Red Hat Summit

Connect

Automatisieren – Automatisieren – Automatisieren

(Sie Ihr OpenShift-Virtualisierung)

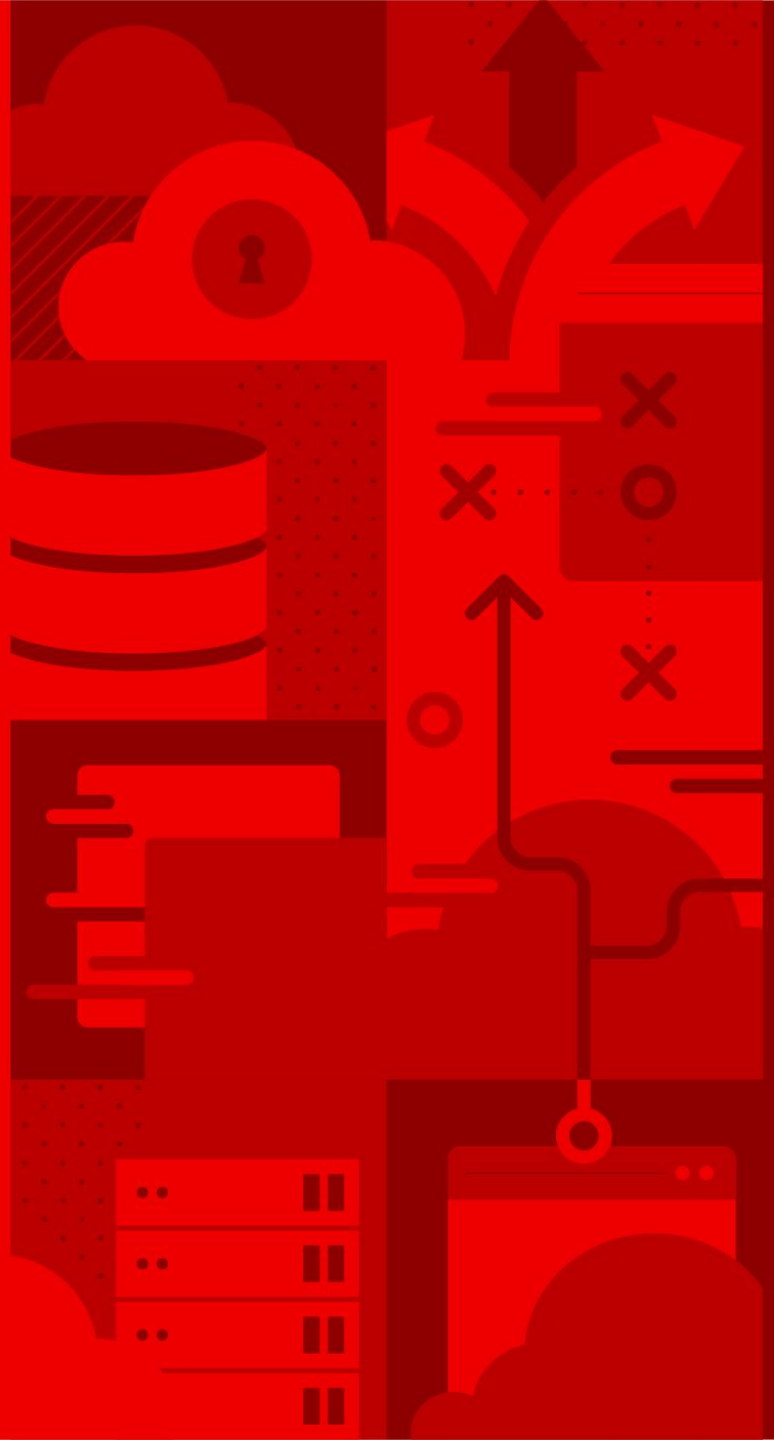
Joachim von Thadden

Senior Principal Specialist Solution Architect, Red Hat

$ whoami

Joachim von Thadden
EMEA Senior Principal Specialist Solution Architect

- based in Germany, near Düsseldorf
- more than 25 years in IT
- more than 30 years working with Linux
- >10 years experience with OpenStack & OpenShift
- 9.5 years at Red Hat



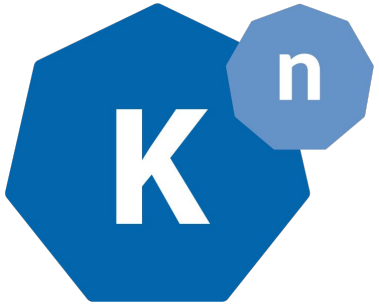
Overview: **Automation** *and/or* GitOps

Red Hat

Standard Kubernetes API

Use whatever you want

Red Hat
Advanced Cluster
Management
for Kubernetes



Standard Kubernetes API

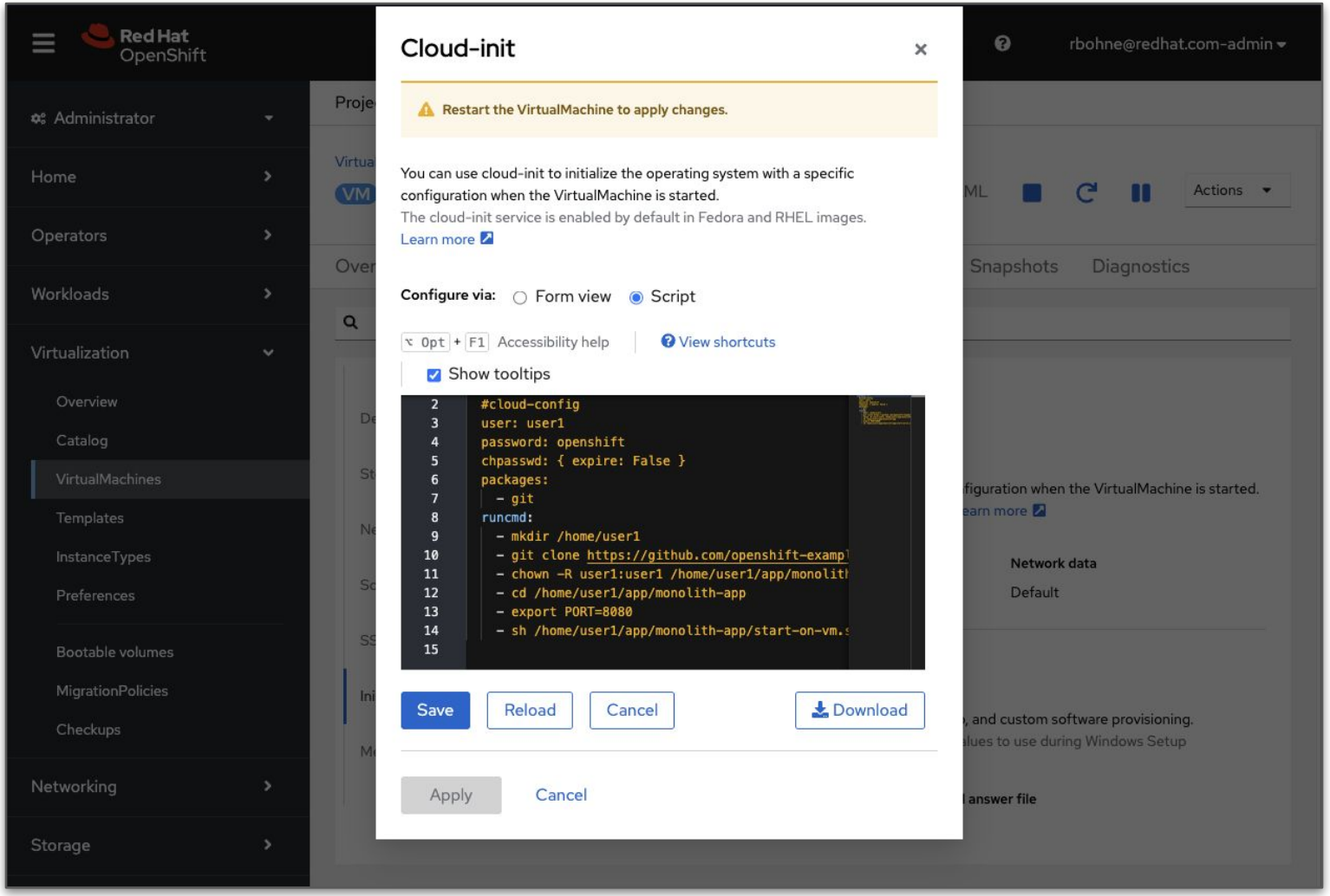
Use whatever you want

cloud-init

argo

Red Hat

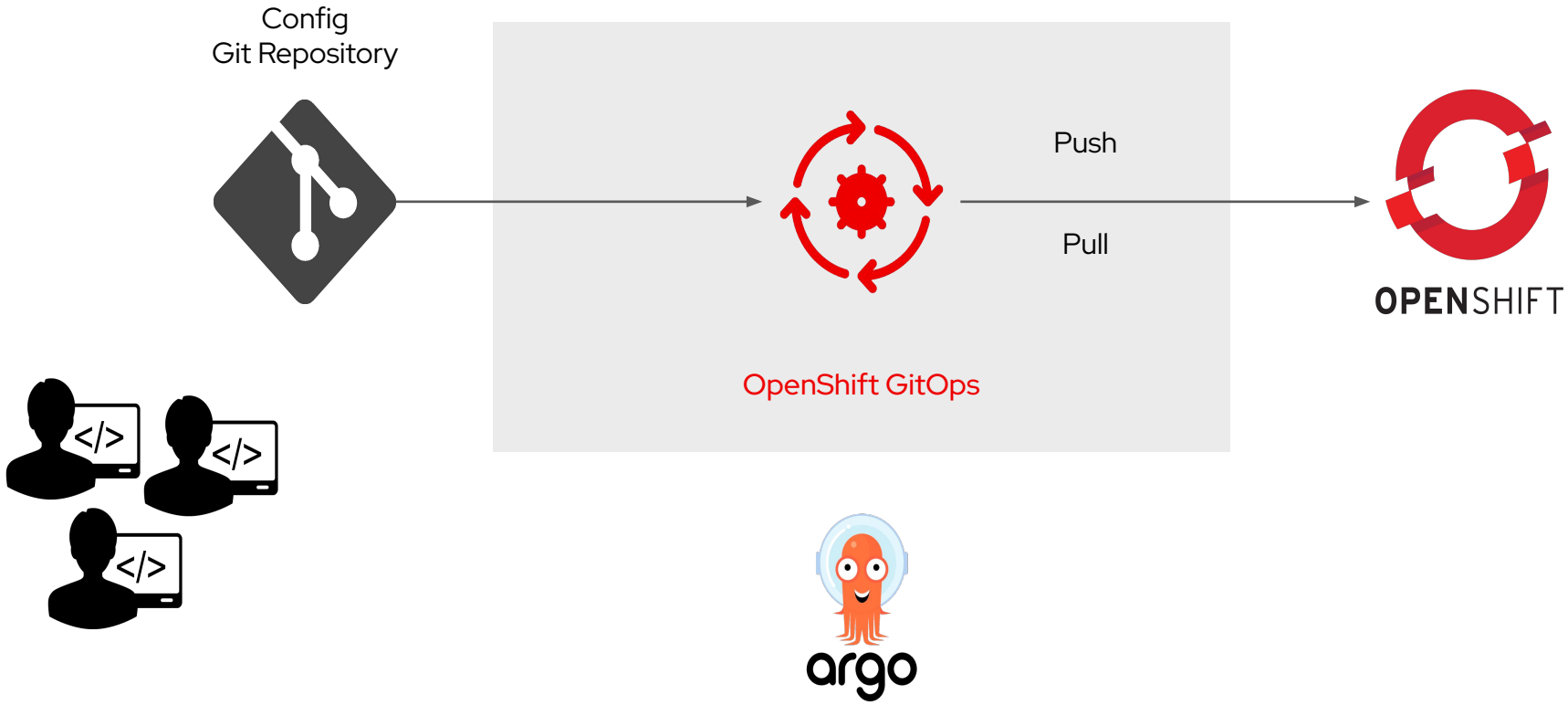
cloud-init



Red Hat

GitOps

e.g. for VM creation or Infrastructure management



https://github.com/stormshift/clusters/tree/main/apps/rdp-service

OpenShift Virtualization

GitOps

Red Hat Summit Connect Darmstadt 2025

Red Hat

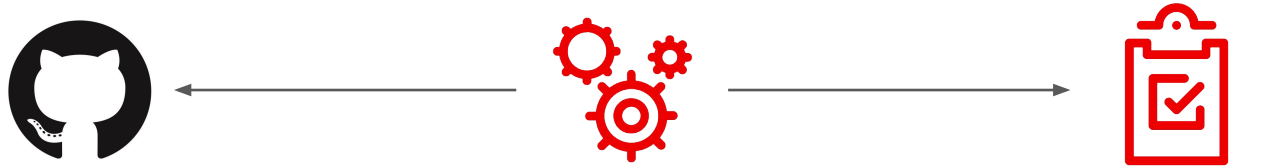
The x-Ops soup is getting real thick these days

1. DevOps - Development + Operations. The OG of this trend.
2. SecOps - Security + Operations. Bridging IT and security teams.
3. DevSecOps - Security baked into DevOps from the start.
4. GitOps - Managing infrastructure and apps using Git as the single source of truth.
5. AIOps - Artificial Intelligence for IT Operations. Using ML/AI to automate and enhance ops.
6. NoOps - Fully automated ops; in theory, no need for humans (in theory...).
7. MLOps - DevOps for machine learning workflows.
8. DataOps - Agile operations for data analytics pipelines.
9. TestOps - Integrating testing into the full lifecycle, tightly coupled with DevOps.
10. QAOps - Focused more directly on quality assurance in the DevOps process.
11. BizOps - Aligning business strategy with tech execution.
12. FinOps - Financial operations, especially for cloud cost management.
13. PeopleOps - HR modernization through tech and process improvements.
14. RevOps - Revenue operations - aligning sales, marketing, and customer success.
15. ComplianceOps - Automating compliance checks into the CI/CD pipeline.
16. RiskOps - Risk management integrated into operational workflows.
17. ModelOps - Operationalizing AI/ML models in production.
18. DesignOps - Scaling design processes and systems across teams.
19. ContentOps - Managing and automating content workflows.
20. ChatOps - Using chat platforms (like Slack) to manage ops and run commands.

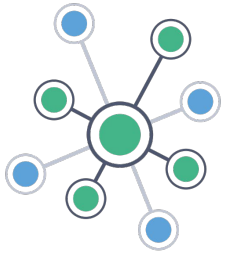
What is GitOps?

Overview

GitOps is a set of practices that leverages Git workflows to manage infrastructure and application configurations. By using Git repositories as the source of truth, it allows the DevOps team to store the entire state of the cluster configuration in Git so that the trail of changes are visible and auditable.



Use cases of using a GitOps Model



Continuous Delivery of application configurations

Apply Release Strategies (Blue/Green, Canary...)

Infrastructure Management in Kubernetes

Disaster Recovery

Sync Secrets

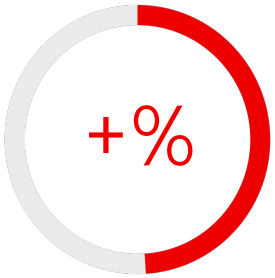
Drift Detection & Auto Remediation

Deploy/Manage Multiple Kubernetes Clusters

Security Handoff Deployments to Devs

Auto-Update Kubernetes YAMLs from a new repository image

Benefits of using a GitOps Model



Deploy faster / Innovation Velocity

Developer Centric
Quick and Easy Recovery (Mean Time To Recover - MTTR)

Secure / Separation of Concerns CI - CD

Auditability / Audit Log outside of Cluster

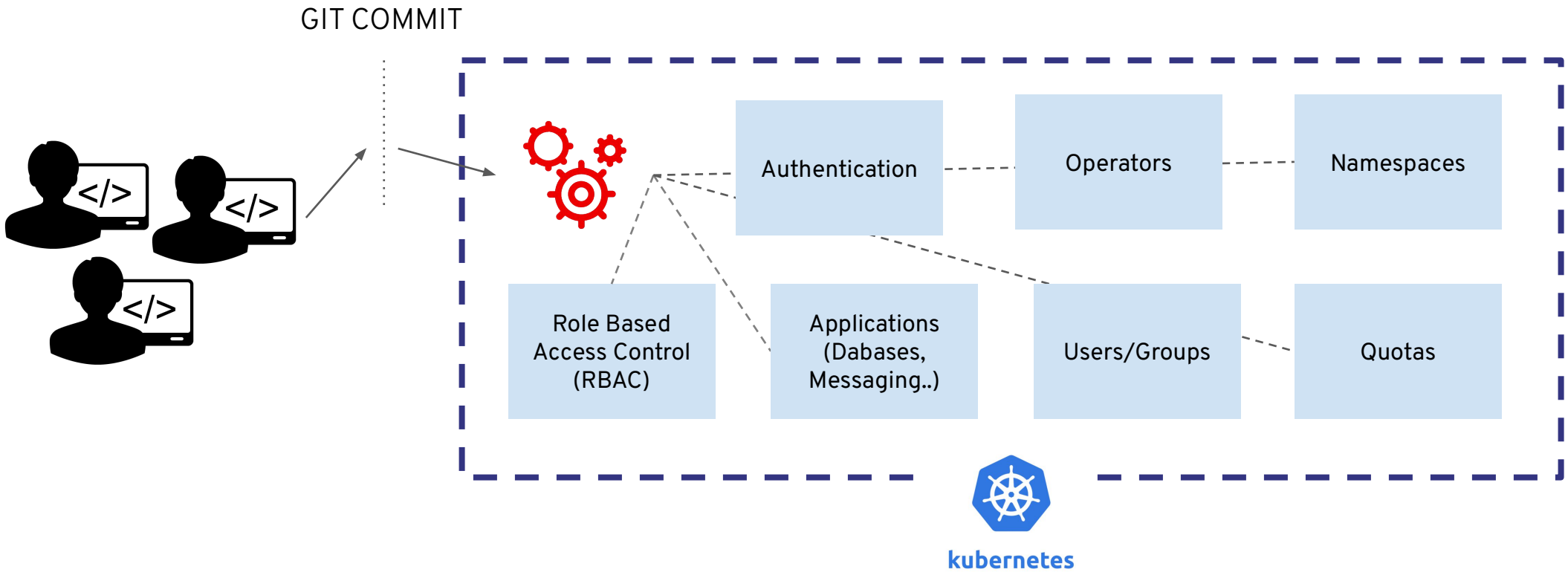
Rollout based on PRs / Rollback with Revert

Code is Reviewed

Observability / Single Source of Truth & Detect Config Drifts
Increase Stability and Reliability

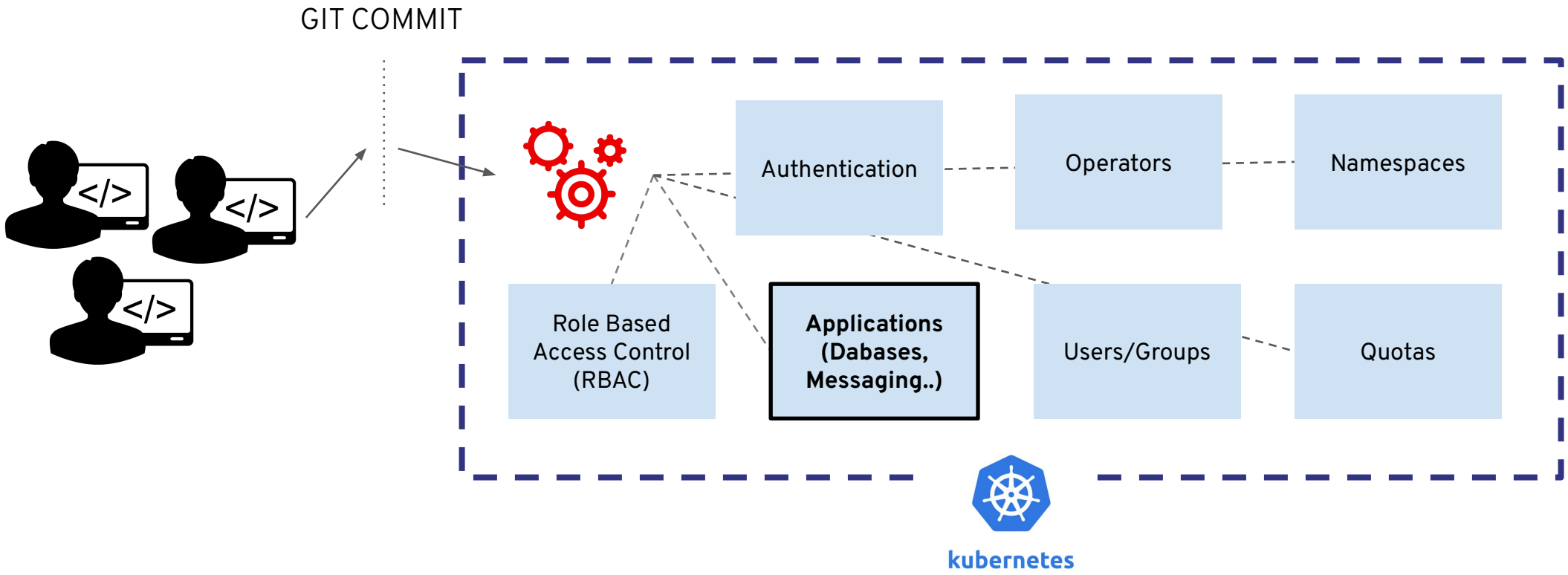
GitOps Infrastructure

Manage Clusters



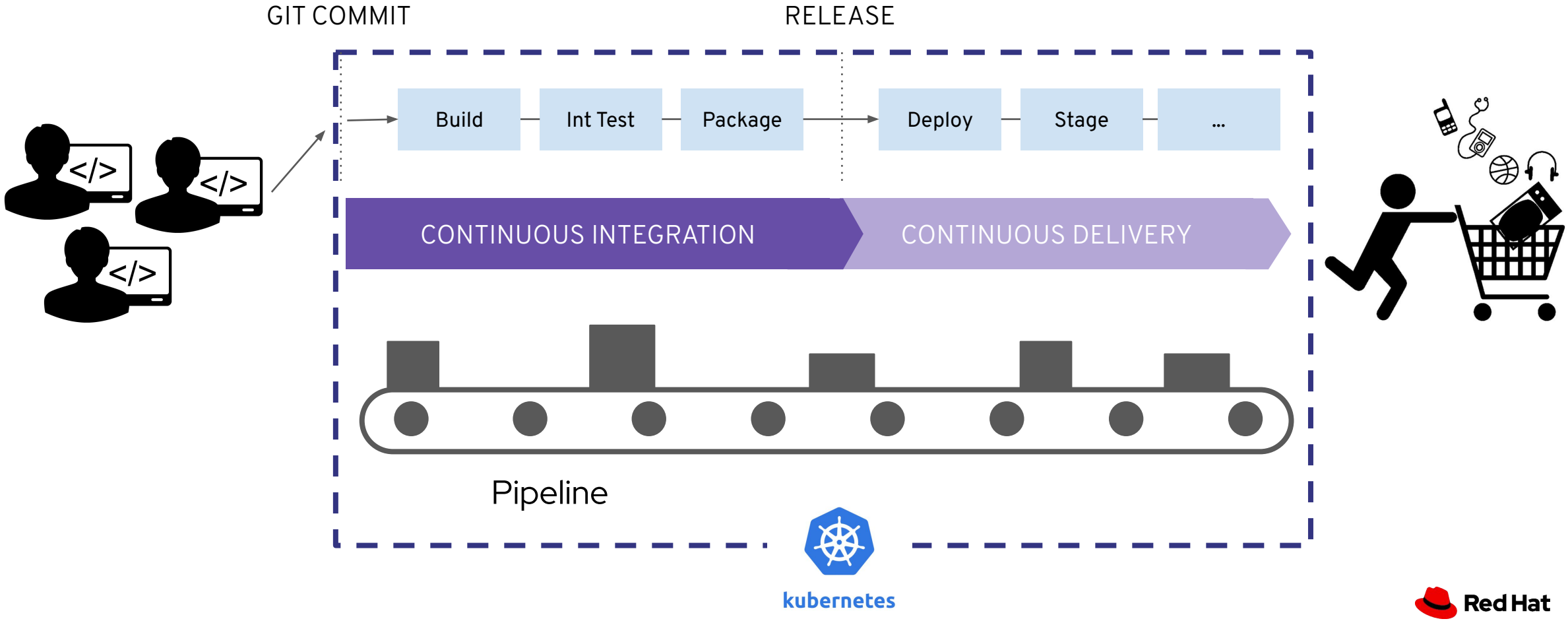
GitOps Infrastructure

Manage Clusters



GitOps Applications

Continuous Integration and Continuous Delivery



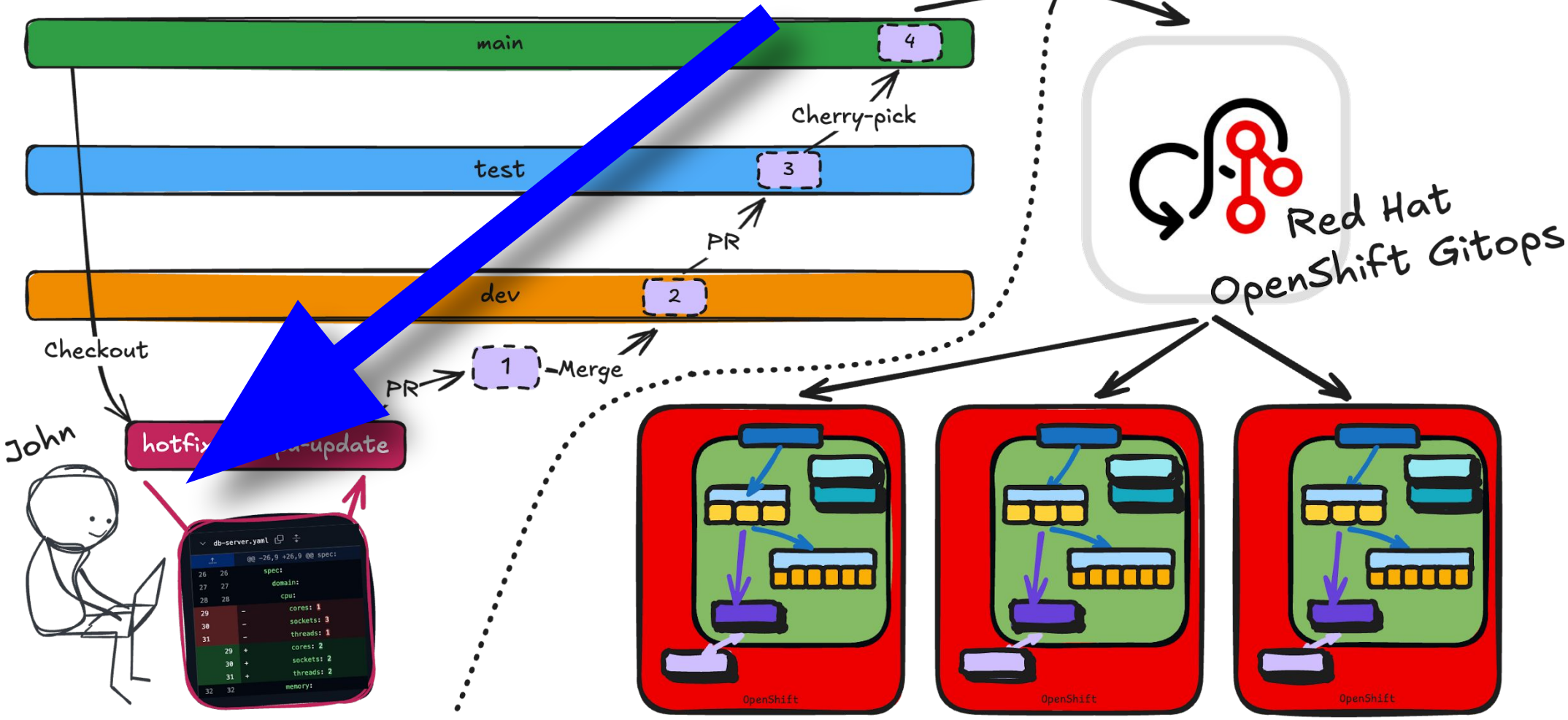
OpenShift Virtualization

GitOps Example

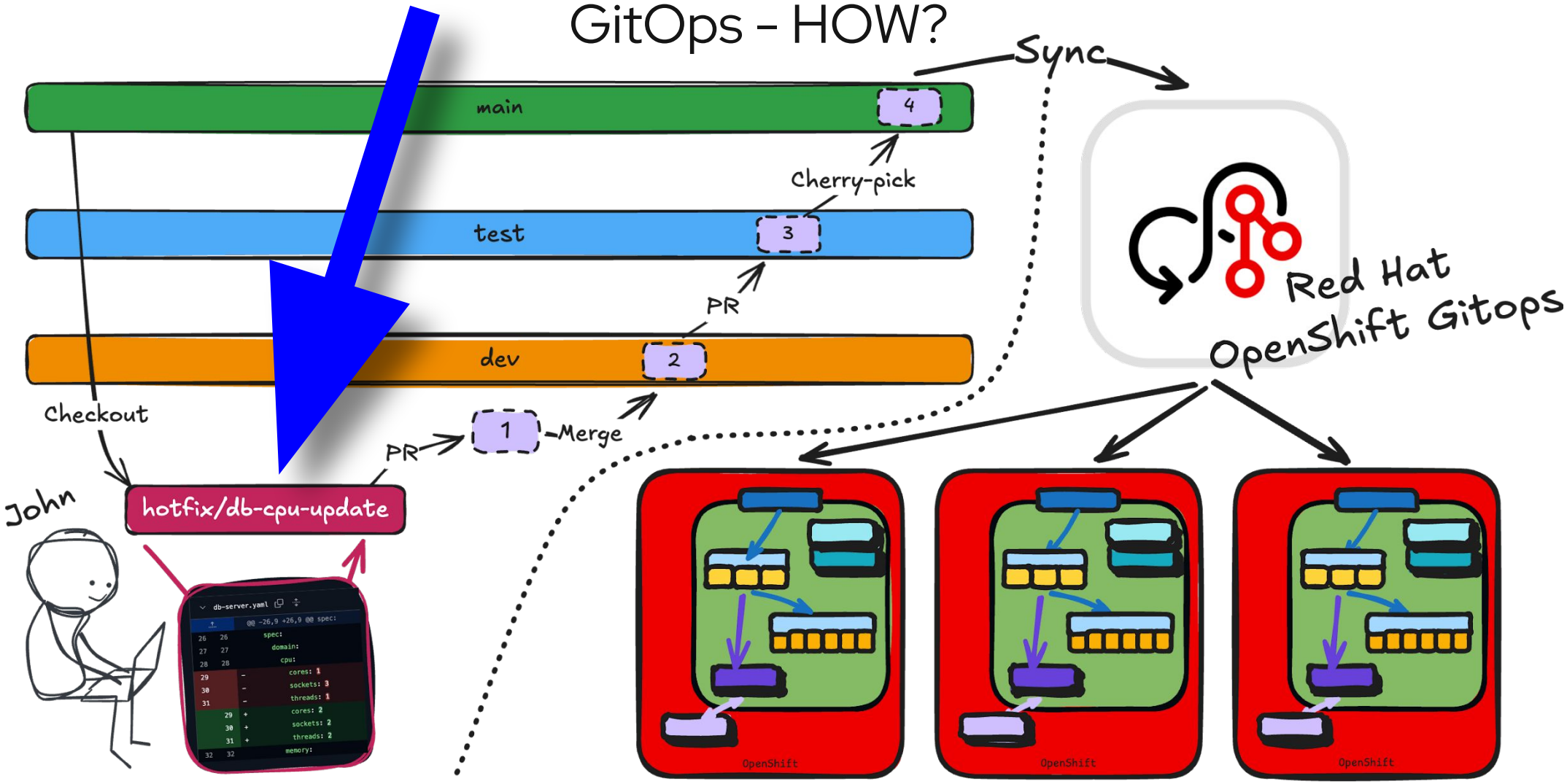
Red Hat Summit Connect Darmstadt 2025

Red Hat

GitOps - HOW?

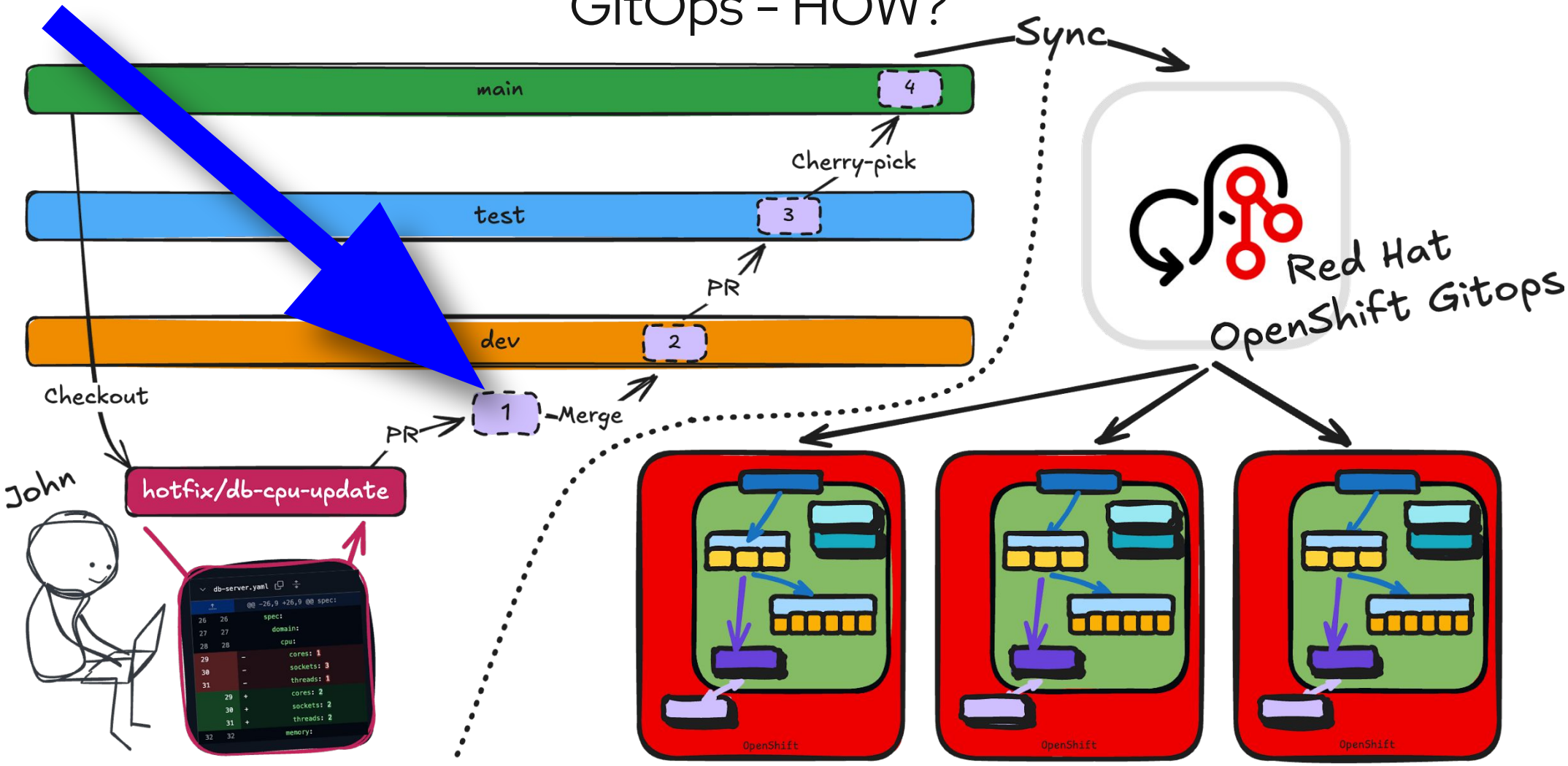


Red Hat


GitOps - HOW?
Sync
main
4
Cherry-pick
test
3
PR
dev
2
Checkout
PR
1
Merge
John
hotfix/db-cpu-update
db-server.yaml
Red Hat
OpenShift Gitops
OpenShift
OpenShift
OpenShift

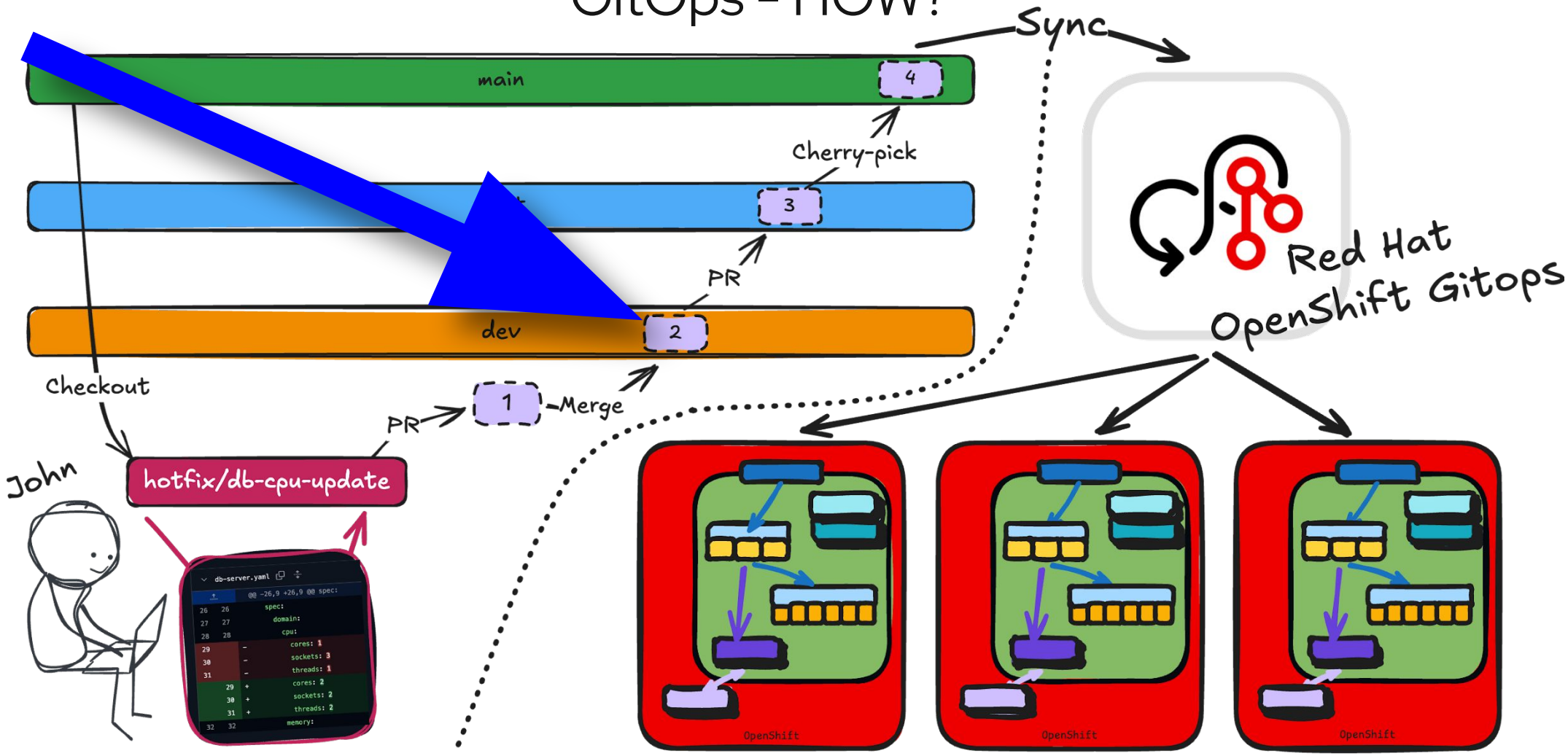
Red Hat

GitOps - HOW?



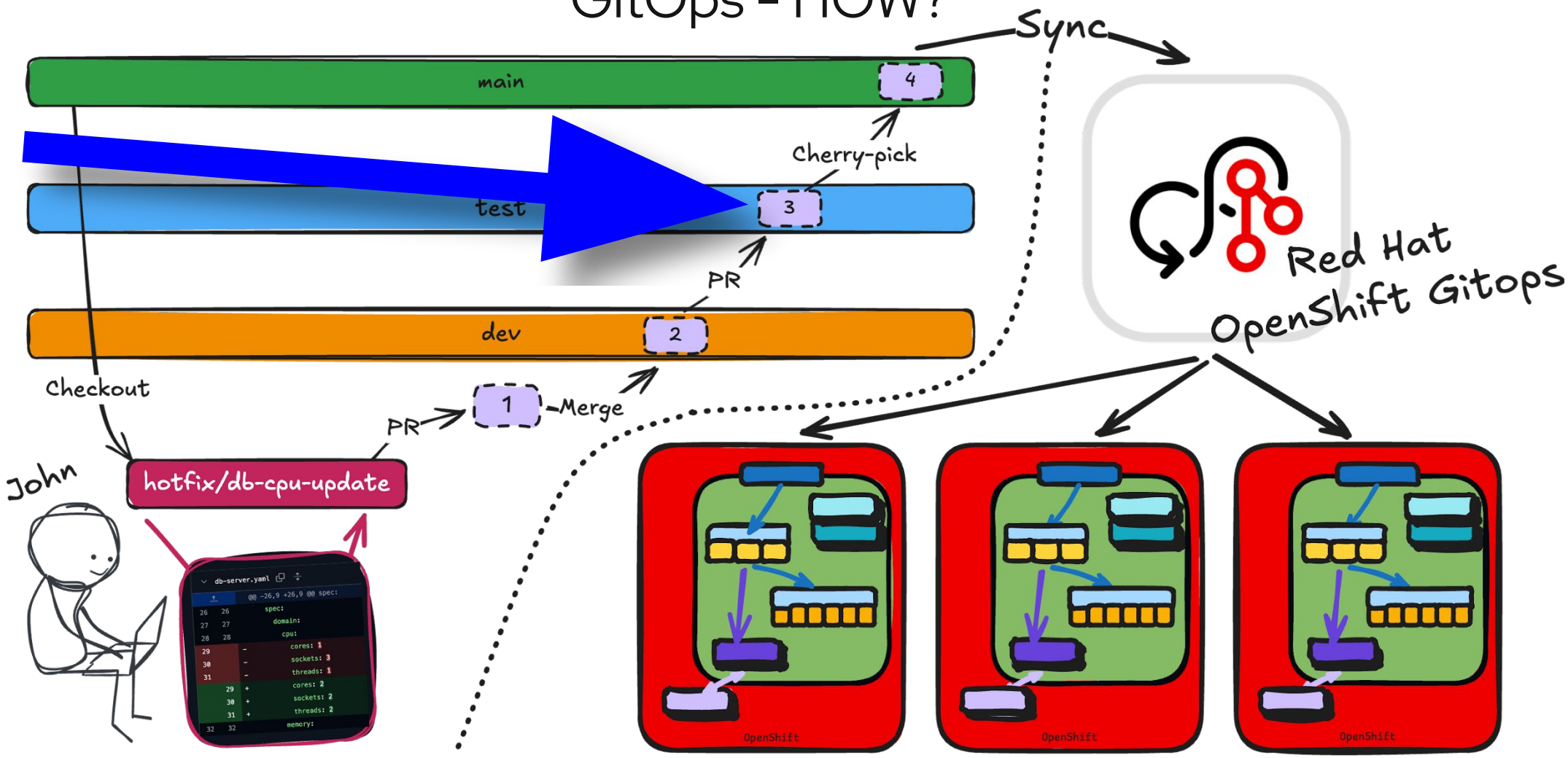
Red Hat

GitOps - HOW?



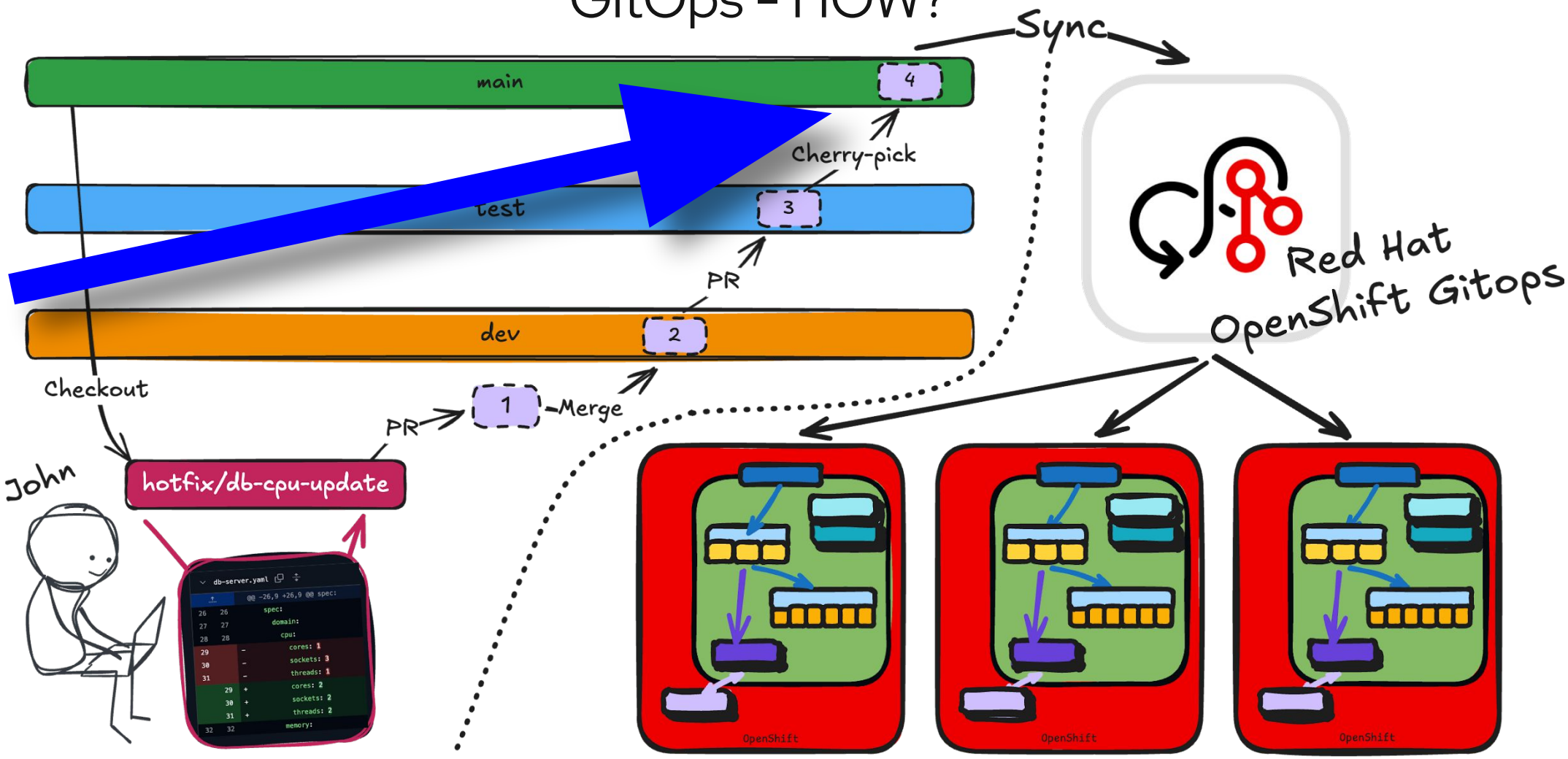
Red Hat

GitOps - HOW?



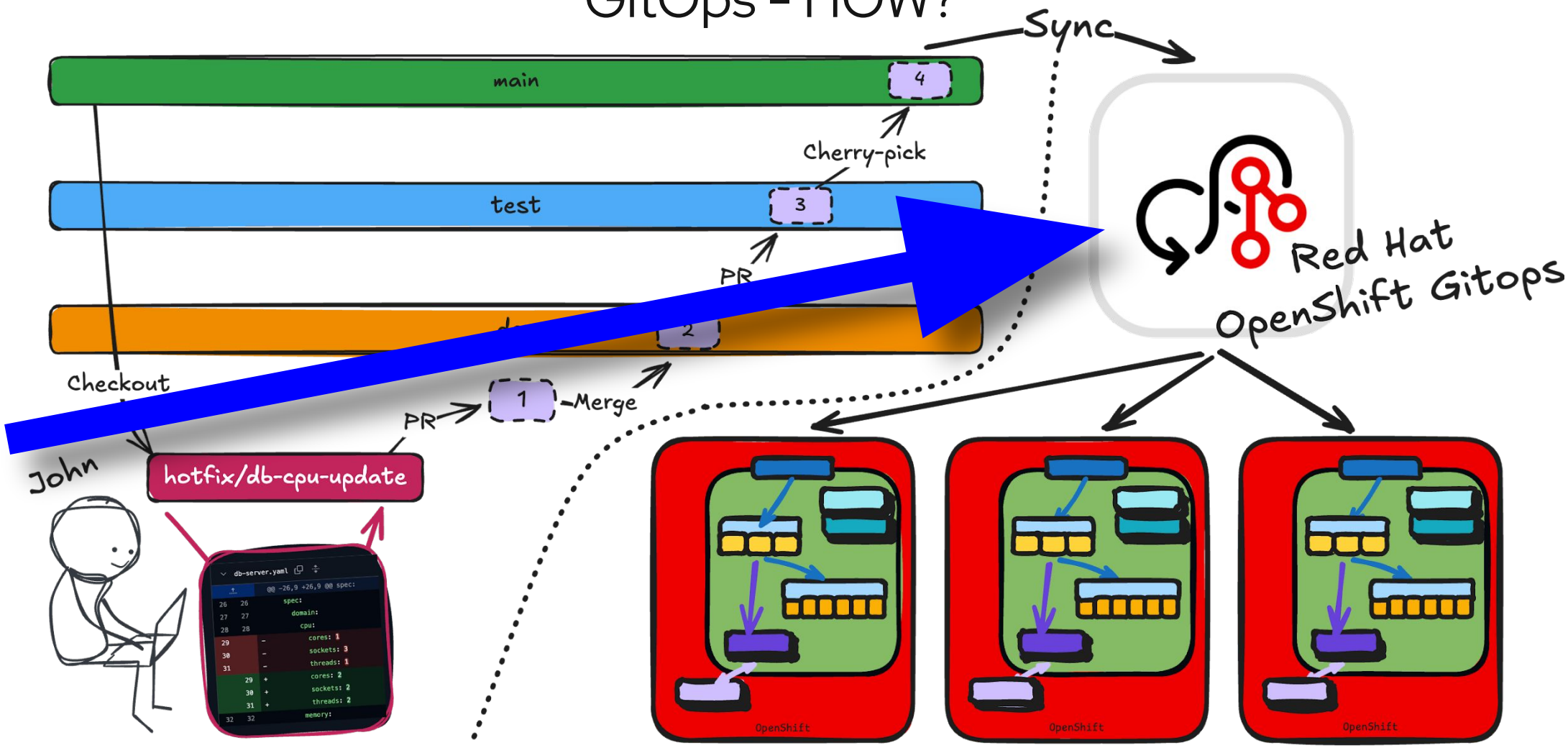
Red Hat

GitOps - HOW?



Red Hat

GitOps - HOW?

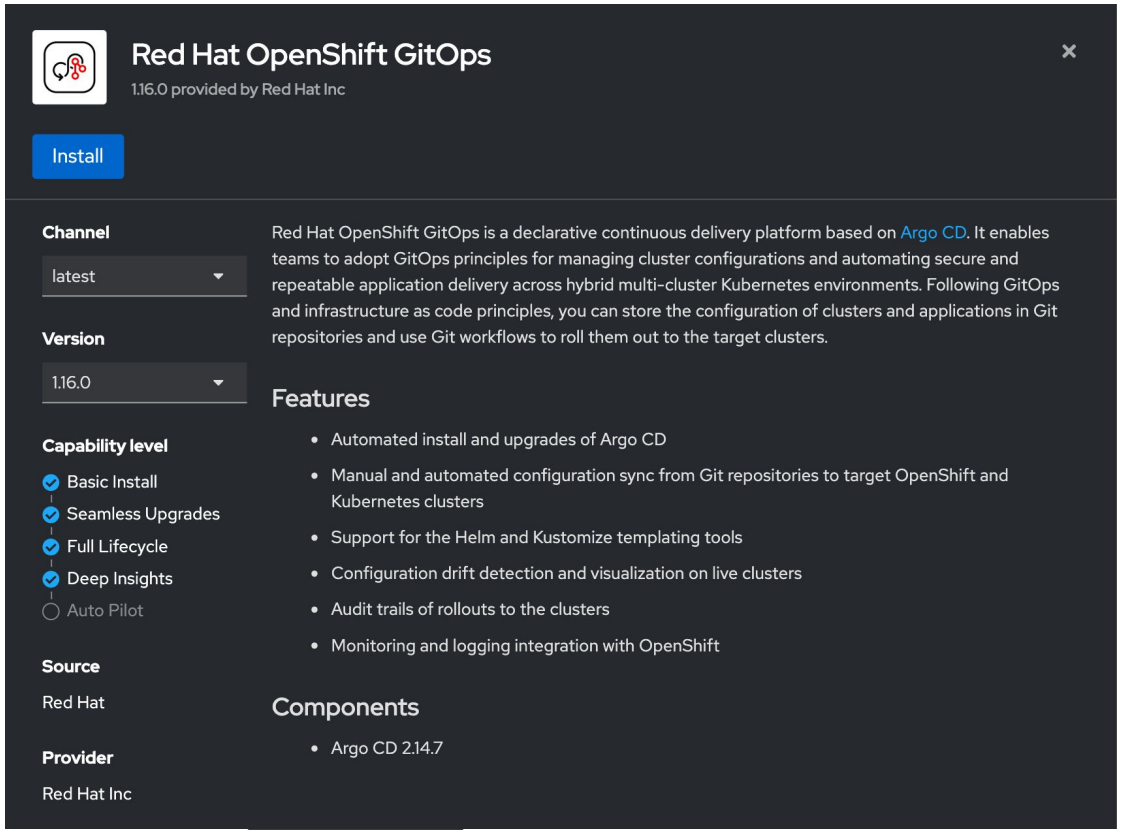


Red Hat

OpenShift GitOps - how to get it?

Red Hat OpenShift GitOps Operator:

- Available from Operators Hub in all OpenShift flavours*
- Tightly integrated
- Full lifecycle management, including seamless upgrades and deep insights
- ArgoCD underneath



* - Red Hat OpenShift Kubernetes Engine and Red Hat OpenShift Virtualization Engine: for VM use cases only

Red Hat

Demo time

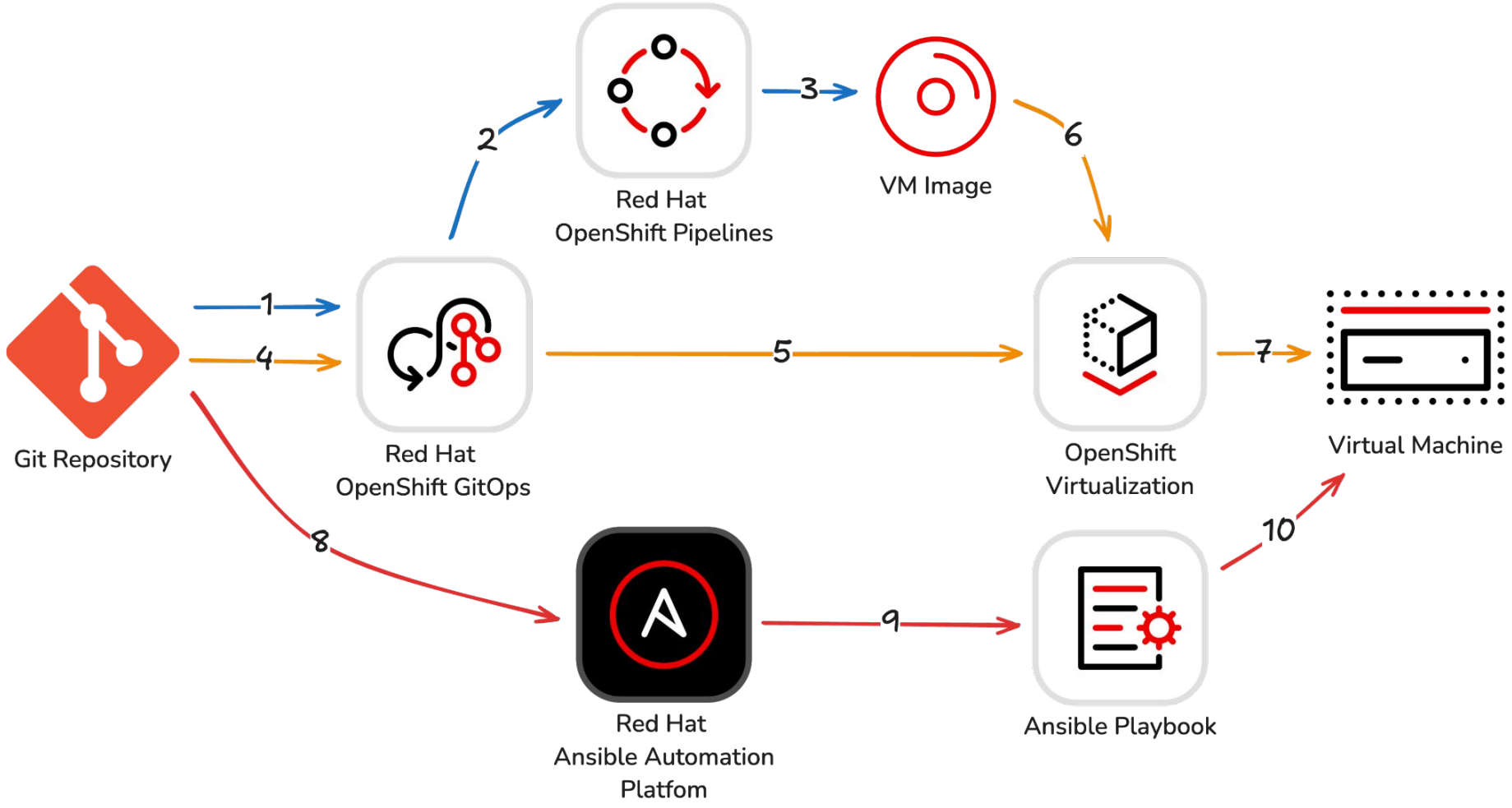
- Look up the Operator.
- Look up the Route for ArgoCD in the gitops Namespace.
- Look at your git repo.
- Login to ArgoCD.
- Create a new app with the prepared git.

Possible points to keep in mind

- State drift - **VMs are traditionally mutable** - users might log in and change configurations manually (network settings, installed software, etc.).
  a. GitOps assumes the cluster state matches what's in Git.
  b. VM changes not represented in Git cause "state drift" — difficult to reconcile.
  c. Immutable infrastructure patterns are harder to enforce on VMs compared to containers.

Remedy: strong config automation (Ansible), limited direct management access

There is more!



Red Hat

Red Hat Summit

Connect

Thank you


linkedin.com/company/red-hat


facebook.com/redhatinc


youtube.com/user/RedHatVideos


twitter.com/RedHat