panels of the Red Hat Advanced Cluster Management for Kubernetes web interface:

- Overview:** Shows a summary of clusters across Google, AWS, Microsoft, and IBM. It includes metrics for Applications (4), Clusters (10), Kubernetes types (1), Regions (5), Nodes (60), and Pods (2513). Below this are three circular dashboards: Cluster compliance (66% compliant), Pods (100% running), and Cluster status (100% ready).
- Governance and risk:** A dashboard for NIST-CSF, NIST SP 800-53, and PCI standards. It shows cluster violations and policy violations. For example, under NIST-CSF, there are 10/10 cluster violations and 8/11 policy violations.
- Application pacman-appl:** A detailed view of an application named "pacman-appl". It shows the deployment pipeline: Ansiblejob (pre-hook), Application pacman-appl, Subscription pacman-appl-subscription-0, Placements pacman-appl-placement-0, Deployment mongo, and Replicaset mongo. It also includes a table of policies and a memory usage chart.

# Unified Multicloud Management

Single Pane for all your Kubernetes Clusters

The screenshot displays the Red Hat Advanced Cluster Management for Kubernetes interface. At the top, there's a navigation bar with the Red Hat logo and the title "Advanced Cluster Management for Kubernetes". Below the navigation bar, there are four cards representing different cloud providers: Google (3 clusters), Amazon (5 clusters), Microsoft (1 cluster), and IBM (1 cluster). In the center, there's a "Summary" section showing 4 applications. On the left, there's a "Cluster compliance" section with a circular progress bar indicating 63% compliant, 120 compliant, and 69 non-compliant. The main area is titled "Cluster management" and shows a table of clusters. The table includes columns for Name, Status, Provider, Distribution, Labels, and Nodes. The clusters listed are: foxtrot-gcp-europe, foxtrot-us-west-1, foxtrot-whiskey, local-cluster, sberens-aro-central, sberens-eks-west, sberens-gke-central, sberens-osd-gcp-central, sberens-roks-south, and sberens-rosa-west. Each cluster entry shows its status (Ready), provider (Google Cloud Platform, AWS, Microsoft Azure, IBM Cloud), distribution (OpenShift 4.6.16, 4.6.9, 4.5.30, etc.), labels (e.g., apps.pacman=deployed, region=europe-west3), and node count (e.g., 6, 13, 2, 3, 7, 3, 7). At the bottom of the cluster management table, there are pagination controls: "1 - 10 of 10", "« «", "1 of 1", and "» »".

- **Centrally** create, update and delete Kubernetes clusters **across multiple** private and public clouds
- **Hibernate / Resume** OCP Clusters across your domain
- **Configure Cluster Sets & Cluster Pools** for simplified OCP cluster management
- Search, find and modify **any** kubernetes resource across the **entire** domain
- **Quickly** troubleshoot and resolve issues across your **federated** domain

# Policy based Governance, Risk, and Compliance

Don't wait for your security team to tap you on the shoulder

Governance

The screenshot displays a governance dashboard with the following sections:

- NIST SP 800-53:** Shows 1/1 Cluster violations and 1/2 Policy violations.
- NIST-CSF:** Shows 1/6 Cluster violations and 1/2 Policy violations.
- HIPAA:** Shows "No violations found" based on industry standards.
- NIST 800-53:** Shows "No violations found" based on industry standards.
- PCI:** Shows "No violations found" based on industry standards.

A central modal window titled "Create policy" is open, containing fields for Name (policy-gatekeeper-operator), Namespace (environment: "dev"), Specifications (Custom specifications), Cluster selector (environment: "dev"), Standards (NIST SP 800-53), Categories (CM Configuration Management), and Controls (CM-2 Baseline Configuration). A note indicates that Inform (Reports the violation, which requires manual remediation) is selected.

To the right, a code editor displays the YAML configuration for the created policy:

```
28 apiVersion: operators.coreos.com/v1alpha1
29 kind: Subscription
30 metadata:
31   name: gatekeeper-operator-product
32   namespace: openshift-operators
33 spec:
34   channels: stable
35   installPlanApproval: Automatic
36   name: gatekeeper-operator-product
37   source: redhat-operators
38   sourceNamespace: openshift-marketplace
39
40 - objectDefinition:
41   apiVersion: policy.open-cluster-management.io/v1
42   kind: ConfigurationPolicy
43   metadata:
44     name: gatekeeper
45   spec:
46     remediationAction: inform
47     severity: Info
48     objectTemplates:
49       - complianceType: musthave
50         objectDefinition:
51           apiVersion: operator.gatekeeper.sh/v1alpha1
52           kind: Gatekeeper
53           metadata:
54             name: gatekeeper
55           spec:
56             audit:
57               logLevel: INFO
58               replicas: 1
59             images: 'registry.redhat.io/rhacm2/gatekeeper-rhel8:v3.3.0'
60             validatingWebhook: Enabled
61             mutatingWebhook: Disabled
62             webhook:
63               emitAdmissionEvents: Enabled
64               logLevel: INFO
65               replicas: 2
66
67           apiVersion: policy.open-cluster-management.io/v1
68           kind: PlacementBinding
69           metadata:
70             name: binding-policy-gatekeeper-operator
71           placement:
72             name: placement-policy-gatekeeper-operator
73             kind: PlacementRule
74             apiGroup: apps.open-cluster-management.io
75             subjects:
76               - name: policy-gatekeeper-operator
77                 kind: Policy
78                 apiGroup: policy.open-cluster-management.io
79
80           apiVersion: apps.open-cluster-management.io/v1
81           kind: PlacementRule
82           metadata:
83             name: placement-policy-gatekeeper-operator
84           spec:
85             clusterConditions:
86               - status: "True"
87               type: ManagedClusterConditionAvailable
88             clusterSelector:
89               matchExpressions:
90                 - {key: environment, operator: In, values: ["dev"]}
```

25

- **Centrally** set & enforce policies for security, applications, & infrastructure
- Quickly **visualize** detailed **auditing** on configuration of apps and clusters
- Perform remediation actions by leveraging **Ansible Automation Platform** integration.
- Built-in **compliance policies** and audit checks, including **GitOps** Integration.
- **Immediate** visibility into your compliance posture based on **your** defined standards

V1



# Advanced Application Lifecycle Management

Simplify your Application Lifecycle

Applications /

Create an application  YAML: On

All fields marked with an asterisk (\*) are mandatory. Fill out the form or edit the YAML directly.

Name\*

Namespace\*

Repository location for resources

Repository types

Select the type of repository where resources that you want to deploy are located

Git

URL\*

Branch

Path

Reconcile option

Set pre and post deployment

Application YAML

```
apiVersion: app.k8s.io/v1beta1
kind: Application
metadata:
  name: newapp
  namespace: default
spec:
  componentKinds:
    - group: apps.open-cluster-management.io
      kind: Subscription
      descriptor: {}
      selector:
        matchExpressions:
          - key: app
            operator: In
            values:
              - newapp
  apiVersion: apps.open-cluster-management.io/v1
  kind: Subscription
  metadata:
    annotations:
      apps.open-cluster-management.io/git-branch: master
      apps.open-cluster-management.io/git-path: s2i
      apps.open-cluster-management.io/reconcile-option: merge
    labels:
      app: newapp
  name: newapp-subscription-1
```

Resource topology | How to read topology

Cluster

Select a cluster to view details

Clusters (2)

foxtrot-gcp-europe

foxtrot-us-west-1

Details

Name: foxtrot-gcp-europe  
Namespace: foxtrot-gcp-europe  
Open cluster console  
Status: ok  
CPU: 12%  
Memory: 7%  
Created: 9 days ago

Name: foxtrot-us-west-1  
Namespace: foxtrot-us-west-1  
Open cluster console  
Status: ok  
CPU: 12%  
Memory: 7%  
Created: 9 days ago

V1

- **Easily** deploy an Application using the **Application Builder**
- Deploy Applications from **Multiple** Sources (GIT / HELM / Object Storage)
- Integrate with **OpenShift GitOps** (Argo CD).
- Automatically **detect and visualize** **Argo CD** Applications in RHACM
- Quickly **visualize** application relationships **across** clusters and those that **span** clusters

# Multicluster Observability

## Overview

- Global Query view with **Grafana** for OCP Clusters
  - Out of the Box multi cluster health monitoring dashboards
  - PromQL compliant - Build your own queries
- **Centralize Alerts** and notifications on the **RHACM Hub**. Forward to 3rd Party Systems (PagerDuty / Slack)
- Centralized **Database**
  - Optimized set of metrics collected from managed clusters
  - Focused on Cluster Management
- Unlimited **Data Retention**
  - Observe Metric trends
  - Set Alert Patterns
  - Supported Object Storage
    - AWS S3 (and compatible)
    - Ceph for on-premise
    - Google Cloud Storage
    - Azure Storage

The screenshot shows a web-based management interface for Red Hat Advanced Cluster Management for Kubernetes. The title bar includes the Red Hat logo and the text "Advanced Cluster Management for Kubernetes". The main content area is titled "Clusters" and displays a list of 39 items. Each item represents a cluster with the following columns:

- Name: A link to the cluster details, such as "cstark-openshift46", "sberens-azure", "spoke50-gke2", etc.
- Status: A color-coded icon indicating the status (e.g., Offline in red, Ready in green).
- Version: The OpenShift version, e.g., "OpenShift 4.6.0-rc.2" or "v1.16.13-gke.401".
- Cloud Provider: The cloud provider, e.g., "cloud=Amazon", "cloud=OpenStack", "cloud=Google", "cloud=EKS".
- Count: The number of nodes, e.g., "+4", "+2", "+1", "+5", "+6".
- Action: A three-dot menu icon for each cluster entry.

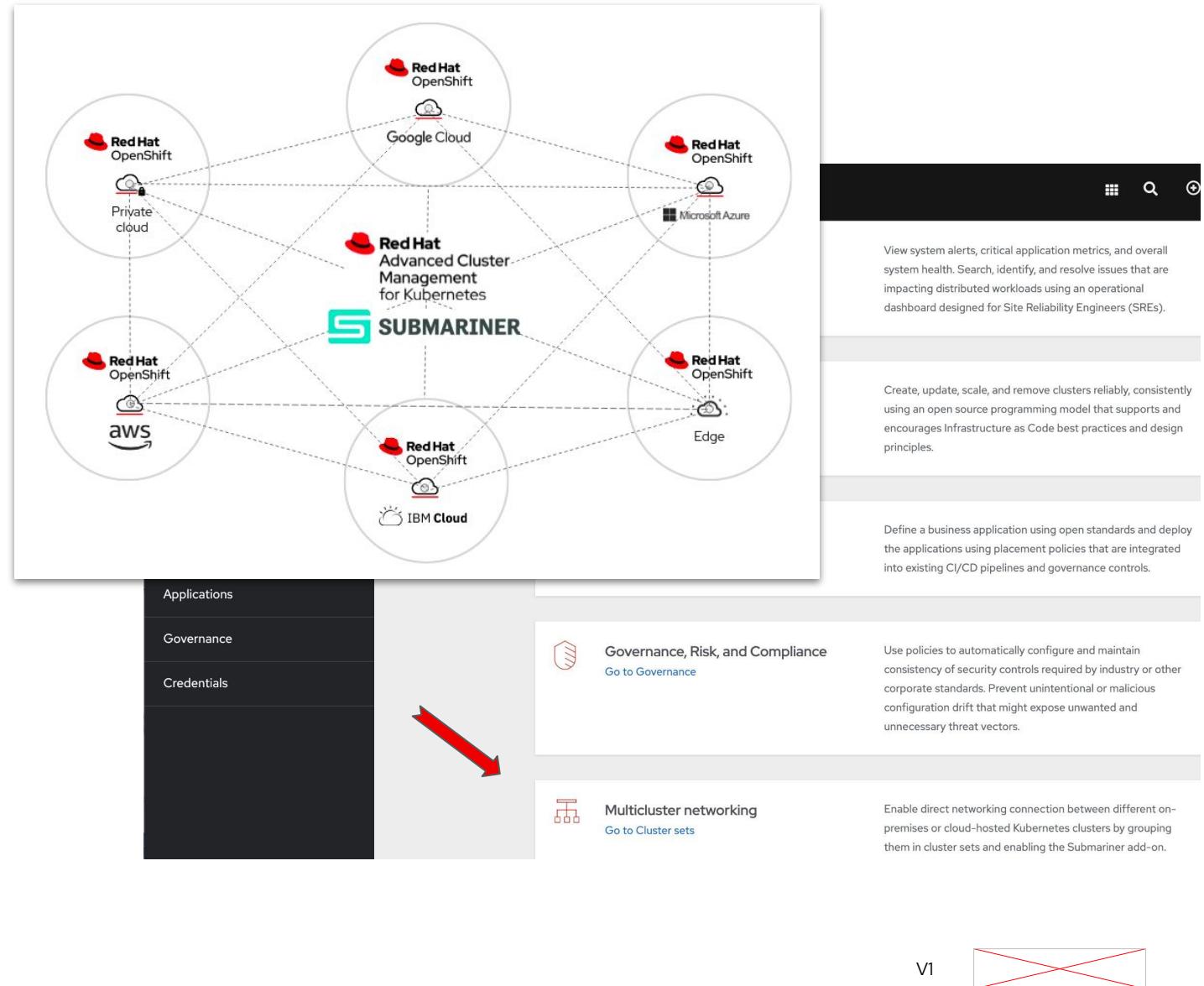
At the bottom of the table, there are pagination controls: "Items per page: 20" and "1 of 2 pages".

Clusters					
<a href="#">stage-3</a>	<a href="#">Detached</a>	-	-	-	
<a href="#">cstark-openshift46</a>	🔴 Offline	OpenShift 4.6.0-rc.2	cloud=Amazon +4	6	⋮
<a href="#">sberens-azure</a>	🔴 Offline	-	cloud=Azure +2	-	⋮
<a href="#">spoke50-gke2</a>	🔴 Offline	v1.16.13-gke.401	cloud=Google vendor=GKE +1	3	⋮
<a href="#">acmcdaan1</a>	🟢 Ready	OpenShift 4.5.2(Upgrade available)	cloud=OpenStack +5	6	⋮
<a href="#">acmcdaan2</a>	🟢 Ready	OpenShift 4.5.2(Upgrade available)	cloud=OpenStack +5	6	⋮
<a href="#">dhaiduce-01</a>	🟢 Ready	OpenShift 4.3.38	cloud=Amazon +6	6	⋮
<a href="#">dhaiduce-02</a>	🟢 Ready	OpenShift 4.3.33(Upgrade available)	cloud=Amazon +6	6	⋮
<a href="#">dhaiduce-03</a>	🟢 Ready	OpenShift 4.5.11(Upgrade available)	cloud=Amazon +6	6	⋮
<a href="#">dhaiduce-04</a>	🟢 Ready	OpenShift 4.4.23(Upgrade available)	cloud=Amazon +6	6	⋮
<a href="#">dhaiduce-eks-eu-central-1</a>	🟢 Ready	v1.14.9-eks-658790	cloud=Amazon vendor=EKS +3	3	⋮
<a href="#">dhaiduce-eks-eu-north-1</a>	🟢 Ready	v1.14.9-eks-658790	cloud=Amazon vendor=EKS +2	3	⋮
<a href="#">dhaiduce-eks-eu-west-1</a>	🟢 Ready	v1.14.9-eks-658790	cloud=Amazon vendor=EKS +2	3	⋮
<a href="#">dhaiduce-eks-eu-west-2</a>	🟢 Ready	v1.14.9-eks-658790	cloud=Amazon vendor=EKS +3	3	⋮
<a href="#">dhaiduce-eks-eu-west-3</a>	🟢 Ready	v1.14.9-eks-658790	cloud=Amazon vendor=EKS +3	3	⋮
<a href="#">installer-test</a>	🟢 Ready	OpenShift 4.5.5(Upgrade available)	cloud=Amazon +5	6	⋮
<a href="#">local-cluster</a>	🟢 Ready	OpenShift 4.5.11(Upgrade available)	cloud=Amazon +5	6	⋮
<a href="#">lubbock</a>	🟢 Ready	OpenShift 4.5.8(Upgrade available)	cloud=Amazon +4	6	⋮
<a href="#">oregon2</a>	🟢 Ready	OpenShift 4.4.26(Upgrade available)	cloud=Amazon +4	6	⋮
<a href="#">sberens-eks1</a>	🟢 Ready	v1.15.11-eks-065dce	cloud=Amazon vendor=EKS +1	2	⋮

# Multicluster Networking

MCN features overview & look ahead

- **ACM MCN, aka 5th pillar**
- Presenting **Submariner**: an CNCF open source project in the form of an **add-on** for RHACM, now generally available
- Enable **direct networking** between Pods in different Kubernetes clusters as well as **Service Discovery**, either on-premises or in the cloud
- Leverage **Cluster Sets** - All done via a group of clusters with a high degree of mutual trust that share services
- **Globalnet** - Support for interconnecting clusters with overlapping CIDRs
- **Future work (subject to change)**
  - ACM Red Hat OpenShift Service mesh integration
  - Discovery Deploy & Configure Federation
  - Custom - upstream Istio, Gloo...



# Benefits

## Red Hat OpenShift and Red Hat Advanced Cluster Management for Kubernetes



### Accelerate development to production

Self-service provisioning allows app dev teams to request clusters directly from a catalog removing central IT as a bottleneck.



### Increase application availability

Placement rules can allow quick deployment of clusters across distributed locations for availability, capacity, and security reasons.



### Reduce costs

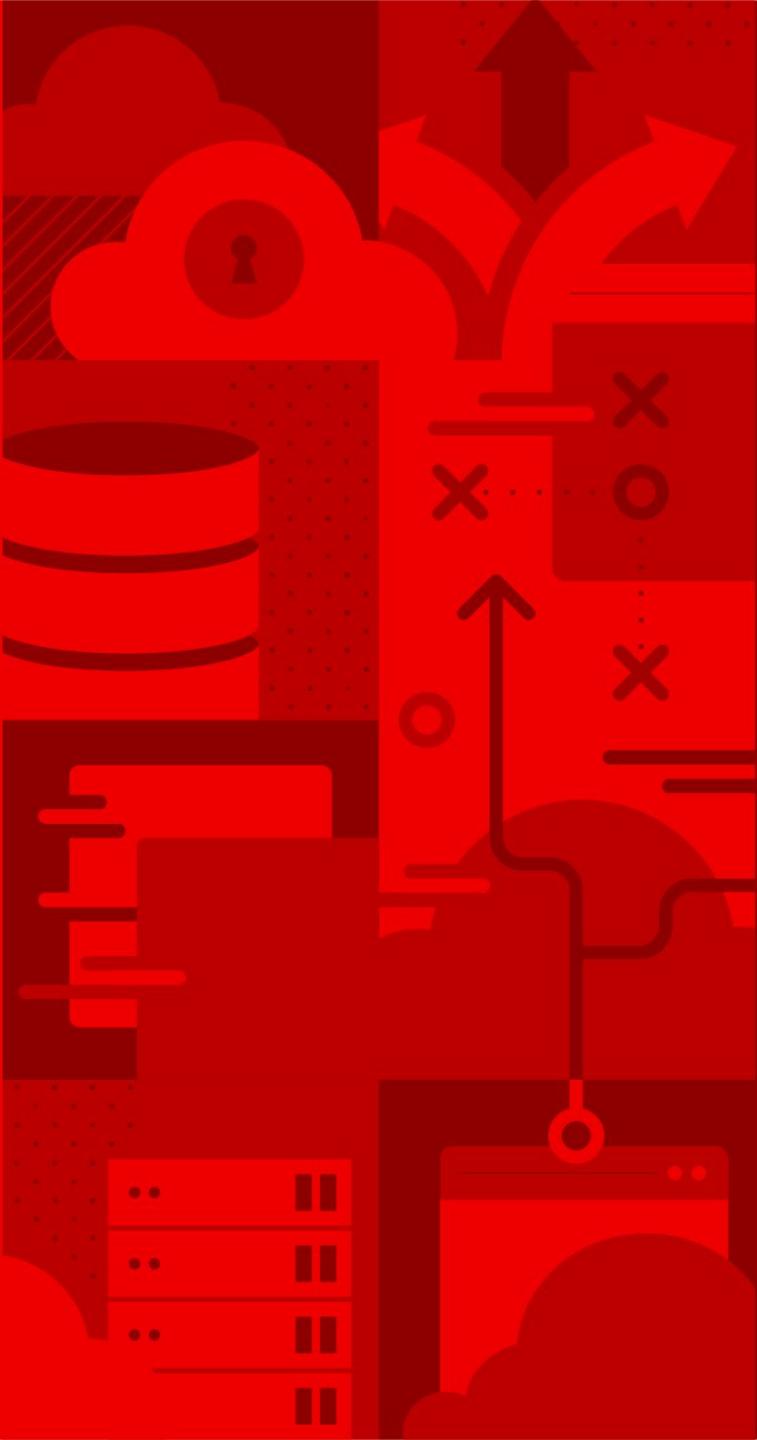
Centralized management of clusters reduces operational cost, makes the environment consistent, and removes the need to manually manage individual clusters.



### Ease Compliance

Policies can be written by the security team and enforced at each cluster, allowing environments to conform to your policy.





# Detailed Use Cases

# Multi-cluster Lifecycle Management



 <p>IT Operations</p>	<p>How do I get a simplified understanding of my cluster health and the impact it may have on my application availability ?</p> <p>How do I automate provisioning and deprovisioning of my clusters?</p>
 <p>DevOps/SRE</p>	<p>How can I manage the life cycle of multiple clusters regardless of where they reside (on-prem, across public clouds) using a single control plane?</p>



# Multi-cluster Lifecycle Management

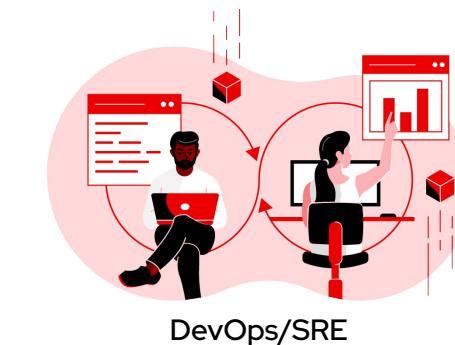
## Overview

- **Full Management of OCP Kubernetes**
  - Provision new OCP 4.8.x and above
  - Manage existing **OCP 3.11 (Limited Support)** 4.8.x and above
  - Support for OCP 4.9+ Single Node (SNO)
- **Public cloud managed kubernetes:** EKS, AKS, GKE, IKS, ROKS, ROSA, ARO, OSD.
  - Deploy Policies and Applications, Search, find and modify kubernetes resources.
- See **high level summaries** across all clusters
  - Misconfiguration
  - Pod status
  - Resource capacity
- **Troubleshoot and resolve** issues across the federated domain
  - See in dashboard or via a list/table form
  - Table shows custom tagging
  - Regions
  - Business Purpose
  - Version

The diagram illustrates the scope of Multi-cluster Lifecycle Management. On the left, two circular icons represent different roles: 'IT Operations' featuring a person working on a laptop with a cloud and server icon, and 'DevOps/SRE' featuring two people working on laptops with a bar chart and a cube icon. An arrow points from the 'IT Operations' icon towards the central cloud icon, and another arrow points from the central cloud icon towards the 'DevOps/SRE' icon. On the right, a screenshot of the 'Red Hat Advanced Cluster Management for Kubernetes' dashboard is shown. The dashboard has a dark header with the Red Hat logo and the title. The left sidebar includes sections for 'Advanced Cluster Management', 'Home' (with 'Welcome' and 'Overview' selected), 'Infrastructure' (with 'Clusters', 'Bare metal assets', 'Automation', and 'Infrastructure environments'), 'Applications', 'Governance', and 'Credentials'. The main content area is titled 'Overview' and displays summary statistics: 2 Clusters, 21 Applications, 2 Clusters, 1 Kubernetes type, 2 Regions, 13 Nodes, and 627 Pods. Below these are three donut charts: 'Cluster compliance' (100% Compliant, 0 Non-compliant), 'Pods' (100% Running, 0 Failed, 0 Pending, 625 Running), and 'Cluster status' (100% Ready, 0 Offline, 2 Ready). At the bottom, there is a section titled 'Cluster issues' showing 3 issues across 2 clusters, categorized by severity: Critical (0), Important (0), Moderate (2), and Low (1).

# Multi-cluster Lifecycle Management

# Creating & Importing clusters

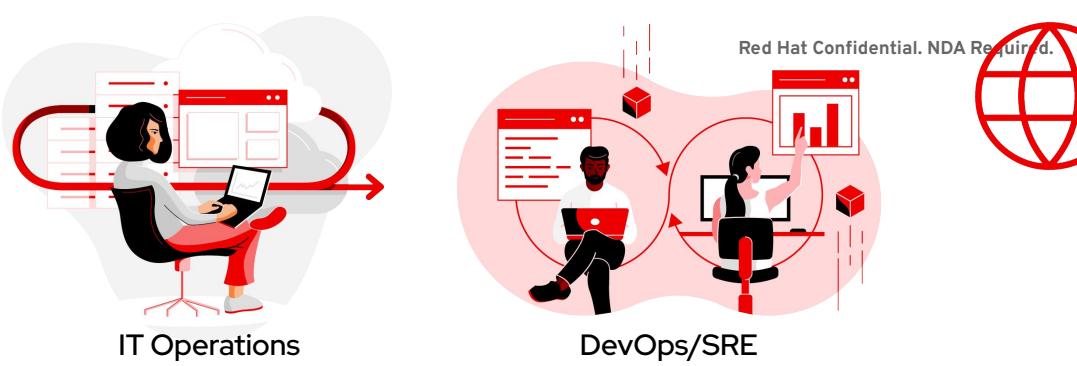


- **Create, Upgrade** and **Destroy** OCP clusters running on **vSphere, Bare-metal** as well as **Public cloud**.
  - Import **OCP Clusters** that can be discovered from **OCM** (OpenShift Cluster Manager)
  - Leverage Hive API for OCP cluster deployment
  - Wizard or YAML based create cluster flow
  - Launch to an OCP Console from ACM
  - Access cluster login credentials and download **kubeadmin** configuration **kubeconfig**
  - Integrate with Ansible Automation Platform
  - Centrally Manage your On-Prem Infrastructure (CIM)

# Multi-cluster Lifecycle Management

## Dynamic Search

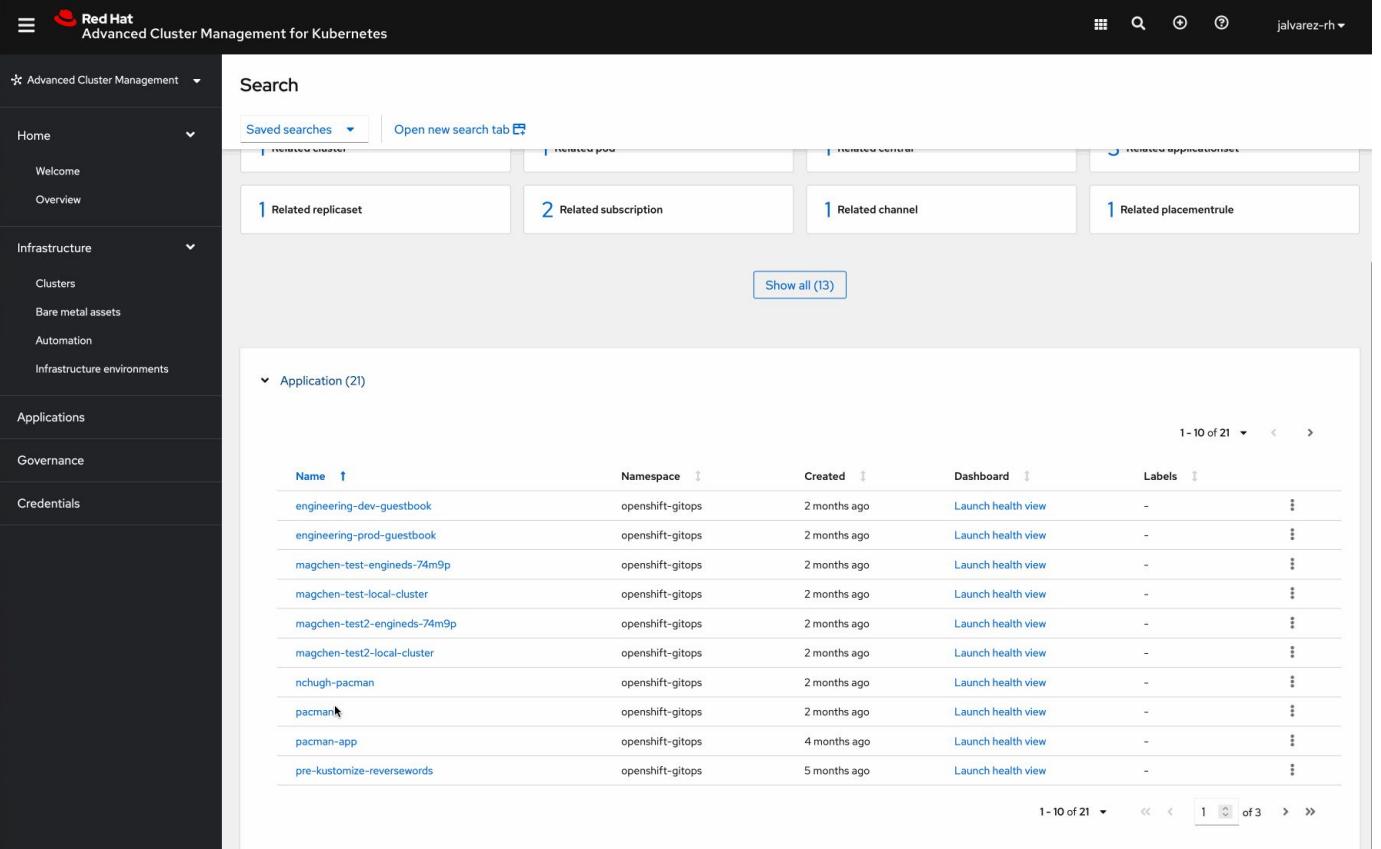
- Troubleshooting across clusters via relationships
- See all **unhealthy** pods
- See related application models to those pods
- See related Persistent Volumes
- See related secrets
- See related **\*any\*** kube resource object category



The diagram illustrates two main roles in the context of multi-cluster lifecycle management:

- IT Operations**: Represented by a woman sitting at a desk with a laptop, working within a cloud environment.
- DevOps/SRE**: Represented by two people (a man and a woman) working on laptops, connected by a red circle, also within a cloud environment.

Red arrows indicate interactions between these roles and the cloud environment, symbolizing how they manage and monitor resources across multiple clusters.



The screenshot shows the Red Hat Advanced Cluster Management for Kubernetes interface. The top navigation bar includes the Red Hat logo and the title "Advanced Cluster Management for Kubernetes". The left sidebar contains a navigation menu with sections like Home, Welcome, Overview, Infrastructure (Clusters, Bare metal assets, Automation, Infrastructure environments), Applications, Governance, and Credentials. The main content area features a "Search" section with tabs for "Saved searches" and "Open new search tab". Below this, there are four cards: "Related replicaset" (1), "Related subscription" (2), "Related channel" (1), and "Related placementrule" (1). A large table titled "Application (21)" lists various applications with columns for Name, Namespace, Created, Dashboard, and Labels. The table includes entries such as "engineering-dev-guestbook", "engineering-prod-guestbook", and "magchen-test-engineds-74m9p". At the bottom right of the interface, there is a footer with pagination controls and a "Show all (13)" button.

# Multi-cluster Observability

## Overview



- Enhanced multi-cluster **OCP** metric aggregation with customized allowlist
  - Enhanced multi-cluster metric aggregation
  - Custom metrics and pre defined metrics
- **Customize** your own Grafana dashboards for fleet management
  - Optimized set of metrics collected from managed clusters
  - Focused on Cluster Management
  - Unlimited Data Retention
  - Set Alert Patterns



# Policy based Governance, Risk and Compliance



 <p>Security OPS</p>	<ul style="list-style-type: none"><li>• How do I ensure all my clusters are compliant with standard and custom policies?</li><li>• How do I set consistent security policies across diverse environments and ensure enforcement?</li><li>• How do I get alerted on any configuration drift and remediate it?</li></ul>
 <p>IT Operations</p>	<ul style="list-style-type: none"><li>• How do I ensure 99.9 % Uptime?</li><li>• How do I drive more innovation at scale?</li></ul>

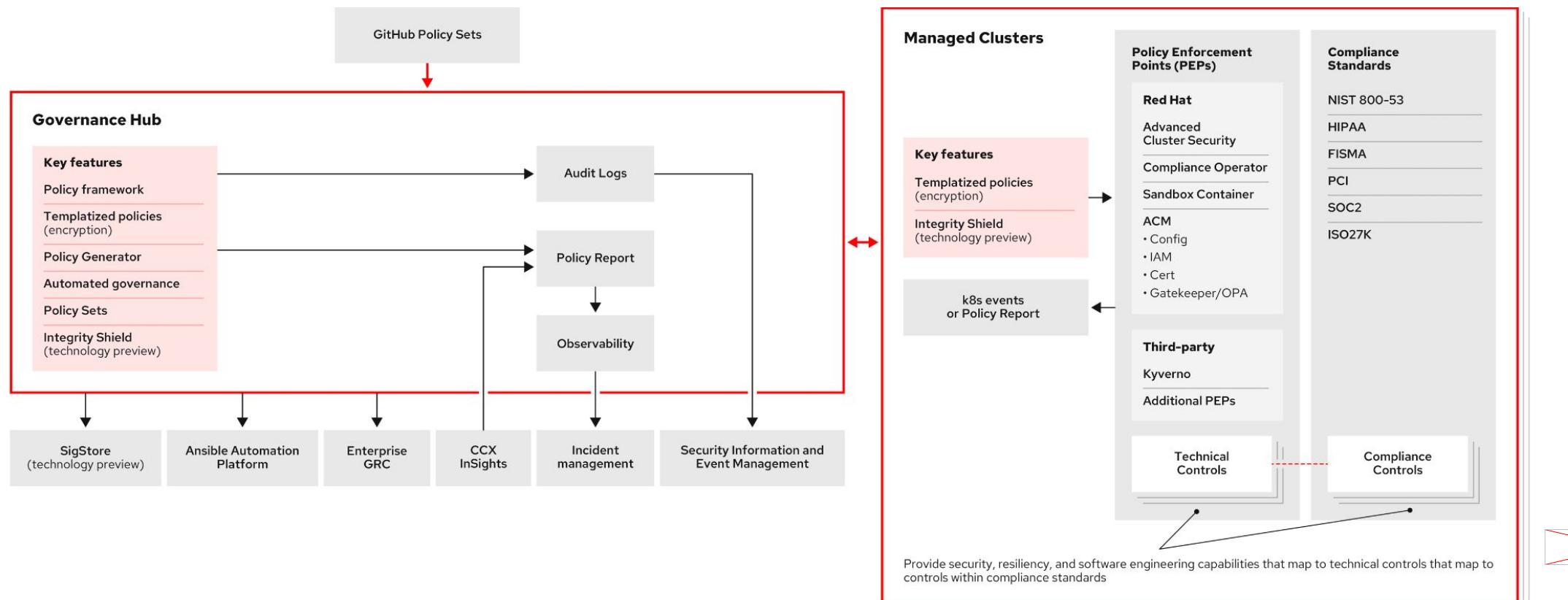


# Policy based Governance, Risk and Compliance

## Overview

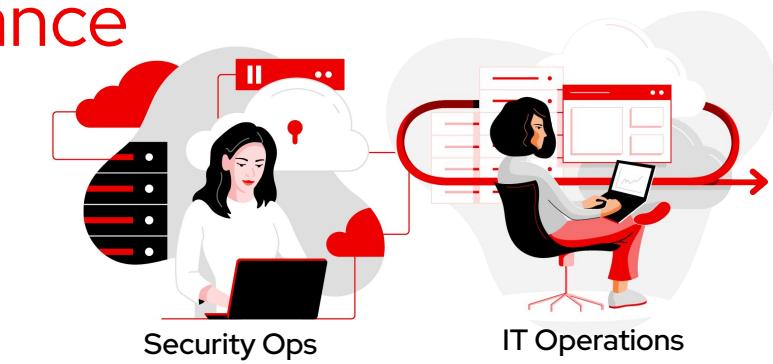
### Managed Cluster and GRC Controllers

- Driven by Kubernetes CRDs and controllers
- Governance capability for managed clusters covering both security and configuration aspects.
- Out of box policies in [GitHub](#) and an extensible policy framework
- Community based policies in [GitHub](#)



# Policy based Governance, Risk and Compliance

Don't wait for your security team to tap you on the shoulder



- Set and enforce policies for security, applications, & infrastructure
- Deep visibility for auditing configuration of apps and clusters
- Unique policy capabilities around compliance
- Categorize violations based on your standards for immediate visibility into your compliance posture
- Integrate with OPA / Gatekeeper & Compliance Operator
- Integrate with Ansible Automation Platform at the Policy Level

The screenshot shows the Red Hat Advanced Cluster Management for Kubernetes interface. The left sidebar has a dark theme with white text. It includes a dropdown menu for 'Advanced Cluster Management', followed by 'Home' with 'Welcome' and 'Overview' options, 'Infrastructure' with 'Clusters', 'Bare metal assets', 'Automation', and 'Infrastructure environments', 'Applications', 'Governance' (which is currently selected), and 'Credentials'. The main content area is titled 'Create policy' with a question mark icon. At the top right of this area are 'Cancel' and 'Create' buttons. The top bar also displays the Red Hat logo and the text 'Advanced Cluster Management for Kubernetes'.

# Policy based Governance, Risk and Compliance

Don't wait for your security team to tap you on the shoulder



- Standard Policies out of the box
  - FISMA
  - HIPAA
  - NIST
  - PCI
- Leverage Different Categories to Represent more standards (if Needed)
- Use Labels to enforce policies against clusters
- Use **inform** to view policy violations
- Use **enforce** to view violations and automatically remediate

A screenshot of the Red Hat Advanced Cluster Management for Kubernetes web interface. The top navigation bar shows the title "Advanced Cluster Management for Kubernetes". On the left, a sidebar menu includes "Home", "Overview", "Infrastructure" (with "Clusters", "Bare metal assets", "Automation", and "Infrastructure environments" sub-options), "Applications", "Governance" (which is selected and highlighted in blue), and "Credentials". The main content area is titled "Create policy" with a "Cancel" and "Create" button at the bottom right. The background features two illustrations: one of a woman labeled "Security Ops" working on a laptop with a server icon, and another of a person labeled "IT Operations" working on a laptop with a cloud and network icons.

# Advanced Application Lifecycle Management

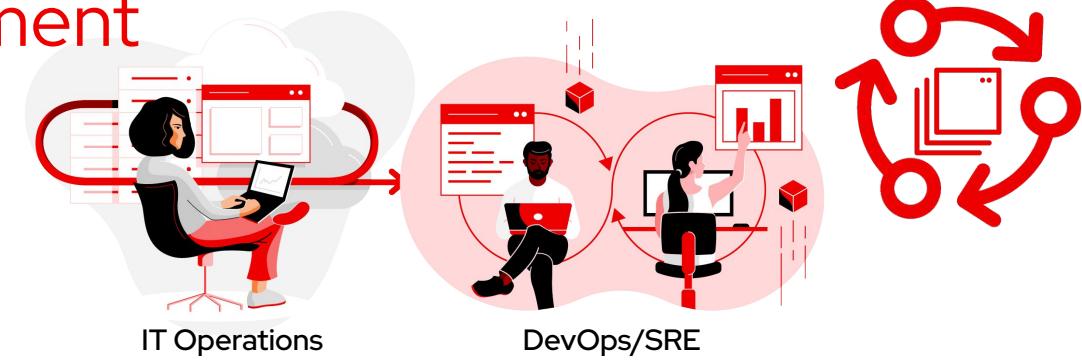
 <p>DevOps/SRE</p>	<ul style="list-style-type: none"><li>• I want to quickly investigate application relationships with real time status, so that I can see where problems are.</li><li>• With the Application Topology view, I can visually inspect application status labels and pod logs to understand if a part of the application is running or not, without having to connect to a cluster and gather any info.</li></ul>
 <p>IT Operations</p>	<ul style="list-style-type: none"><li>• I want new clusters to be deployed with a set of known configurations and required applications.</li><li>• With the assignment of a label at cluster deploy time, the necessary configurations and applications will be automatically deployed and running without any additional manual effort.</li></ul>



# Advanced Application Lifecycle Management

Simplify your Application Lifecycle

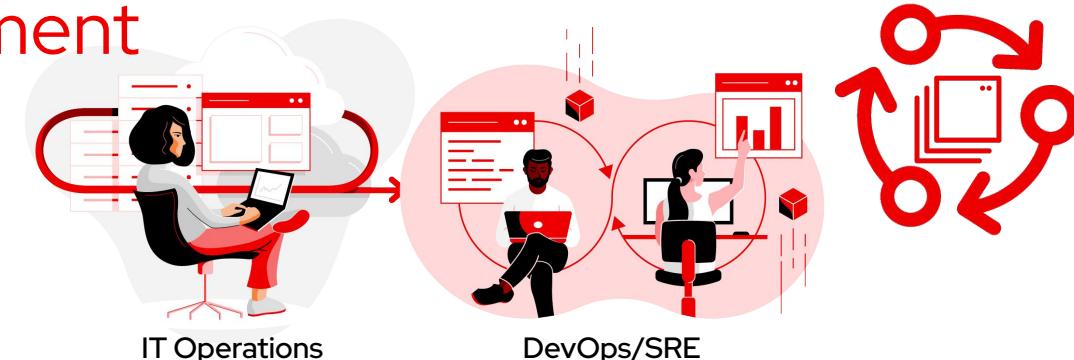
- Deploy Applications at Scale
- Deploy Applications from Multiple Sources (GitOps / HELM /ObjectStorage)
- Quickly Visualize Application Relationships
- Integrate with the Red Hat Ansible Automation Platform
- Visualize Argo CD Applications in RHACM (Local and Remote)
- Support for Application Sets (ArgoCD)



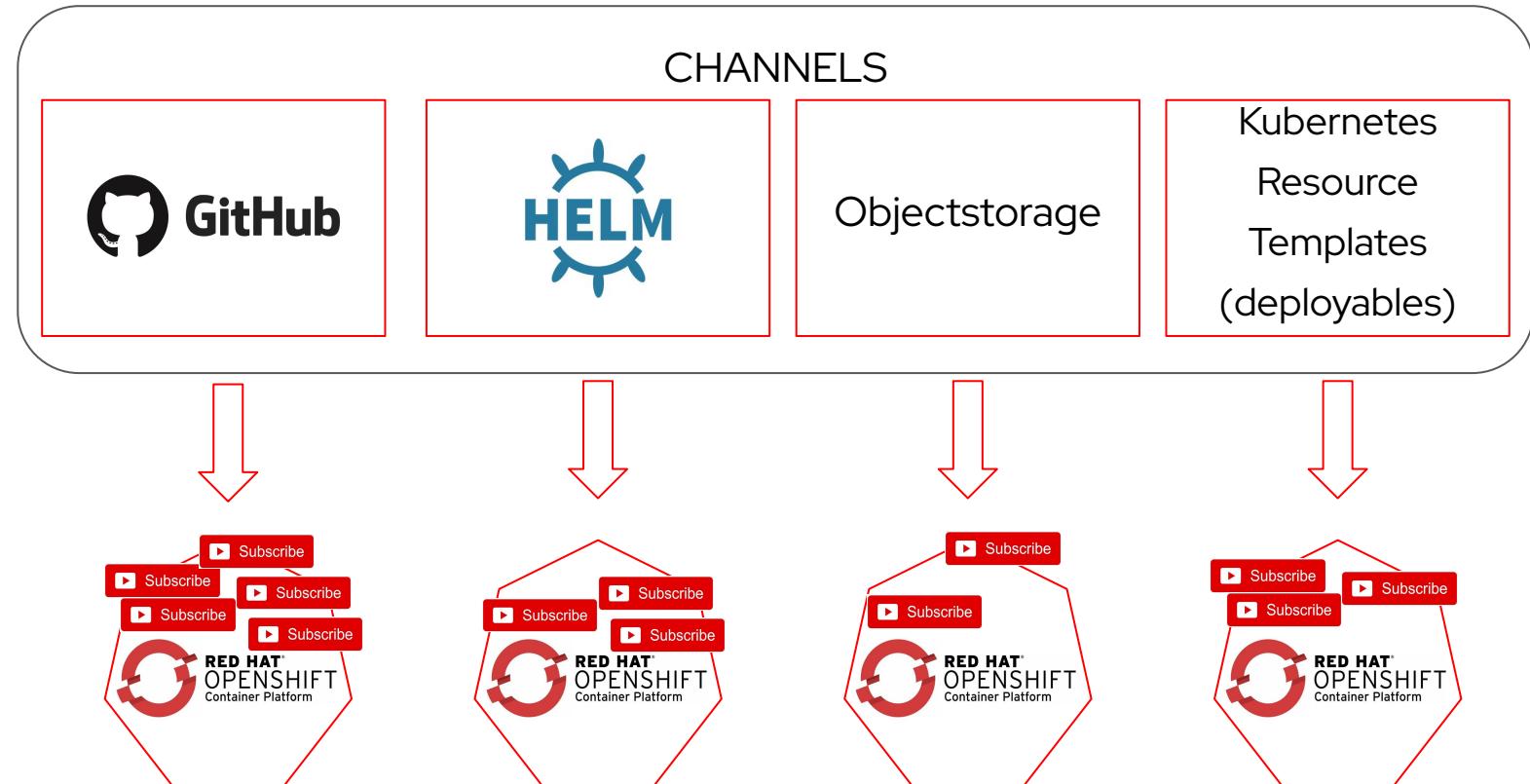
The screenshot displays the Red Hat Application Catalog Manager (RHACM) interface. On the left, a table titled 'Applications' lists various application resources, including 'guestbook', 'engineering-dev-guestbook', 'engineering-prod-guestbook', 'magchen-test', 'magchen-test2', 'nchugh-pacman', 'pacman', 'pacman-app', 'pre-kustomize-reversewords', 'rhacm-in40', 'rhacm-op10', and 'spring-petclinic'. The 'Type' column shows entries like 'Argo CD ApplicationSet' and 'Discovered'. The 'Namespace' column indicates namespaces such as 'guestbook', 'default', 'pacman-app', 'sre', 'open-cluster-management', and 'spring-petclinic'. On the right, a detailed cluster visualization shows a complex network of components including 'Application pacman-app', 'Ansiblejob', 'Subscription pacman-app', 'Placements pacmas-dev-clusters', 'Cluster foxtrot-gcp-us-west-1', 'Service mongo', 'Route foxtrot-pacman', 'Route pacman', 'Deployment mongo', 'PersistentVolumeClaim mongo-storage', and 'Replicaset mongo' and 'Replicaset pacman'. The visualization uses colored nodes (green, blue, red) and arrows to represent relationships between these components. A sidebar on the right shows cluster details for 'foxtrot-gcp-europe' and 'foxtrot-us-west-1'.

# Advanced Application Lifecycle Management

Subscriptions bring enterprise to Kubernetes



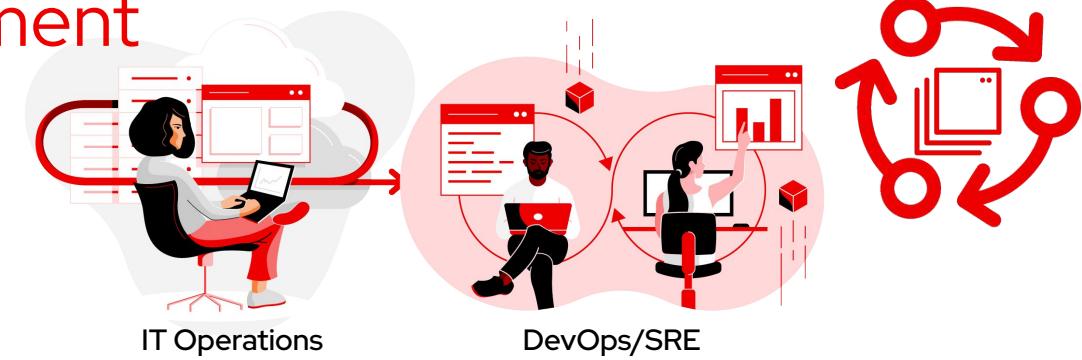
- Extending the best of Enterprise into a desired state methodology
- Time Windows: New releases during your maintenance windows
- Orchestrate actions with the integration of Ansible Automation Platform
- Seamlessly integrate with ArgoCD



# Advanced Application Lifecycle Management

## GitOps - Git as source of truth

- **Create, modify & delete**, just as you would any source code. Git becomes your source of truth controlling your data center.
- Have a record of **who, what & when** for every change precipitated in your environments
- Through code Reviews & Approvals, take full control of all changes to your data center(s)
- Restore your environment, via the Git commit history (system of record)



A screenshot of a GitHub repository page titled "open-cluster-management / demo-subscription-gitops". The repository is private, has 57 commits, 3 branches, 0 packages, 0 releases, and 1 contributor. The latest commit was 360046f 11 minutes ago. The repository contains examples of GitOps, specifically for Bare Metal Assets, Blue-Green Application Management, and Placement Rules. A note at the bottom encourages users to reach out to [jpacker@redhat.com](mailto:jpacker@redhat.com) or Slack @jpacker in coreos.slack.com for help.

Demonstrate Subscriptions via Git Ops

Branch: master New pull request Create new file Upload files Find file Clone or download

inpacker Merge branch 'master' of github.com:open-cluster-management/demo-subs... 11 minutes ago

blueGreen Update README.md yesterday

bma Updates 2 days ago

placement Merge branch 'master' of github.com:open-cluster-management/demo-subs... 15 minutes ago

.gitignore Add directories 2 days ago

CONTRIBUTE.md Updates 15 minutes ago

LICENSE Updates 15 minutes ago

README.md Update README.md 13 minutes ago

README.md

This repository contains examples of GitOps

Examples

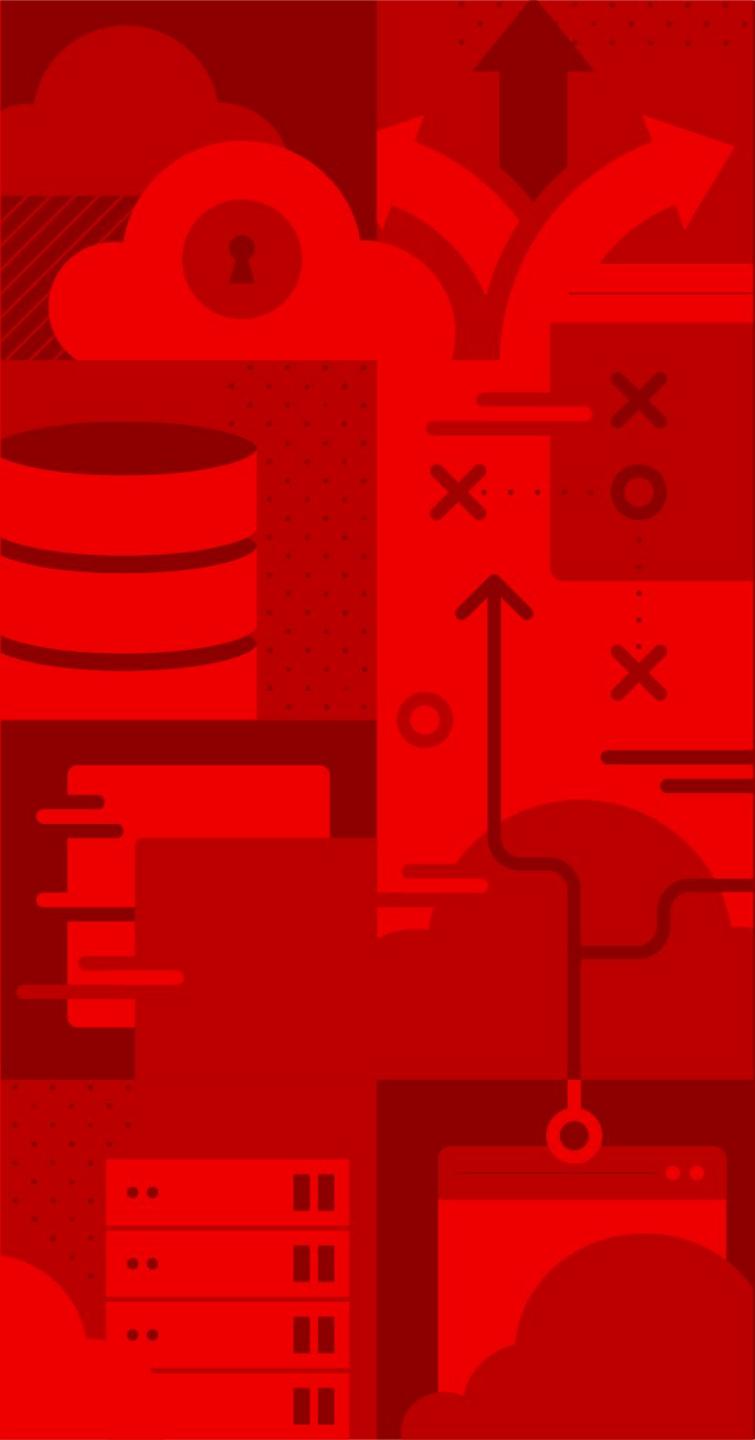
1. Bare Metal Assets via gitops
2. Blue-Green Application Management via gitops
3. Placement Rules example

Help

Reach out to [jpacker@redhat.com](mailto:jpacker@redhat.com) or Slack @jpacker in coreos.slack.com for help

<https://github.com/open-cluster-management/demo-subscription-gitops>





# How ACM works with OpenShift

# Draw Me a Picture!

## Advanced Cluster Management

### Multi-cluster Management

Creation : Discovery : Policy : Compliance : Configuration : Workloads

## OpenShift Container Platform

Manage Workloads

Build Cloud-Native Apps

Developer Productivity

#### Platform Services

Service Mesh : Serverless Builds : CI/CD Pipelines Full Stack Logging Chargeback

#### Application Services

Databases : Languages Runtimes : Integration Business Automation 100+ ISV Services

#### Developer Services

Developer CLI : VS Code extensions : IDE Plugins Code Ready Workspaces CodeReady Containers

## OpenShift Kubernetes Engine

### Cluster Services

Automated Ops : Over-The-Air Updates : Monitoring : Registry : Networking : Router : KubeVirt : OLM : Helm

### Kubernetes

### Red Hat Enterprise Linux & RHEL CoreOS



Physical



Edge



Virtual



Private cloud



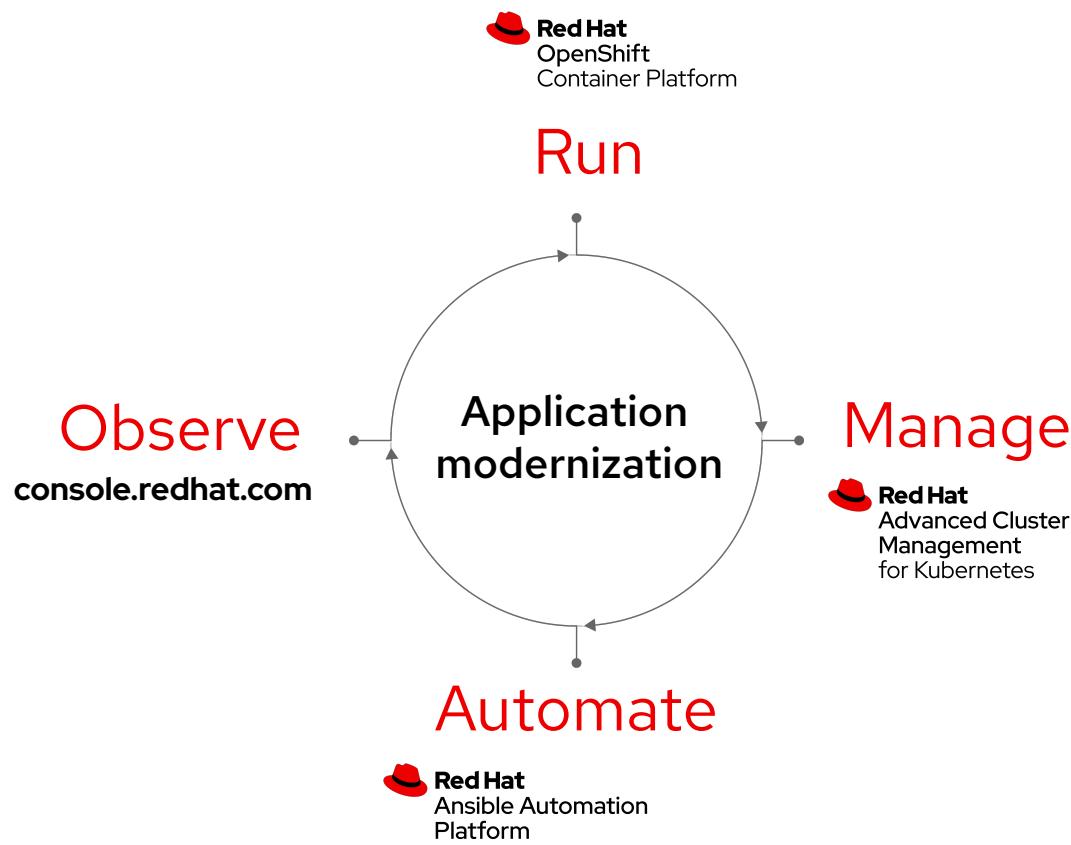
Public cloud



Managed cloud  
(Azure, AWS, Google, IBM, Red Hat)



# Supporting application modernization



## Run

- Cluster services: metrics, chargeback, registry, logging
- Advanced infrastructure functionality
- Dev services: dev tools, automated builds, CI/CD, IDE

## Manage

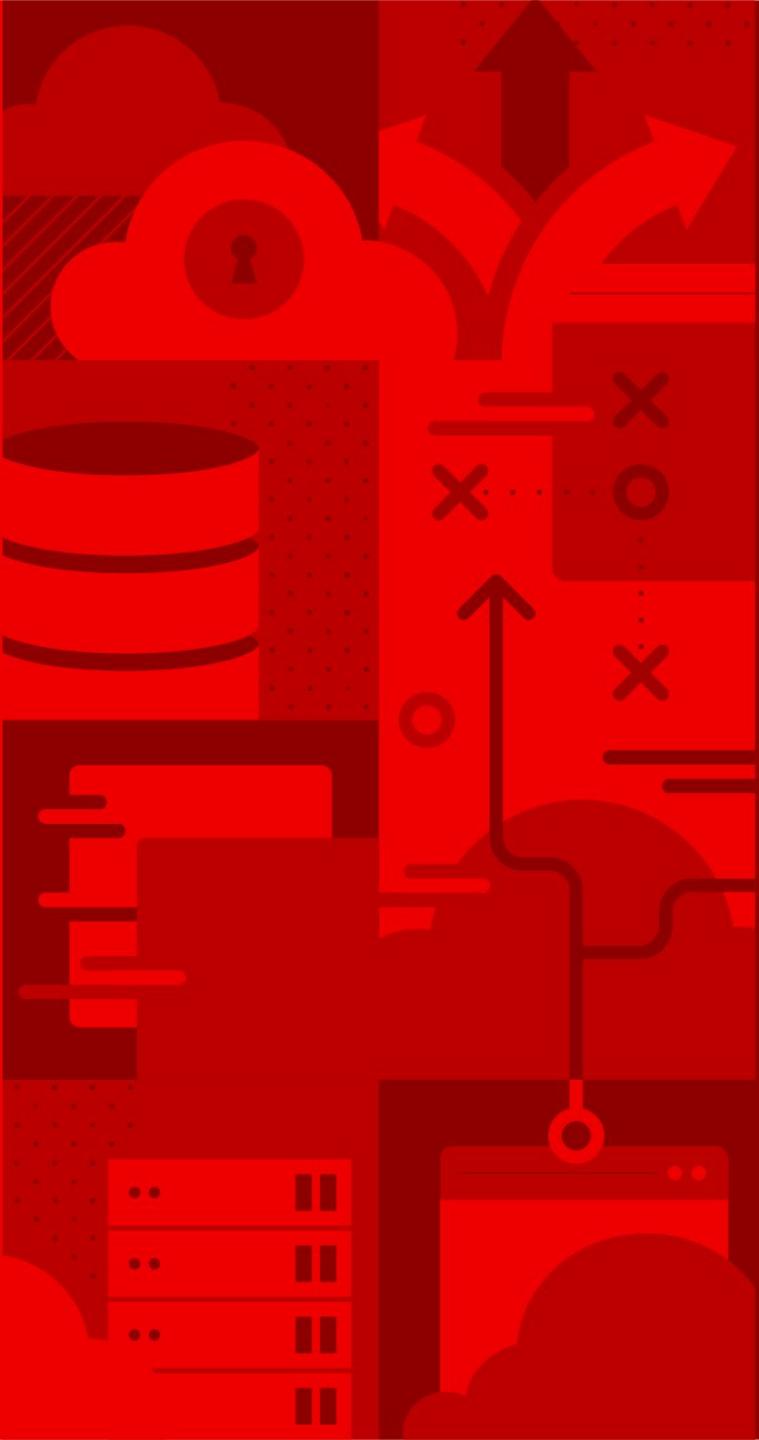
- Multicloud and Kubernetes lifecycle management
- Policy-based governance, risk, and compliance
- Application lifecycle management

## Automate

- Configuration management
- Workflow orchestration
- Network and security automation
- Automation analytics
- Certified content
- Automation services catalog

## Observe

- Red Hat Insights for OpenShift
- Connected customer experience
- Subscription watch

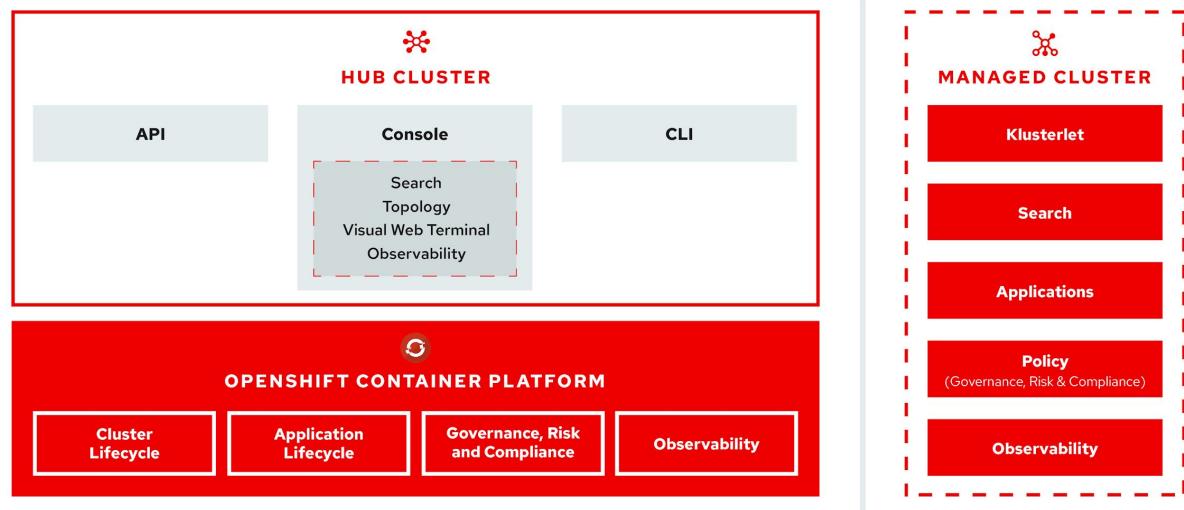


# Architecture

# Architecture overview



IT Operations



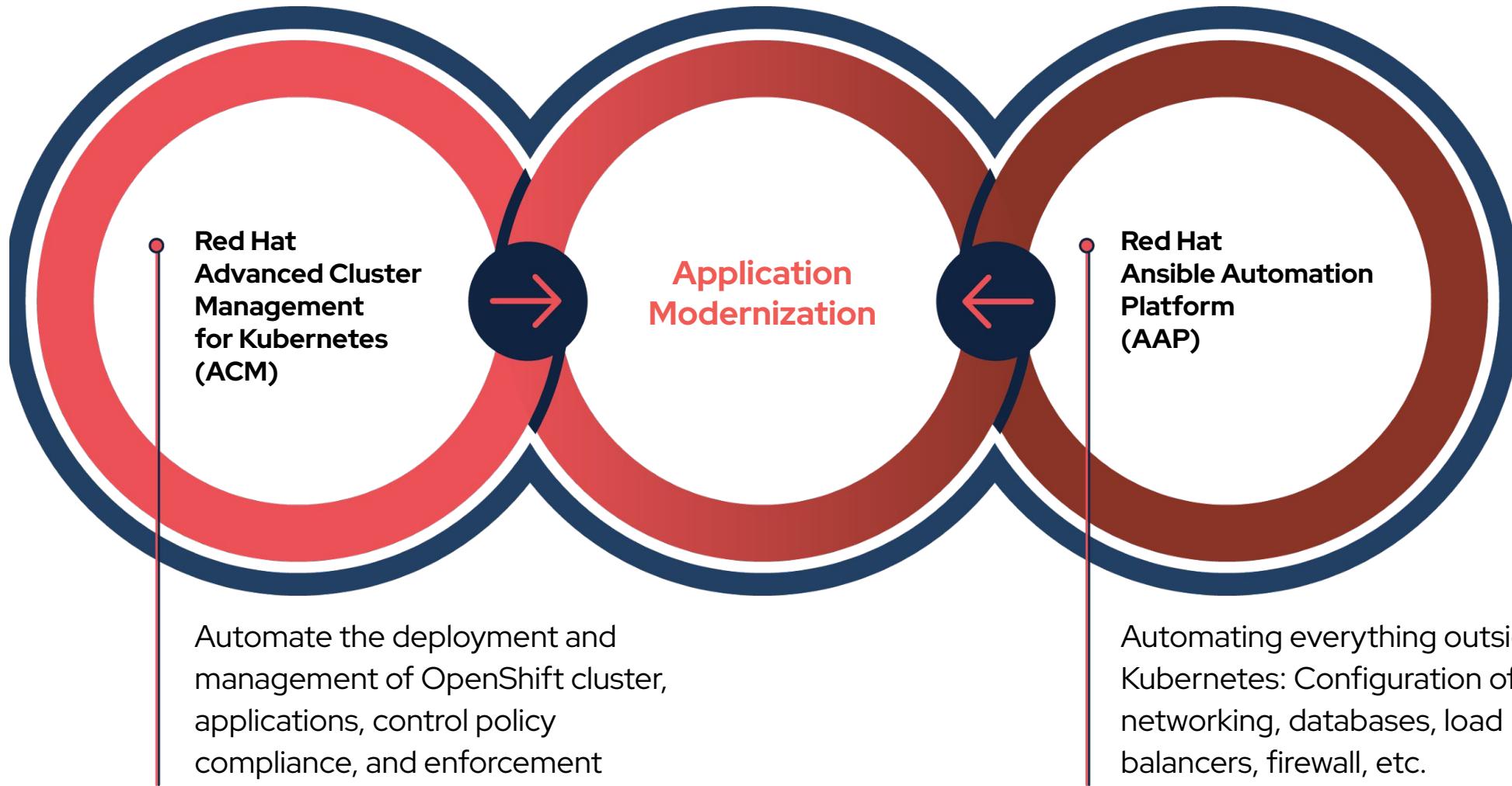
## Hub architecture and components

Red Hat Advanced Cluster Management uses the **multicloud-hub** operator and runs in the **open-cluster-management** namespace

## Managed cluster architecture and components

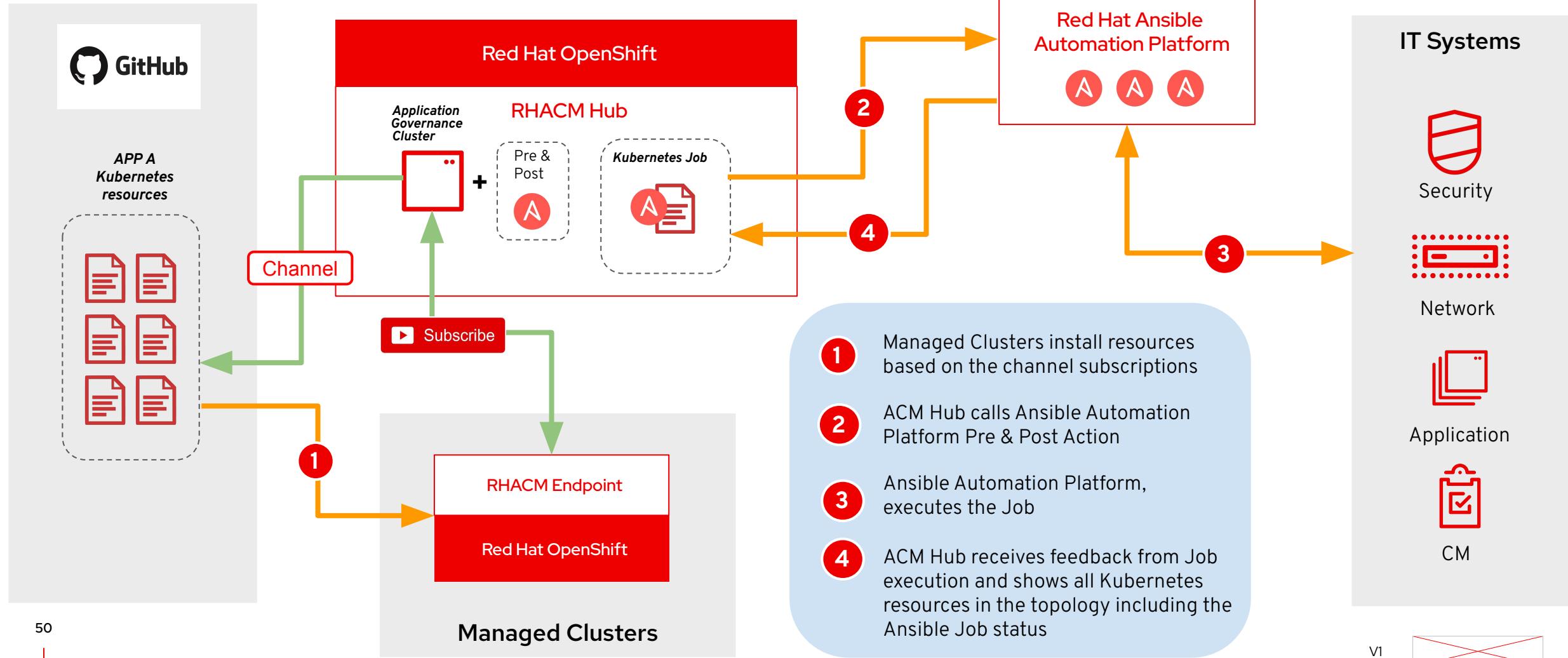
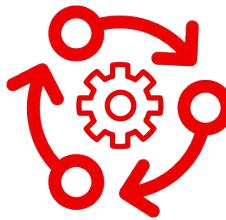
Red Hat Advanced Cluster Management managed clusters use the **multicloud-endpoint** operator which runs in the **open-cluster-management** namespace

# Application modernization driven by Automation of Kubernetes and beyond....



# Complete Automation Platform

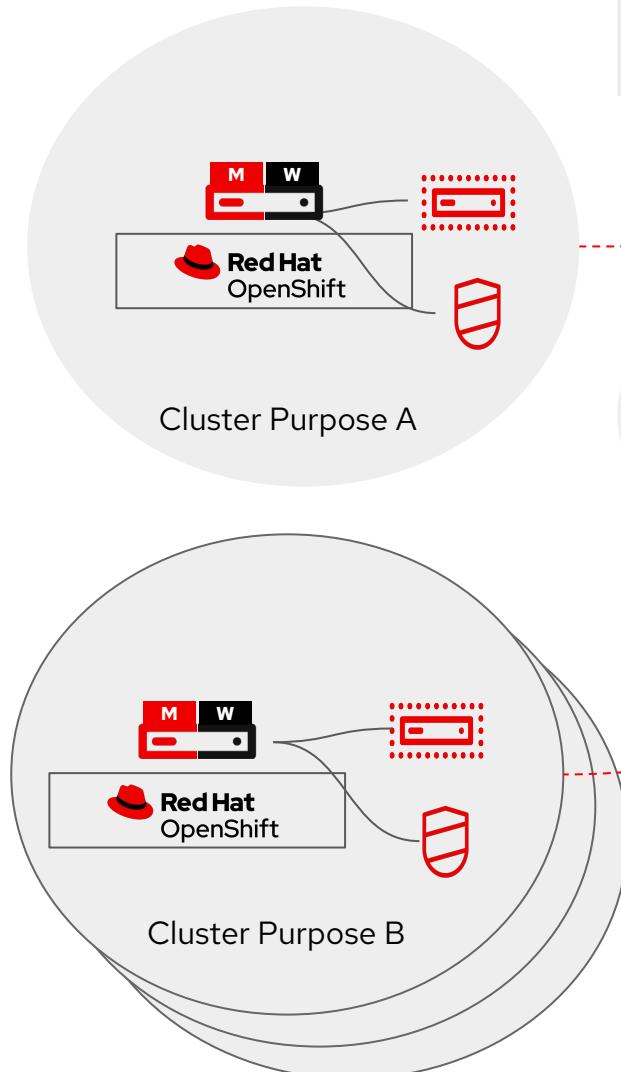
ACM - Ansible Automation Platform Integration through all the use-cases



# Complete Automation Platform

Use Case:

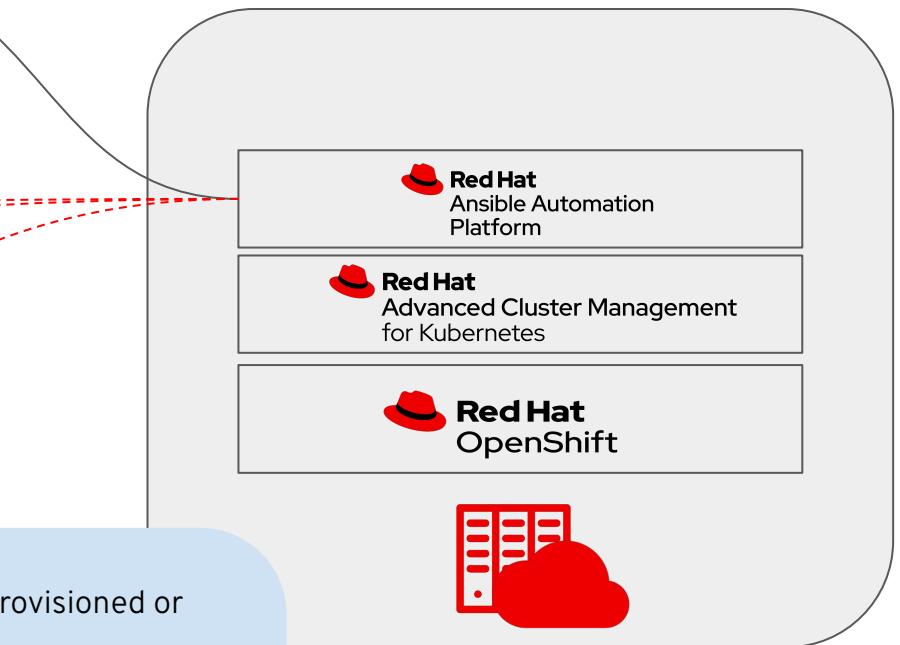
## Cluster Lifecycle Management



- 1 Managed Clusters provisioned or imported
- 2 ACM hub call Ansible Automation with Template Job ID define in Provisioning Pre & Post Action
- 3 Ansible Automation executes Job to update security, storage and more
- 4 Provisioning status and Ansible job execution represented in RHACM

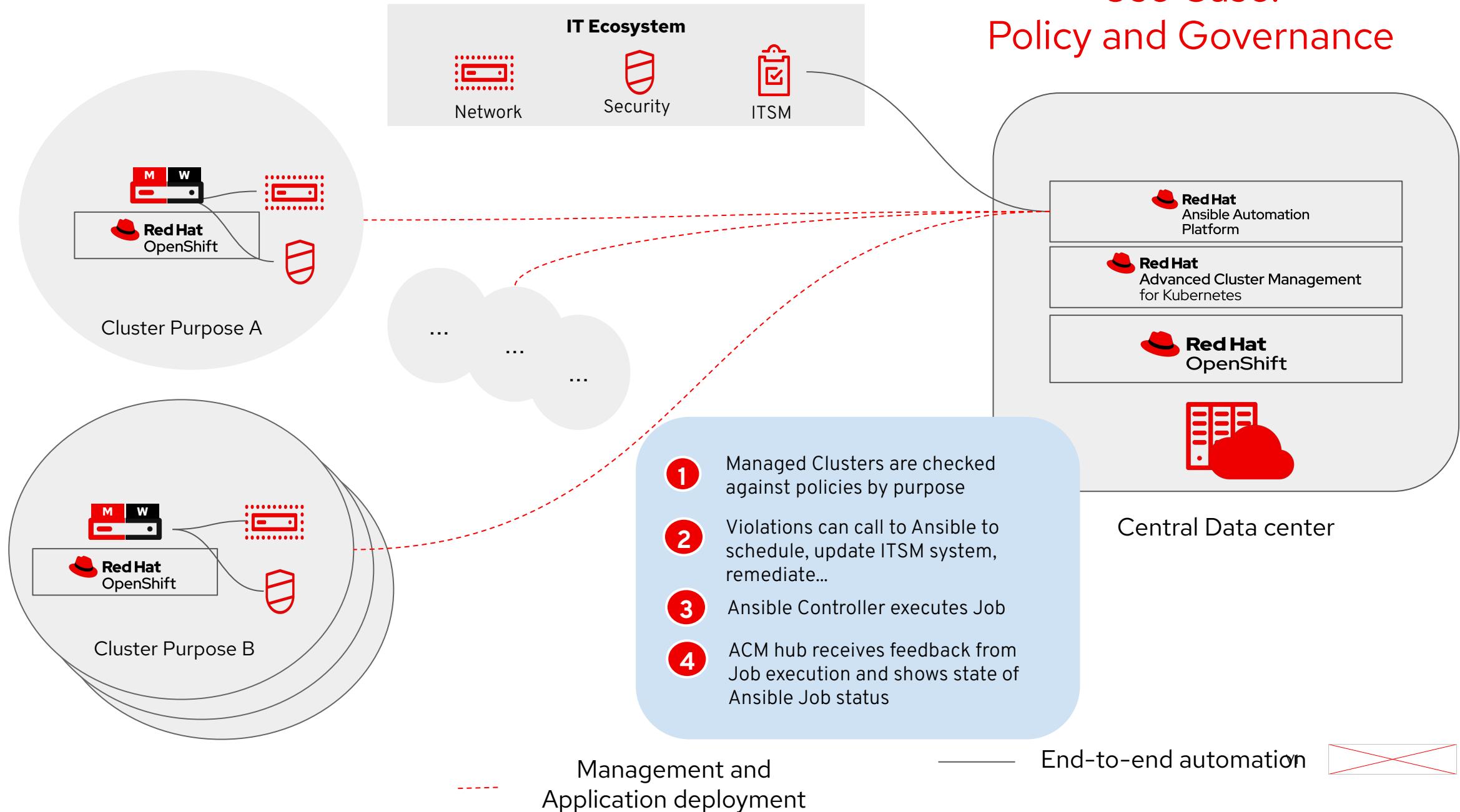
Management and Application deployment

End-to-end automation

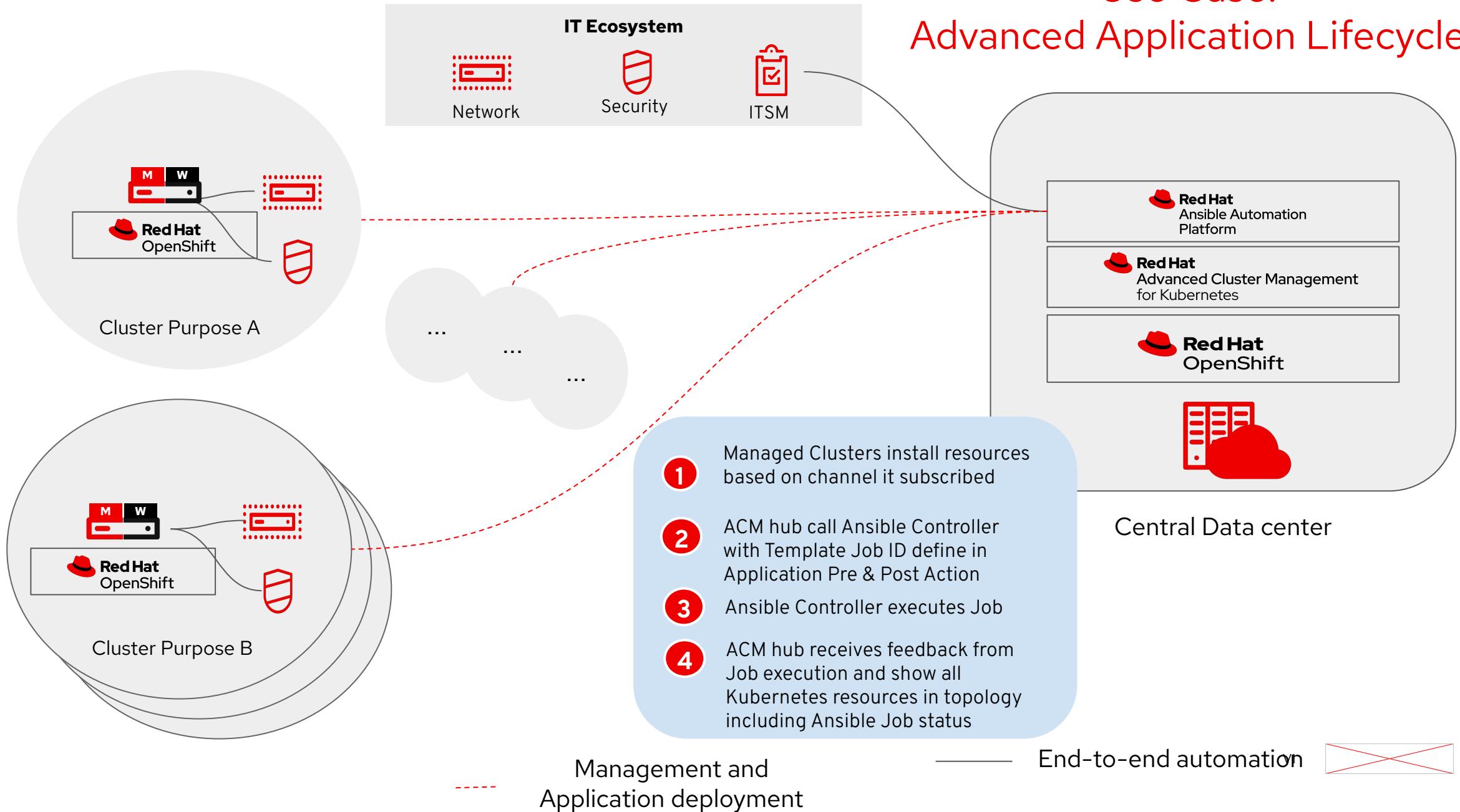


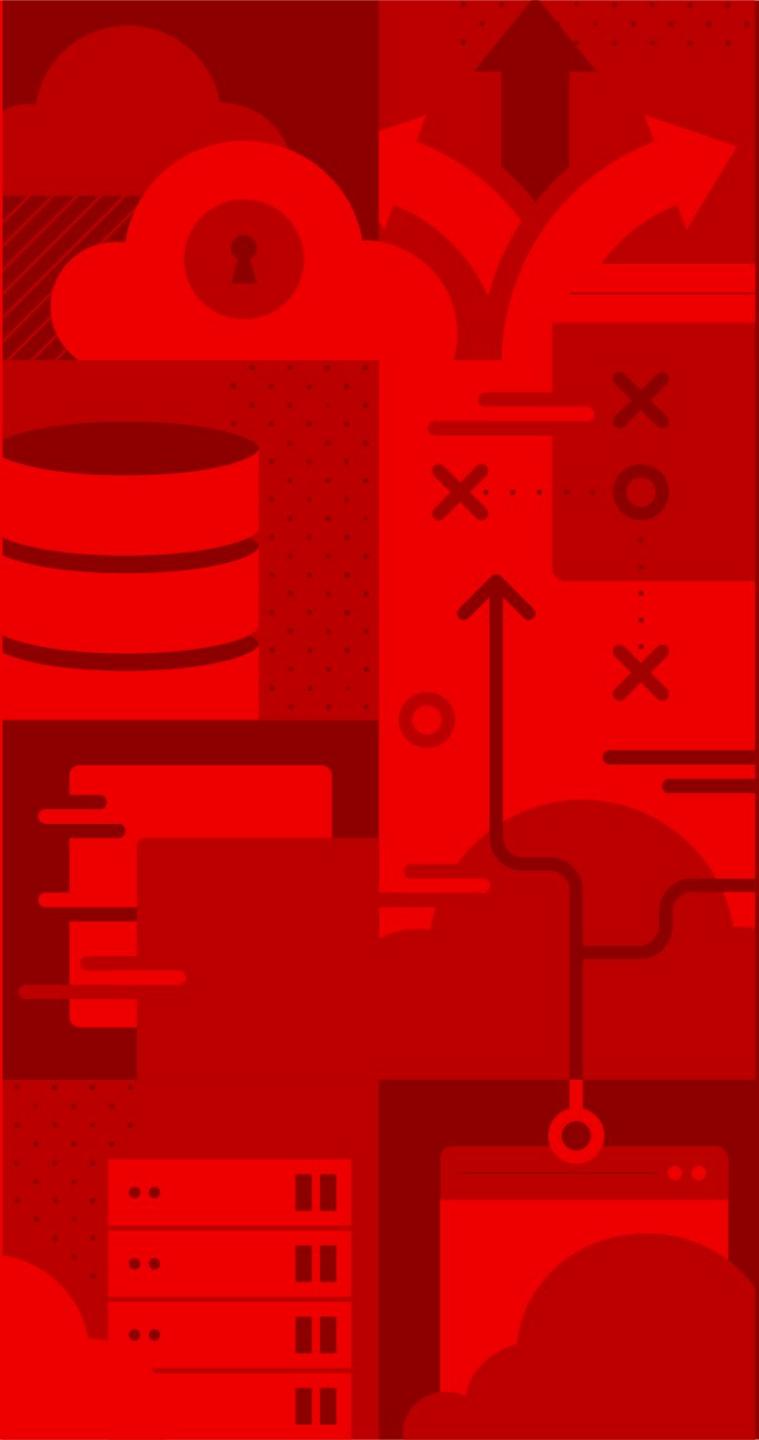
# Complete Automation Platform

## Use Case: Policy and Governance



# Complete Automation Platform





# Installation

# Installation and Foundation

## Operator-based installation for Hub cluster

### Hub Cluster

- Operator based installation
- Available on OperatorHub
- Requires OCP 4.8.x - **Latest**

### Full Lifecycle Management of OCP clusters

- Deploy OpenShift 4.8.x - **Latest**

### Import and Management of OCP clusters

- OpenShift 3.11\*, OpenShift 4.8.x - **Latest**
- Cloud hosted OCP: ROSA / OSD / ARO / ROKS

### Import and Limited Management for cloud Kubernetes

- EKS, AKS, GKE, IKS

### High Availability

- Supports OCP Availability Zone
- Limitation for Search component based on RedisGraph

### Resource Requirements

- **Test:** 3 master, 3 workers, 6 vCPU and 16GB RAM
- **Production:** 3 masters, 3 workers, 16 vCPU and 24GB RAM\*

\* Production requirements vary based on number of clusters in the management domain and types of workloads being run.

\* vCPU/RAM Numbers are per node.



IT Operations

The screenshot shows the Red Hat OpenShift Container Platform interface. On the left, a sidebar menu includes Home, Operators (selected), OperatorHub (selected), Installed Operators, Workloads, Networking, Storage, Builds, Monitoring, Compute, User Management, and Administration. The main area is titled 'Overview' under 'Cluster'. It displays the Cluster API Address (https://api.demo-east-v4618-4z595.demo.red-chesterfield.com:6443), Cluster ID (ec8c4bae-d19b-420f-b7f4-2ada7ac66f16), and Provider (AWS). It also shows the OpenShift Version (4.6.18) and Update Channel (stable-4.6). Below this, there's a 'Status' section with a green checkmark for 'Cluster' and 'Control Plane', and a yellow warning for 'Operators' (1 pending). A 'Cluster Utilization' chart shows resource usage over the last hour for CPU, Memory, Filesystem, Network Transfer, and Pod count. To the right, a 'Cluster Inventory' section lists 6 Nodes, 270 Pods, 2 Storage Classes, and 1 PVC. At the bottom, a log viewer shows recent events from the cluster, including policy updates and install timeouts. The top right corner shows the user Jimmy Alvarez and a 'Quick start available' button.

# Installation and foundation

Operator install for managed cluster



IT Operations



## Managed cluster

The **multicloud-endpoint** operator controls the deployment of components on the managed cluster.

### List of included components:

- ▶ Application manager
- ▶ Connection manager
- ▶ Work manager
- ▶ Policy controller
- ▶ Search collector
- ▶ Service registry
- ▶ IAM policy controller
- ▶ Certificate policy controller
- ▶ CIS policy controller



# Role-Based Access Control

## How to control user access



- RBAC in RHACM is based on kubernetes concepts and is enforced through openshift.
- Cluster-Admin Role is an Openshift super-user role and can perform all actions cluster-wide.
- Additional Roles are available out of the box to assign users Admin, Edit or View level access to RHACM artifacts, for more please see the [documentation](#)

Role	Description
open-cluster-management:cluster-manager-admin	A user with cluster-wide binding to this role, is an RHACM super user can perform any action on RHACM resources
open-cluster-management:admin:managed-cluster-x	A user with cluster binding to this role, has admin access to ManagedCluster "X" resource
open-cluster-management:view:managed-cluster-x	A user with cluster-wide binding to this role, has view access to ManagedCluster "X" resource
OCP Default admin / edit / view roles	A user with namespace binding to these roles has access to resources like policies, applications etc in that namespace or ManagedCluster. A user with cluster-wide binding to these roles has access to resources like policies, applications etc in all namespaces or for all ManagedClusters.

# Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.



[linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)



[youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)



[facebook.com/redhatinc](https://www.facebook.com/redhatinc)



[twitter.com/RedHat](https://twitter.com/RedHat)