CI/CD Automation with Ansible

Kevin Tatro

Senior Solutions Architect

Kevin Tatro

Senior Solutions Architect

ktatro@redhat.com
M: 248-459-2440
East Lansing, MI

Before Red Hat

- Insurance Industry
- Red Hat Customer
- Application and Solution Architecture
- DevOps adoption
- Application Development - Java EE, Spring Boot

Agenda

What we’ll discuss today

- Overview of Ansible Automation Platform
- Introduction to CI/CD
- Ansible and CI/CD Workflows
- Demo

Red Hat® Ansible® Automation Platform

Red Hat Ansible Automation Platform

Improve efficiency

Maximize current resources while investing and involving your entire IT organization.

Increase productivity

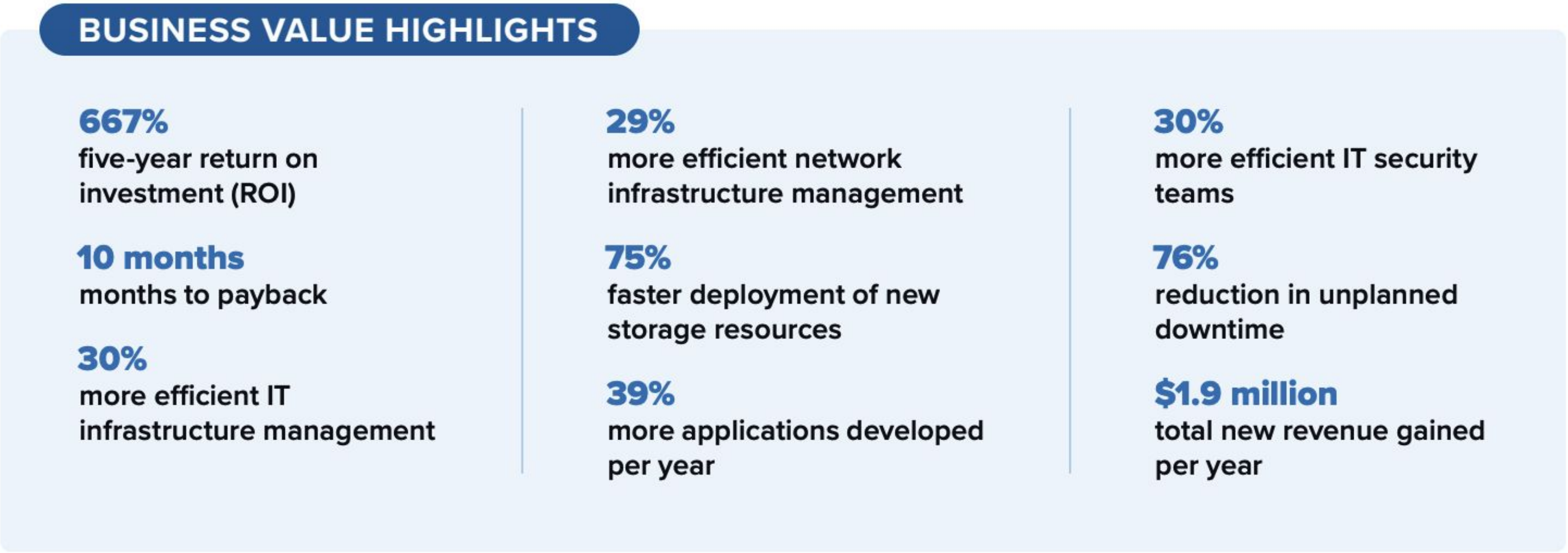
Accelerate business outcomes and consolidate tools across the entire hybrid cloud infrastructure.

Control risk and expenses

Increase compliance while minimizing errors and reducing the cost of production.

Business value of automating with Ansible Automation Platform[1]

Red Hat
Ansible Automation
Platform



BUSINESS VALUE HIGHLIGHTS

667%
five-year return on investment (ROI)

10 months
months to payback

30%
more efficient IT infrastructure management

29%
more efficient network infrastructure management

75%
faster deployment of new storage resources

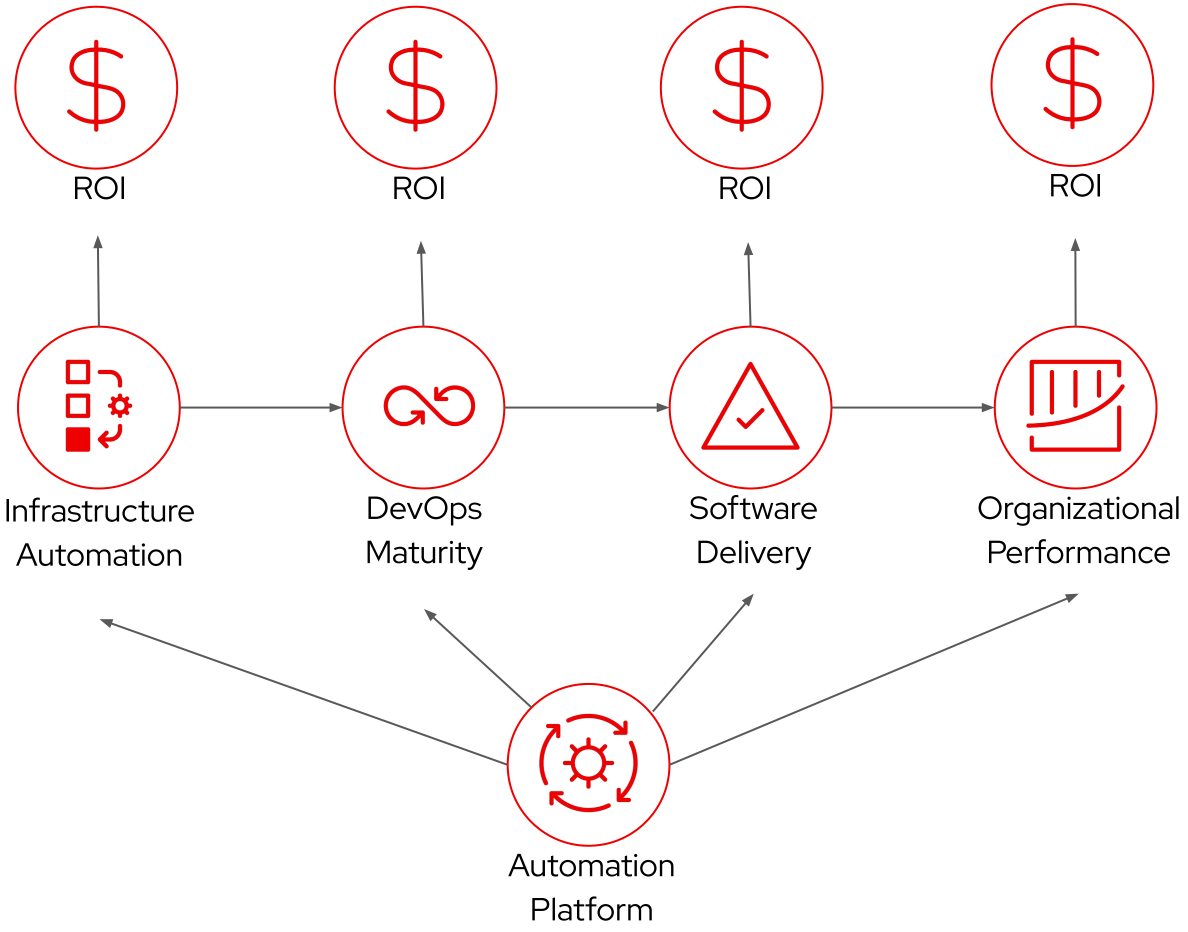
39%
more applications developed per year

30%
more efficient IT security teams

76%
reduction in unplanned downtime

$1.9 million
total new revenue gained per year

[1] Harsh Singh and Stephen Elliot. "The Business Value of Red Hat Ansible Automation Platform," IDC, sponsored by Red Hat, October 2021.


ROI
ROI
ROI
ROI
Infrastructure Automation
DevOps Maturity
Software Delivery
Organizational Performance
Automation Platform

Why the Ansible Automation Platform?

Powerful

Orchestrate complex processes at enterprise scale.

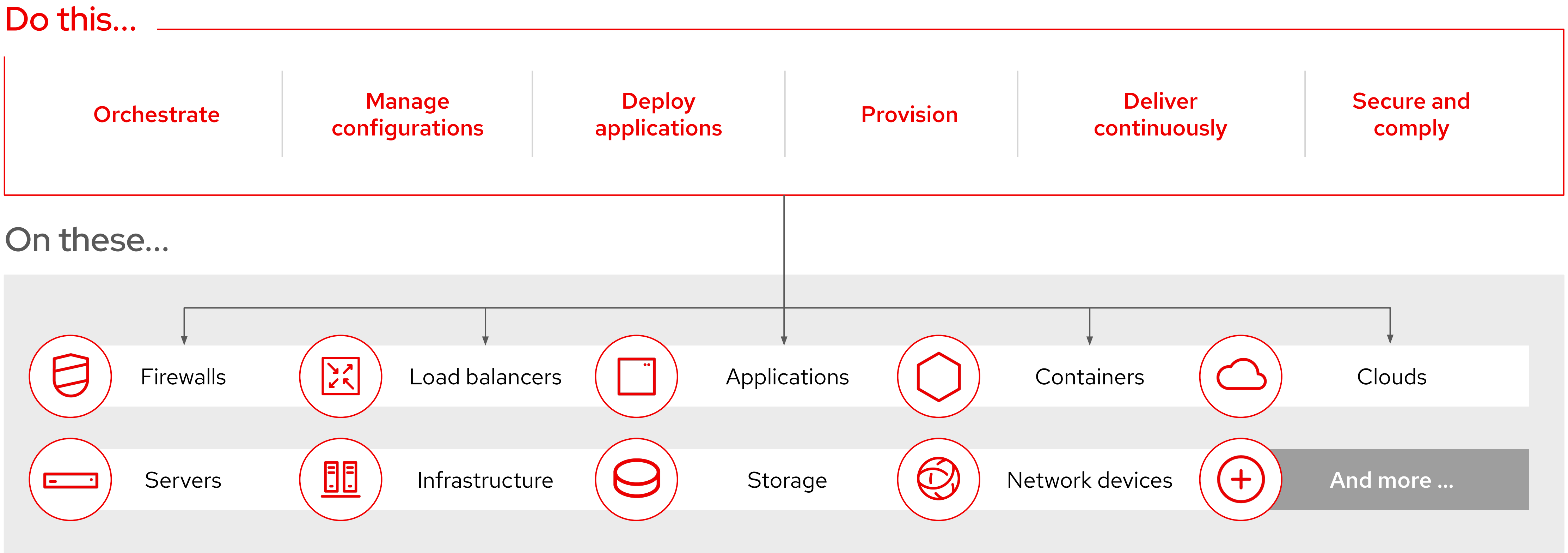
Simple

Simplify automation creation and management across multiple domains.

Agentless

Easily integrate with hybrid environments.

Automate the deployment and management of automation for your entire IT footprint



Do this...

Orchestrate

Manage configurations

Deploy applications

Provision

Deliver continuously

Secure and comply

On these...

Firewalls

Load balancers

Applications

Containers

Clouds

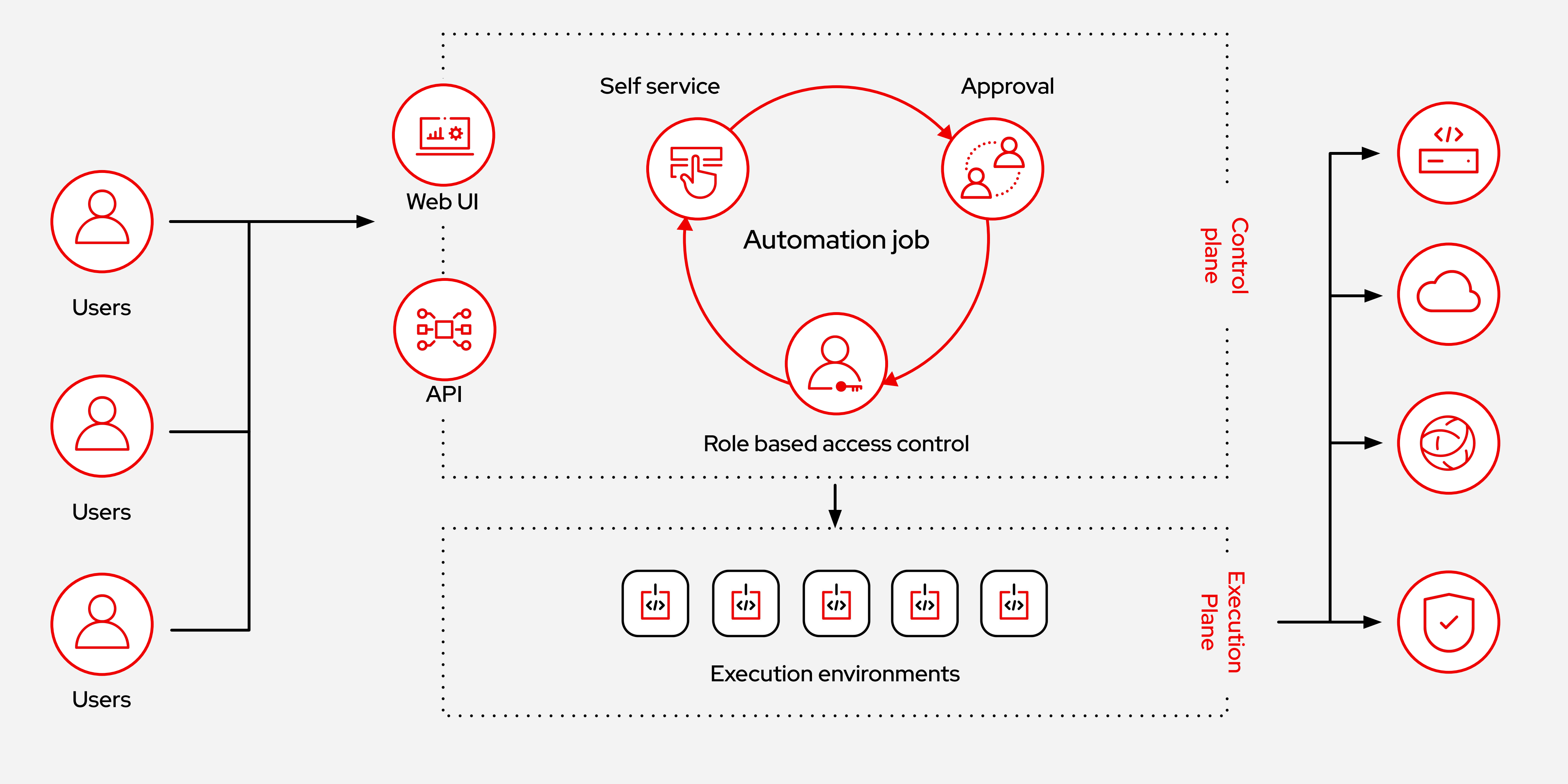
Servers

Infrastructure

Storage

Network devices

And more ...


Users
Users
Users
Web UI
API
Self service
Approval
Automation job
Role based access control
Control plane
Execution environments
Execution Plane

CI/CD Overview

CI/CD Concepts

Continuous Integration

Continuous Integration (CI) is a development practice where developers integrate code into a shared repository frequently, preferably several times a day. Each integration can then be verified by an automated build and automated tests.

Continuous Delivery

Continuous delivery (CD) is an extension of continuous integration since it automatically deploys all code changes to a testing and **manually** to production environment after the build stage.

Continuous Deployment

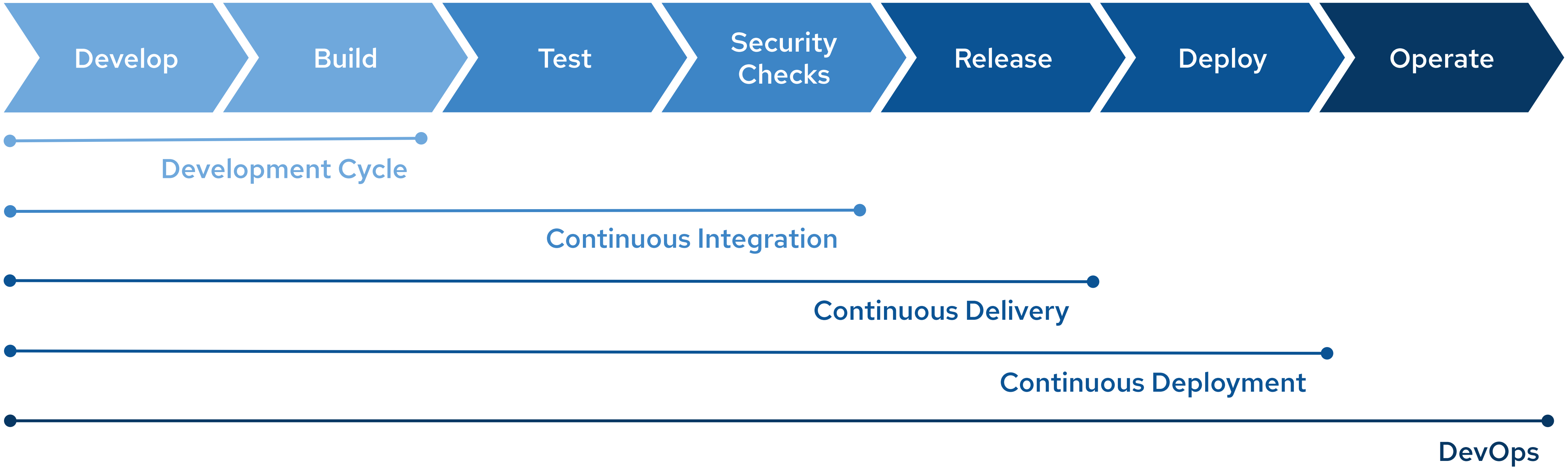
Continuous deployment (CD) is a strategy for software releases wherein any code that has been committed which passes the automated testing phase is automatically released into the production environment, making changes that are visible to the software's users.

DevOps

A set of practices that combines software development and IT operations to accelerate the systems development life cycle and provide continuous delivery with high software quality

Stages in Software Development

A key DevOps principle for automation, consistency and reliability



CI/C (Delivery or Deployment)?

- Can be either, depending on the environment you are working in
- It’s a journey, and will vary from project to project and team to team even within a single organization
- What are the costs and benefits to these approaches and what does that mean for my organization/customer?
- How strong is my automated testing culture?
- Where/how are my bugs discovered?
- What regulatory requirements do I have to meet?

Continuous Integration

Cost

- Mature unit testing for new development
- Tooling for continuous integration
- Frequent code check-ins and changes to developer workflows

Benefit

- Building strong regression testing into your codebase
- Bugs are discovered sooner
- Building and testing processes are automated and streamlined
- Less manual testing effort required

https://www.atlassian.com/continuous-delivery/principles/continuous-integration-vs-delivery-vs-deployment

Continuous Delivery

Cost

- Even more mature unit testing for new development
- Strong integration testing
- Potentially additional environments for smoke testing
- Less control over what code is released when (embrace feature toggles)

Benefit

- Release day becomes automated, no more manually cherry picking commits, rebuilding, packaging specific code for release.
- Shorter release cycles
- Build confidence in releases, less trepidation on release day

https://www.atlassian.com/continuous-delivery/principles/continuous-integration-vs-delivery-vs-deployment

Continuous Deployment

Cost

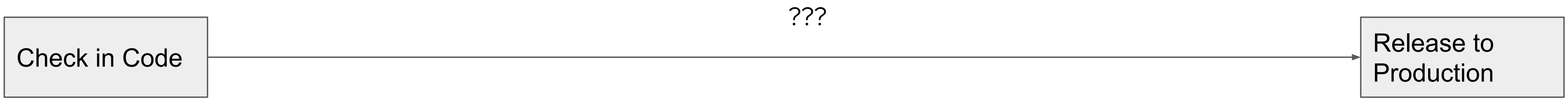
- Highest level of end-to-end automated testing
- Cannot live without feature toggles
- Additional effort to understand, document, explain, and comply with compliance and audit requirements

Benefit

- Deployment deltas get smaller, easier to fix
- New features get to production fast
- Very short customer feedback loops

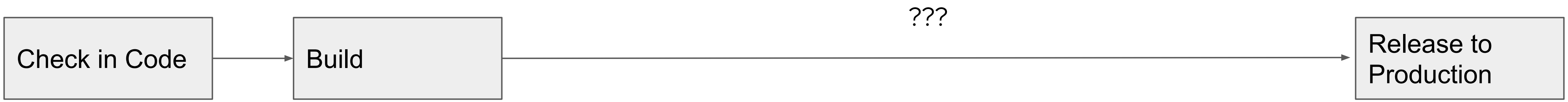
https://www.atlassian.com/continuous-delivery/principles/continuous-integration-vs-delivery-vs-deployment

Ansible and CI/CD Workflows


Check in Code

???

Release to Production

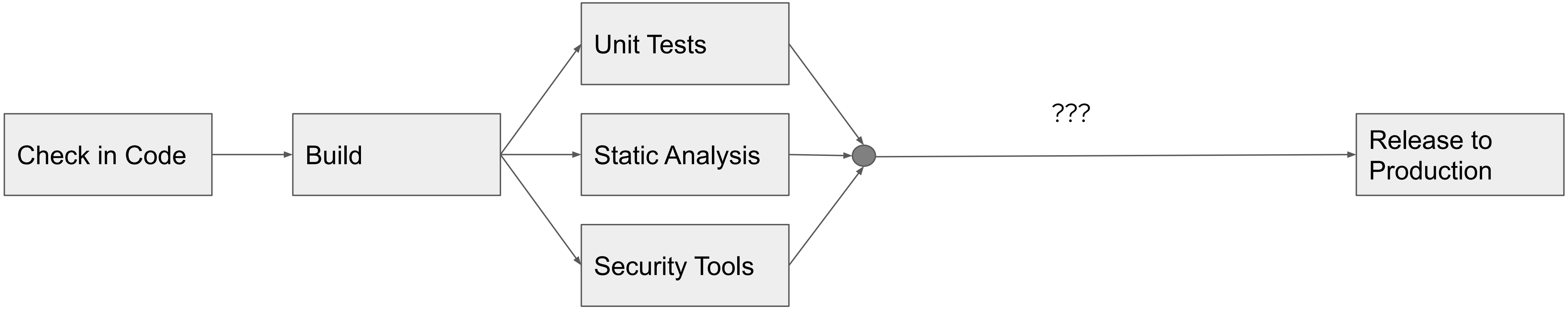


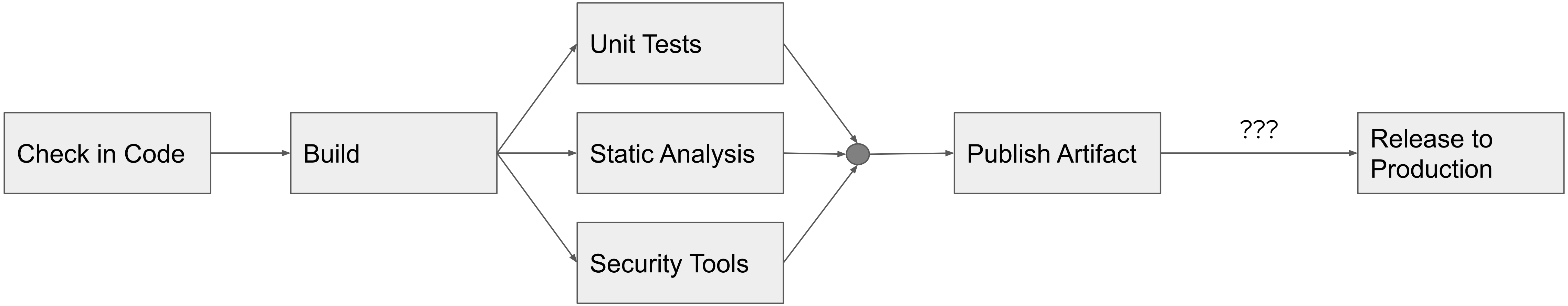
Check in Code

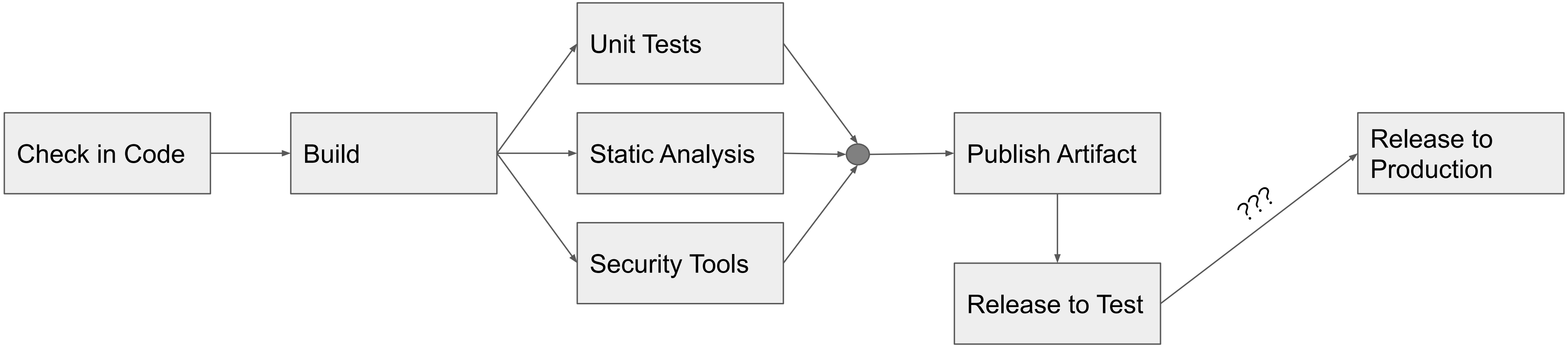
Build

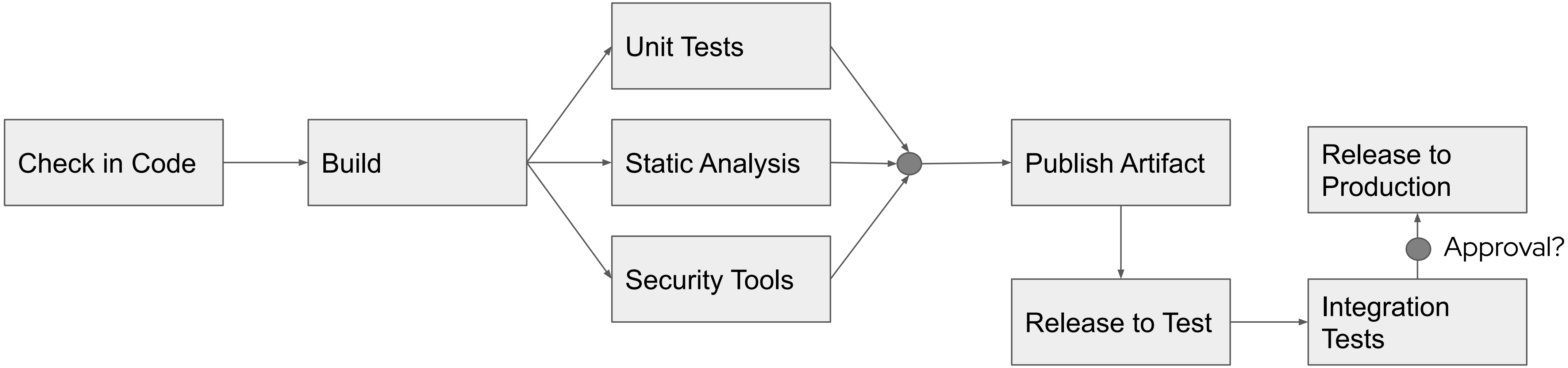
???

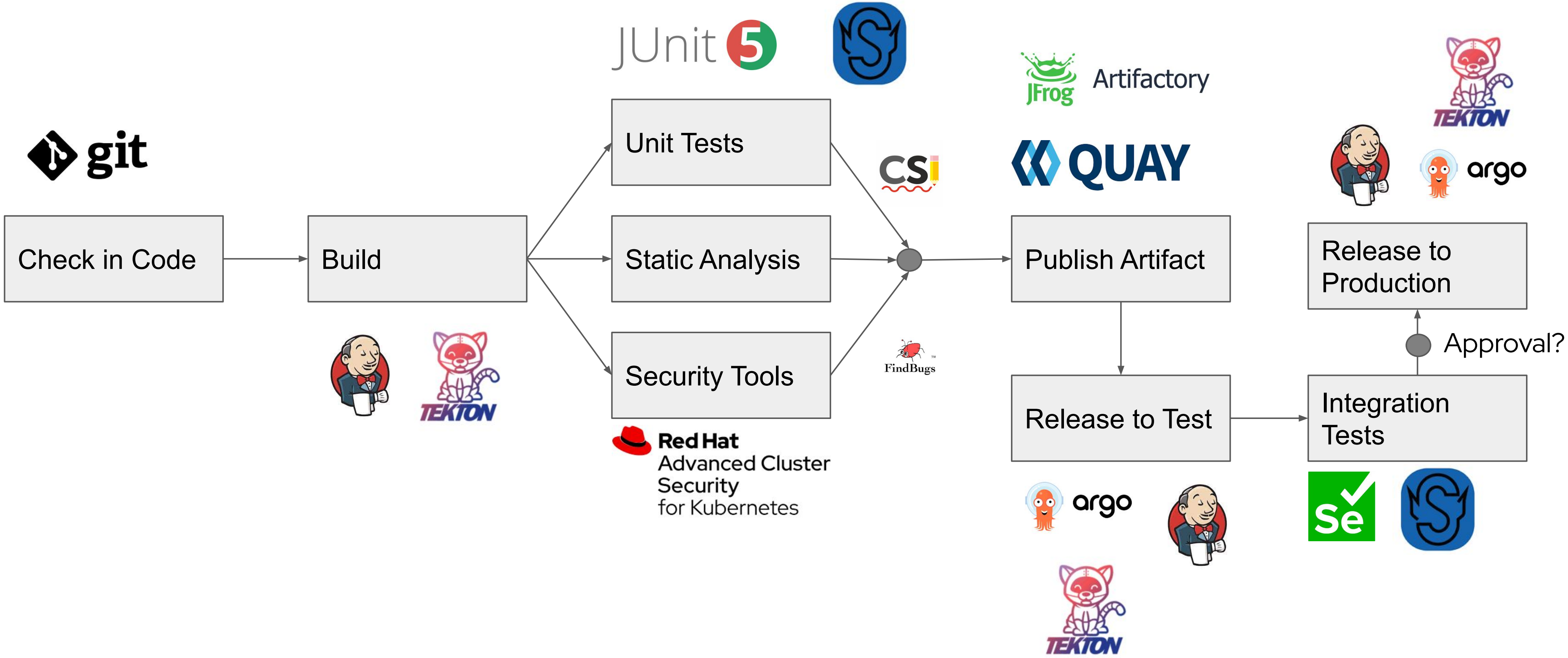
Release to Production

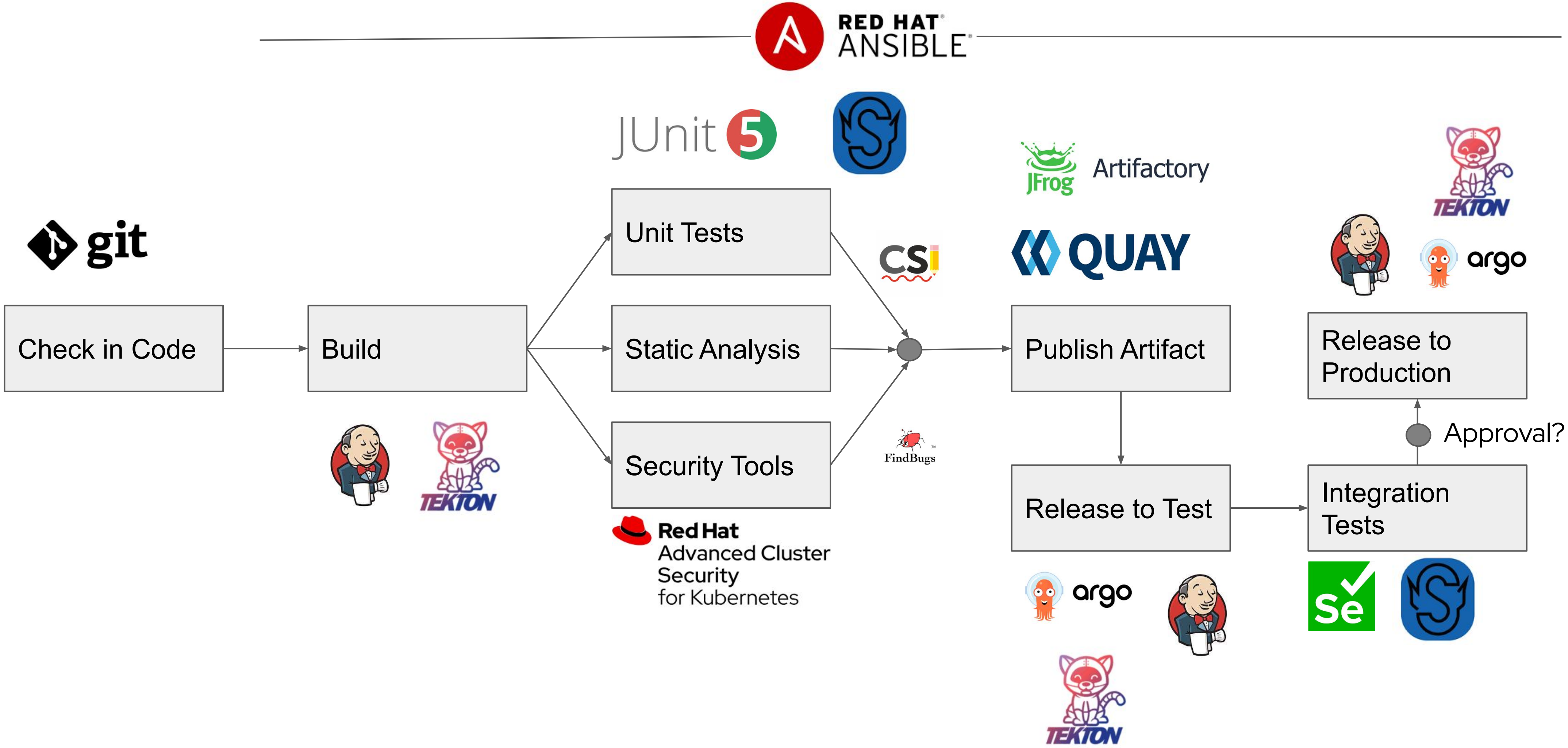

Check in Code
Build
Unit Tests
Static Analysis
Security Tools
???
Release to Production

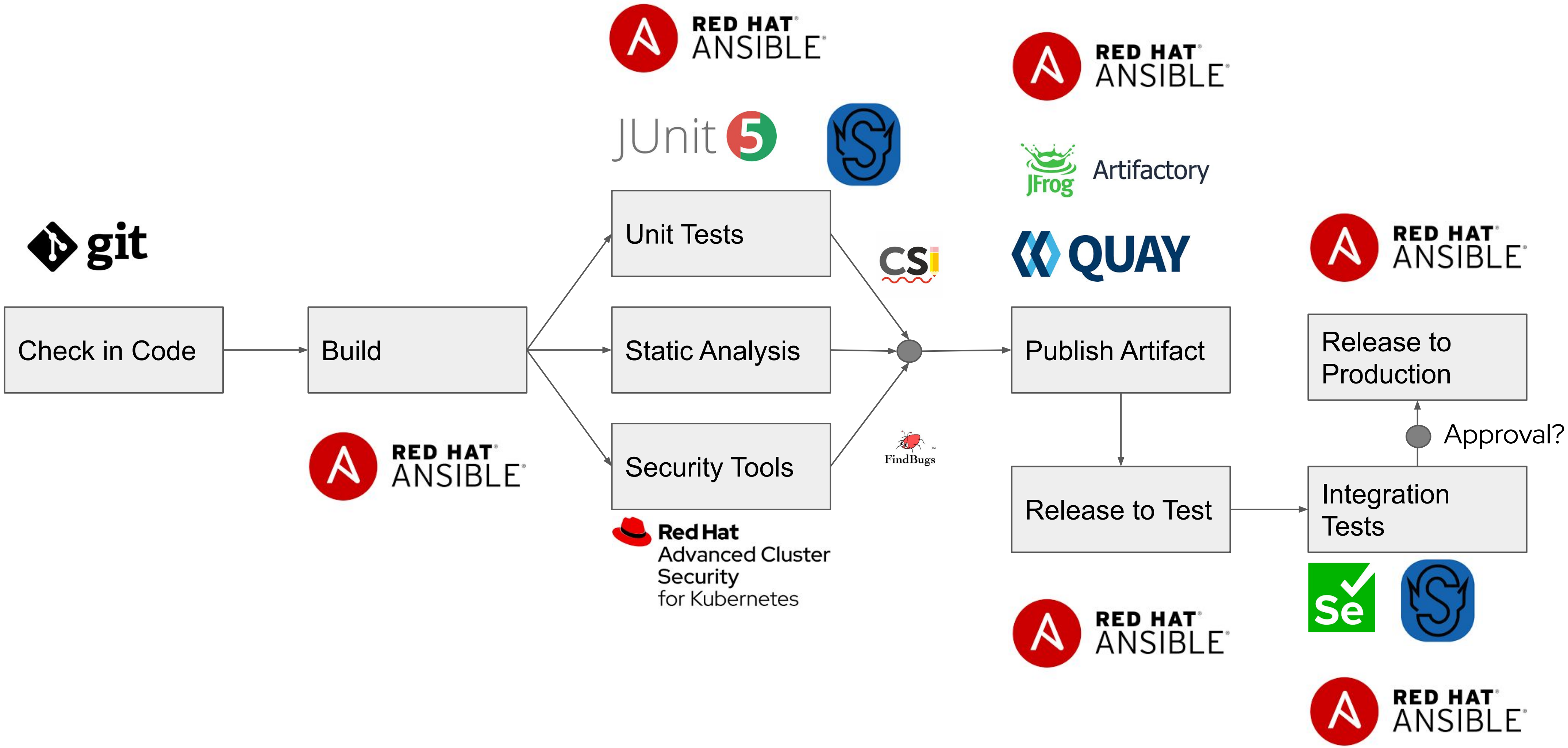

Check in Code
Build
Unit Tests
Static Analysis
Security Tools
Publish Artifact
???
Release to Production

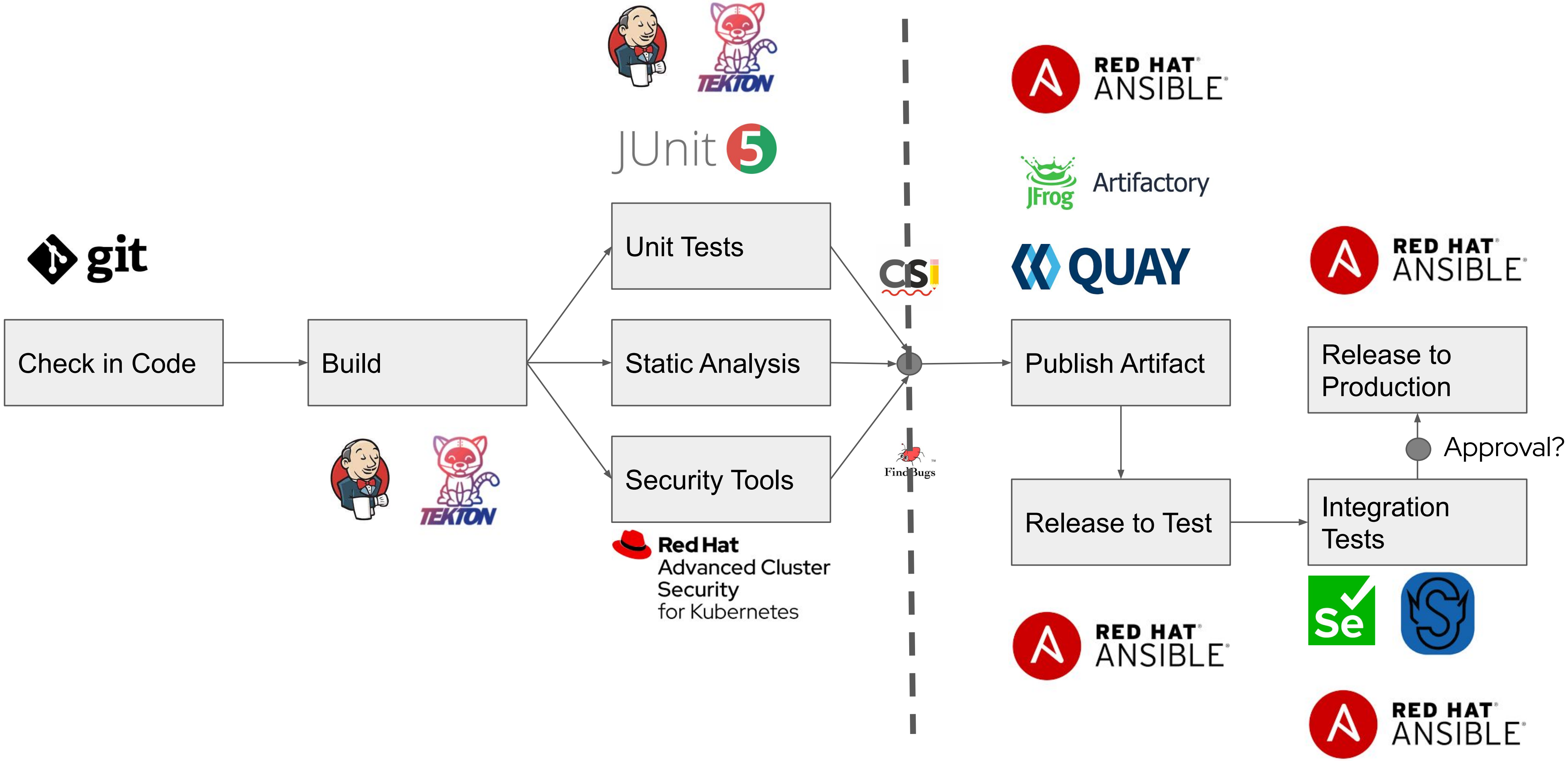

Check in Code
Build
Unit Tests
Static Analysis
Security Tools
Publish Artifact
Release to Test
???
Release to Production


Check in Code
Build
Unit Tests
Static Analysis
Security Tools
Publish Artifact
Release to Test
Integration Tests
Approval?
Release to Production


git
Check in Code
Build
JUnit 5
Unit Tests
Static Analysis
Security Tools
CSi
FindBugs
Red Hat
Advanced Cluster
Security
for Kubernetes
TEKTON
Artifactory
JFrog
QUAY
Publish Artifact
Release to Test
argo
TEKTON
Integration
Tests
Se
Approval?
Release to
Production
argo
TEKTON


RED HAT ANSIBLE
git
Check in Code
Build
TEKTON
JUnit 5
Unit Tests
Static Analysis
Security Tools
CSi
FindBugs
Red Hat Advanced Cluster Security for Kubernetes
JFrog Artifactory
QUAY
Publish Artifact
Release to Test
argo
TEKTON
Integration Tests
Se
Approval?
Release to Production
argo
TEKTON


git
Check in Code
Build
RED HAT ANSIBLE
RED HAT ANSIBLE
JUnit 5
Unit Tests
Static Analysis
Security Tools
Red Hat Advanced Cluster Security for Kubernetes
CSI
FindBugs
RED HAT ANSIBLE
JFrog Artifactory
QUAY
Publish Artifact
Release to Test
RED HAT ANSIBLE
RED HAT ANSIBLE
Release to Production
Approval?
Integration Tests
Se
RED HAT ANSIBLE



Check in Code

Build

Unit Tests

Static Analysis

Security Tools

Publish Artifact

Release to Test

Integration Tests

Approval?

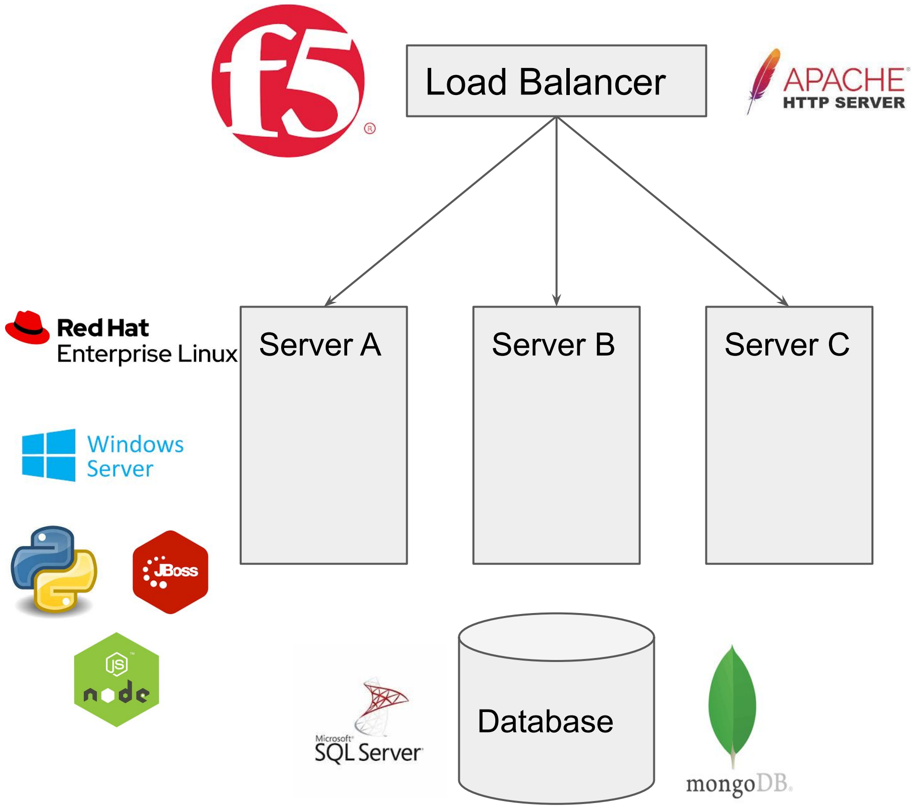
Release to Production

Diving into an Abstraction

Release to <Environment>

- Every environment is going to be different
- Environments are going vary within a single organization or a even a single development team
- Releasing to Production is going to vary based on the type of application and the where it is deployed (i.e. Containerized Application on Openshift, COBOL application on Mainframe, JBoss Application running on a VM)
- A release could be as simple as updating a deployment YAML or very complex...

Complex Deployment Scenario

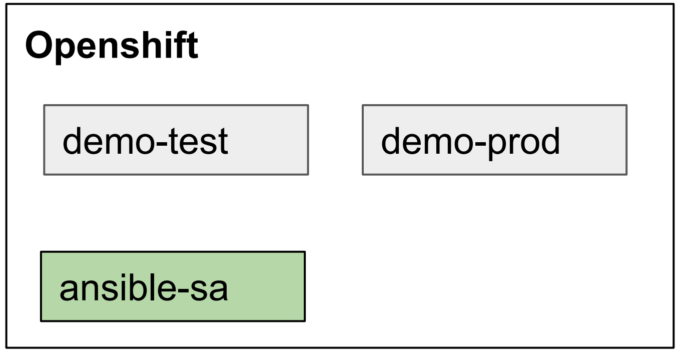


1. Take Snapshot of Database Server
2. Remove Server A from Load Balancer configuration
3. Stop Application Server on Server A
4. Run Custom Cleanup Script on Server A
5. Copy new artifact to Server A
6. Restart Server A
7. Perform Health Check on Server A
8. Add Server A back to load balancer configuration
9. Repeat 2-8 for Server B and Server C

RED HAT ANSIBLE

Demo

Environment

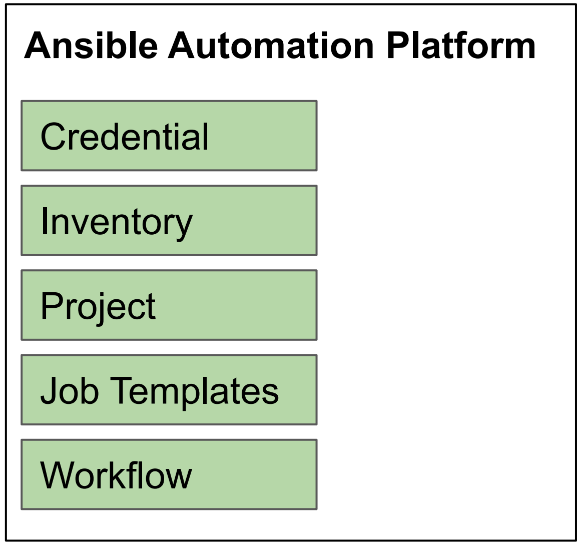


Openshift

demo-test

demo-prod

ansible-sa



Ansible Automation Platform

Credential

Inventory

Project

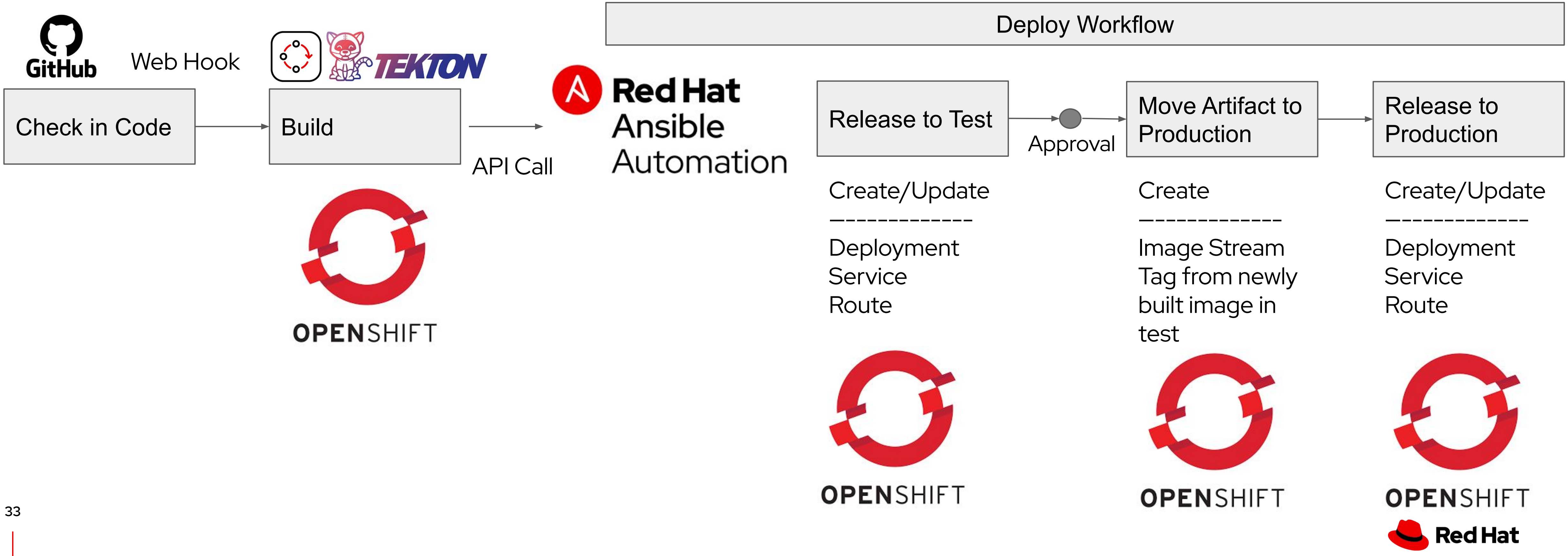
Job Templates

Workflow

https://github.com/ktatro-rh/ansible-ci-cd-demo

Red Hat Ansible Automation

1. Create Namespaces
2. Create Pipeline, custom tasks, and required secrets



Deploy Workflow

GitHub

Web Hook

TEKTON

Check in Code

Build

API Call

Red Hat Ansible Automation

OPENSHIFT

Release to Test

Approval

Move Artifact to Production

Release to Production

Create/Update

Deployment
Service
Route

Create

Image Stream
Tag from newly
built image in
test

Create/Update

Deployment
Service
Route

OPENSHIFT

OPENSHIFT

OPENSHIFT

Red Hat

Demo

Additional Resources

1. Ansible Interactive Labs
2. DevOps & CI/CD with Automation Controller Hands-on Lab
3. Ansible CI/CD Whitepaper (pdf)
4. Use-case: DevOps with Red Hat Ansible Automation Platform
5. Ansible Certified Content Collections
6. IDC: Business Value of Ansible
7. Demo GitHub Repository

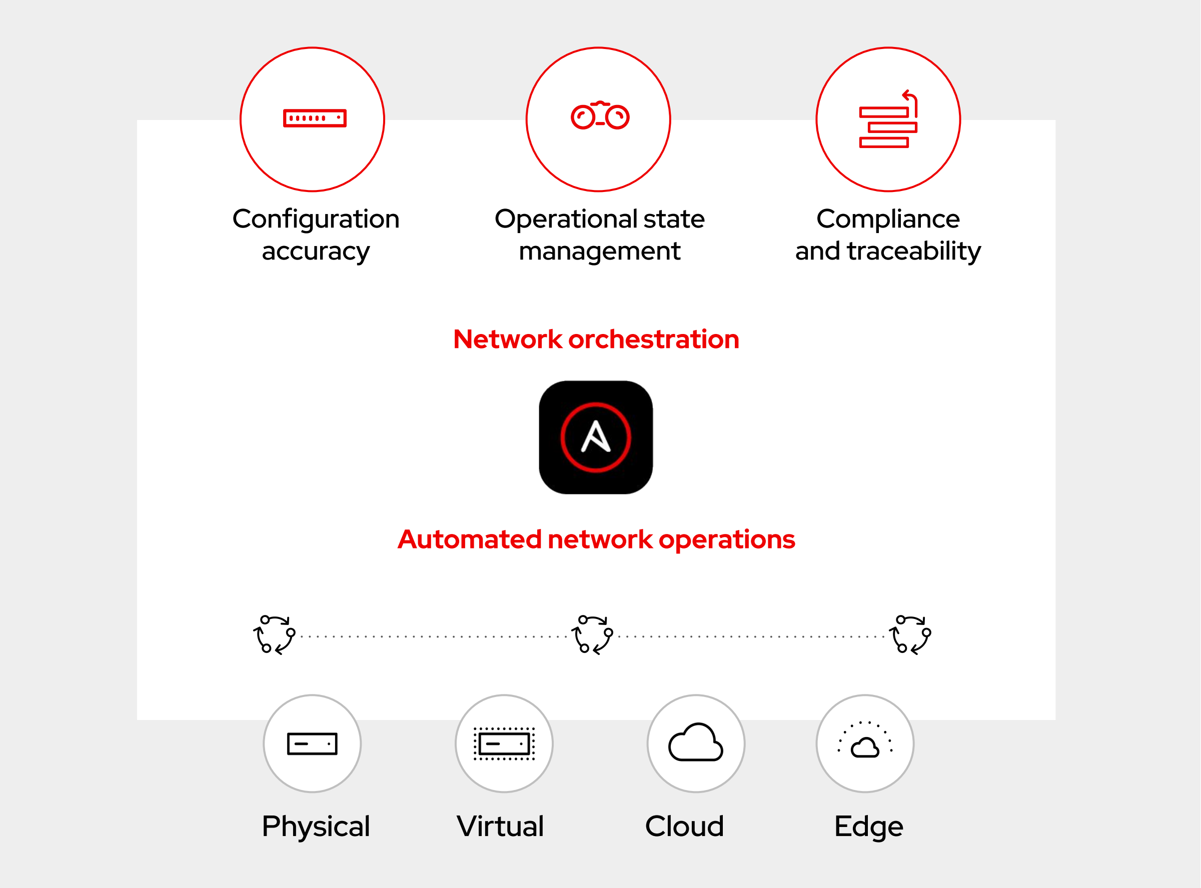
Thank you

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
- youtube.com/user/RedHatVideos
- facebook.com/redhatinc
- twitter.com/RedHat

Appendix
Common Use Cases

Ansible network automation. Next-gen network operations.


Configuration accuracy
Operational state management
Compliance and traceability
Network orchestration
Automated network operations
Physical
Virtual
Cloud
Edge

Configuration accuracy

- Config backup and restore
- Scoped configuration management

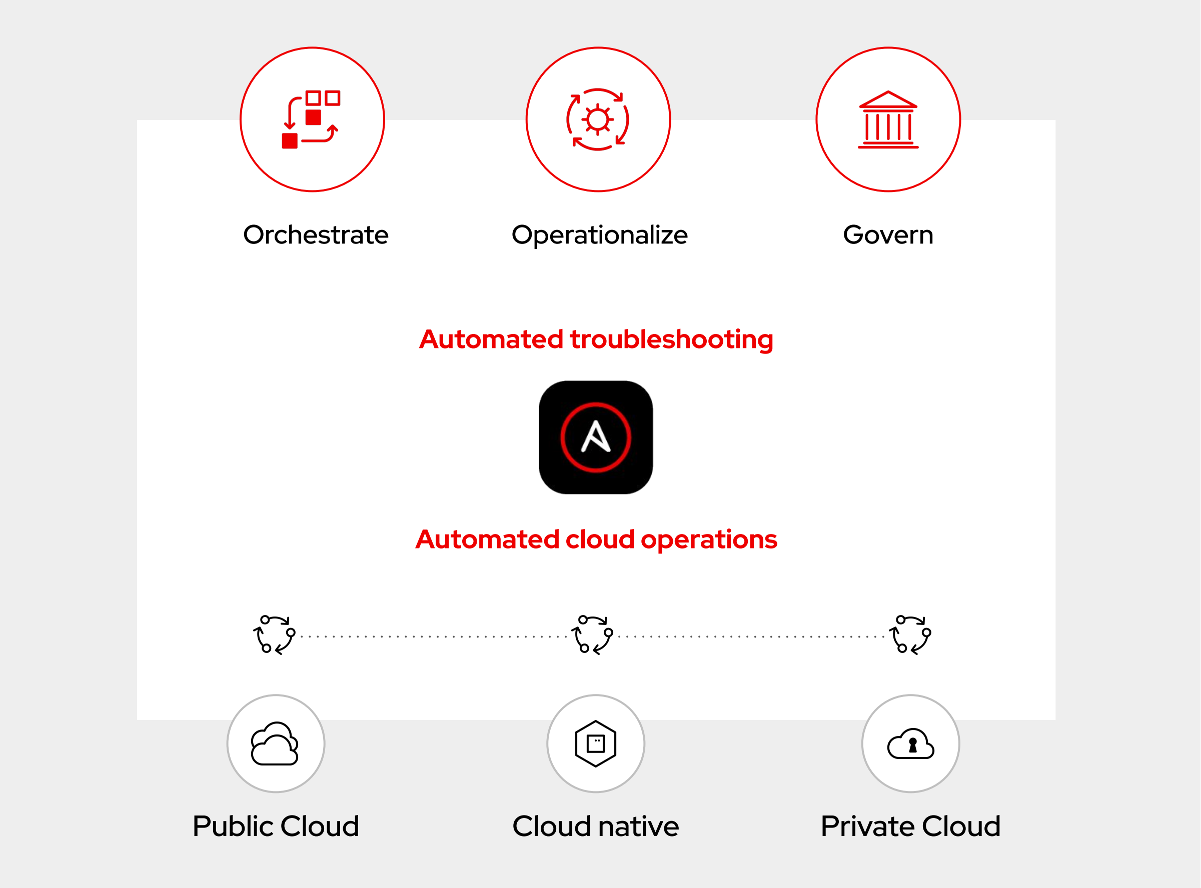
Operational state management

- Dynamic documentation
- Automated NetOps

Compliance and traceability

- Operational state validation
- Network compliance

Ansible hybrid cloud automation. Tame your clouds.



Orchestrate

- Deployment and retirement
- Infrastructure coordination
- Cloud migration

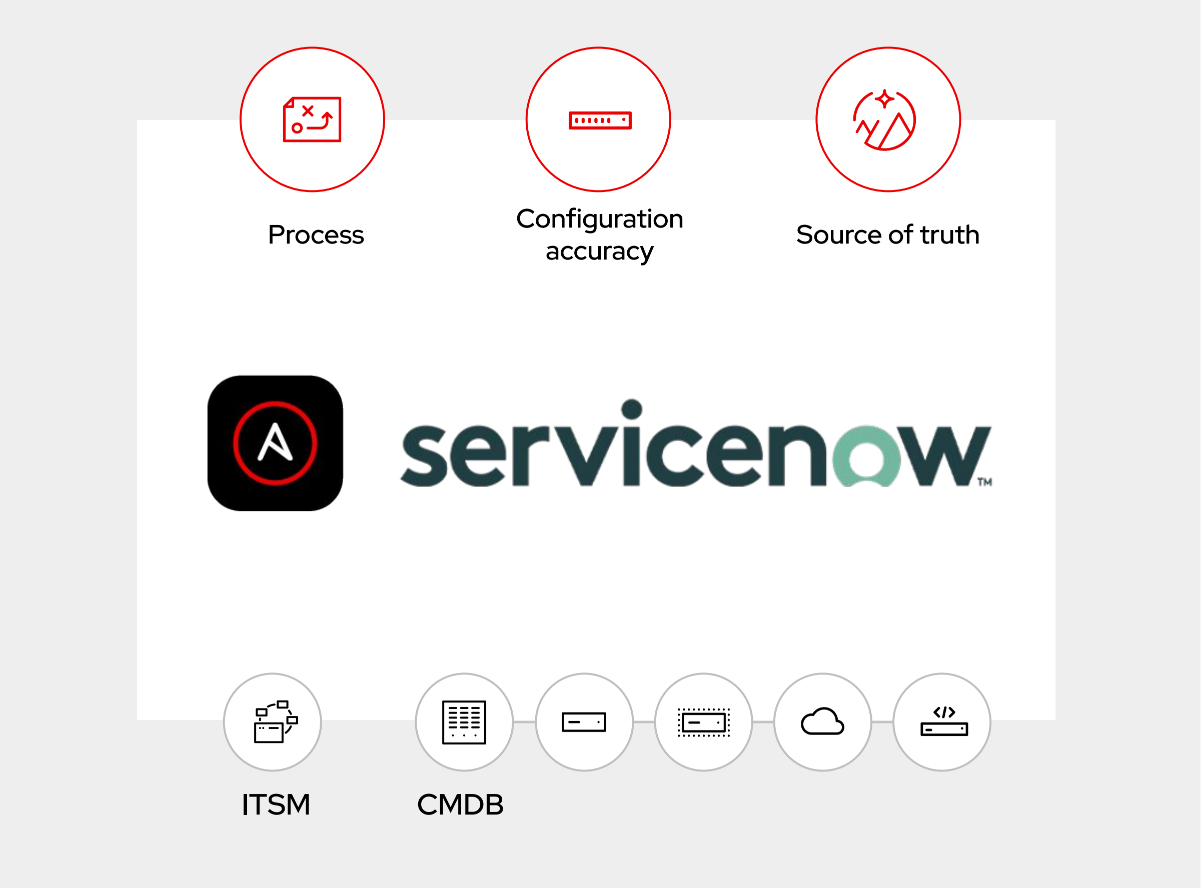
Operationalize

- Infrastructure visibility
- Cloud operations
- Automated troubleshooting

Govern

- Business continuity
- Cost management
- compliance

Ansible Automation Platform for ServiceNow solution.



Process

Configuration accuracy

Source of truth

ITSM

CMDB

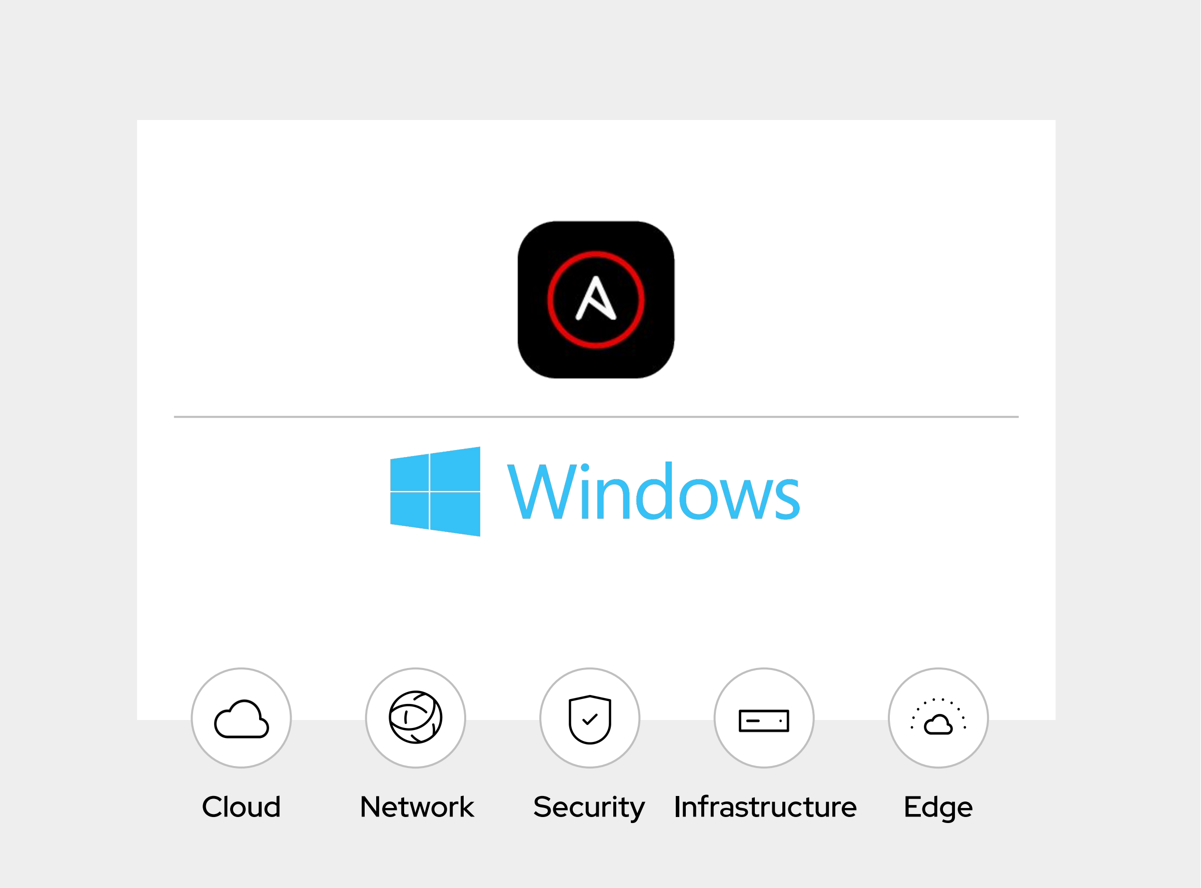
IT Service Management (ITSM)

- Create and update records
  - Incident, problem, problem task, change request
- Assign items to user accounts
- Attach files to records
- Advanced queries of record types
- Support for custom mappings (modified choice lists)

Configuration Management Database (CMDB)

- Advanced queries of configuration items
- Update configuration items after automated changes
- CMDB as inventory source for automation
- Batch modifications of configuration items

Ansible Automation Platform for Windows.



Windows

Cloud

Network

Security

Infrastructure

Edge

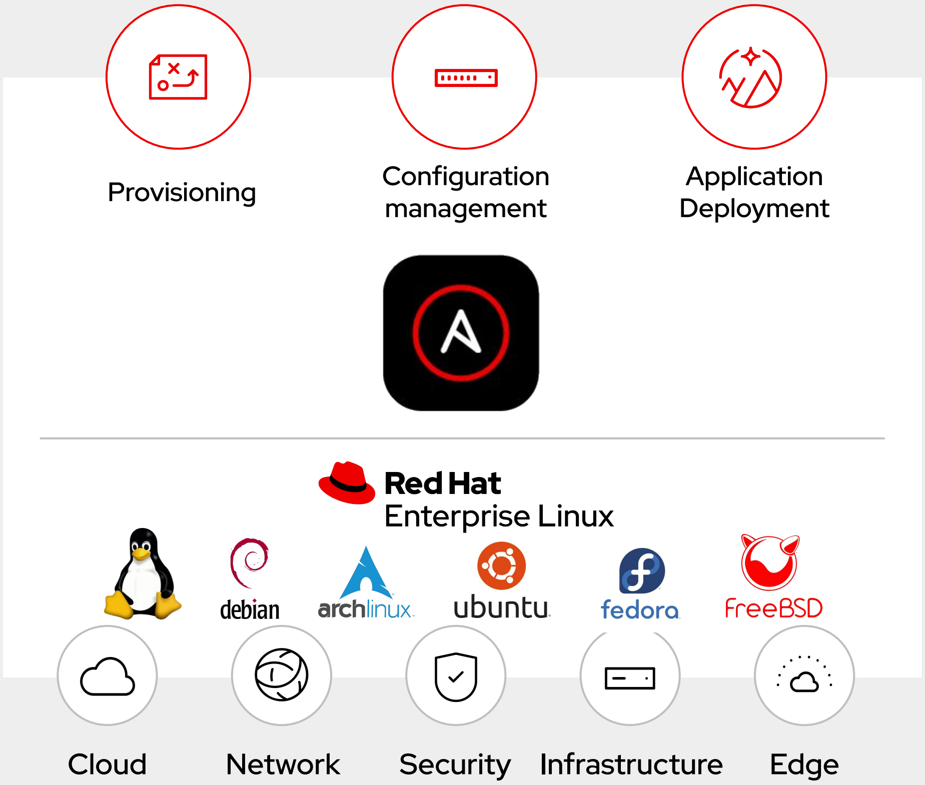
Day 1 configuration

- Install and uninstall MSIs, .exes
- Chocolatey package manager integration
- Start, stop, and manage Windows services
- Template and apply registry updates
- Flexible authentication support

Day 2 operations

- Manage and install Windows updates across reboots
- Create and manage local users
- Create and manage domain controller/member server state
- Manage certificates
- Fetch files from remote hosts
- Push and execute Powershell scripts
- Leverage Powershell DSC resources

Ansible Automation Platform for Linux.



Provisioning

- Work seamlessly with bare metal, virtualized and cloud infrastructure
- Easily patch, upgrade and maintain Linux servers
- Automation can handle reboots and ad-hoc changes

Configuration Management

- Built-in support for Linux file and user management
- Full integration of Jinja2 templating library
- Support for Idempotence across numerous Linux modules

Application Deployment

- Full support of dnf, yum and apt packaging tools
- Start, stop, and manage Linux services
- Check operational state and verify application deployments

Ansible Automation Platform for Red Hat OpenShift.

Red Hat
OpenShift 4

Cloud

Network

Security

Infrastructure

Edge

Infrastructure coordination

- Coordination of existing off-cluster IT infrastructure and services with cloud-native systems

Lifecycle management

- Automation of Red Hat OpenShift infrastructure and applications lifecycle management

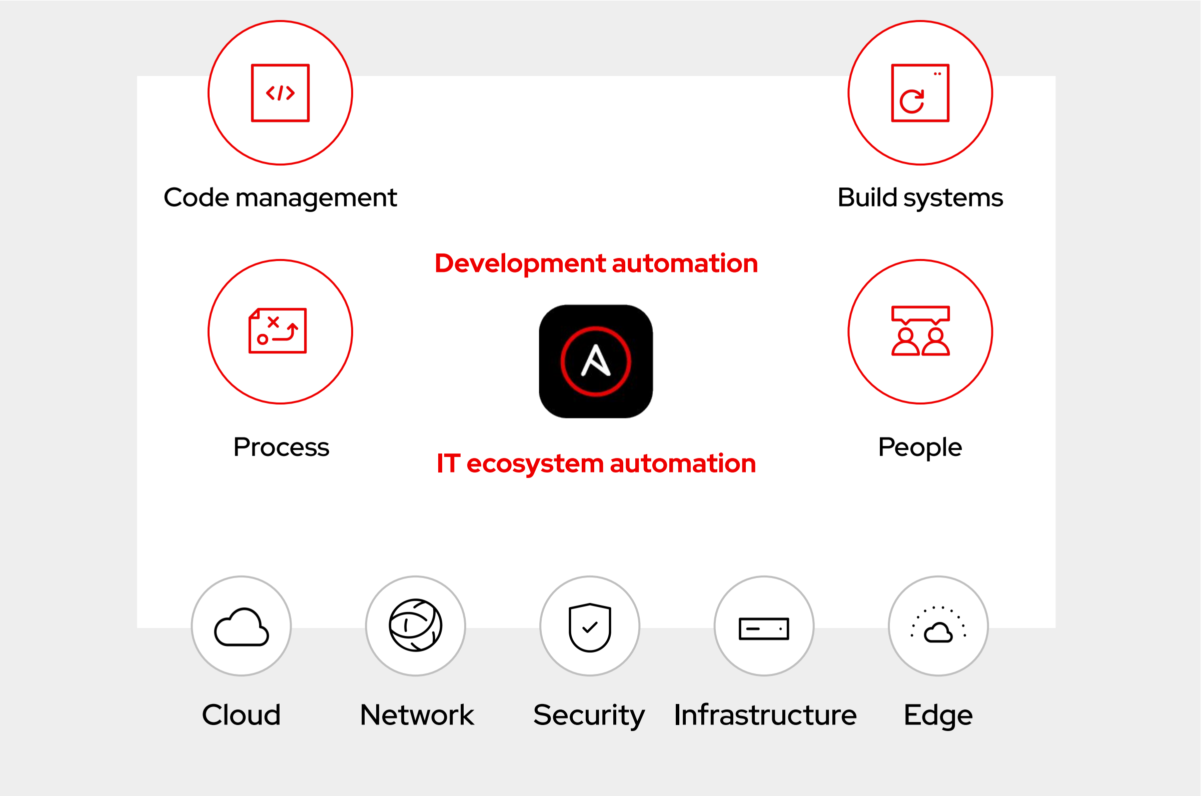
Day 2 configuration

- Repeatable “last mile” automation for Red Hat OpenShift cluster configuration and management

Business continuity

- End-to-end business continuity and disaster recovery (DR) automation of Red Hat OpenShift clusters

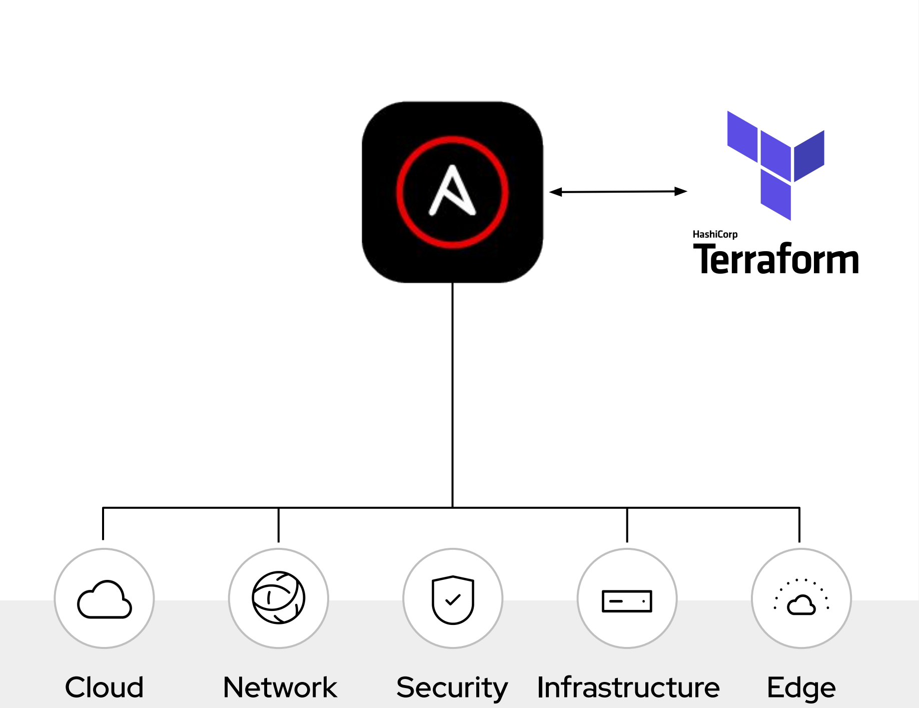
Ansible Automation Platform and DevOps.
Deliver consistently, reliably, and rapidly.



Accelerate DevOps throughout your IT ecosystem

- **Align to business goals** with outcome-based IT process automation.
- **Orchestrate and integrate** using the API and webhooks
- **Simple, pervasive automation language** used across the IT ecosystem.
- **Extend capabilities** using Ansible Content Collections.
- **Improve CI/CD security and governance** with approvals, auditing, and RBAC
- **Speed up development cycles** with consistent dev, test, and prod environments

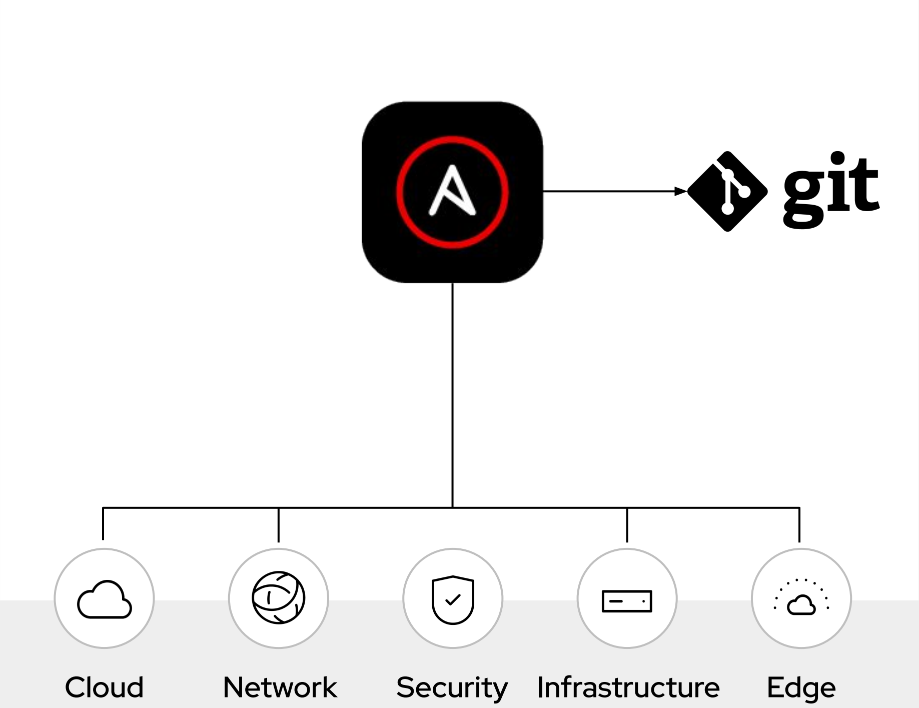
Everything as code. Everything is possible.



Infrastructure and configuration as code

- **Ansible creates** infrastructure manifests and triggers the provisioning and deprovisioning of infrastructure.
- **Dynamic inventory** management provides access to newly provisioned infrastructure without manual intervention.
- **Combine tool chains** to deliver infrastructure as code and configuration as code. Allowing for complete management of infrastructure life cycles with post-provisioning tasks.

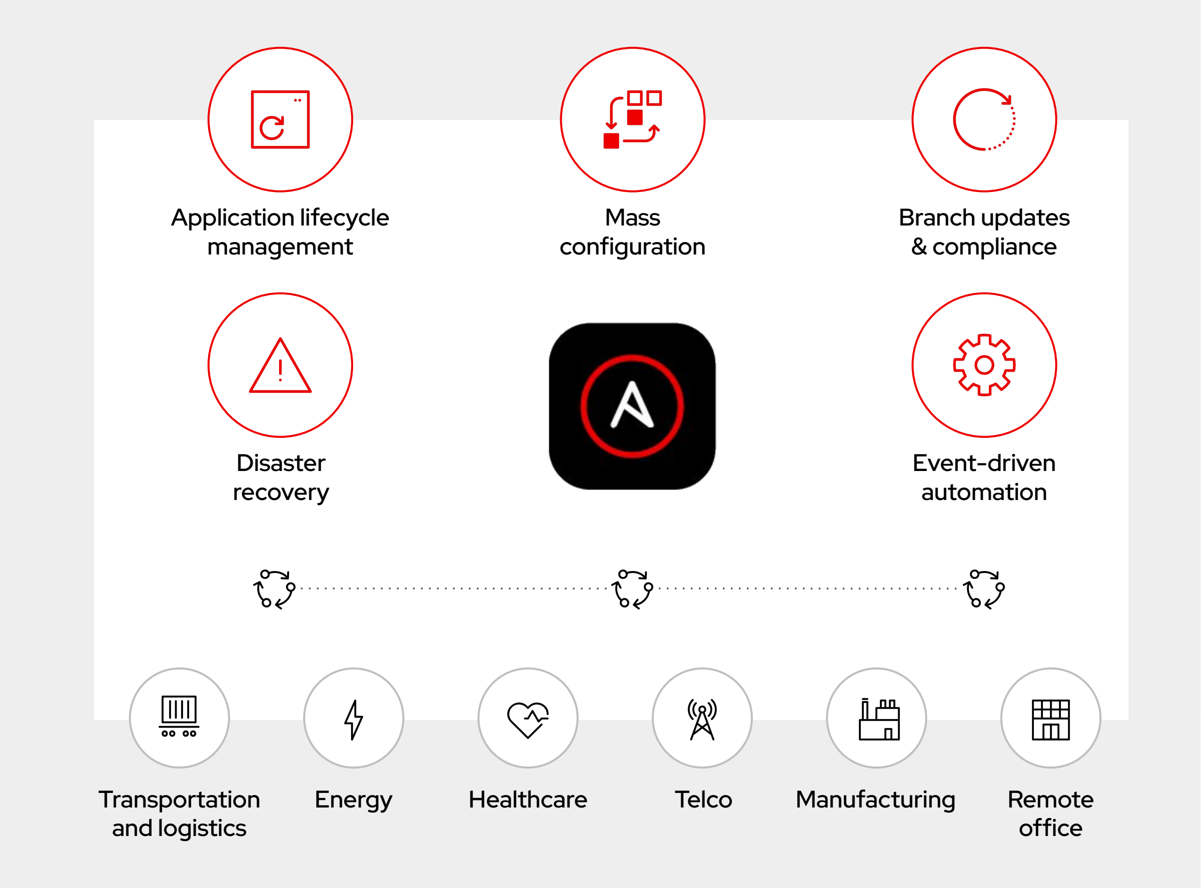
GitOps. IaC, merge requests, and CI/CD.



Ansible Automation Platform and GitOps

- **GitOps** is a framework that uses Git repositories as a single source of truth (SoT) to deliver infrastructure as code.
- **Infrastructure as Code (IaC)** is stored within a Git repositor as the SoT.
- **Merge Requests (MRs)** is the change mechanism for any updates to the infrastructure.
- **Continuous Integration/Continuous Delivery (CI/CD)** automates the infrastructure updates.
- **Ansible triggers and automates** the IaC updates via Ansible workflows while acting as a CI/CD tool and providing complete management of the entire lifecycle day-2 operations.

Automating edge enterprise use cases.



Why Ansible for Automation at the Edge?

- **Automate scale and complexity** with a consistent platform from the datacenter to the edge across heterogeneous estates
- **Facilitates IT/OT convergence**
- **Provides predictability and repeatability** to automate anything with programmatic API or Linux OS