

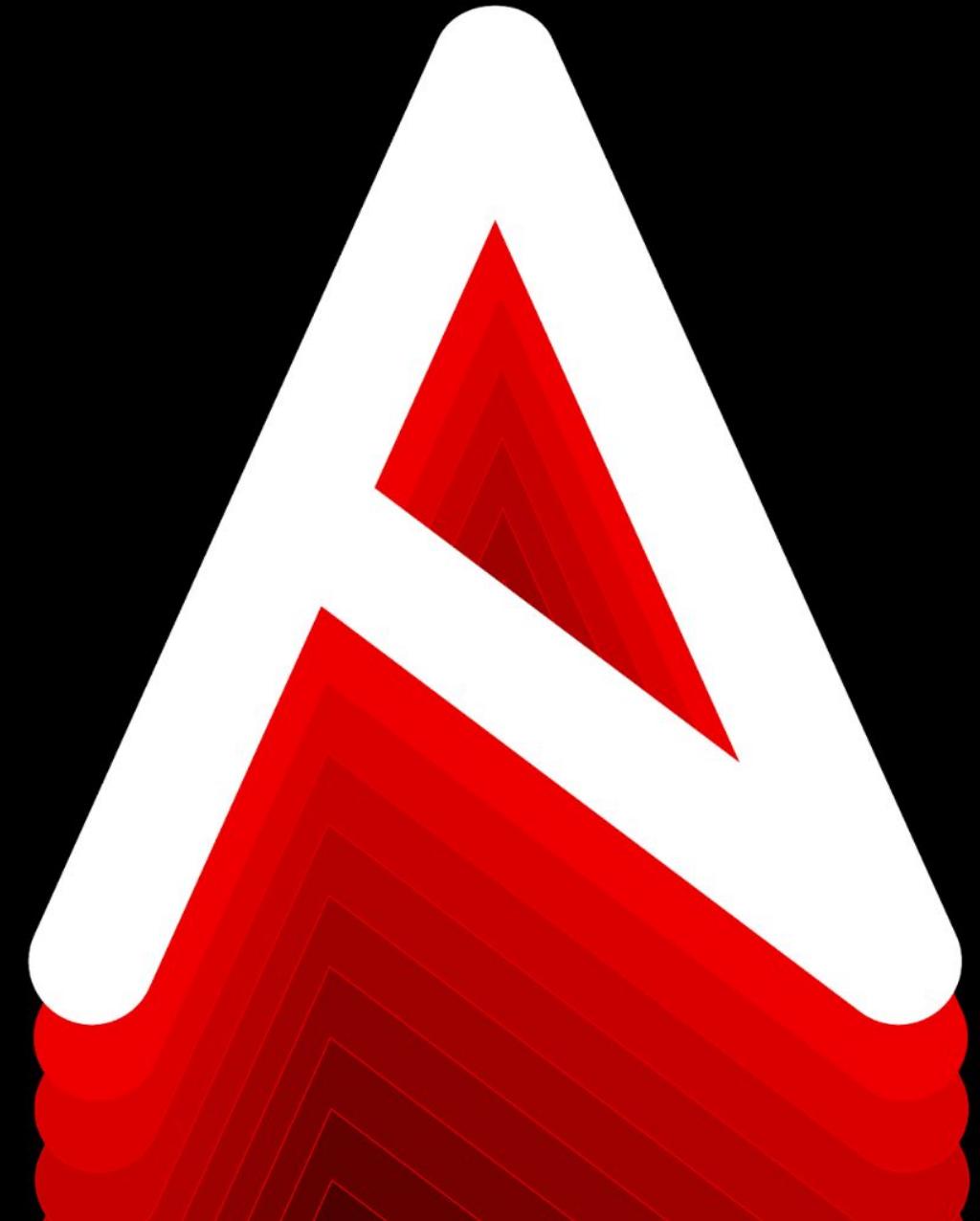


# Ansible Automation Platform

**Partner Development Workshop**

8.2. - 10.2.2022

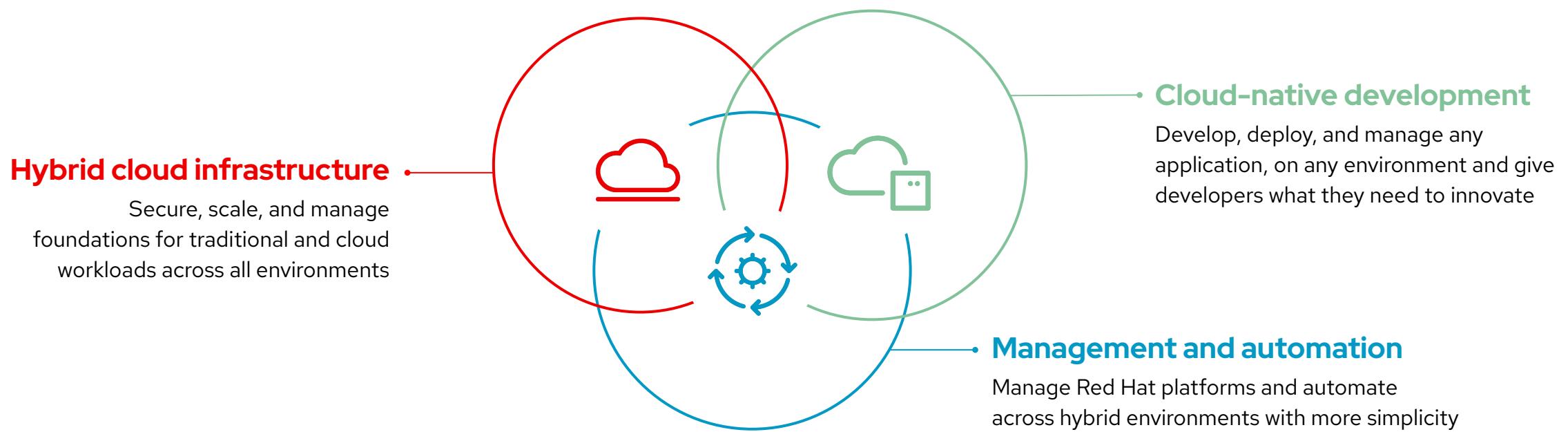
**Andreas Stolzenberger**  
Principal Solution Architect



# The three pillars of our business

## Open hybrid cloud

Red Hat's strategy and vision for its portfolio of software, tools, and services built in the open source development model and designed for future architectures that are open, secure, and agile across hybrid, multicloud



# Product development model

1

## Participate

We participate in and create community-powered upstream projects.

2

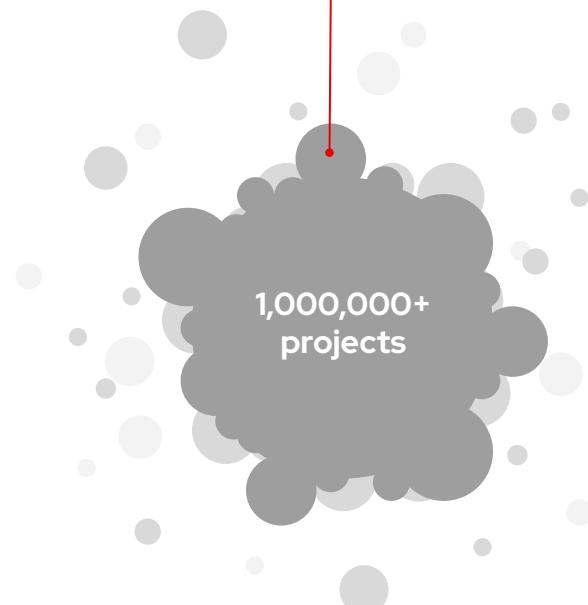
## Integrate

We integrate upstream projects, fostering open community platforms.

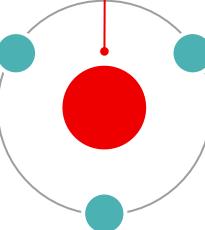
3

## Stabilize

We stabilize and commercialize these platforms, including a rich ecosystem of services and certifications.



1,000,000+ projects

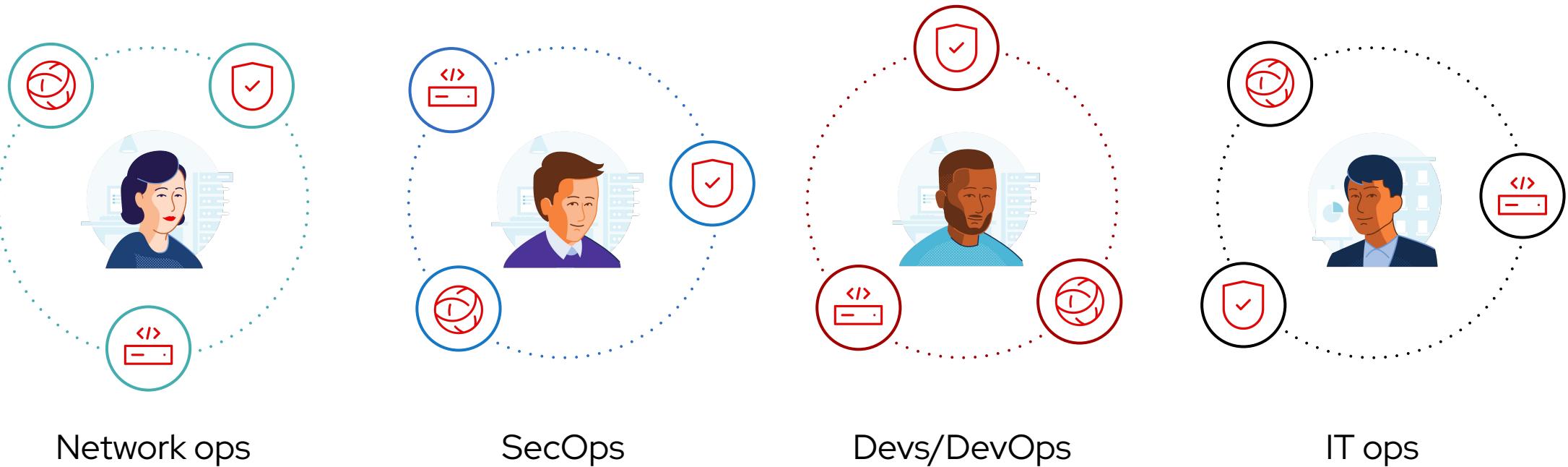


---

# Red Hat Ansible Automation Platform Intro

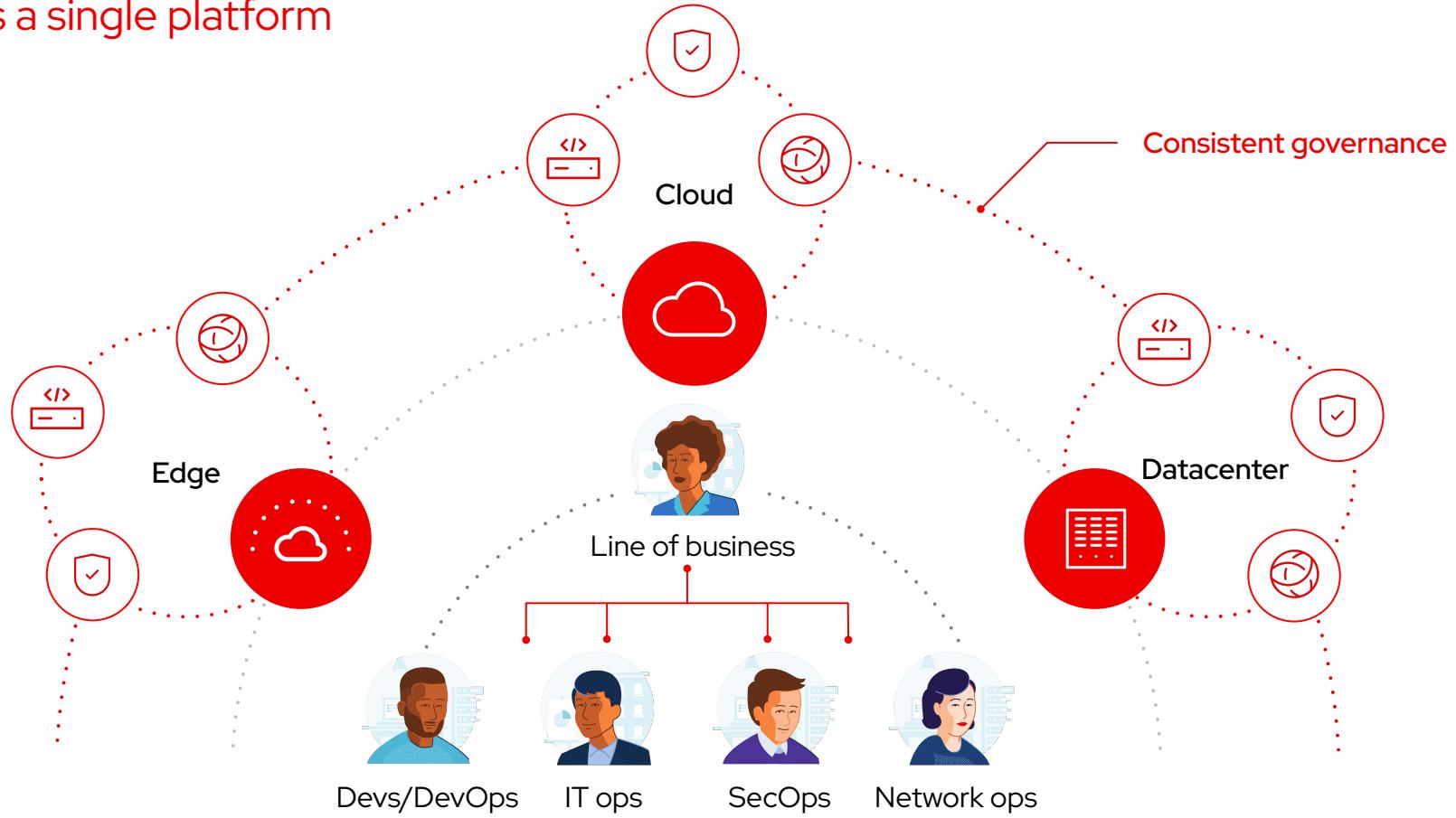
# Many organizations share the same challenge

Too many unintegrated, domain-specific tools



# Break down silos

Different teams a single platform



## Processes

**Optimize  
MTTR**



Operational  
efficiency

**Grow  
Margin**



Cost  
Reduction

**Accelerate  
TTM**



Faster  
Innovation

**Increase  
CSAT**



Enhance Customer  
Journey

**Mitigate  
Risks**



Increase  
Consistency

# Products

## Disposable Infrastructure



Accelerate operations

## CI/CD



Faster Deployments

## DevSecOps



Security from inception

## Continuous Improvement



Boost efficiency

## Reduce Cost



Increased reliability

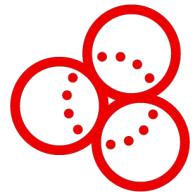
# People

## Culture of Collaboration



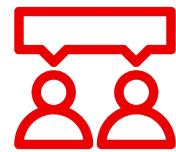
Learn from Mistakes

## Bridge Silos



Cross Organization

## Common Language



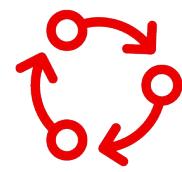
Common Tools

## Time to Innovate



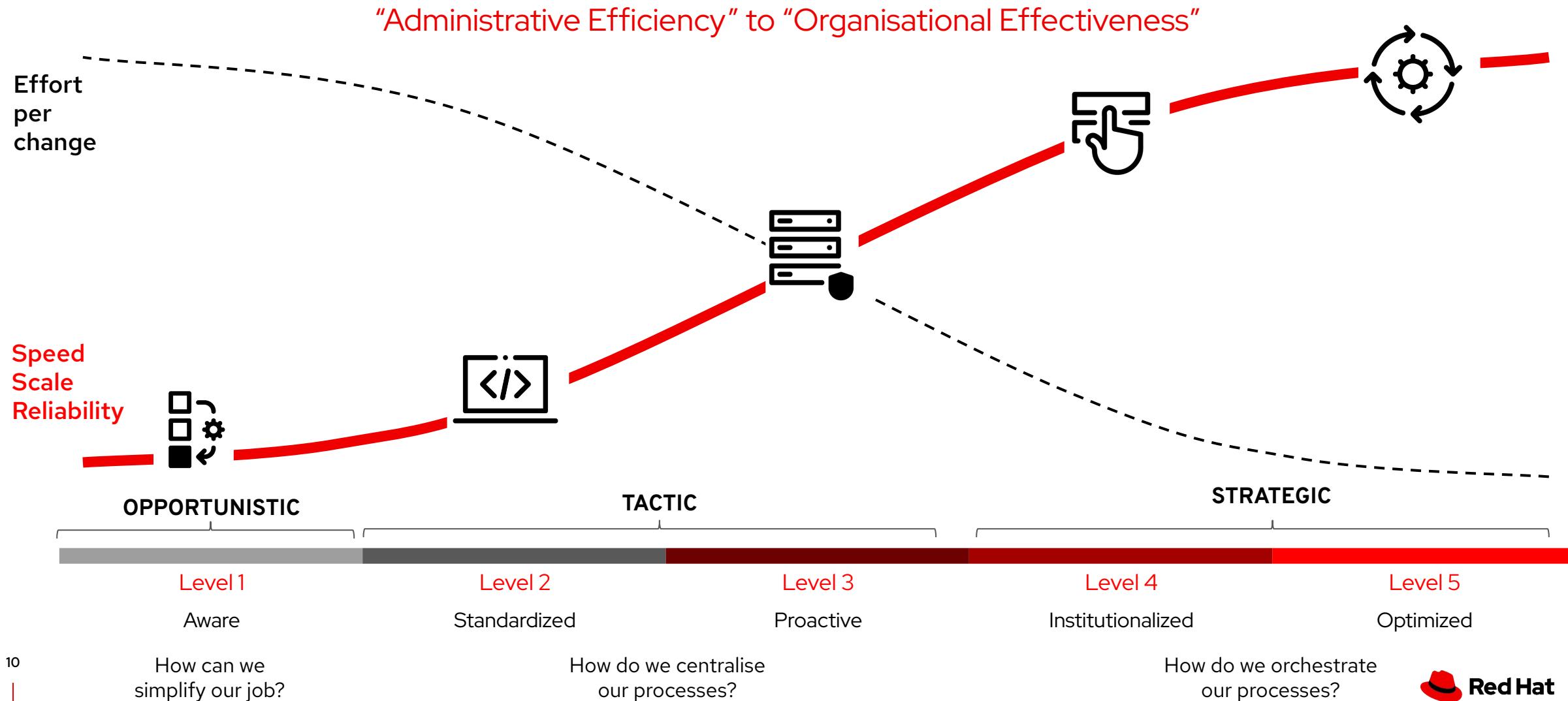
Unblock Resources

## Ongoing Journey

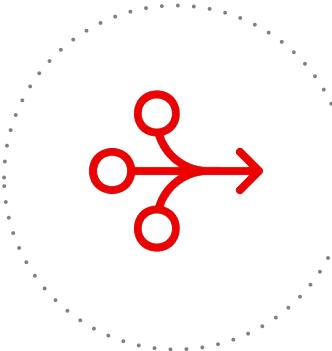


Automation is never “done”

# Where are we going?

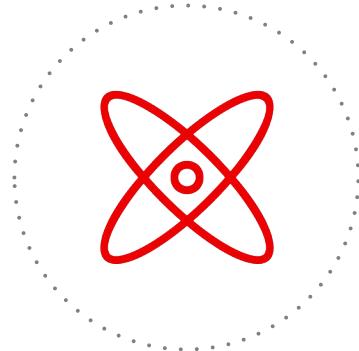


# The Ansible Automation Platform



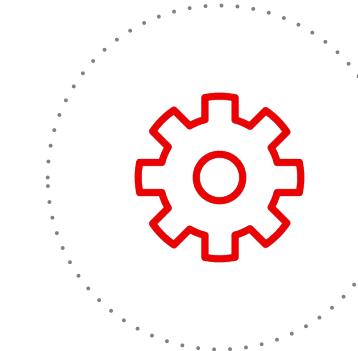
## Simple

Simplify automation creation and management across multiple domains.



## Powerful

Orchestrate complex processes at enterprise scale.



## Agentless

Easily integrate with hybrid environments.

# Automate Everything

Your entire IT footprint

Do this...

Orchestrate      Manage configurations      Deploy applications      Provision / deprovision      Deliver continuously      Secure and comply

On these...



Firewalls



Load balancers



Applications



Containers



Virtualization platforms



Servers



Clouds



Storage



Network devices



And more ...

# Ansible Automates Technology you use

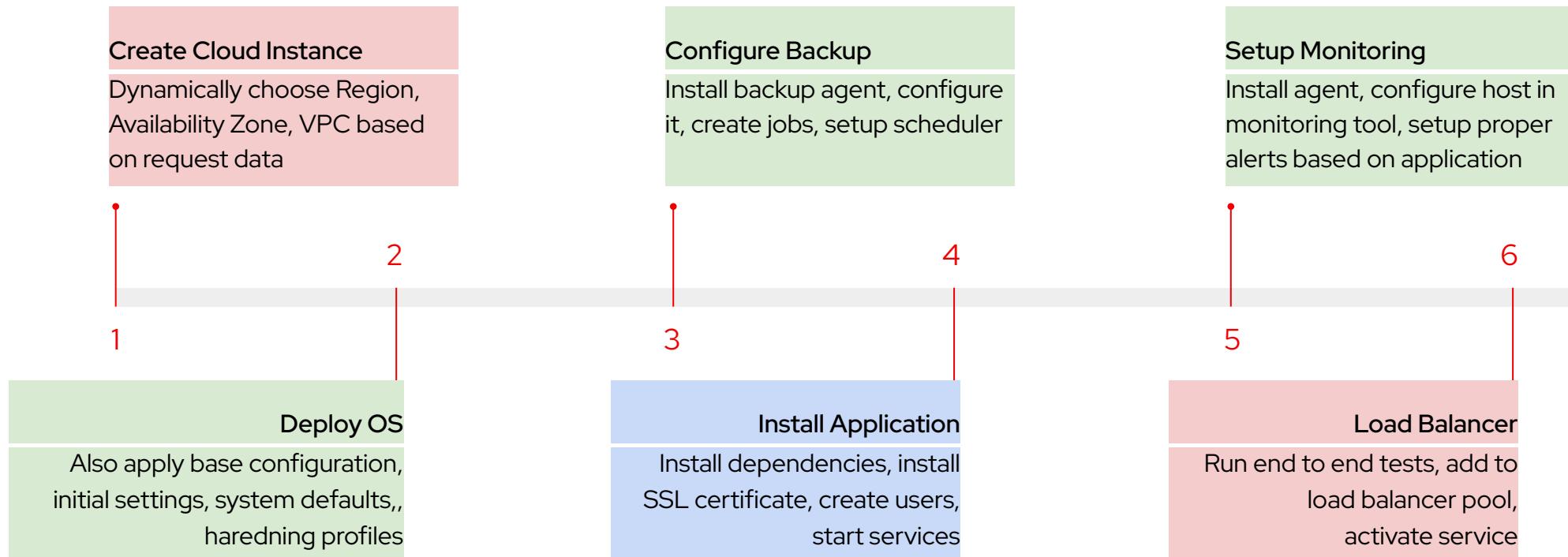
<b>Cloud</b>	<b>Virt &amp; Container</b>	<b>Windows</b>	<b>Network</b>	<b>Security</b>	<b>Monitoring</b>
AWS	Docker	ACLs	A10	Checkpoint	Dynatrace
Azure	VMware	Files	Arista	Cisco	Datadog
Digital Ocean	RHV	Packages	Aruba	CyberArk	LogicMonitor
Google	OpenStack	IIS	Cumulus	F5	New Relic
OpenStack	OpenShift	Regedits	Bigswitch	Fortinet	Sensu
Rackspace	<b>+more</b>	Shares	Cisco	Juniper	<b>+more</b>
<b>+more</b>		Services	Dell	IBM	
<b>Operating Systems</b>	<b>Storage</b>	Configs	Extreme	Palo Alto	<b>Devops</b>
		Users	F5	Snort	Jira
		Domains	Lenovo	<b>+more</b>	GitHub
		<b>+more</b>	MikroTik		Vagrant
			Juniper		Jenkins
Linux			OpenSwitch		Slack
Windows	<b>+more</b>		<b>+more</b>		<b>+more</b>
<b>+more</b>					

---

# Ansible Use Cases and Examples

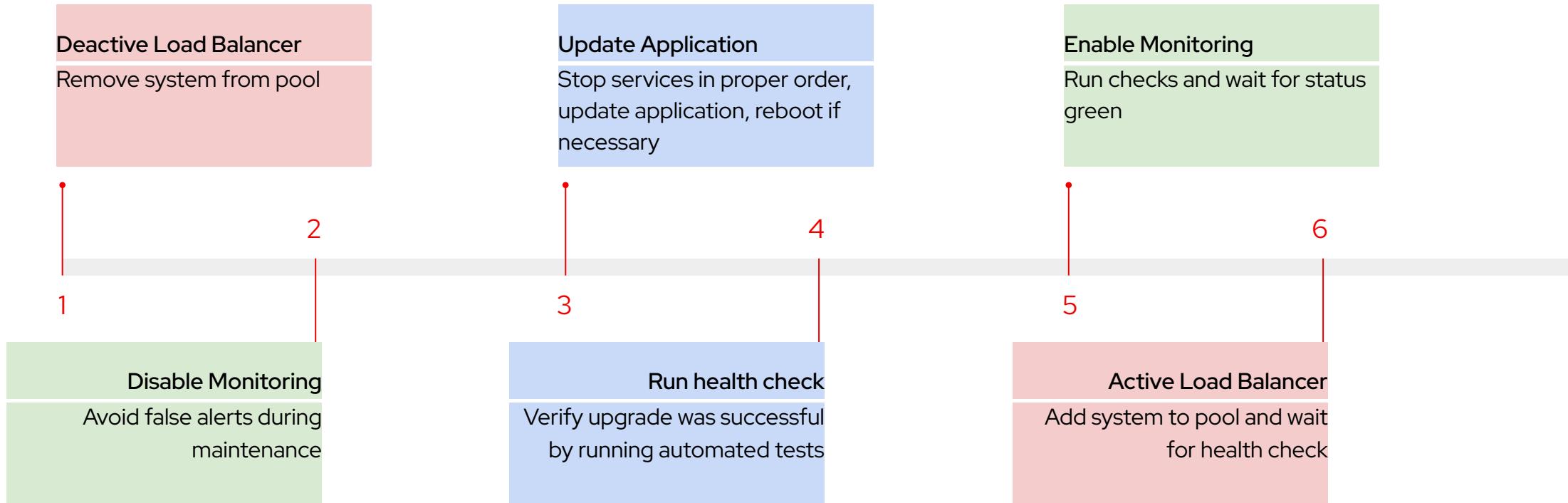
# End to End Provisioning

Full stack deployments



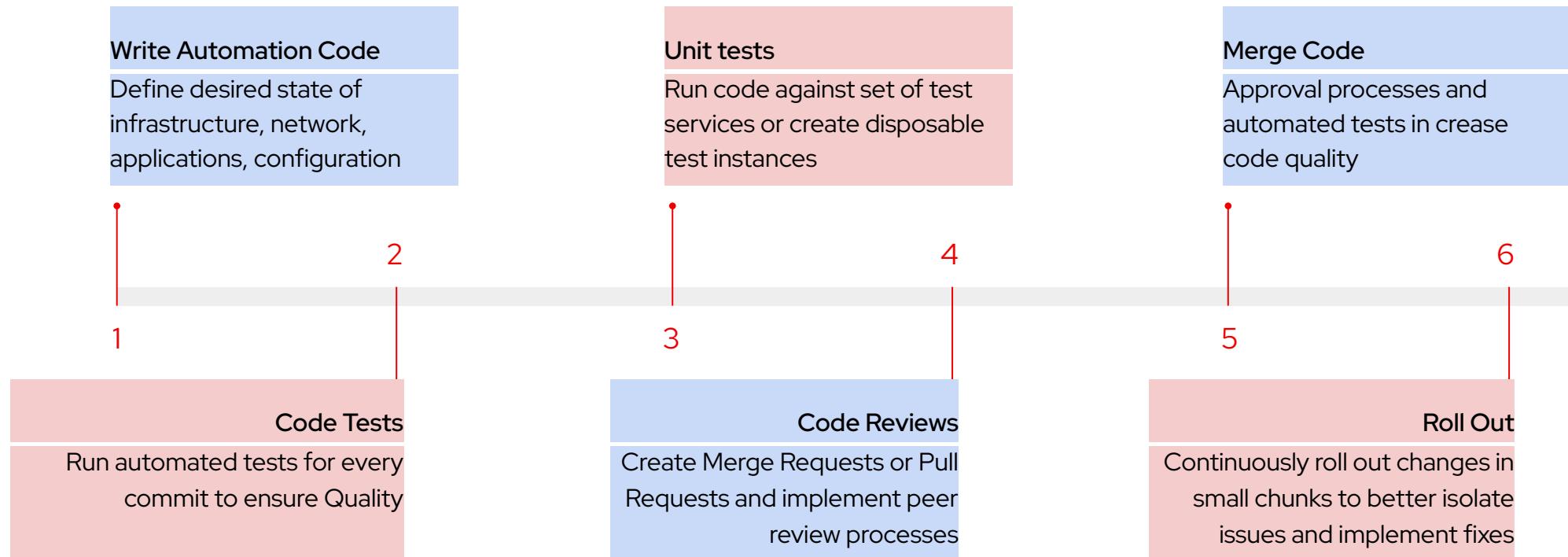
# Operation and Management

## Maintenance and Patching



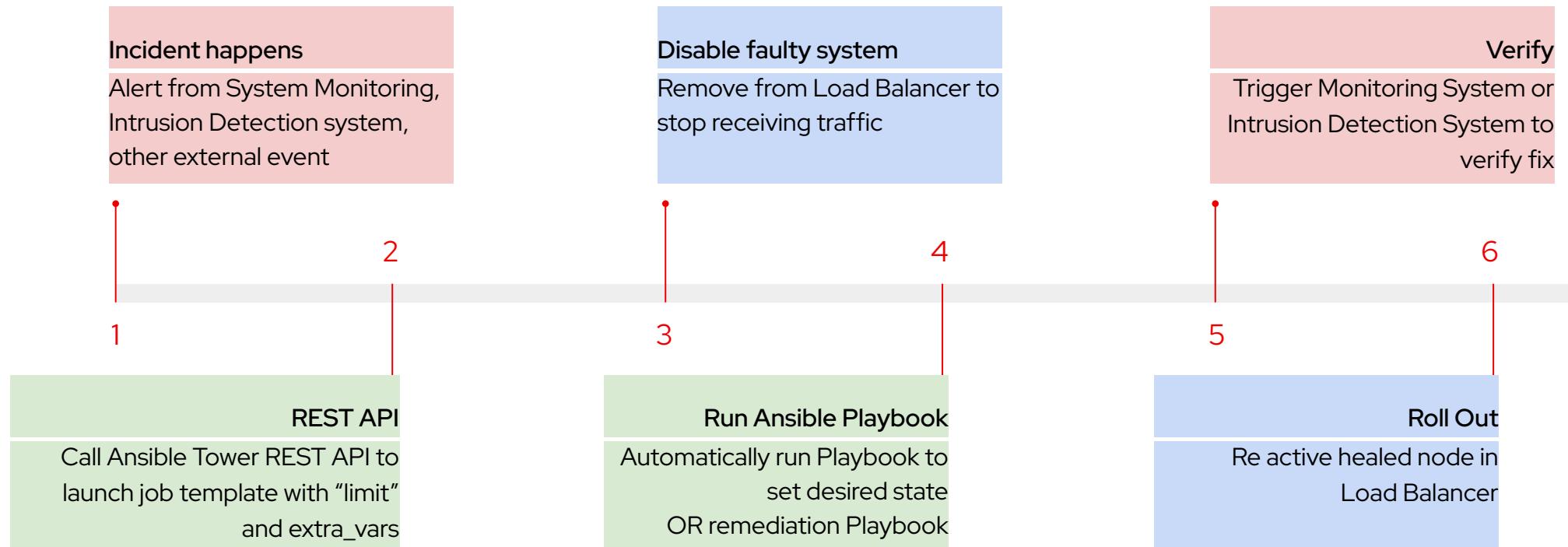
# Continuous Improvement

## Infrastructure as Code



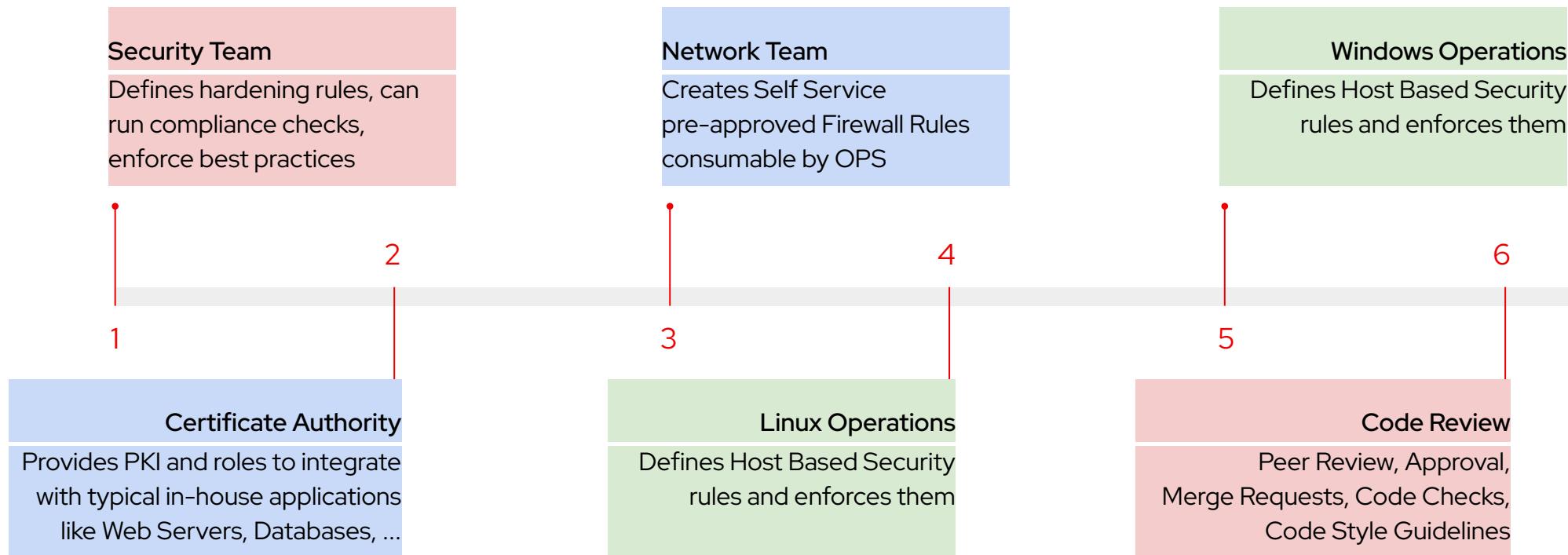
# Incident Response

## Self Healing Infrastructure



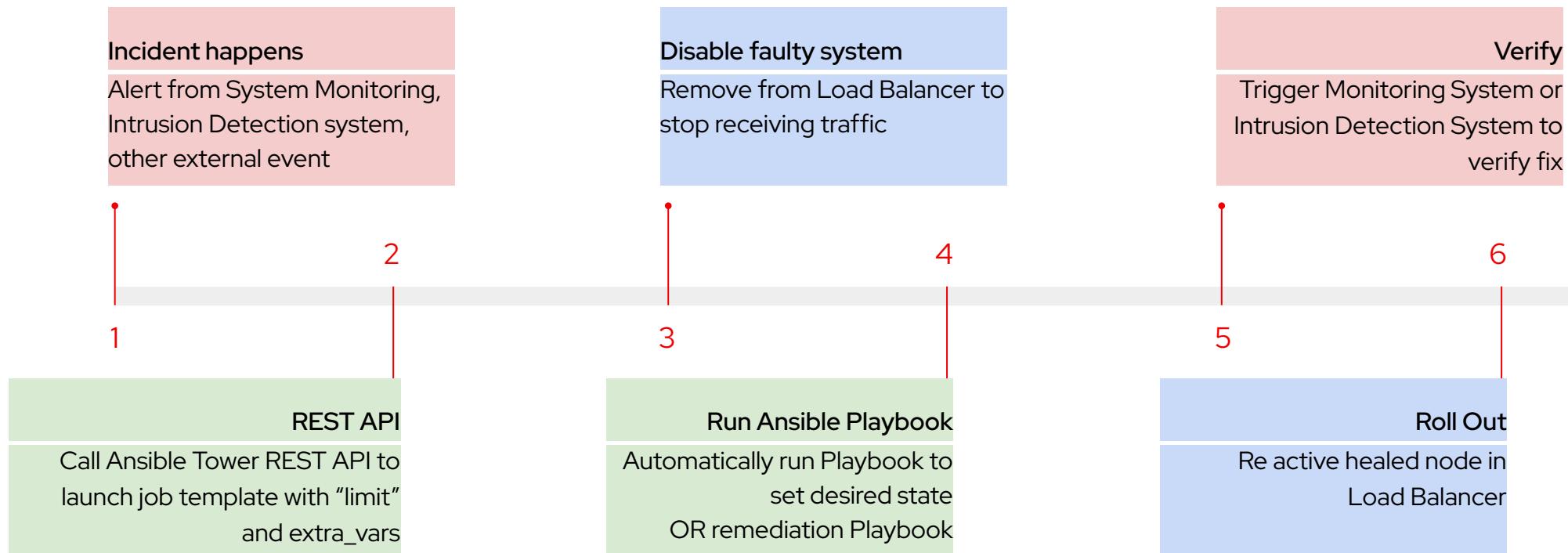
# DevSecOps

Security becomes routine



# DevSecOps

Security becomes routine





*"Adopting Red Hat Ansible Automation has not only changed how our networks are managed, but also sparked a cultural transformation within our organization."*

**Bart Dworak**

Software Engineering Manager, Infrastructure and Operations, Network



---

# **Ansible Automation Platform**

## **Value of a Subscription**

### **(or “Free-to-Fee”)**



**“Ansible is rock-solid, why should I pay for it?”**

**“Automation is not mission/business critical”**

Why Pay?

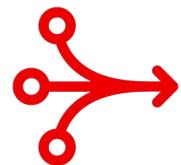
# Enterprise Automation is Business Critical

Don't think so? Think Again



## Business Advantages

- Speed and agility
- Reduce costs
- Quickly adapt to business demands
- Respond to complex threats

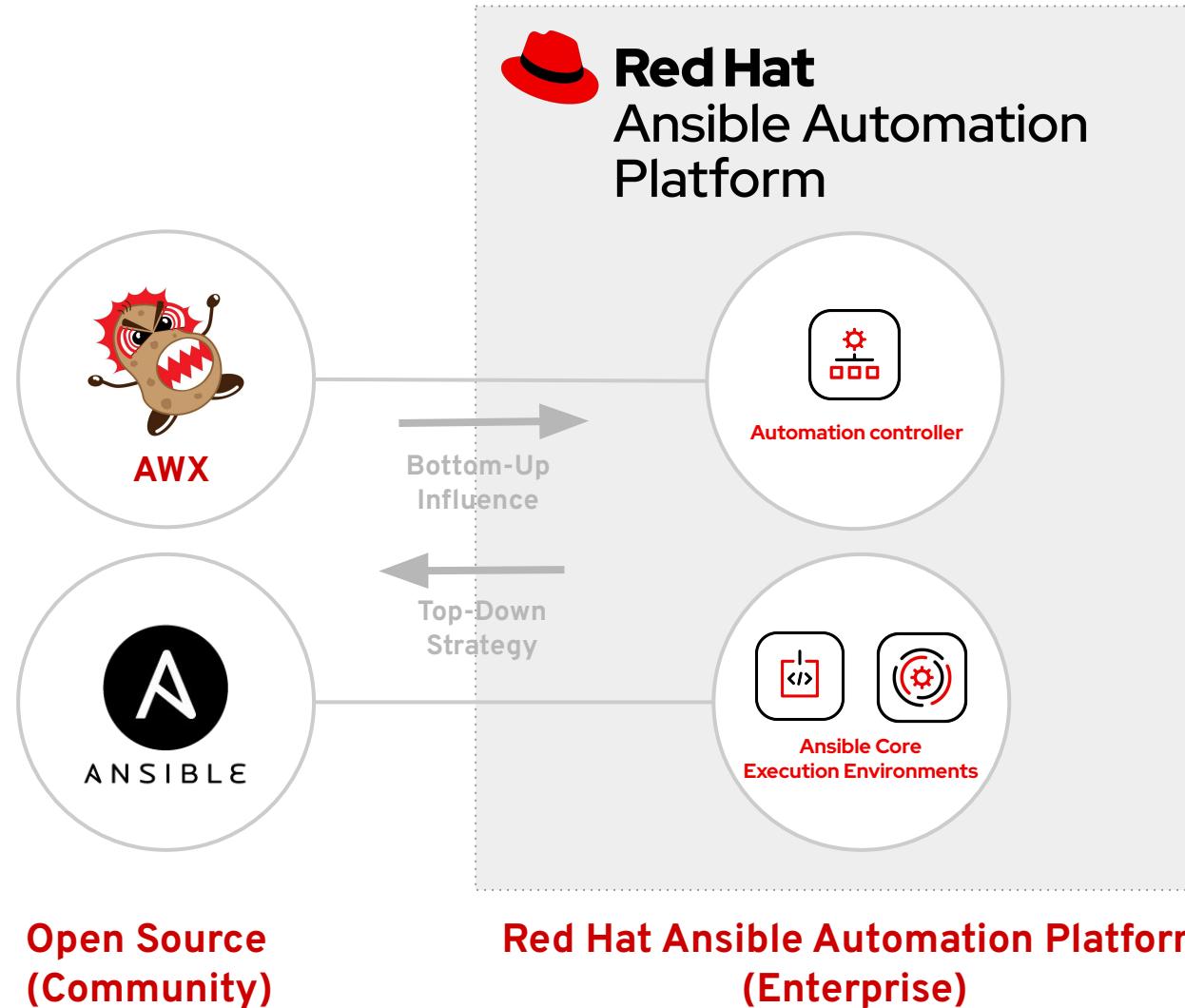


## Enterprise Ready

- Hardening & backporting for production environments
- Supported Upgrade Paths
- Defined life cycle and support SLAs
- Technical support & Open Source Assurance
- Certified Ansible Content

Why Pay?

# From Project to Product



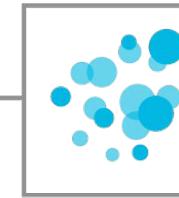
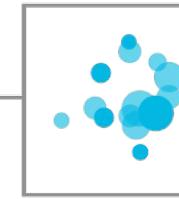
Open source

## ASSEMBLE

galaxy-ng



ansible-builder  
ansible-runner  
ansible/awx-ee  
awx-operator  
container registry  
...



1,000,000+  
PROJECTS



## BUY



**Red Hat**  
Ansible Automation  
Platform

### Participate

We participate in and create community-powered upstream projects.

### Integrate

We integrate upstream projects, fostering open community platforms.

### Stabilise

We commercialise these platforms together with a rich ecosystem of services and certifications.

Why Pay?



# Red Hat Ansible Automation Platform

vs

## free/community offerings

	RED HAT ANSIBLE AUTOMATION PLATFORM	COMMUNITY PROJECTS
STABILITY & SECURITY	Enterprise quality for production Defined product lifecycle	Constant development Frequent changes No full Quality Engineering
SUPPORT & SERVICE	Support SLAs Services Training	Self/community support only No training, certification, or consulting services
INTEGRATIONS & PARTNERS	Large Red Hat partner ecosystem Partner integrations Certified partners and content	No partner ecosystem Just the community No business partner focus or processes
SCALABILITY & UPGRADES	Automation controller clustering Automation Mesh Supported and maintained upgrade path Clear product life cycles	AWX clustering limited No isolated instances/zones No upgrade path, migration only Rolling release model
DISTRIBUTION METHODS	Installation on-prem or cloud Fully supported on RHEL Fully supported on Openshift	AWX only as containerized application No official production grade packages or containers



## Red Hat Ansible Automation Platform



### Ansible Content Collections

A new method of building and consuming Ansible content



### Automation services catalog

A venue for developers and business users to manage, provision, and retire automation resources

# One subscription. One platform.



### Automation Hub

Private content repository for sharing automation



### Automation Analytics

Analytical insights into the automation runtime

# Subscription Rules

- **Ansible Automation Platform is subscribed per managed node**
  - E.g. Virtual Machines, Containers
  - Endpoints that sit behind API
  - Device Management Systems
- **Edge cases can be a bit more complicated.** Examples:
  - If Ansible talks to OpenShift, only OCP nodes need to get a sub
  - If Ansible connects “into” containers, it’s an endpoint/node that needs a sub
  - Ansible talking to another management system (e.g. Cisco ACI) sitting in front of the actual managed nodes

## Red Hat Ansible Automation Platform customers

Healthcare	Financial services	Oil and gas	Tech	Government	Other
    	     	 	       	 	    

---

# Red Hat Ansible Automation Platform **How does it work**



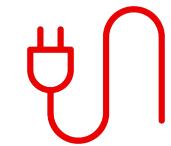
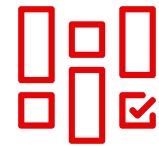
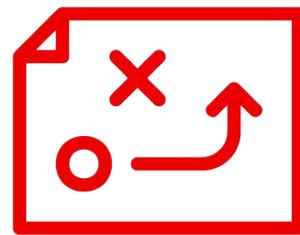
```
---
- name: install and start apache
  hosts: web
  become: yes

  tasks:
    - name: httpd package is present
      yum:
        name: httpd
        state: latest

    - name: latest index.html file is present
      copy:
        src: files/index.html
        dest: /var/www/html/

    - name: httpd is started
      service:
        name: httpd
        state: started
```

# What makes up an Ansible playbook?



Plays

Modules

Plugins

# Ansible plays

What am I automating?



## What are they?

Top level specification for a group of tasks.  
Will tell that play which hosts it will execute on  
and control behavior such as fact gathering or  
privilege level.



## Building blocks for playbooks

Multiple plays can exist within an Ansible  
playbook that execute on different hosts.



# Ansible inventories

Where am I automating?



## What are they?

Inventories are used to list managed hosts, organize them in groups and define variables.  
Hosts can be in multiple or even nested groups.



## Static and dynamic inventories

Static inventories are a great way to get started, in larger and more volatile environments, dynamic inventories can query cloud provider, virtualization platform, CDMDB's and more.



# Ansible modules

The “tools in the toolkit”



## What are they?

Parametrized components with internal logic,  
representing a single step to be done.  
The modules “do” things in Ansible.



## Language

Usually Python, or Powershell for Windows  
setups. But can be of any language.



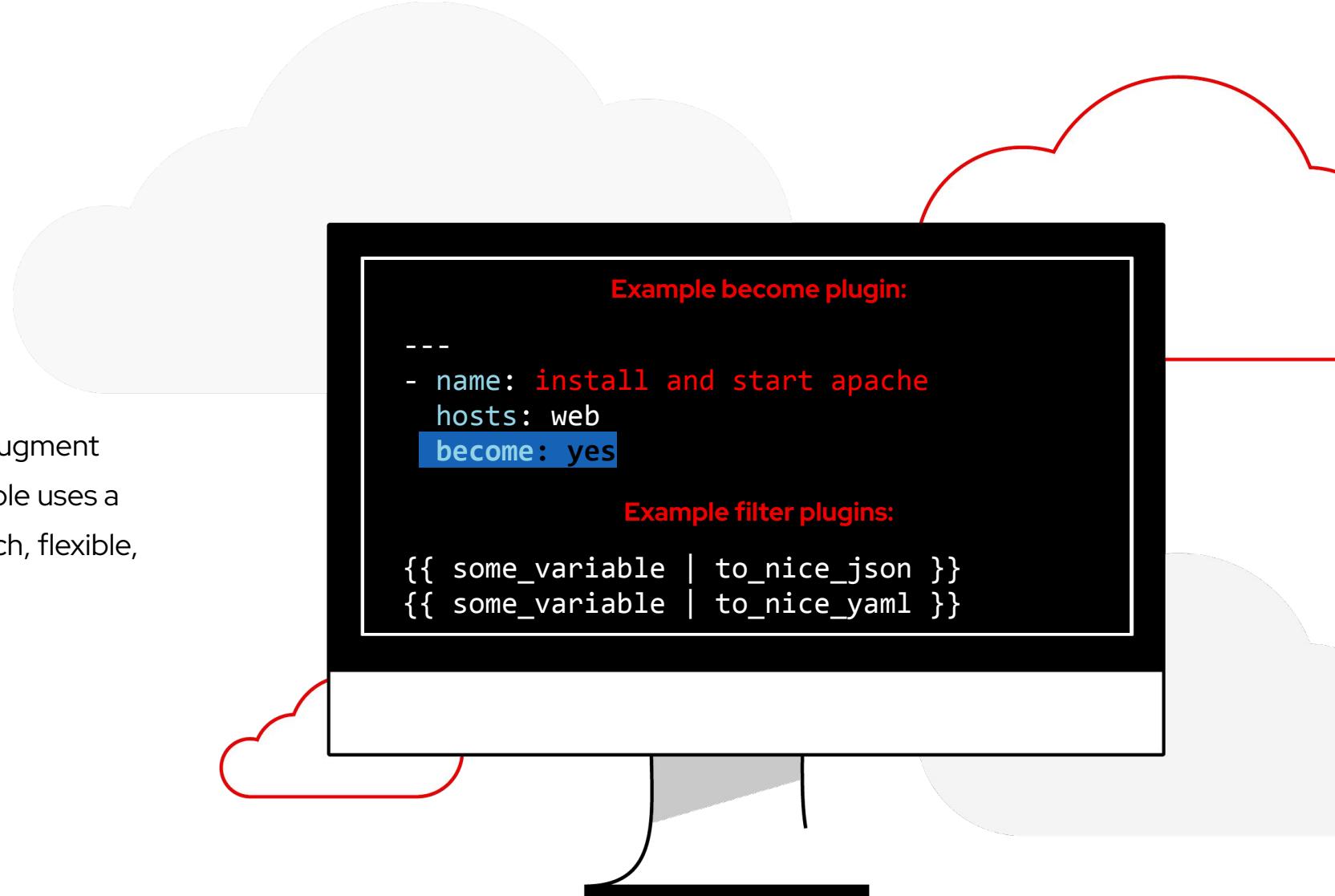
# Ansible plugins

The “extra bits”



## What are they?

Plugins are pieces of code that augment Ansible's core functionality. Ansible uses a plugin architecture to enable a rich, flexible, and expandable feature set.



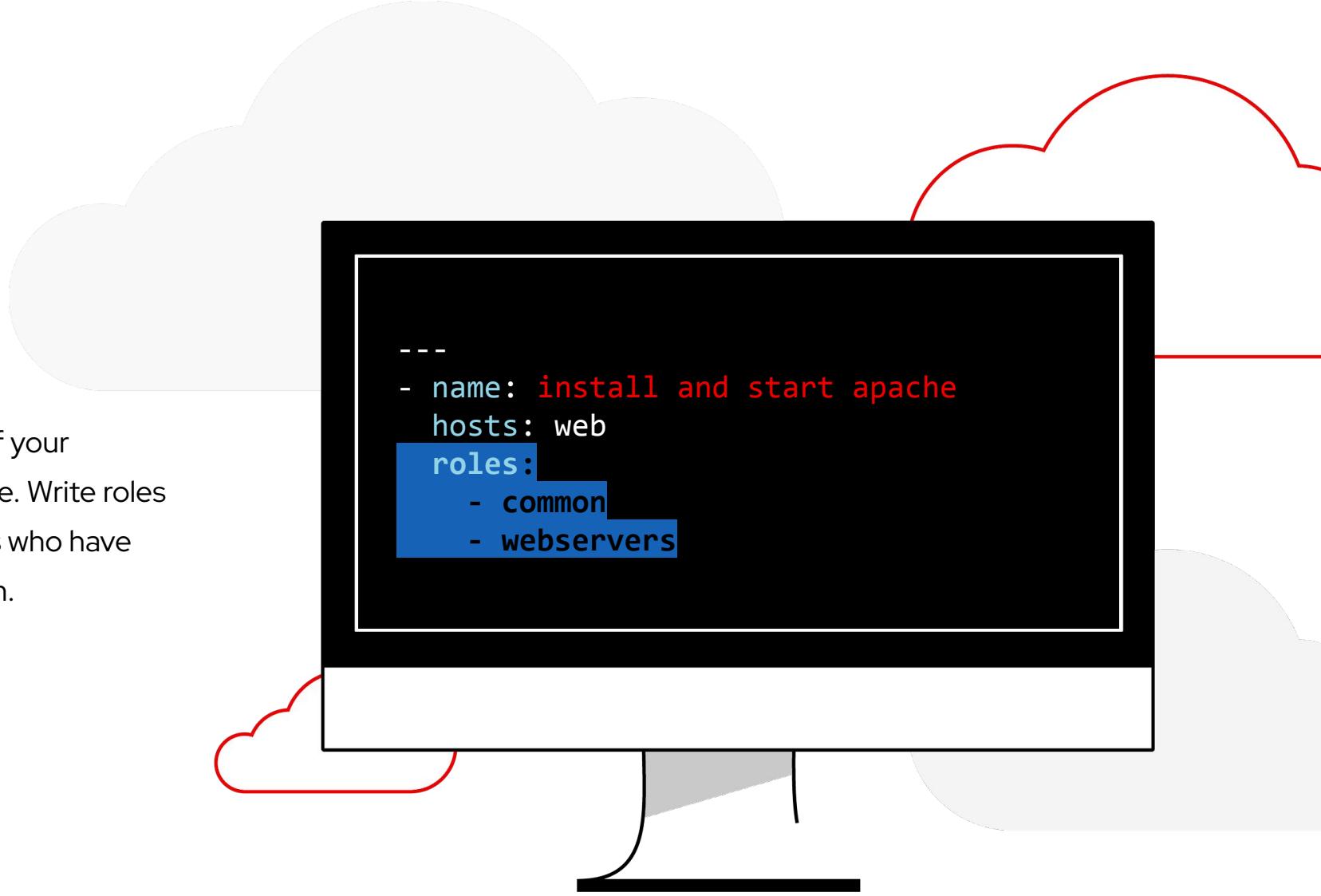
# Ansible roles

Reusable automation actions



## What are they?

Group your tasks and variables of your automation in a reusable structure. Write roles once, and share them with others who have similar challenges in front of them.



# Collections

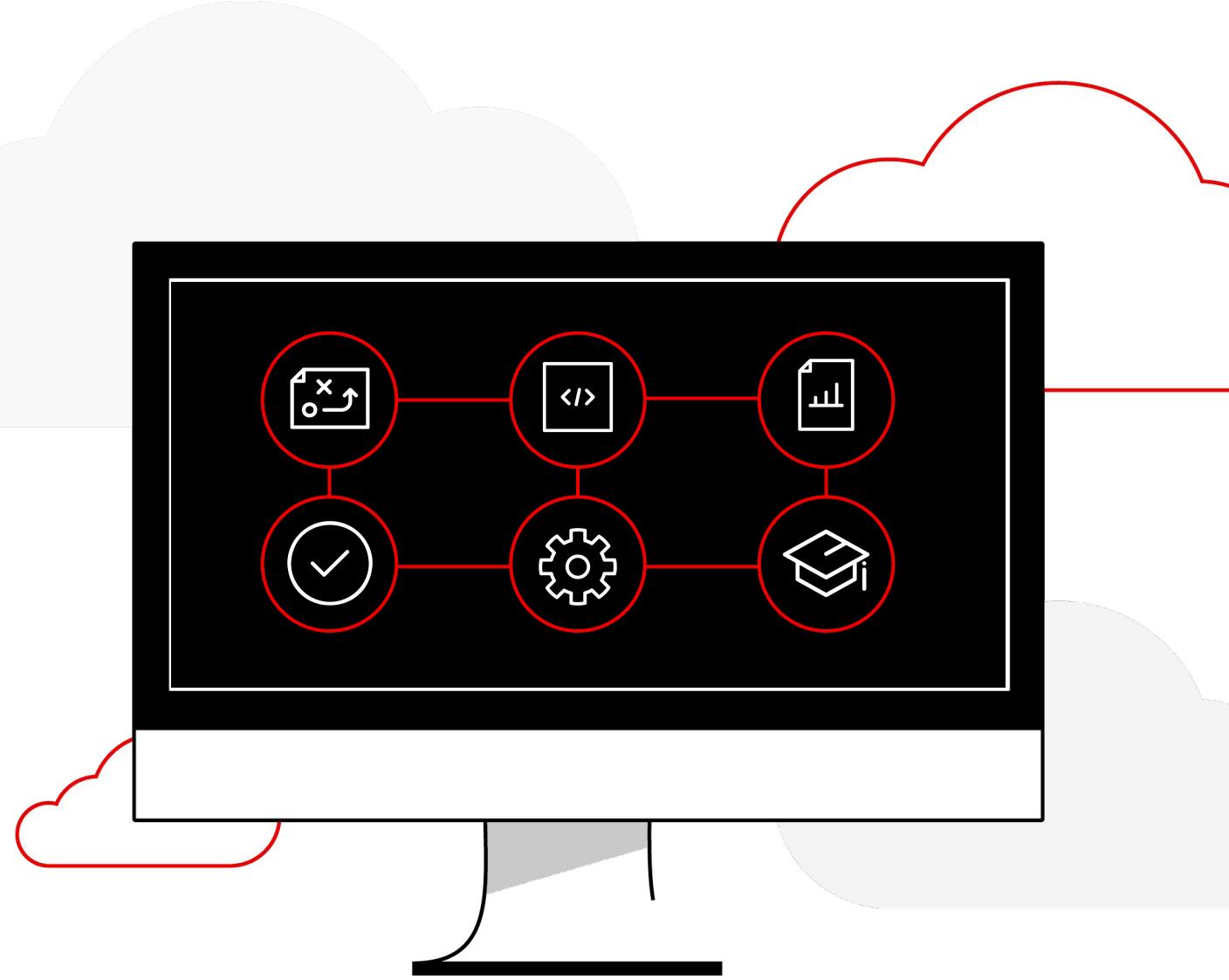
Simplified and consistent content delivery



## What are they?

Collections are a data structure containing automation content:

- ▶ Modules
- ▶ Playbooks
- ▶ Roles
- ▶ Plugins
- ▶ Docs
- ▶ Tests



---

# Red Hat Ansible Automation Platform 2



**Red Hat**  
Ansible Automation  
Platform

## **Automate everything.**

Develop, operate and consume automation at scale

## **Automate everywhere.**

Flexible, portable automation for the hybrid cloud.

## **Automate for everyone.**

Deliver expansive automation drives results

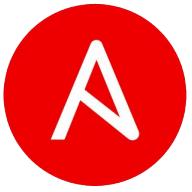
# Changes in Version 2

**Ansible Tower** and **Ansible Engine** are **no more**.

The product that we sell, distribute, and support is  
**Red Hat Ansible Automation Platform.**



## Major component changes



AAP 1.2



AAP 2.1

### Create

- ▷ Private Automation Hub 4.3
- ▷ Ansible Content Collections\*
- ▷ ansible-playbook



- ▷ **Private Automation Hub 4.4**
- ▷ **Ansible Content Collections**
- ▷ **ansible-navigator**
- ▷ **VSCode developer tooling**

### Operate

- ▷ Ansible Tower 3
- ▷ Ansible Engine 2.9



- ▷ **Automation controller 4**
- ▷ **Automation execution environments**

### Scale

- ▷ Isolated Nodes



- ▷ **Automation Mesh**

\*Optional with AAP 1.2

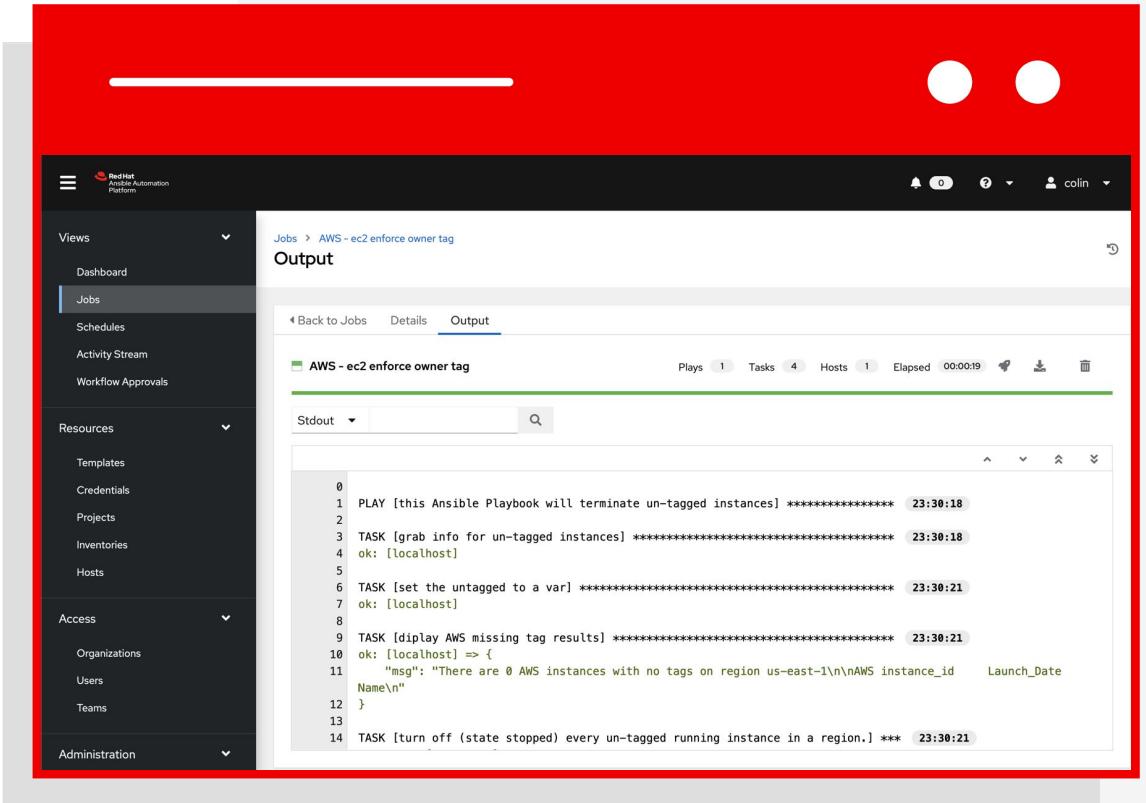
Red = New AAP component or major change

# Automation controller overview

# A playbook run

## Where it all starts

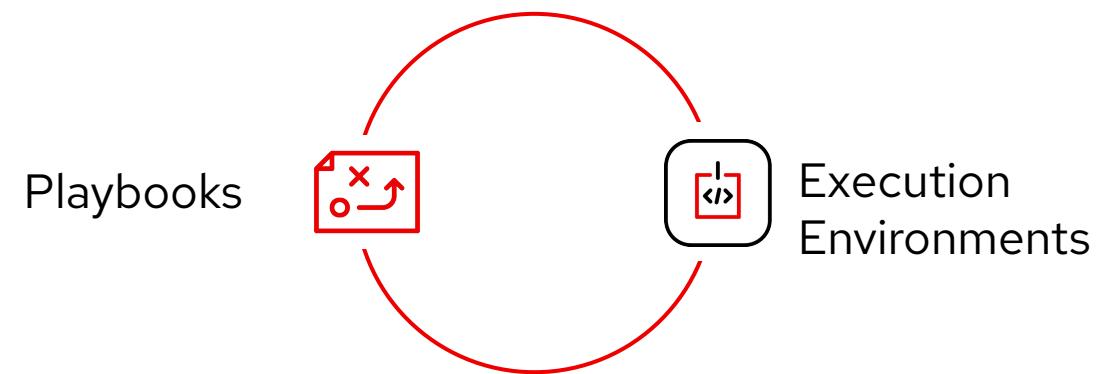
- ▶ A playbook is interpreted and run against one or multiple hosts – task by task. The order of the tasks defines the execution.
- ▶ In each task, the module does the actual work.



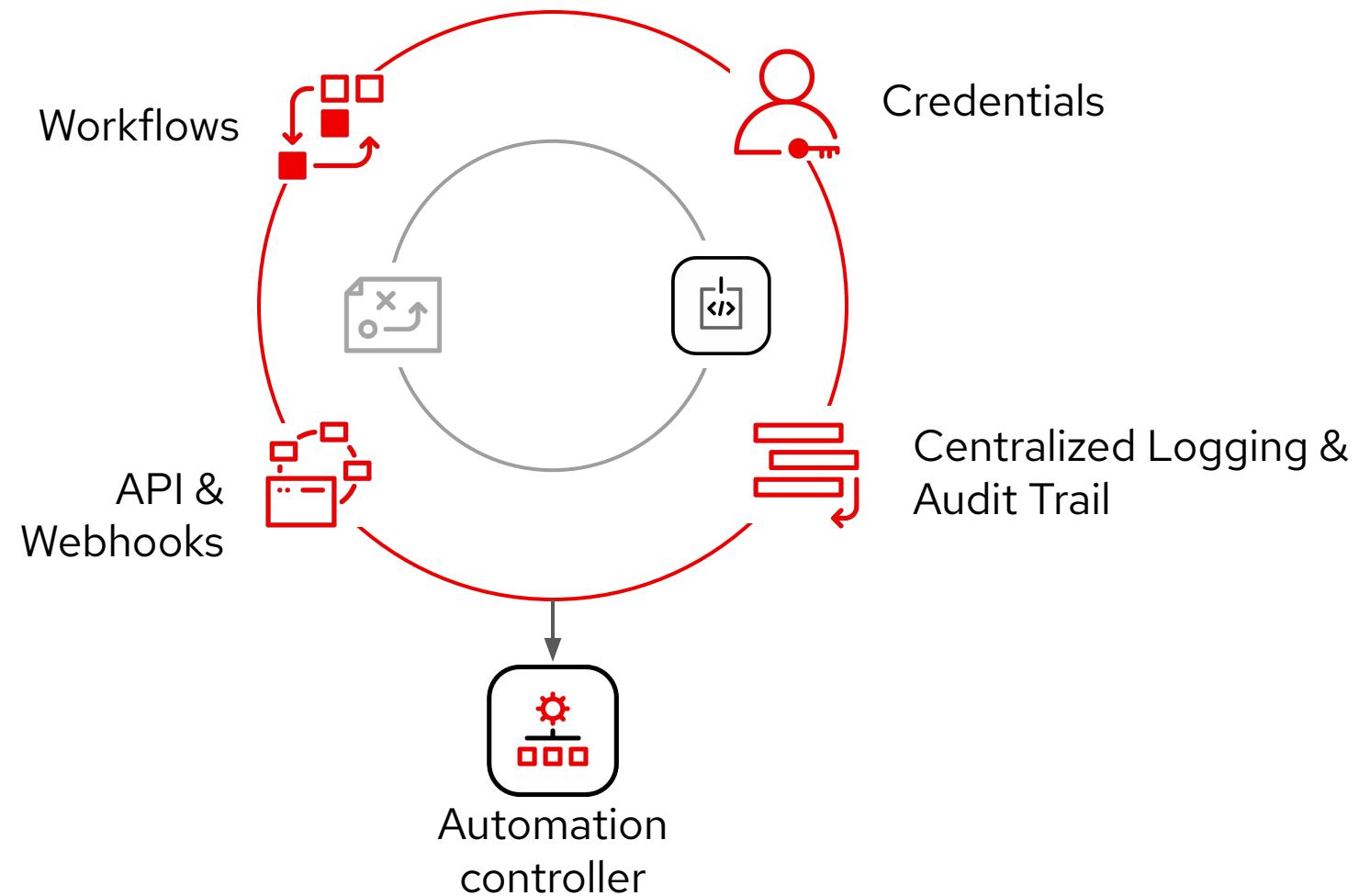
The screenshot shows the Red Hat Ansible Platform web interface. On the left is a dark sidebar with navigation links: Views (Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals), Resources (Templates, Credentials, Projects, Inventories, Hosts), Access (Organizations, Users, Teams), and Administration. The main area has a red header bar with two white circles on the right. Below the header, the URL is "Jobs > AWS - ec2 enforce owner tag" and the tab "Output" is selected. The output pane shows the command-line logs of a playbook run:

```
0
1 PLAY [this Ansible Playbook will terminate un-tagged instances] *****
2
3 TASK [grab info for un-tagged instances] *****
4 ok: [localhost]
5
6 TASK [set the untagged to a var] *****
7 ok: [localhost]
8
9 TASK [display AWS missing tag results] *****
10 ok: [localhost] => {
11   "msg": "There are 0 AWS instances with no tags on region us-east-1\nAWS instance_id Launch_Date Name\n"
12 }
13
14 TASK [turn off (state stopped) every un-tagged running instance in a region.] ***
```

# Components of Automation



# Anatomy of Automation Operation



# Execution of content

## Running at the core

- ▶ The central execution of automation content is managed and done either via central cluster..
- ▶ Can also sync git repositories, takes care of execution environments, collections, credentials, inventory and logging.
- ▶ Full audit trail of the execution, including what version of content was executed, what variable values were provided, etc.

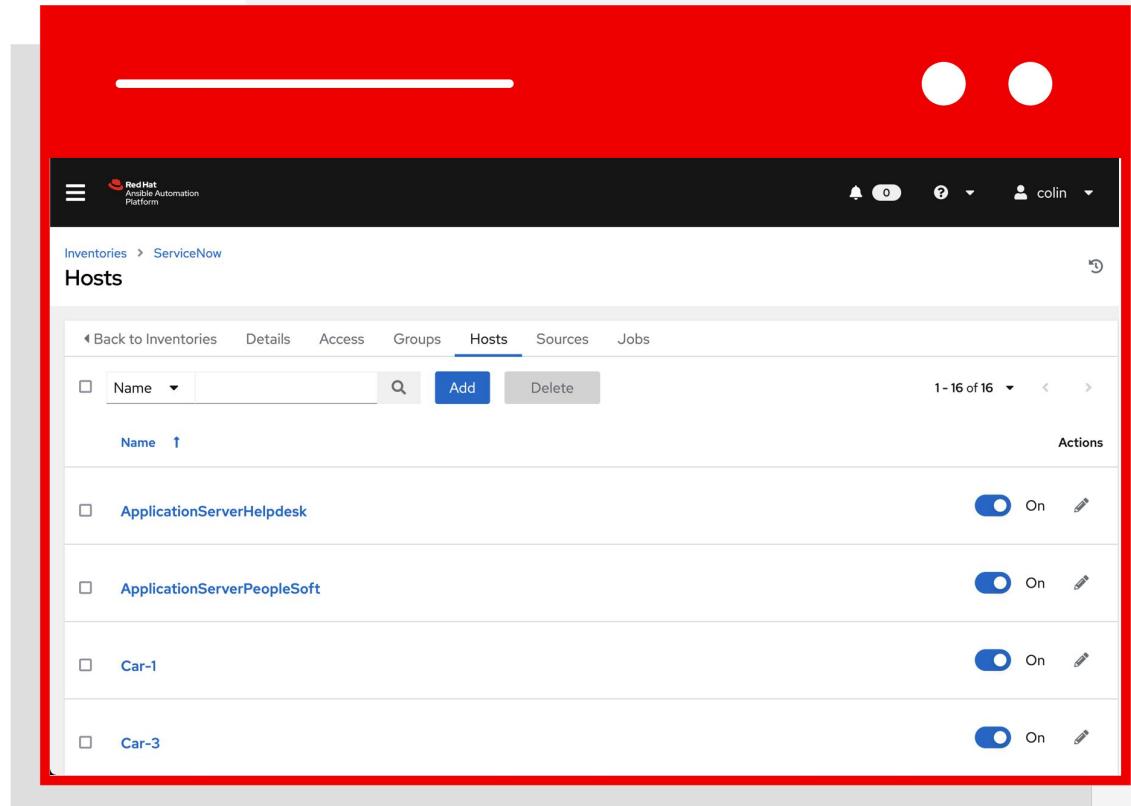
The screenshot shows the 'Details' tab for a template in the Red Hat Ansible Automation Platform. The template is named 'AWS - ec2 enforce owner tag'. Key details include:

Name	AWS - ec2 enforce owner tag	Description	stop all instances without an owner tag
Job Type	run	Organization	Public Cloud Group
Inventory	no inventory - API	Project	aws ec2 operational project
Execution Environment	aws_ee	Playbook	playbooks/no_tags.yaml
Forks	0	Verbosity	0 (Normal)
Timeout	0	Show Changes	Off
Job Slicing	1	Created	4/15/2021, 9:59:03 AM by seanc
Last Modified	1/5/2022, 4:25:40 PM by admin		
Enabled Options	Concurrent Jobs		
Credentials	Cloud: Sean AWS		

# Inventories and credentials

## How to talk to others

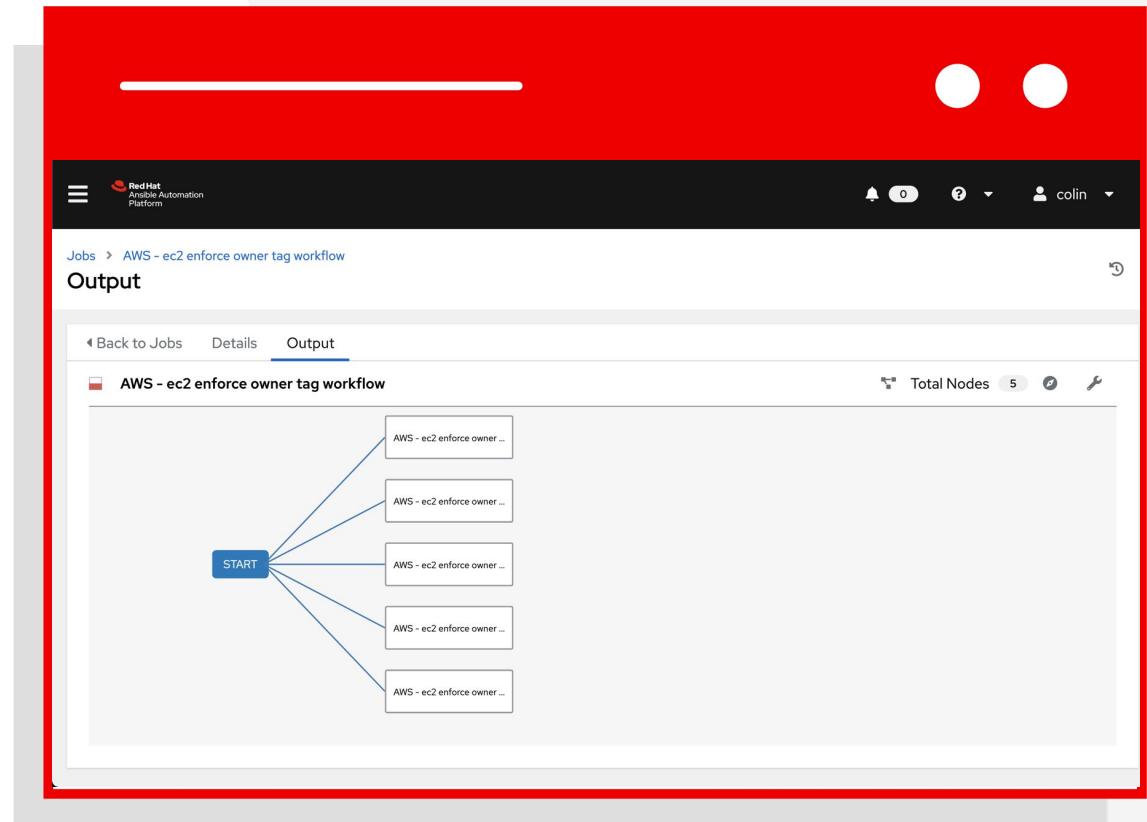
- ▶ An inventory is a collection of hosts (nodes) with associated data and groupings that the automation platform can connect to and manage:
  - Nodes
  - Groups
  - Can be static or dynamic
  - Smart inventories possible
- ▶ And what usernames and passwords do you use during connection? That is kept in the credentials.



# Workflows

Combine automation to create something bigger

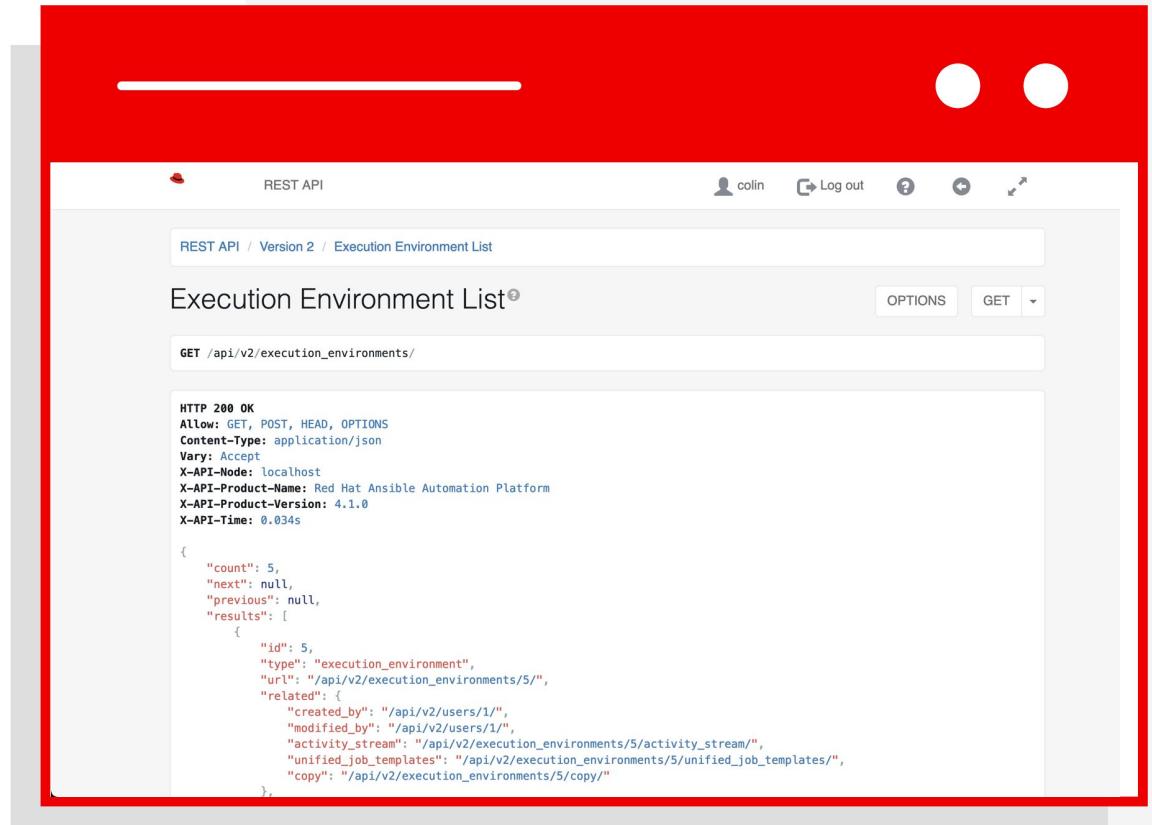
- ▶ Workflows enable the creation of powerful holistic automation, chaining together multiple pieces of automation and events.
- ▶ Simple logic inside these workflows can trigger automation depending on the success or failure of previous steps.



# API

## Integration of automation into larger workflows

- ▶ The API provides programmatic access to the automation via a defined interface.
- ▶ Underneath it is still powered by the same bits and pieces which are at the core: workflows, inventories, etc.
- ▶ It offers simple integration into other tools like ITSM, SOAR, etc.



The screenshot shows a REST API browser interface. At the top, there's a header with a user icon labeled 'colin', a 'Log out' button, and some other icons. Below the header, the title 'REST API / Version 2 / Execution Environment List' is displayed. To the right of the title are buttons for 'OPTIONS' and 'GET'. A red box highlights the main content area. The content area has a title 'Execution Environment List' and a sub-section for the 'GET /api/v2/execution\_environments/' endpoint. Below this, there's a status message 'HTTP 200 OK' followed by various headers: 'Allow: GET, POST, HEAD, OPTIONS', 'Content-Type: application/json', 'Vary: Accept', 'X-API-Node: localhost', 'X-API-Product-Name: Red Hat Ansible Automation Platform', 'X-API-Product-Version: 4.1.0', and 'X-API-Time: 0.034s'. The main response body is a JSON object representing a list of execution environments:

```
{  
    "count": 5,  
    "next": null,  
    "previous": null,  
    "results": [  
        {  
            "id": 5,  
            "type": "execution_environment",  
            "url": "/api/v2/execution_environments/5/",  
            "related": {  
                "created_by": "/api/v2/users/1/",  
                "modified_by": "/api/v2/users/1/",  
                "activity_stream": "/api/v2/execution_environments/5/activity_stream/",  
                "unified_job_templates": "/api/v2/execution_environments/5/unified_job_templates/",  
                "copy": "/api/v2/execution_environments/5/copy/"  
            }  
        },  
        {  
            "id": 1,  
            "type": "execution_environment",  
            "url": "/api/v2/execution_environments/1/",  
            "related": {  
                "created_by": "/api/v2/users/1/",  
                "modified_by": "/api/v2/users/1/",  
                "activity_stream": "/api/v2/execution_environments/1/activity_stream/",  
                "unified_job_templates": "/api/v2/execution_environments/1/unified_job_templates/",  
                "copy": "/api/v2/execution_environments/1/copy/"  
            }  
        },  
        {  
            "id": 2,  
            "type": "execution_environment",  
            "url": "/api/v2/execution_environments/2/",  
            "related": {  
                "created_by": "/api/v2/users/1/",  
                "modified_by": "/api/v2/users/1/",  
                "activity_stream": "/api/v2/execution_environments/2/activity_stream/",  
                "unified_job_templates": "/api/v2/execution_environments/2/unified_job_templates/",  
                "copy": "/api/v2/execution_environments/2/copy/"  
            }  
        },  
        {  
            "id": 3,  
            "type": "execution_environment",  
            "url": "/api/v2/execution_environments/3/",  
            "related": {  
                "created_by": "/api/v2/users/1/",  
                "modified_by": "/api/v2/users/1/",  
                "activity_stream": "/api/v2/execution_environments/3/activity_stream/",  
                "unified_job_templates": "/api/v2/execution_environments/3/unified_job_templates/",  
                "copy": "/api/v2/execution_environments/3/copy/"  
            }  
        },  
        {  
            "id": 4,  
            "type": "execution_environment",  
            "url": "/api/v2/execution_environments/4/",  
            "related": {  
                "created_by": "/api/v2/users/1/",  
                "modified_by": "/api/v2/users/1/",  
                "activity_stream": "/api/v2/execution_environments/4/activity_stream/",  
                "unified_job_templates": "/api/v2/execution_environments/4/unified_job_templates/",  
                "copy": "/api/v2/execution_environments/4/copy/"  
            }  
        }  
    ]  
}
```

# Role-based access control

## How to manage access

- ▶ Role-based access control system:

Users can be grouped in teams, and roles can be assigned to the teams.

- ▶ Rights to edit or use can be assigned across all objects.

- ▶ All backed by enterprise authentication if needed.

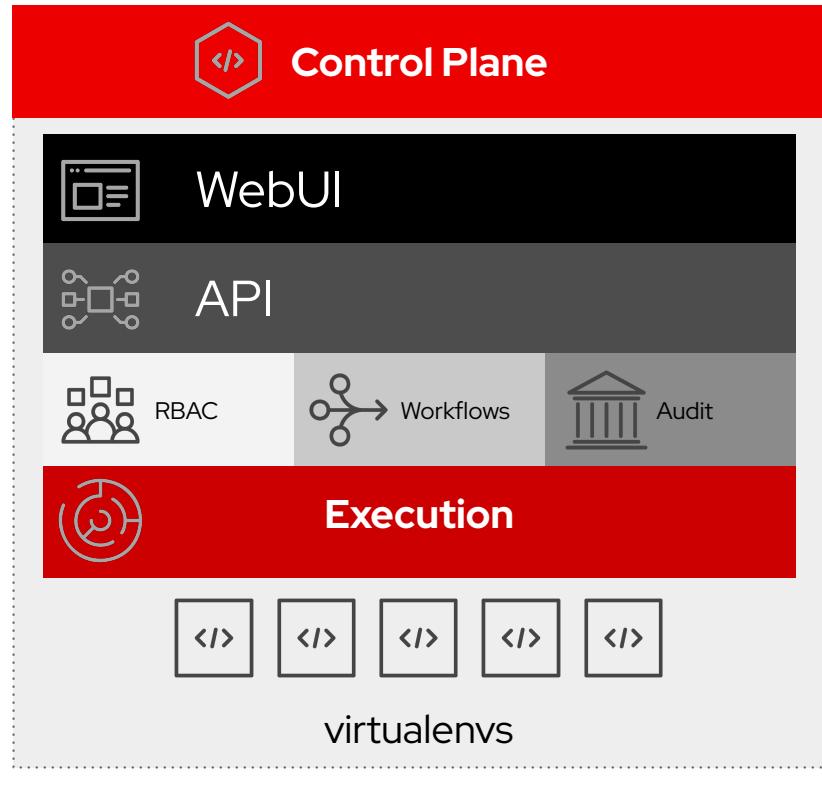
The screenshot shows the Red Hat Ansible Platform web interface. On the left is a dark sidebar with various navigation options: Views (Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals), Resources (Templates, Credentials, Projects, Inventories, Hosts), Access (Organizations, Users, Teams), and Administration. The 'Inventories' option under Resources is currently selected. The main content area has a red header bar with two white circles on the right. Below the header, the title 'Access' is displayed above a table. The table has columns for Username, First name, Last name, and Roles. It lists six users: admin, ajay, andriusb, colin, brandt, and another entry for ajay. All users are listed as 'System Administrator' under the 'User Roles' column. At the bottom of the table, it says '1-5 of 9 items' and 'of 2 pages'.

Username	First name	Last name	Roles
admin			User Roles: System Administrator
ajay	Ajay	Chenampara	User Roles: System Administrator
andriusb	Andrius	Benokraitis	User Roles: System Administrator
colin	Colin	McNaughton	User Roles: System Administrator
brandt	Craig	Brandt	User Roles: System Administrator
ajay			User Roles: System Administrator

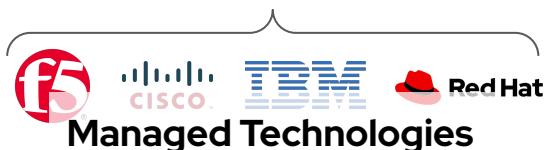
---

# Automation for an **agile** world

# Ansible Automation Platform 1 Architecture



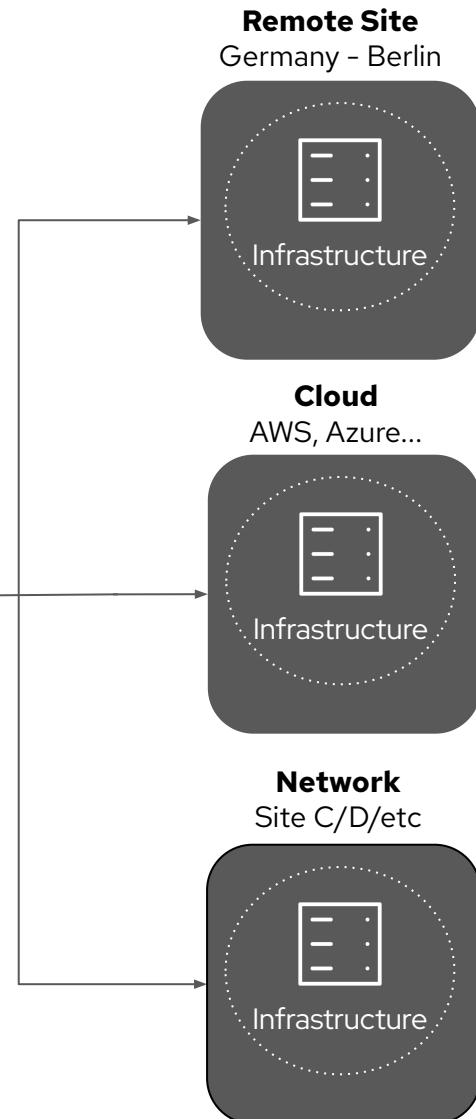
- ▶ The artist formerly known as Ansible Tower
- ▶ Centralized, monolithic application
- ▶ Control and execution cluster capacity shared
- ▶ Closely coupled to the database and requires low latency
- ▶ Deployment limitations: Poor scalability, rigid architecture
- ▶ Isolated node constraints: Extremely sensitive to latency, low resilience to connection disruption



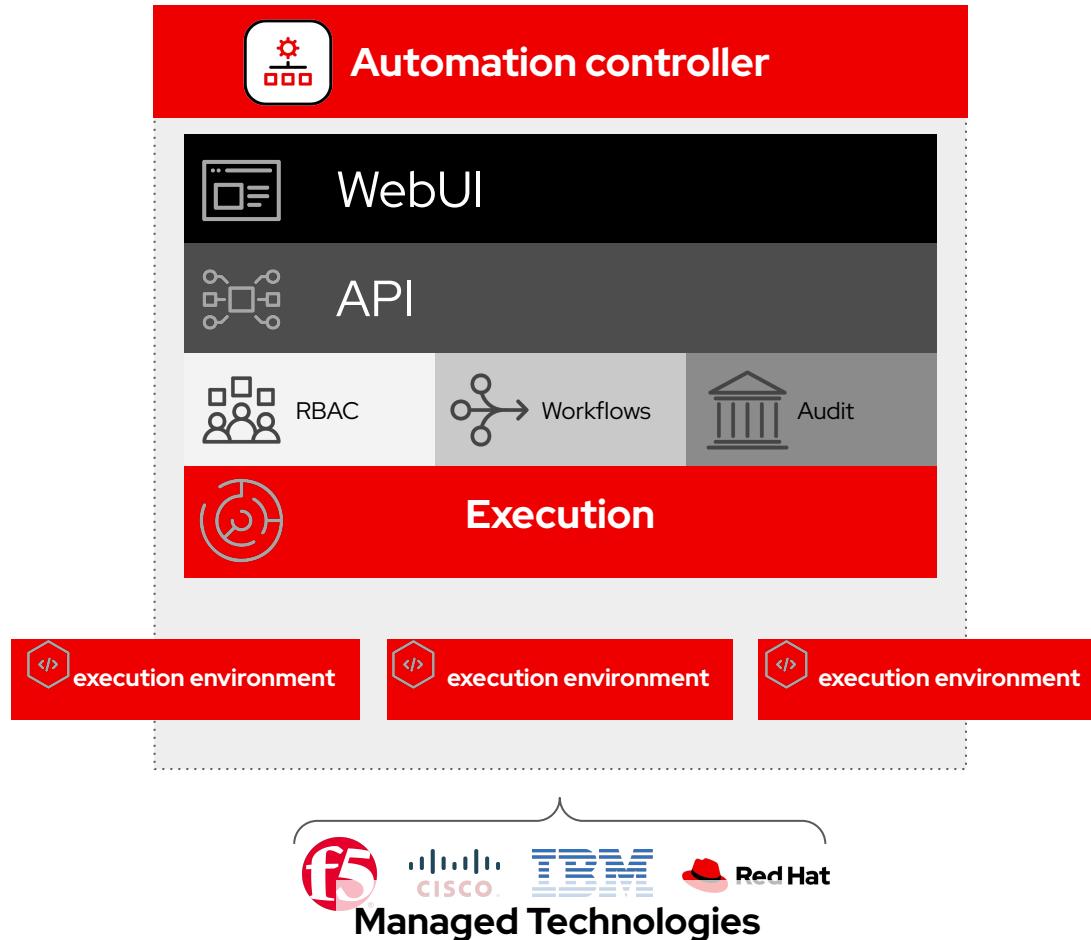
## Manage



Ansible Tower

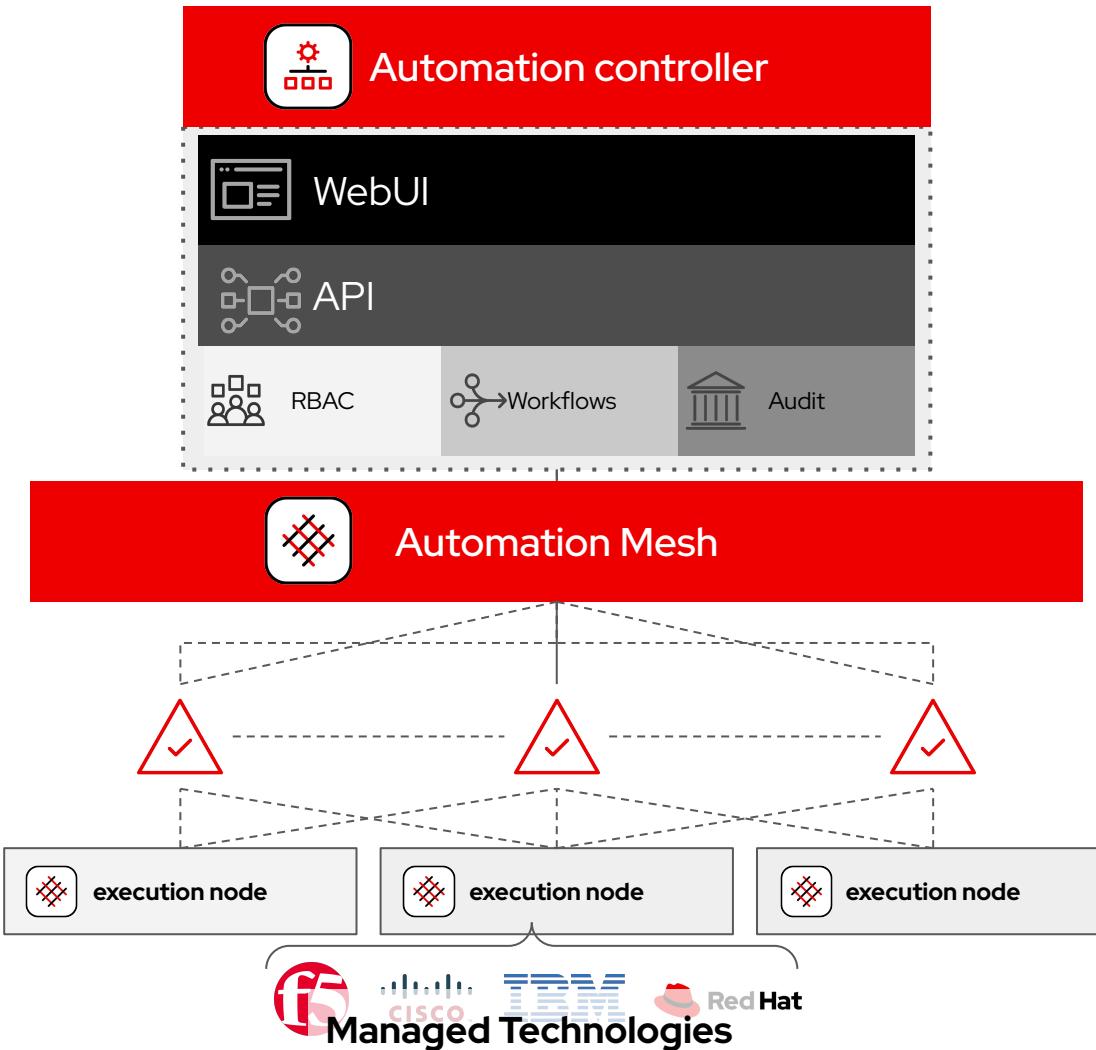


# Ansible Automation Platform 2.0 Early Access

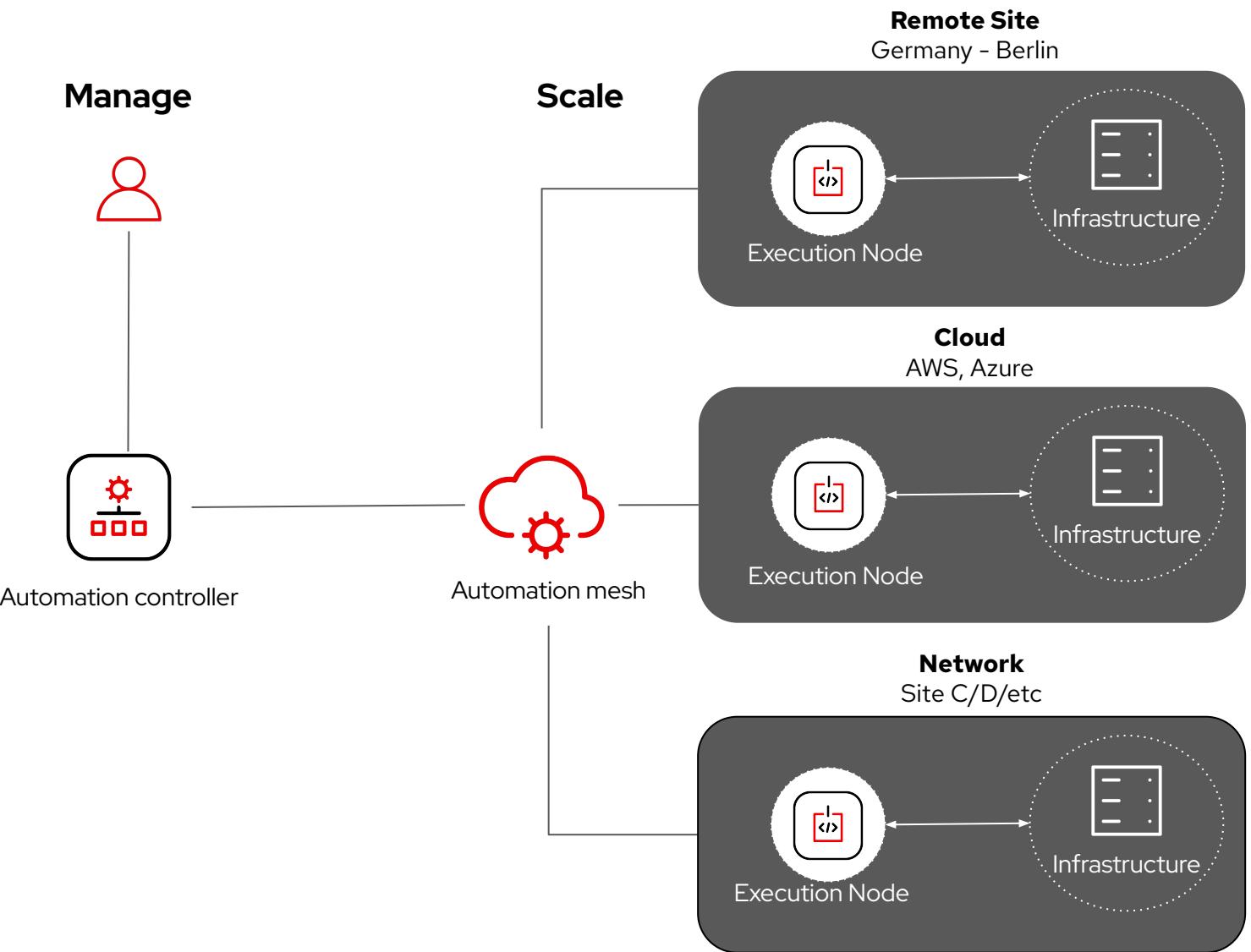


- ▶ Started to modularize
- ▶ **Control node still contains control plane and execution plane**
- ▶ Containerized execution environments

# Ansible Automation Platform 2.1 - GA since December 2021



- ▶ Dynamic cluster capacity, Cluster capacity scales independently
- ▶ **Decoupled execution and control plane**, deploy execution capacity where it's needed
- ▶ **Execution plane resilient to latency** and connection interruptions
- ▶ Natively build redundant mesh topologies
- ▶ Centralized management with automation controller



# Automation Mesh

# What is automation mesh?

Simple, flexible and reliable scaling of execution capacity

## Automate at a global scale

Simple, flexible and reliable way to scale automation of large inventories across diverse network topologies and platforms.

## Distributed overlay network

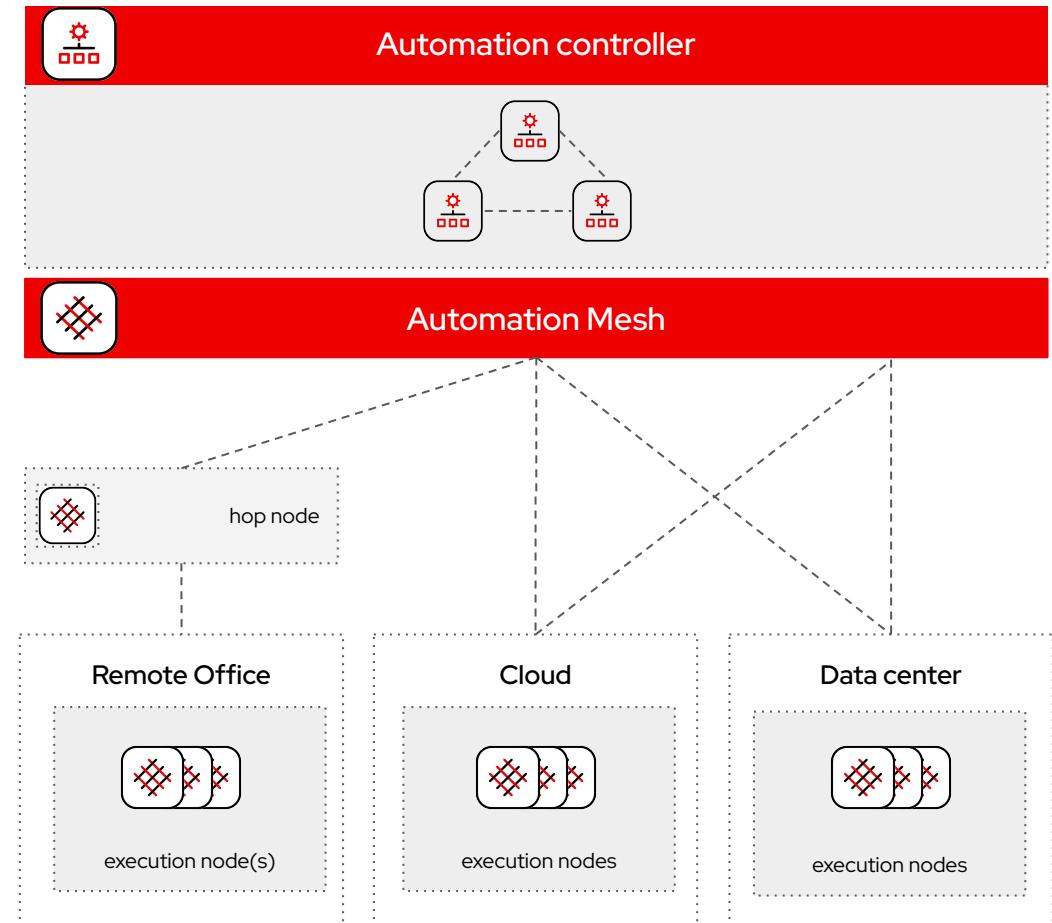
Overlay network which eases distributing automation execution  
Establishes peer-to-peer connections between execution nodes  
across existing networks

## Flexible architecture

Flexible architecture offers more design choices compared to isolated nodes

## Execution node health

Health checks performed on execution nodes



# Automation mesh node types

## Hybrid node

Default for controller nodes

Performs controller runtime functions and automation execution

## Controller node

Dedicated to controller runtime functions

Execution capabilities disabled

## Execution node

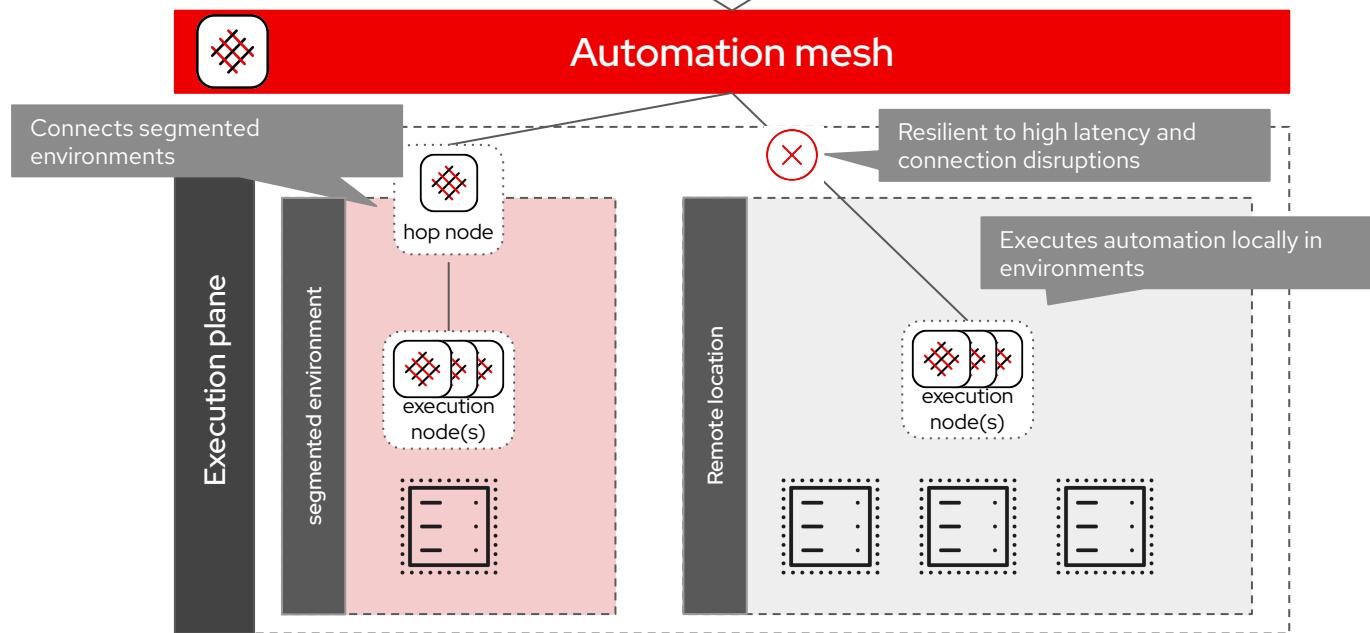
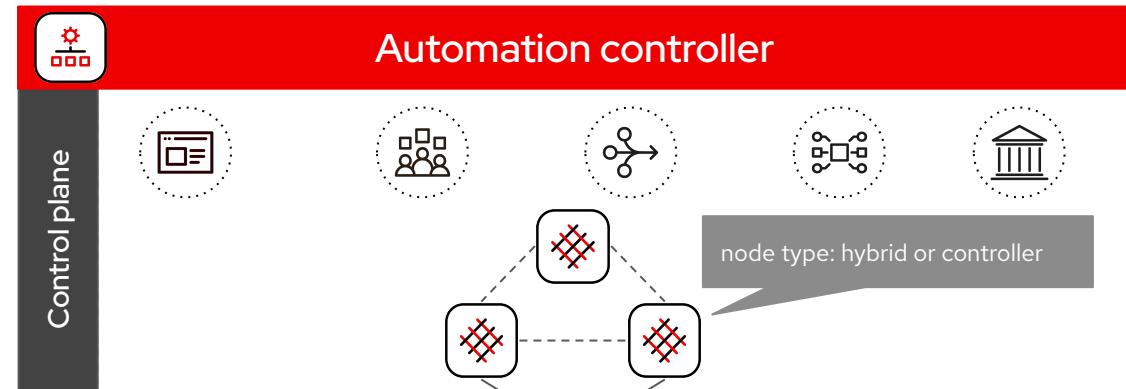
Dedicated to run automation on behalf of controller

Job isolation via podman and execution environments

## Hop node

Dedicated to route traffic to other execution nodes

Cannot execute automation



# Ansible Automation Mesh in 2.1

## Mesh replaces Isolated Nodes

Isolated Nodes are discontinued.

Functionality is provided by automation mesh



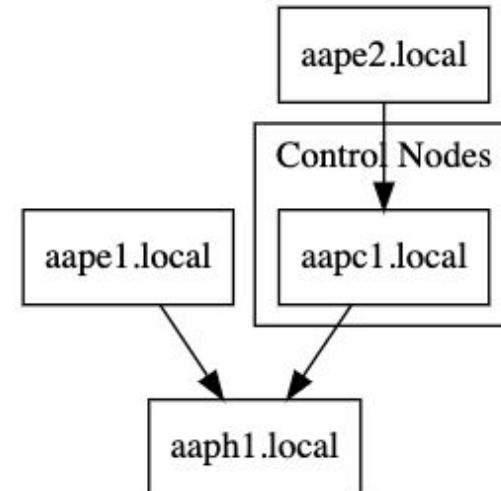
## Mesh installation

Mesh is installed using inventory installer method

New sections added to inventory file for automation mesh configuration

Installer performs sanity checks on mesh configuration

Flexible configuration and design options



## Health checks

Health checks inspect execution node capacity

Health check intervals differ based on mesh node type

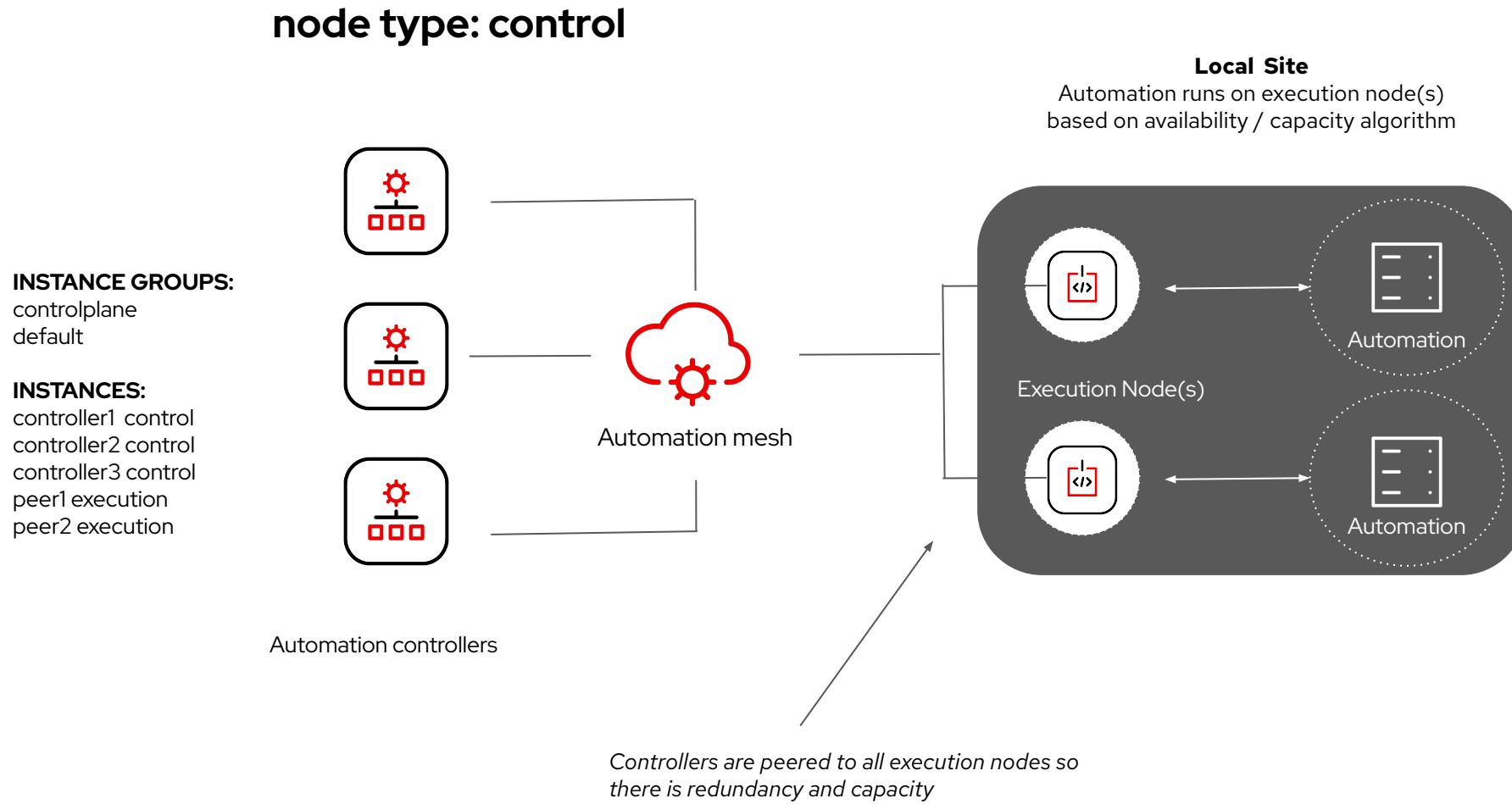
## Graphing

AAP2 installer can be used to create Graphviz dot file:

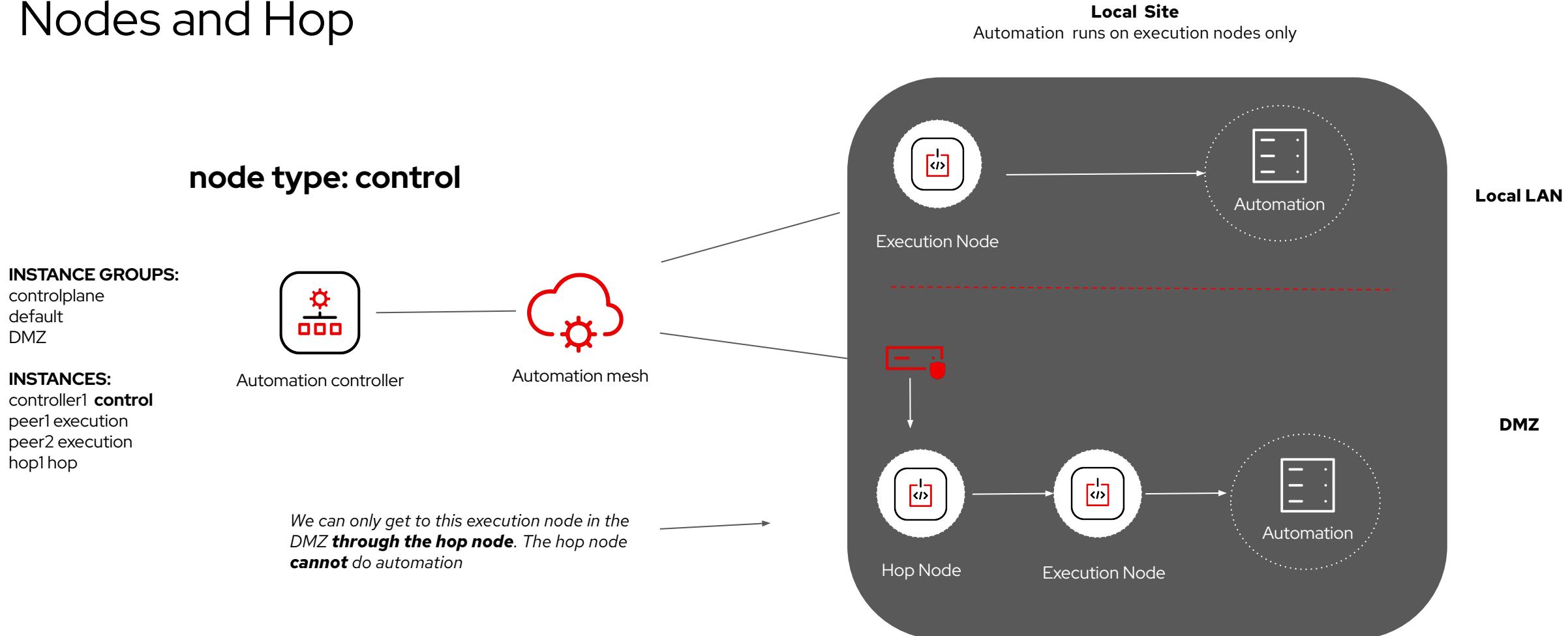
-> Incl installation: `./setup.sh -e generate_dot_file=/tmp/mesh.dot`

-> Only graph file creation: `./setup.sh --tag generate_dot_file`

# Scaleable Starter Setup: Multi Controllers/Execution Nodes



# Scaleable Starter Setup: Single Controller/Multi Execution Nodes and Hop

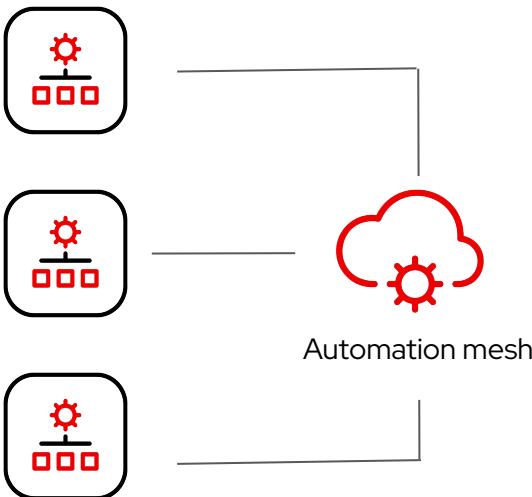


# Multi DC Setup: flat, connected cross site network

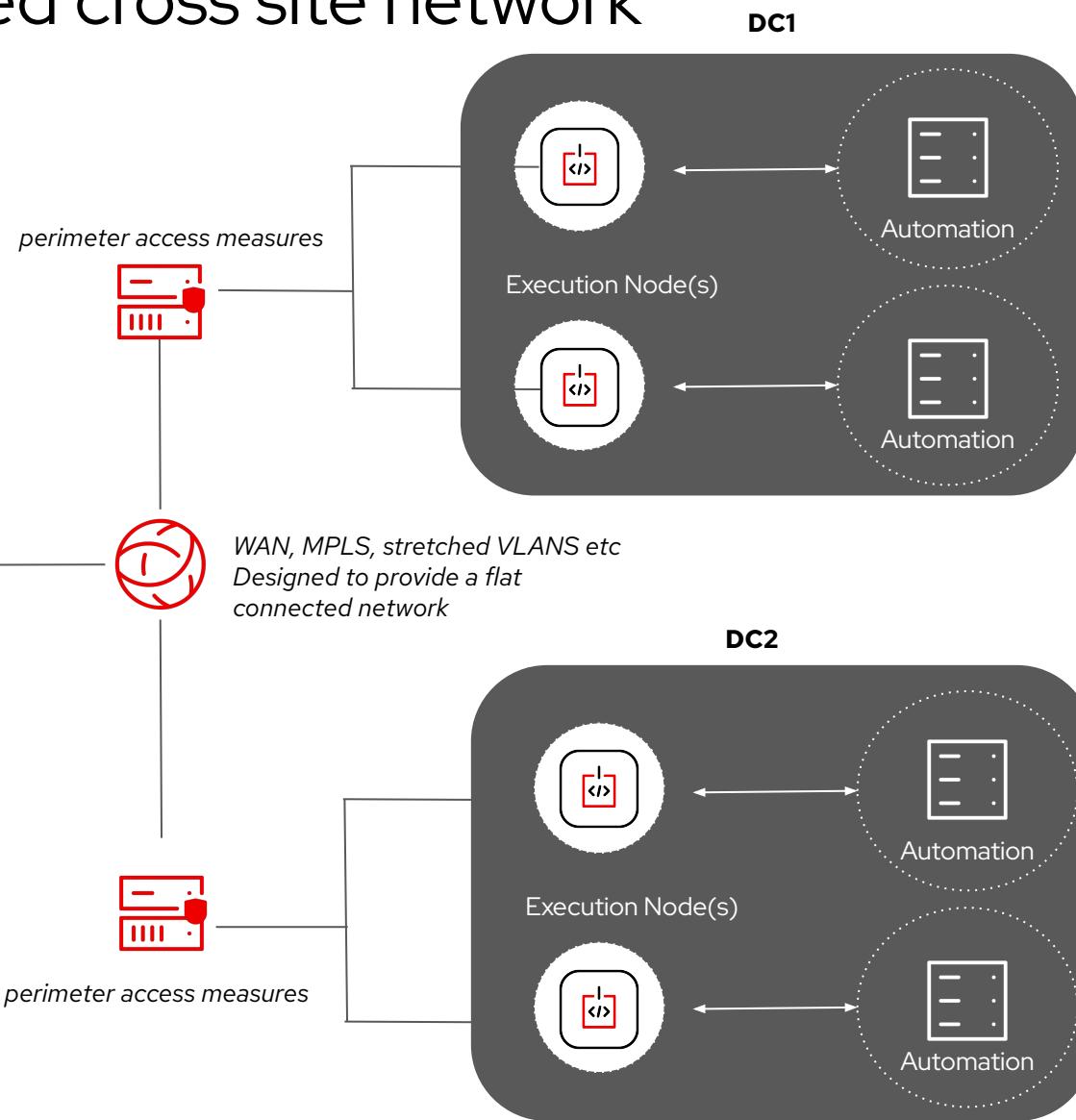
**node type: hybrid or control**

**INSTANCE GROUPS:**  
controlplane  
default  
DC1  
DC2

**INSTANCES:**  
controller1-n  
peer1 execution [DC1]  
peer2 execution [DC1]  
peer3 execution [DC2]  
peer4 execution [DC2]



*Controllers could be in either DC and used for execution as well or just control plane.*

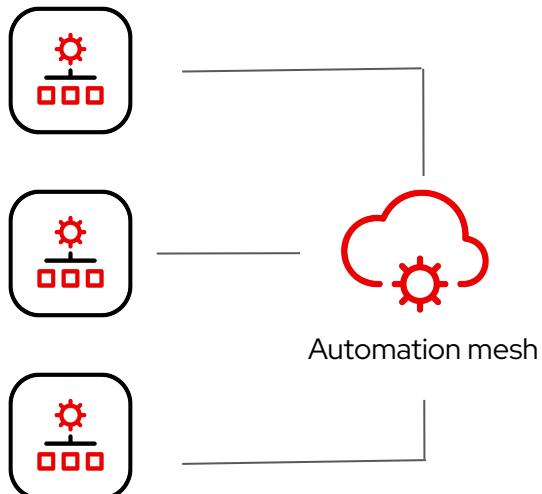


# Multi DC Setup: multiple zoned network segregation $DC_n$

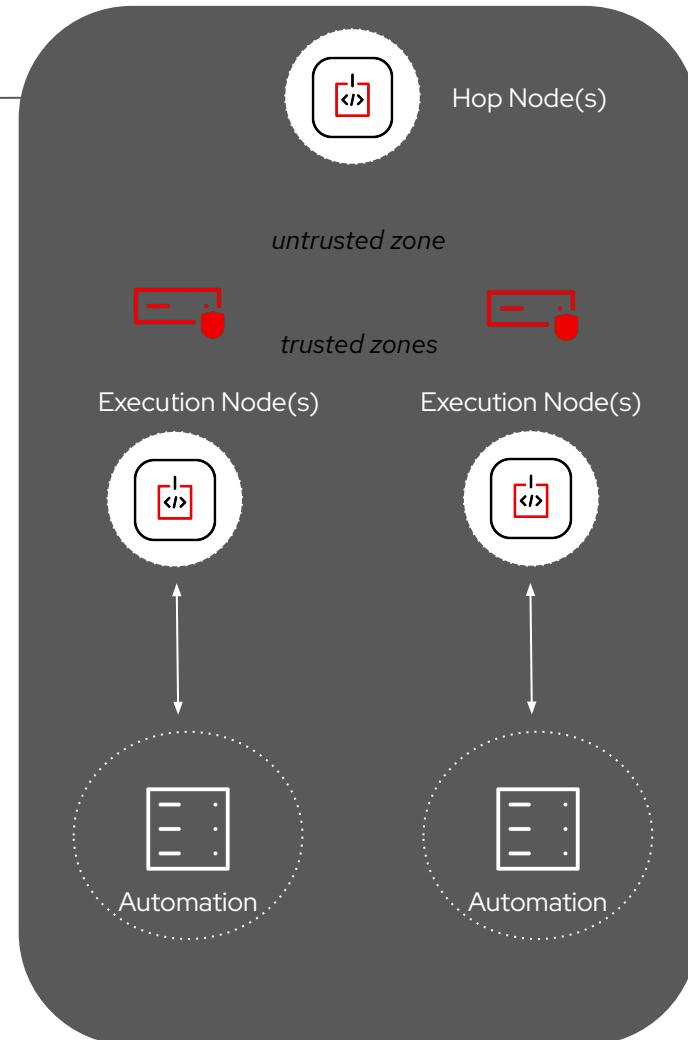
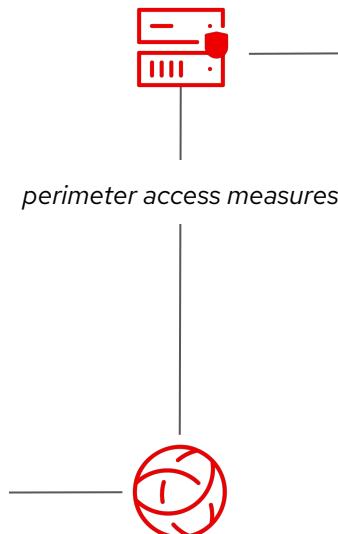
**node type: hybrid or control**

**INSTANCE GROUPS:**  
controlplane  
default  
DC1  
DC2

**INSTANCES:**  
controller1-n  
hop1-n  
peerz1-n execution [DC $n$ -zone1]  
peerz2-n execution [DC $n$ -zone2]



*Controllers could be on corporate LAN or in either DC and used for execution as well or just control plane.*



# Automation for Developers and IT Operations

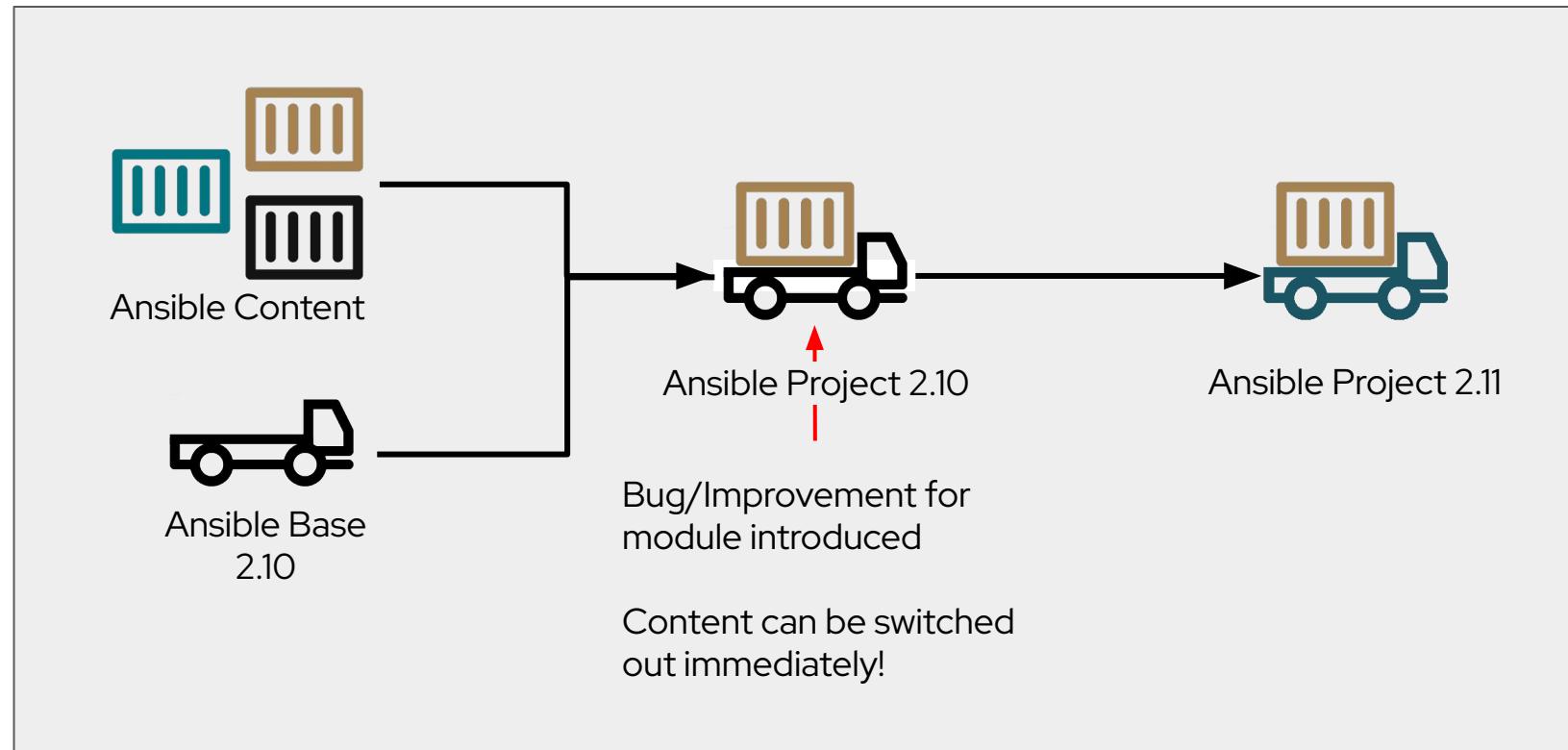
# Refresher: Ansible Content Collections

# Collections: Simplified and Consistent Content Delivery

## What are they?

Collections are a data structures containing automation content:

- ▶ Modules
- ▶ Playbooks
- ▶ Roles
- ▶ Plugins
- ▶ Docs
- ▶ Tests



# Ansible Content Delivery

The screenshot shows the Ansible Galaxy interface for the 'ibm' collection. At the top, there's a navigation bar with links for 'About', 'Help', 'Documentation', and 'Login'. Below the header, the 'ibm' collection is displayed with its logo, name, location (Armonk, New York, U.S.), and GitHub link. A 'Login to Follow' button is present. The collection has 6 collections, including:

- blockchain\_platform**: Ansible collection for building Hyperledger Fabric networks using the IBM Blockchain Platform. Current Version: 0.0.22 uploaded a day ago. 23 Modules, 3 Roles, 13 Plugins. Tags: ibm, enterprise, blockchain, dt, hyperledger, fabric, blockchain.
- ibm\_zos\_core**: The IBM z/OS core collection includes connection plugins, action plugins, modules, filters, sample playbooks and ansible-doc to automate tasks on z/OS. Current Version: 1.1.0-beta1 uploaded 19 days ago. 9 Modules, 0 Roles, 14 Plugins. Tags: ibm, z, zos, z\_os, core, zos\_core, ibm\_zos\_core, data\_set, jcl, uss, mvs.
- ibm\_zos\_ims**: The IBM z/OS IMS collection includes action plugins, modules and sample playbooks to automate tasks for IBM IMS. Current Version: 1.0.0-beta2 uploaded 4 days ago. 4 Modules, 0 Roles, 4 Plugins. Tags: ibm, z, zos, z\_os, core, zos\_core, data\_set, jcl, uss, mvs, ims, zos\_ims.

Ansible Galaxy  
galaxy.ansible.com

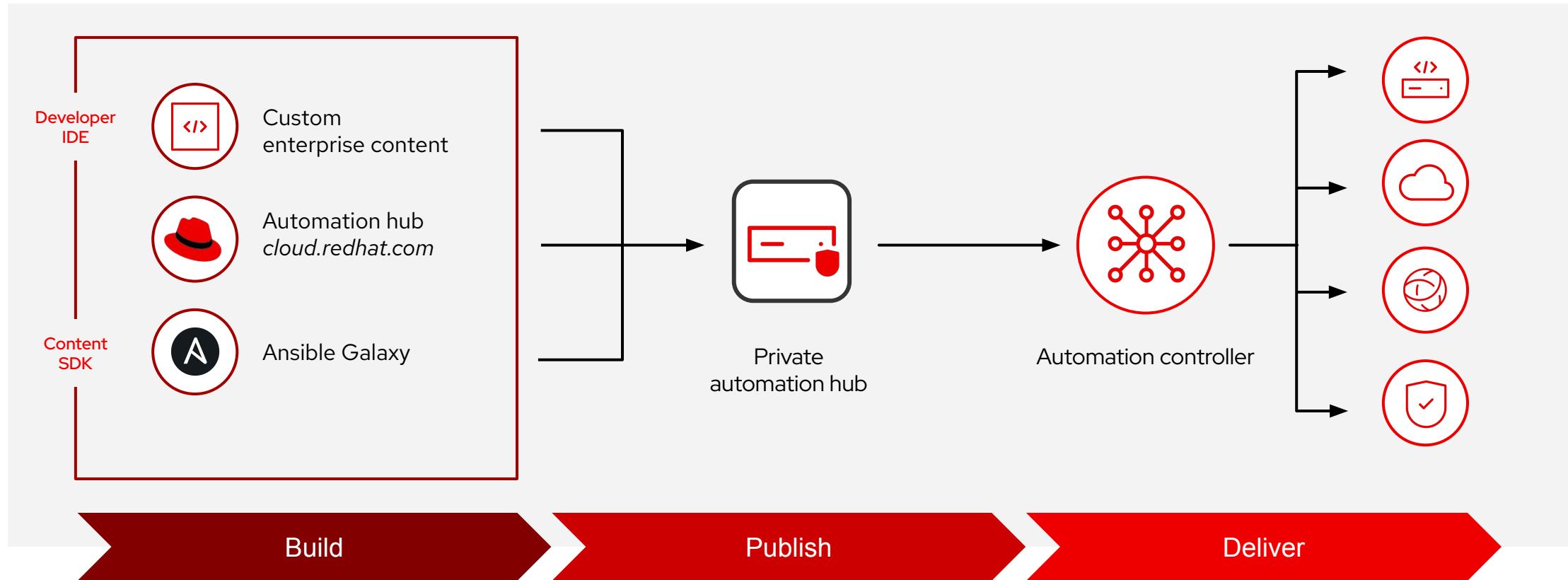
The screenshot shows the Red Hat Ansible Automation Hub interface. The left sidebar includes 'Red Hat Ansible Automation Platform', 'Automation Analytics', 'Automation Hub' (selected), 'Collections', 'Partners', 'My namespaces', 'Automation Services Catalog', and 'Documentation'. The main area is titled 'Collections' with a search bar containing 'ibm'. It lists several collections:

- ibm**: Certified. ibm\_zos\_core: Provided by IBM. The IBM z/OS core collection includes connection plugins, action plugins, modules, filters, sample playbooks and ansible-doc to automate tasks on z/OS. 4 Modules, 0 Roles, 6 Plugins.
- IBM**: Certified. spectrum\_virtualize: Provided by IBM. Ansible Collections for IBM Spectrum Virtualize. 6 Modules, 0 Roles, 1 Plugin.
- ibm**: Certified. radar: Provided by IBM. Ansible Collection of content aiming to automate c... 7 Modules, 0 Roles, 2 Plugins.
- Pure Storage**: Certified. flasharray: Provided by Pure Storage. Collection of modules to manage Pure Storage Flash... 35 Modules, 0 Roles, 2 Plugins.

Ansible Automation Hub  
cloud.redhat.com



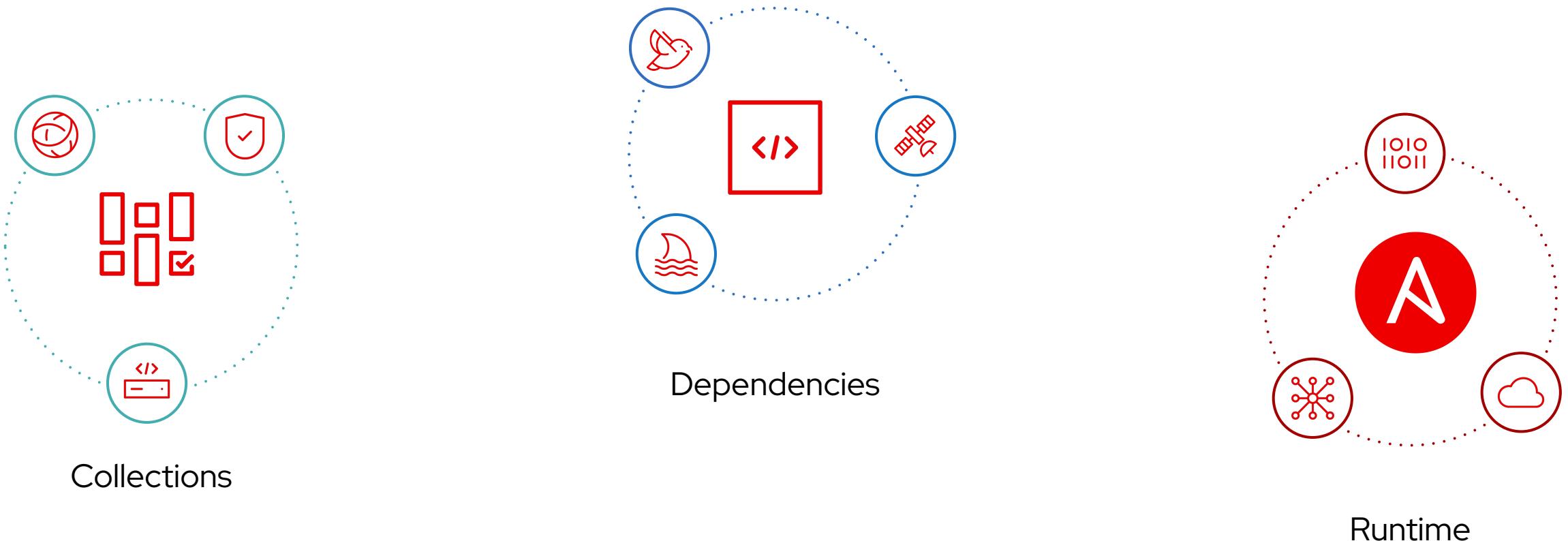
# Private Automation Hub



# Automation execution environments

# Many technologies, different life cycles

How to keep runtime environment, collections and dependencies aligned?



# Automation execution environments

## Challenge



Management of Ansible dependencies, multiple Python virtualenvs increases complexity of automation.

---

## Solution & Business Value



Standardized package format to build, deploy and run Ansible content, provides reliable, reproducible and scalable execution layer.

---

## Technical Implementation



Bundle together all required Collections, corresponding RPM or PIP3 dependencies, and a minimal Ansible version in a single container.

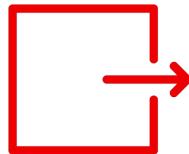
# Limitations of Python Virtual Environments

They don't work for the enterprise



## Tooling

Python Virtual Environments are not part of the Red Hat Ansible Automation Platform, they are Python constructs meant for Python developers.



## Portability

Python Virtual Environments are unique to a single system and hard to replicate on another system.



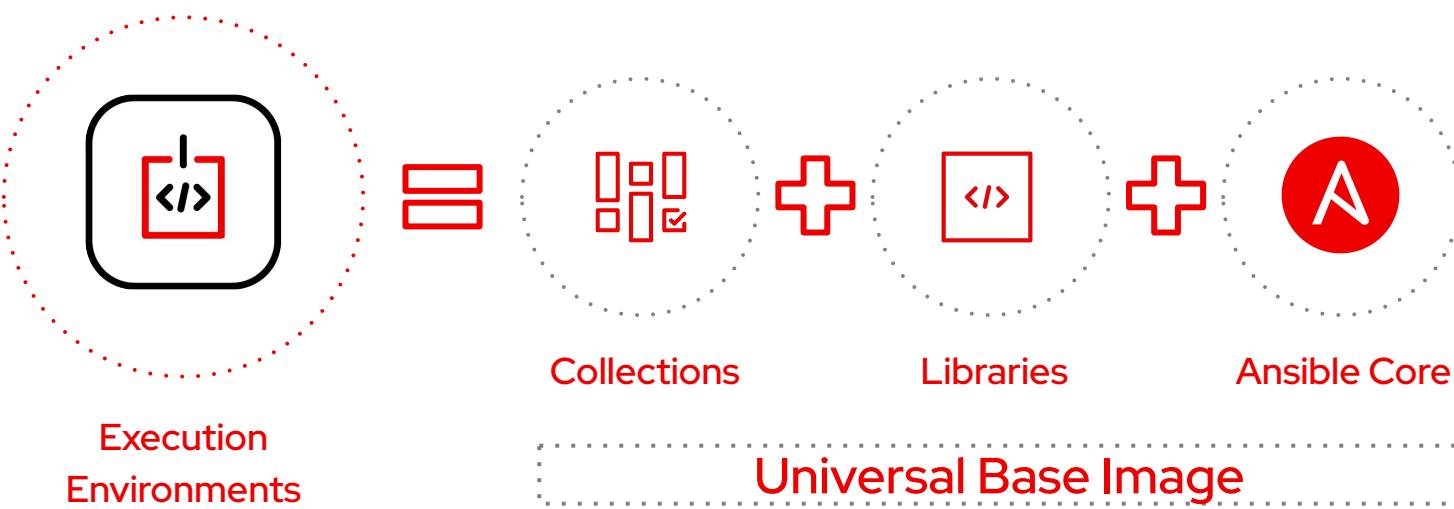
## Maintenance

Python Virtual Environments may have dozens of Python dependencies and become increasingly hard to manage and maintain overtime.



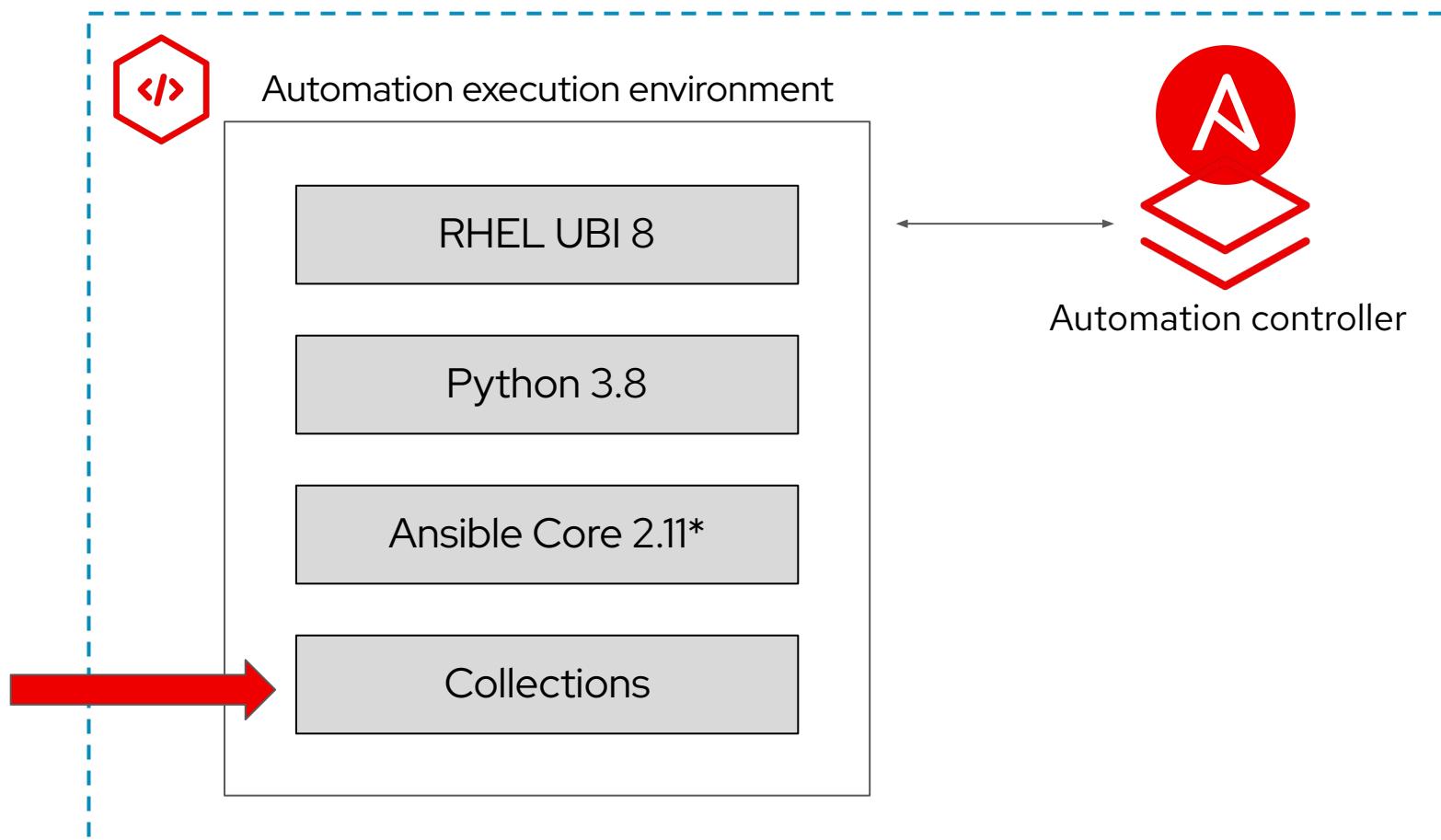
# Automation Execution Environments

Components needed for automation, packaged in a cloud-native way



# Example Packaging with automation execution environments

-  amazon.aws Collection
-  ansible.utils Collection
-  arista.cvp Collection
-  azure.azcollection Collection
-  ibm.qradar Collection
-  redhat.satellite Collection



\*includes other Ansible dependencies/packages

## Available automation execution environments with AAP 2

- ▶ **Minimal** (ee-minimal-rhel8) - Contains Ansible Core 2.11 and doesn't contain any Collections.
- ▶ **Supported** (ee-supported-rhel8) - This is the default image. It is built on top of the minimal image and contains content supported by Red Hat.
- ▶ **Compatibility** (ee-29-rhel8) - Contains Ansible 2.9 "batteries included" and is best for customers migrating from Ansible Automation Platform 1.2.

# Execution environment builder

# Custom EEs: Execution environment builder

Execution environment builder is a tool that aids in the creation of execution environments.

No existing tooling for building Automation execution environments.

## Challenge

## Solution

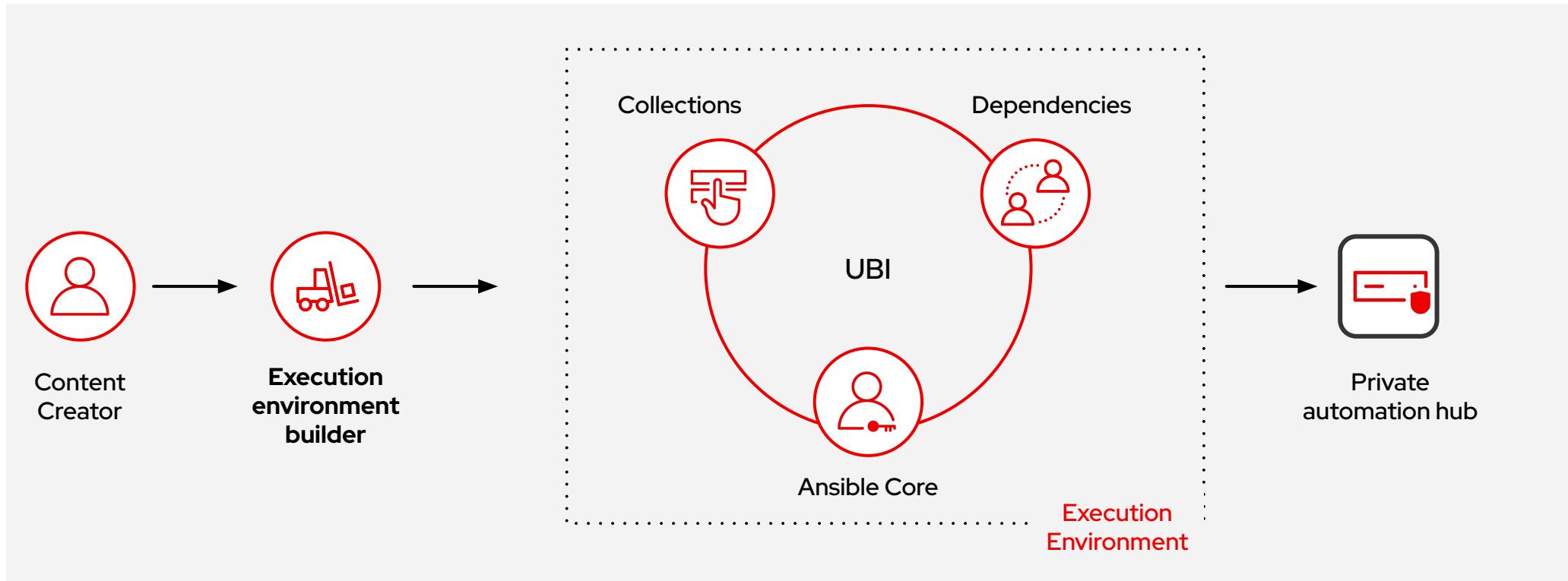
Execution environment builder allows organizations to customize and build their own execution environments with the collections and dependencies they need.

## Implementation

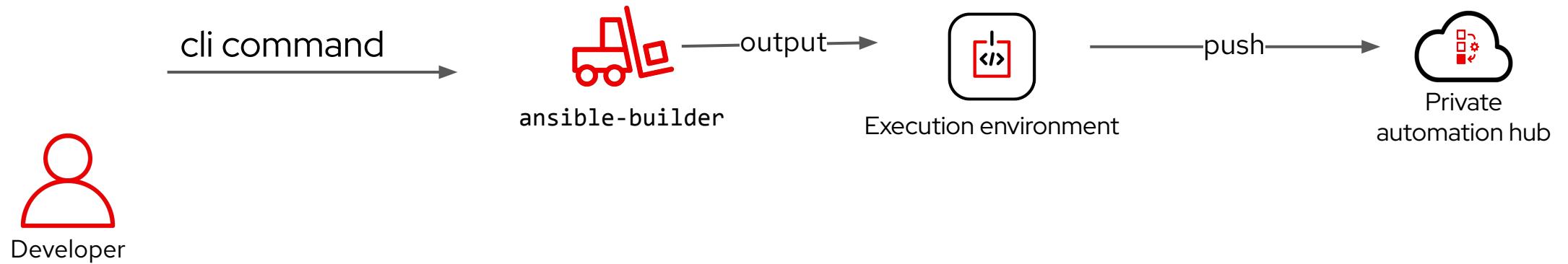
Execution environment builder is a Python application that will produce the container image along with any other files that need to be added to the image.

# Build, create, publish

Development cycle of an automation execution environment



# Adapting execution environments



# Automation content navigator

## Automation content navigator

Top-level interface for Platform enterprise developers

### Challenge

Containerized execution introduces new challenges for developing, testing, and deploying Ansible content destined for automation controller.

---

### Solution

Provides a more cohesive, more consistent, predictable, developer experience for content destined to be run on the Automation Platform.

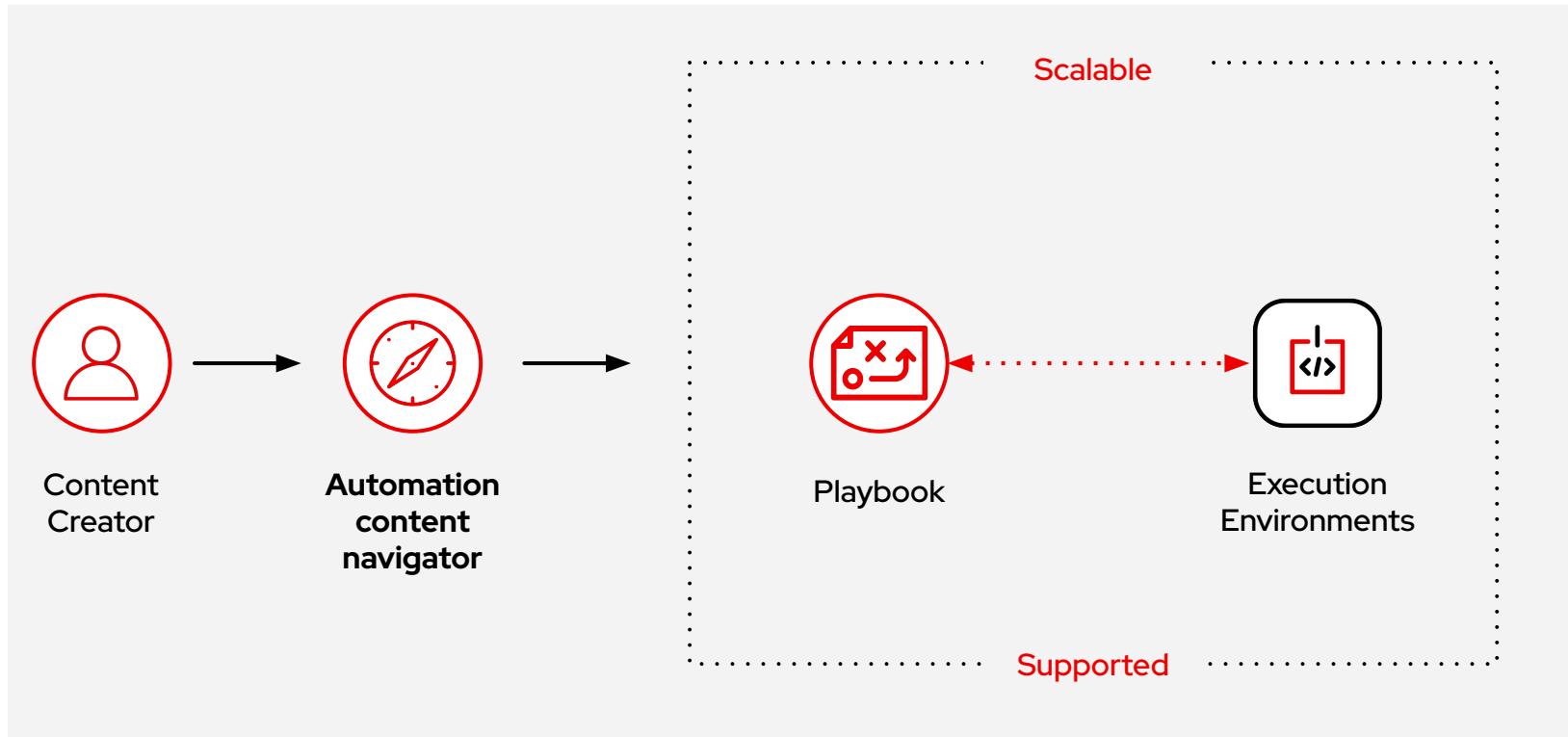
---

### Implementation

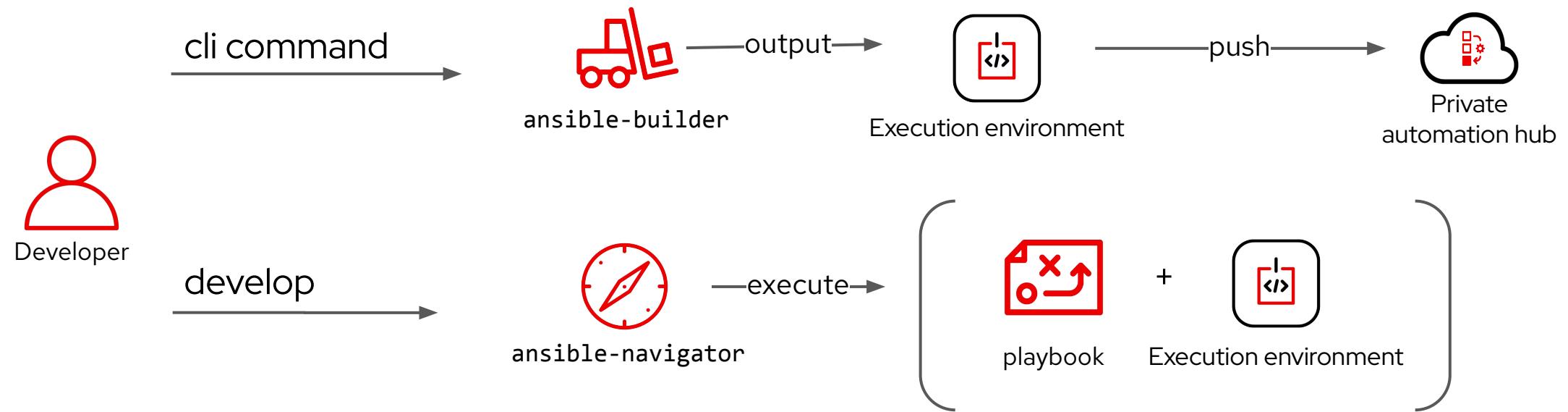
Automation content navigator is a Python application bundled with Ansible Automation Platform.

# Develop, test, run

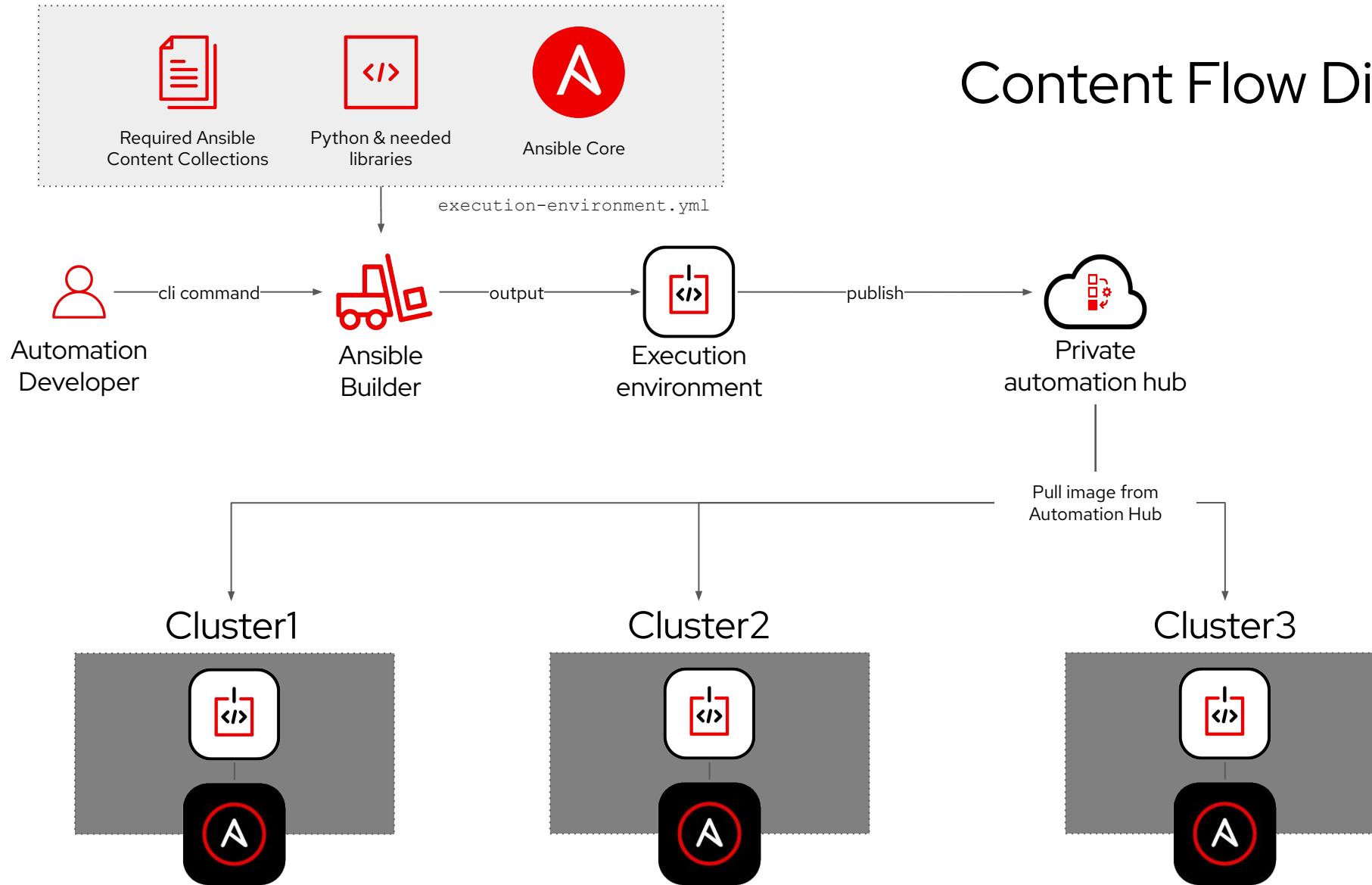
How to develop, test and run containerized Ansible content



# Content Development with Execution Environments



- ✓ Supported Tooling
- ✓ Portable
- ✓ Scalable



## Content Flow Diagram

---

# Trust Certified Content

# Automation Hub

## Trusted content source

### Customer controlled

Sync certified Ansible content from **Automation hub** to on-premises private automation hub, or download directly.

### Private content

Manage the lifecycle and internal distribution of in-house Ansible content within Private Automation hub.

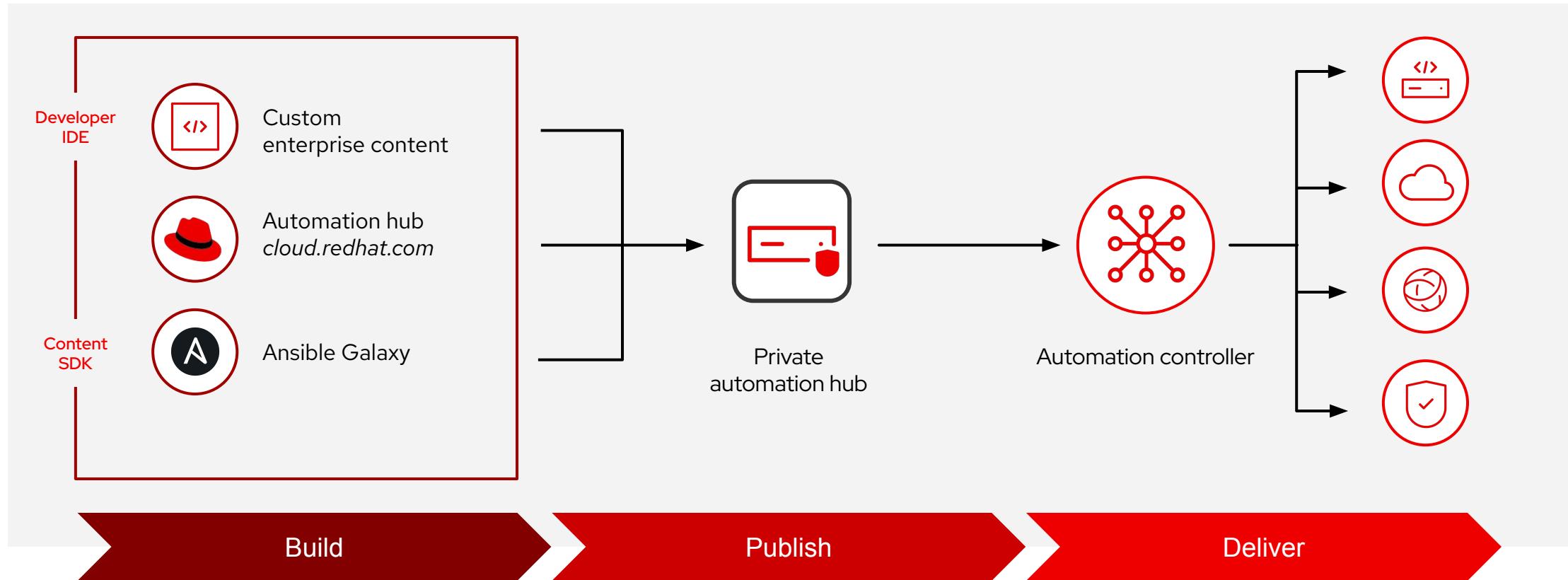
### Customizable Content Catalog

Supply internal users with approved content in one controlled location in Private Automation hub.

The screenshot shows the Red Hat Hybrid Cloud Console interface. On the left, a sidebar menu for the Ansible Automation Platform includes options like Overview, Automation Hub (selected), Collections, Partners, Repo Management, Connect to Hub, Automation Services Catalog, Insights, Reports, Savings Planner, Automation Calculator, Organization Statistics, Job Explorer, Clusters, Notifications, and Documentation. The main content area is titled 'Collections' and lists 102 items. Each item is a card with a provider logo, name, certification status (e.g., Certified), provider information, description, and statistics for Modules, Roles, and Plugins. A 'Sync' toggle switch is present for each collection. The cards are arranged in a grid format.

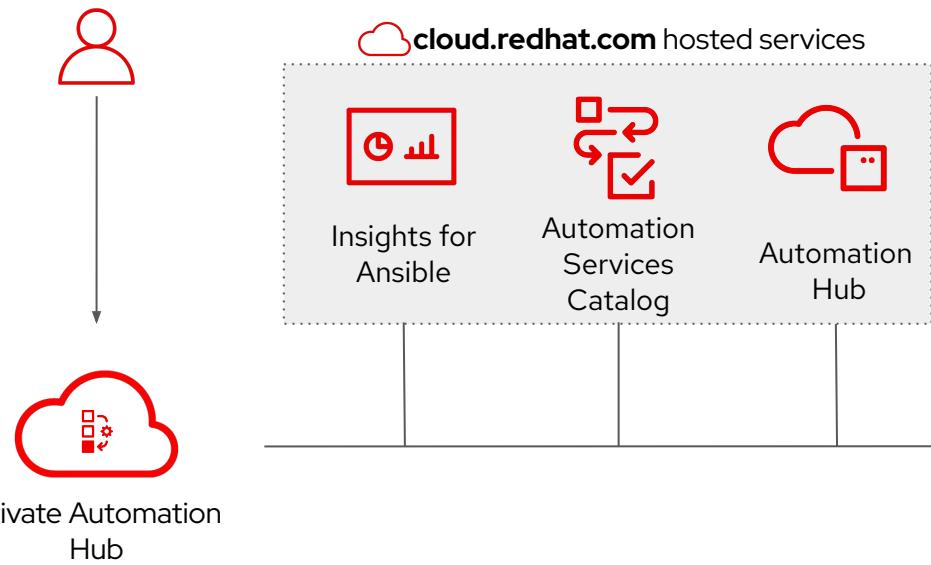
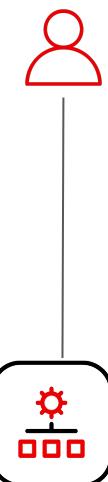
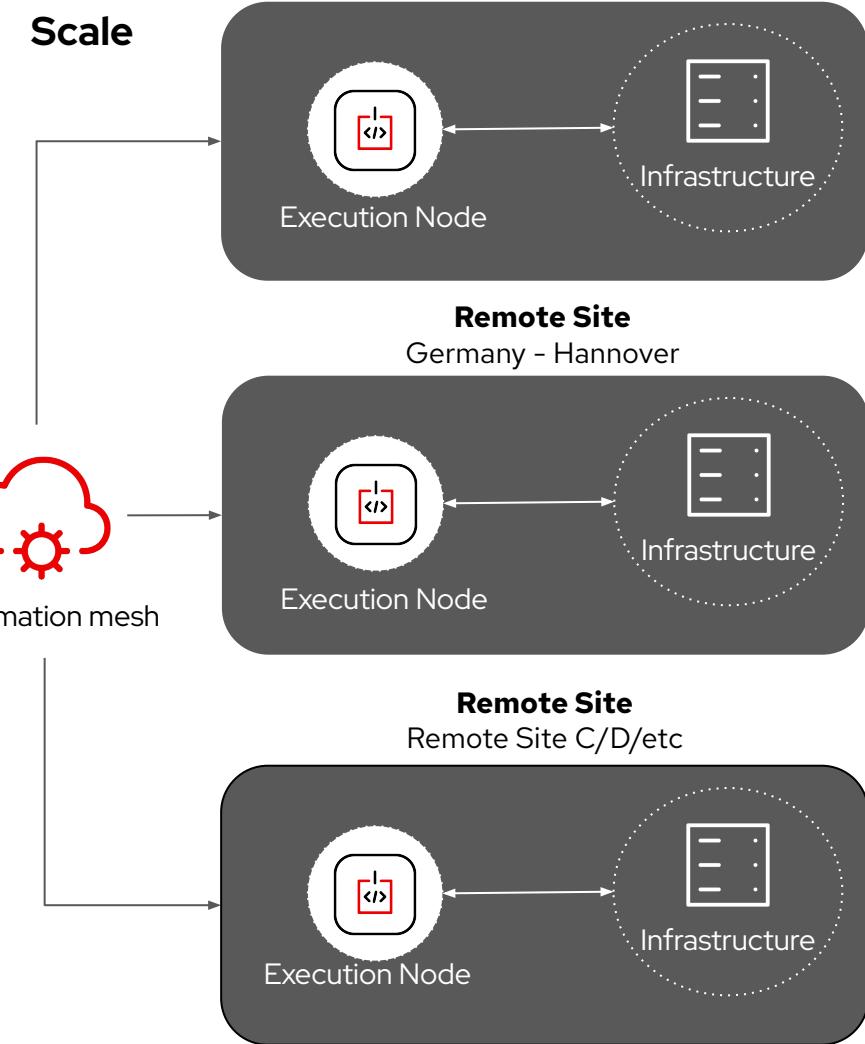
# Private Automation Hub

Value of private Automation Hub



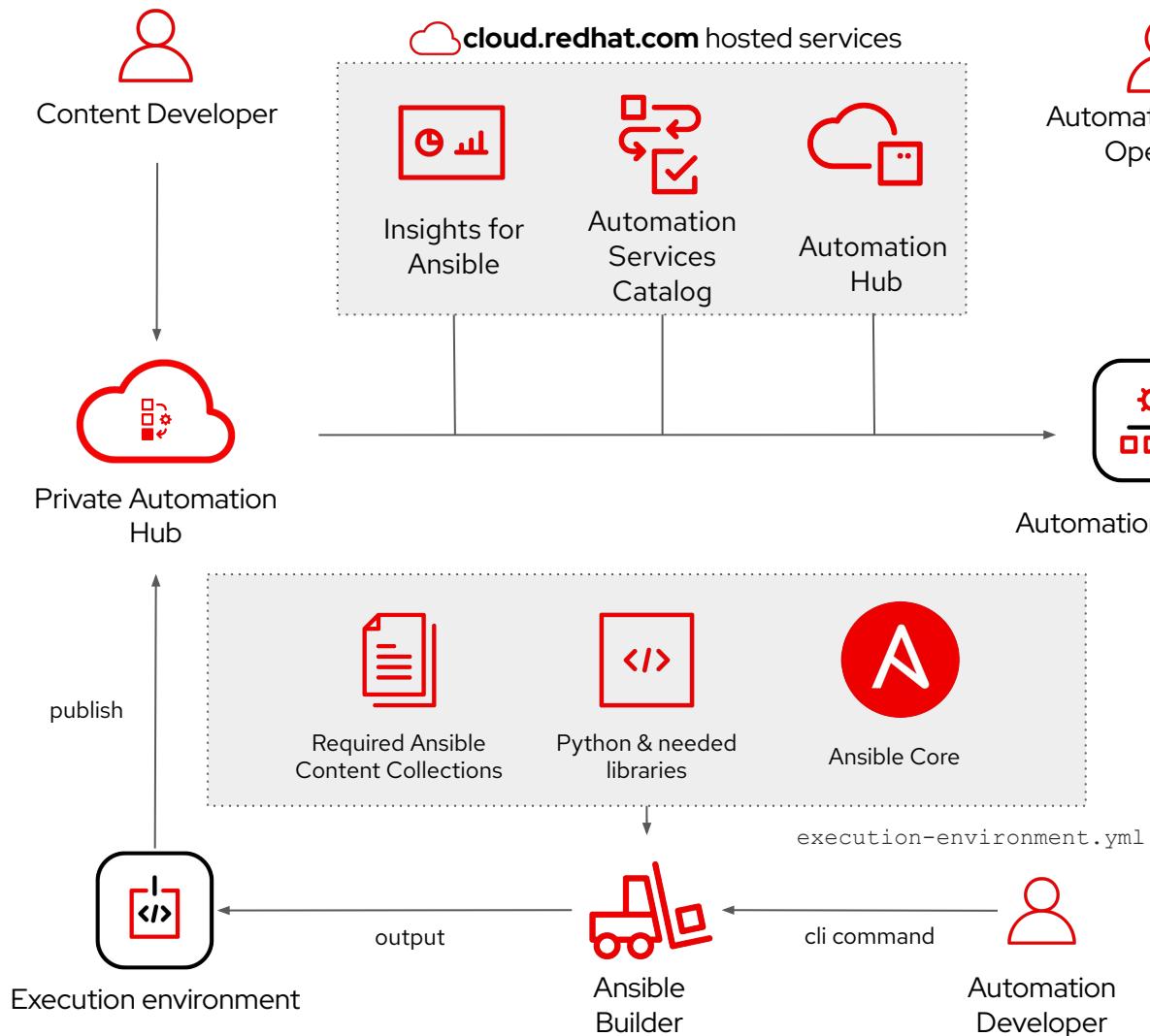
---

# The Big Picture

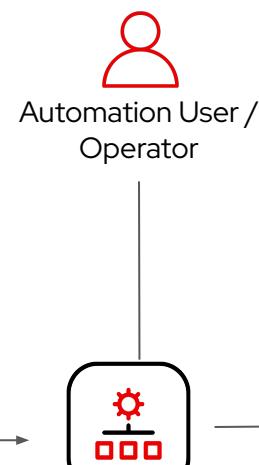
**Create****Manage****Scale**

## Ansible Automation Platform 2

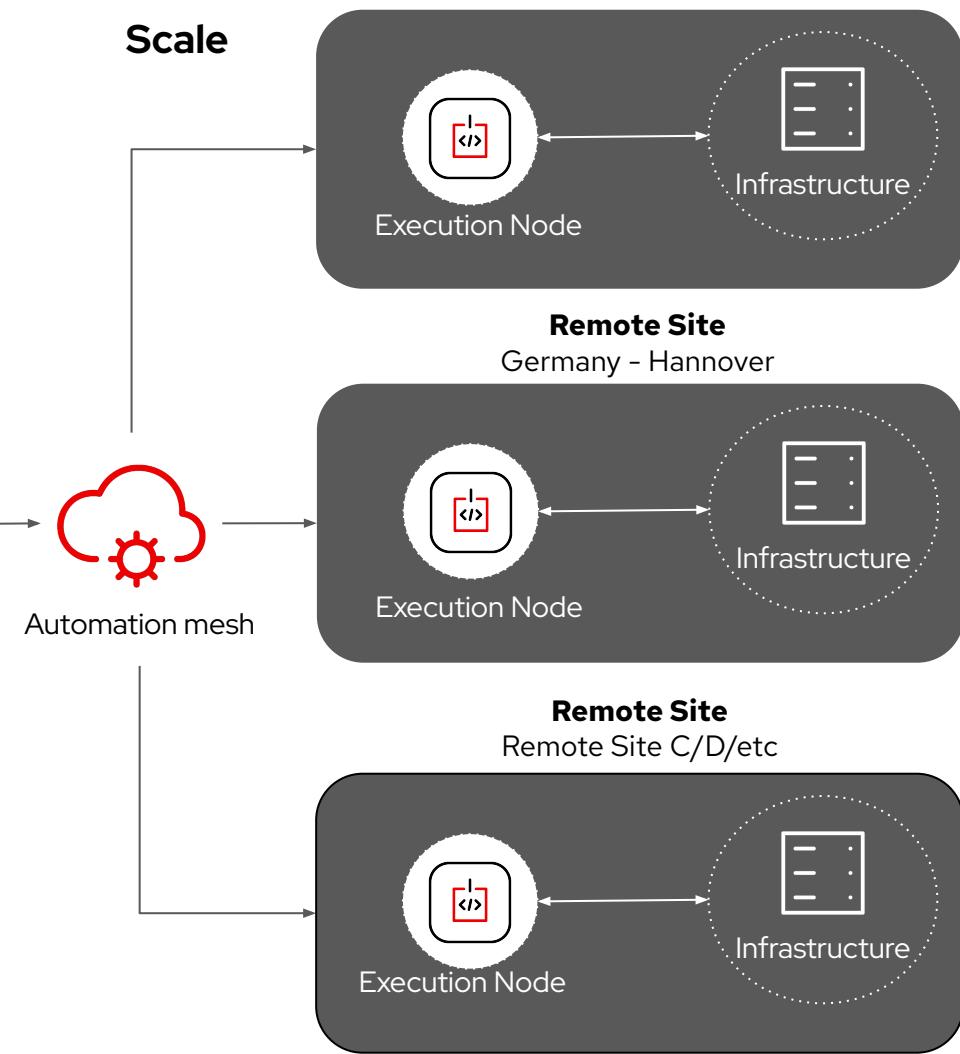
### Create



### Manage



### Scale



---

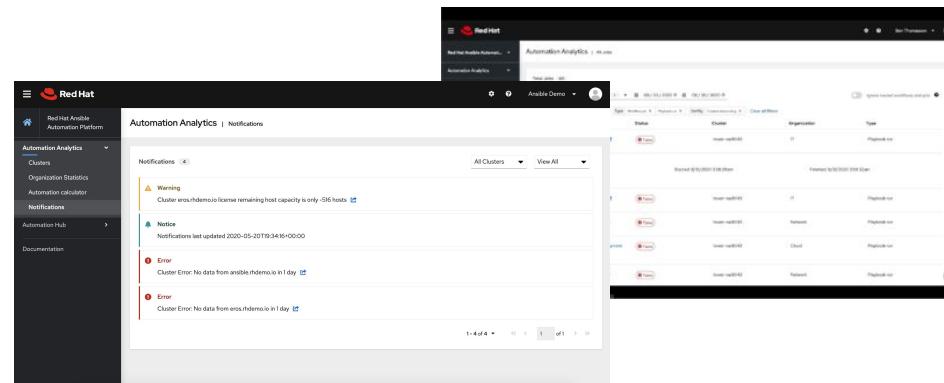
# (More) Cloud Services Make Automation Available to Everyone

# Even More Cloud Services

Bring the power of automation to your whole organization

## Red Hat Insights for Red Hat Ansible Automation Platform

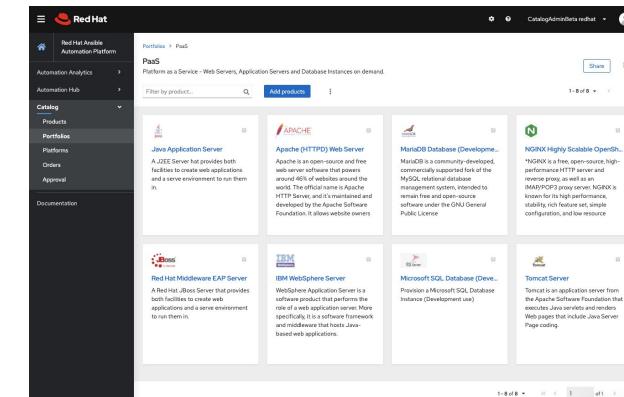
Automation Analytics with visual dashboard, health notifications, and organizational statistics. Added features Advisor, Drift & Policies for automation controllers.



The screenshot shows the Red Hat Insights interface. On the left is a dark sidebar with navigation links: Red Hat Ansible Automation Platform, Automation Analytics (selected), Clusters, Organization Statistics, Automation Calculator, Notifications (selected), Automation Hub, and Documentation. The main area is titled 'Automation Analytics' and 'Notifications'. It displays a list of notifications: a warning about a license capacity limit, a notice about notification updates, and two errors related to cluster data collection. Below the notifications is a table showing cluster status across different regions: North America, Europe, and Asia Pacific, with columns for State, Cluster, Organization, and Type.

## Automation Services Catalog

The automation services catalog is part of the Red Hat Ansible Automation Platform, giving developers and business users access to their automation. It ensures compliance and procurement requirements around automation are met.



The screenshot shows the Red Hat Automation Services Catalog. The left sidebar includes links for Portfolios, Automation Analytics, Automation Hub, Catalog (selected), Products, Portfolios, Platforms, Orders, Approval, and Documentation. The main content area is titled 'PasS' (Platform as a Service) and lists various services: Java Application Server, Apache (HTTPD) Web Server, MySQL Database (Development), and NGINX Highly Scalable OpenSource. Each service card provides a brief description and a link to more details.

---

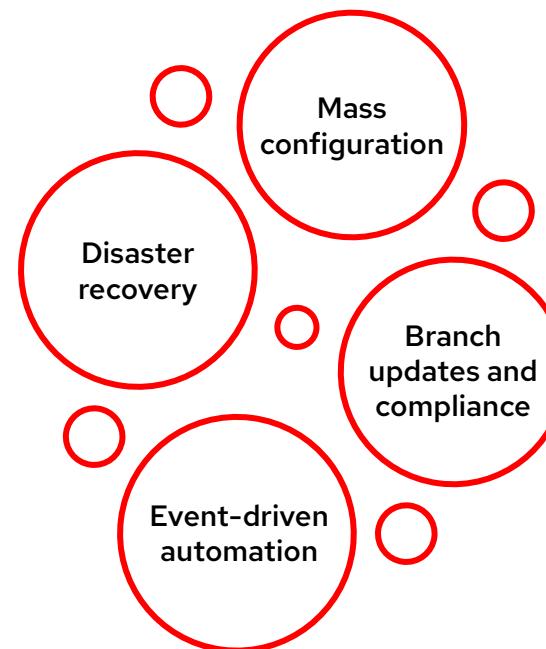
# Ansible for Edge Use Cases

# 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

## Use cases



## Example industries



Transportation and logistics



Energy



Healthcare



Telco

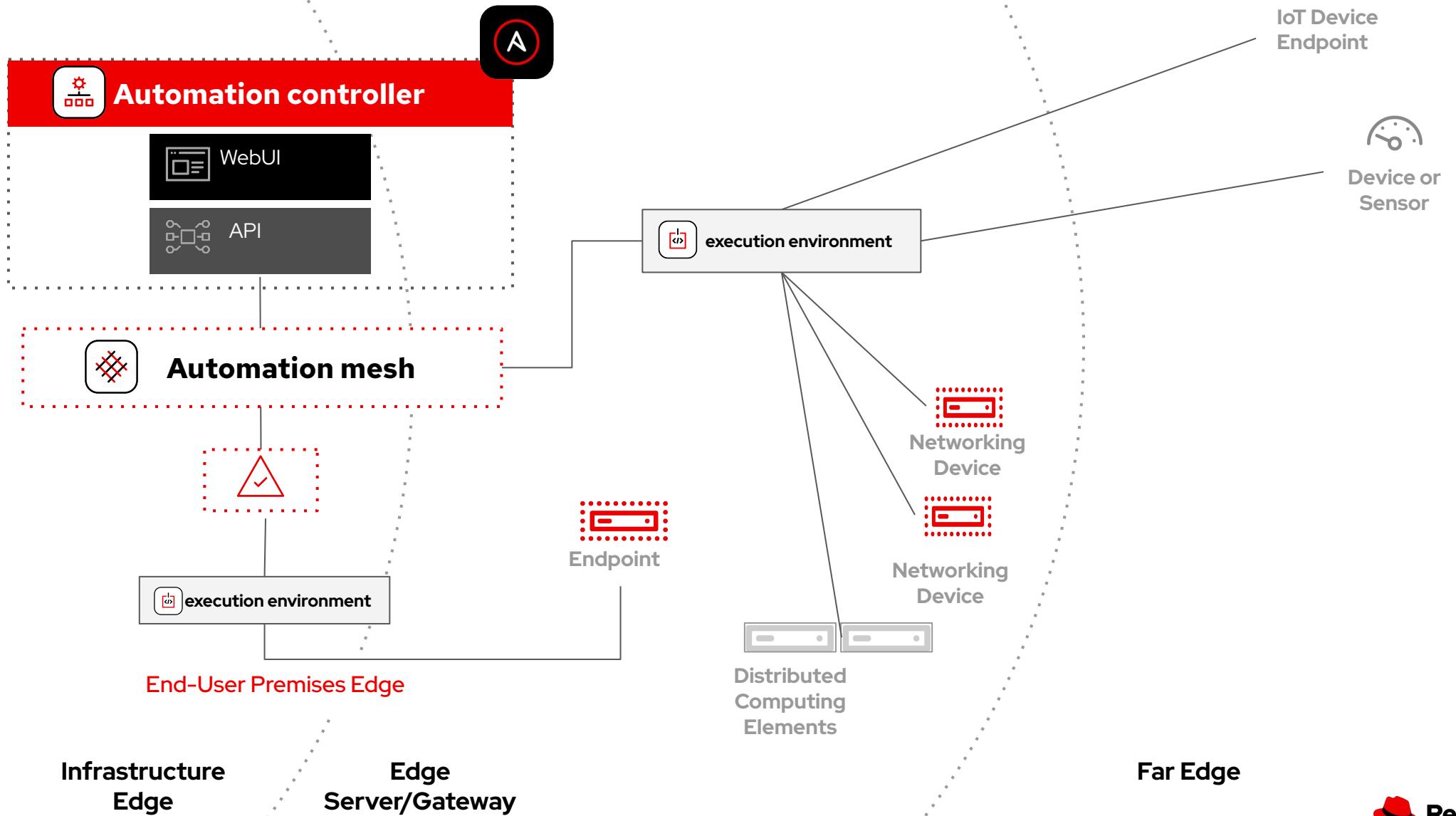


Manufacturing



Remote office

# Bring Ansible to the Edge with automation mesh



# Extending automation to the edge



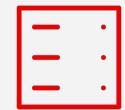
Device or  
sensor



Edge endpoint



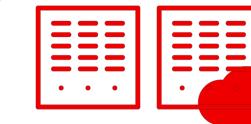
Edge gateway,  
network



Edge server



Regional  
datacenter



Core datacenter



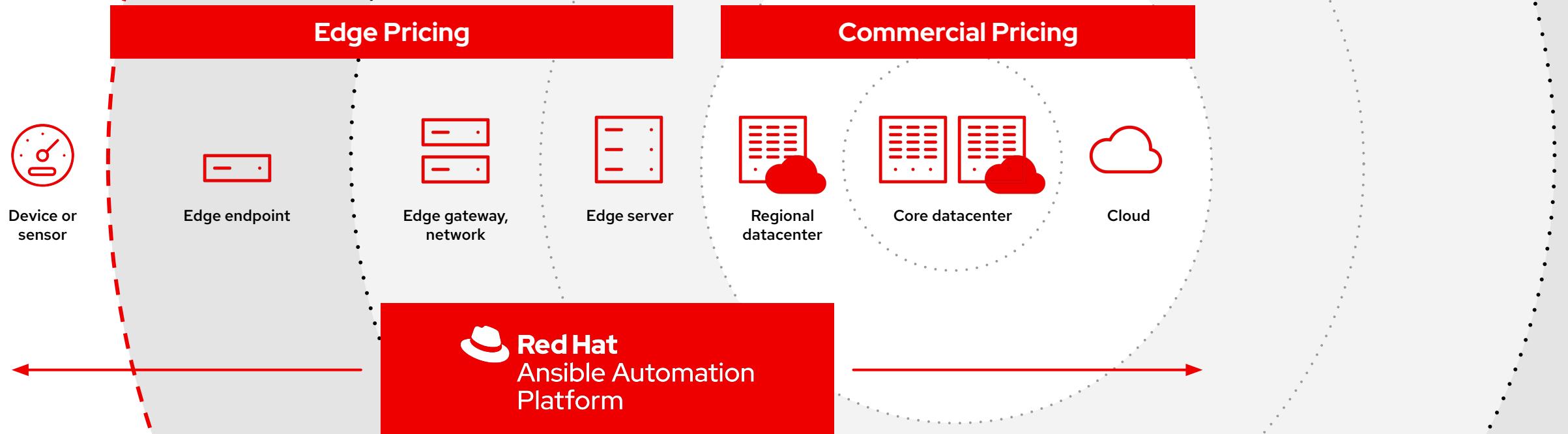
Cloud



**Red Hat**  
Ansible Automation  
Platform

Ansible for Edge

# Extending automation to the edge



# Red Hat Ansible Automation Platform

---

## Commercial SKUs

---

### Enterprise central, remote data center, and cloud

**Central:** Large purposeful equipment hosting locations

**Regional:** Smaller capacity in regional locations purposeful equipment hosting

---

---

## Edge SKUs

---

### Distributed edge

- ▶ Between datacenters and far edge remote sites
- ▶ Typically includes datacenter equipment deployed in branches, stores, or other properties that are not purpose build equipment hosting locations

### Edge endpoints

- ▶ Closest to the users or use case the device is managed
- ▶ Typically covers large numbers of common lightweight devices (radio devices, sensors, POS systems) usually single purpose

Commercial and edge framework

---

# Automation for Red Hat's Open Hybrid Cloud

# Ansible Automation Platform Operator

## Ansible Automation Platform on Red Hat OpenShift

AAP2 runs everywhere: on servers, virtual machines and cloud-native clusters running Red Hat OpenShift.

### The **Red Hat Ansible Automation Platform Operator**

- ▶ integrates Ansible Automation Platform with Red Hat OpenShift
- ▶ enables cloud-native deployment of automation clusters
- ▶ allows easier management and migration



---

# WiP: Users, Teams, Rights & Roles

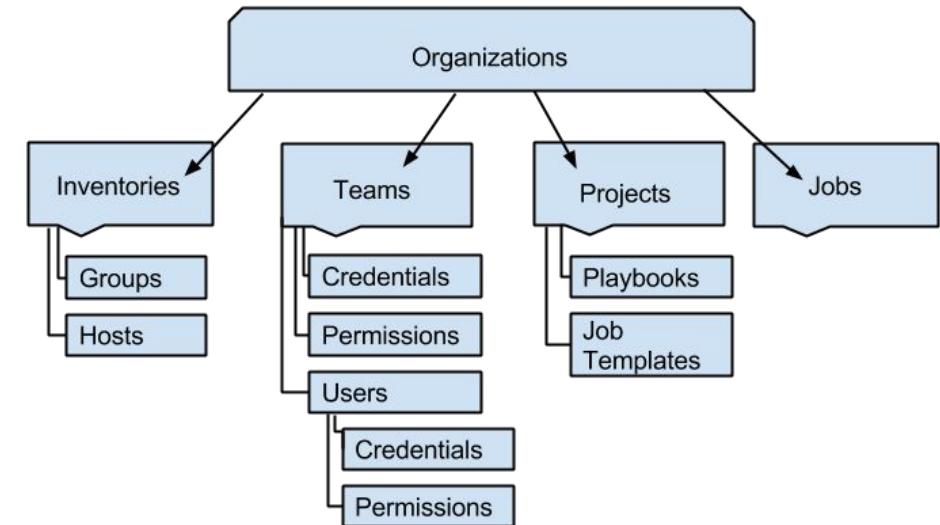
# Automation Controller Organizations

## ▶ Organizations

- logical collection of Users, Teams, Projects, Inventories
- highest level in controller's object hierarchy

## ▶ Creating Organizations

- Only the superuser/admin can
- The **Max Hosts** parameter sets an upper limit on the number of hosts.



# Role Based Access Control

- ▶ **Organizations:** Defines a tenancy that can reflect internal organizational structures.
- ▶ **Teams:** Within organizations to group users.
- ▶ **Users:** Users typically belong to teams. What the user can do controlled/defined using roles.
- ▶ **Roles:** Roles define what actions a user may perform. Roles are build-in and predefined.
- ▶ **Permissions:** Define what objects a team or user can access. Objects are:
  - Job templates, Workflow job templates, Credentials, Inventories, Projects, Organizations
  - Rights are Admin, Execute, Read

---

# **High Availability and Disaster Recovery**

# Basic Implementations

- ▶ **Backup & restore**
  - Re-installing Automation Controller and restoring the back-up using the installation script
- ▶ **Leveraging underlying infrastructure**
  - Use the virtualization management capabilities to ensure availability
  - Latency restrictions still apply during failover!

# Automate the automation

- ▶ Assumption: No data should live on Automation Controller
  - logs are forwarded to a central logging system
  - playbooks are stored in Git repositories
  - inventory sources are stored in Git repositories and/or external sources
  - users and groups are managed by enterprise authentication systems
  - credentials are stored in external credential management systems
- ▶ The only data unique to the Automation Controller are job templates, workflows, projects, inventory definitions and permissions.
- ▶ These data can be described and thus created automatically using the official Tower collection.

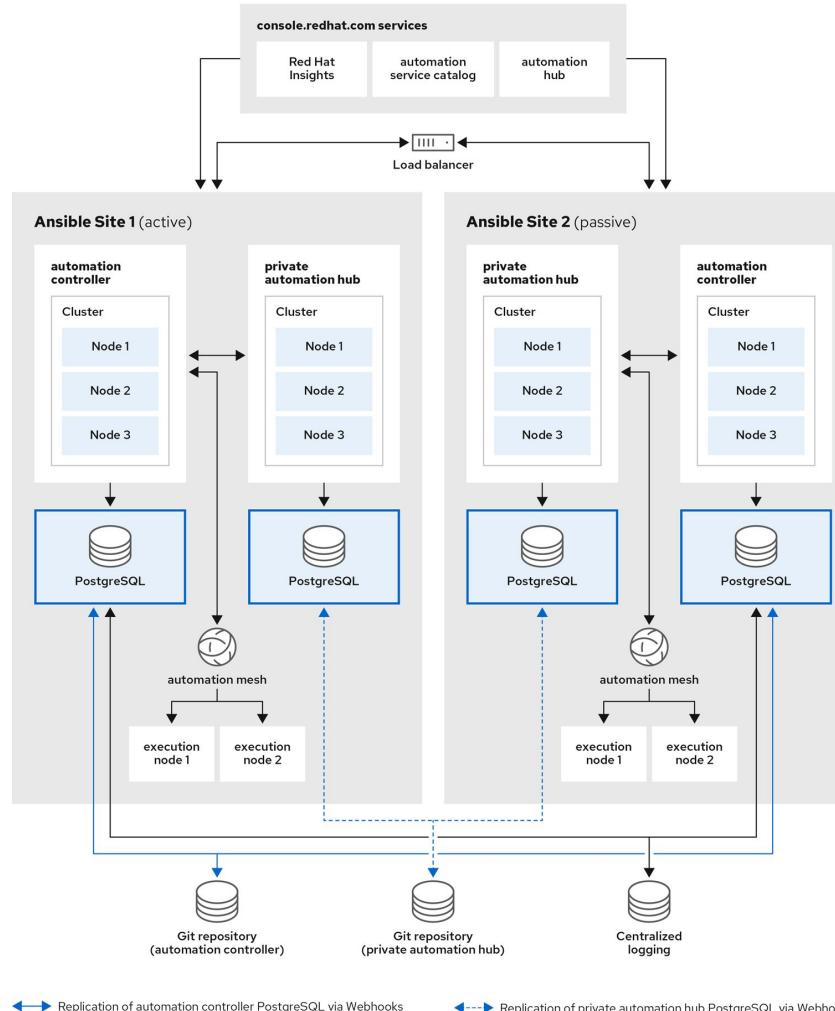
# Automate the automation

- ▶ Advantages
  - Flexible: there is no need to recover what was lost.
  - Refactoring or clean-up? Just destroy, adapt, re-create.
  - Easy to create additional clusters fully automatically. Think test cluster.
  - Restore skills are available anyway: the skills needed in the daily job using Ansible.
- ▶ Downsides
  - Cannot be applied if users are allowed to manually create their own structures.
    - As a workaround Tower collection provides export and import modules.
  - ▶ Such a setup can be restored within one hour or even less, Depending on infrastructure and complexity.

## Database failover with streaming replication

- ▶ Between two data centers to failover from data center A to B, streaming replication of the database can be used.
- ▶ Primary database is set up in data center A and a replica is set up in data center B.
- ▶ If data center A fails, the streaming link is cut, and the replica is promoted to primary and the old primary is shut down.
- ▶ At data center B, a deployment of the Tower environment is run, pointing towards the new primary database.
- ▶ Streaming replication is not covered by Red Hat support.

# New 2.1 Reference Architecture using GitOps



---

# Update Considerations

# Migrating to Red Hat Ansible Automation Platform 2

- ▶ **Running AAP 1.2 on RHEL 7?**
  - Migrate AAP 1.2 environment to RHEL 8.3+ as part of upgrade
  - Or run on OCP 4
- ▶ **Running AAP 1.1 or older?**
  - Must update to AAP 1.2 first in order to then update to AAP 2
- ▶ **Isolated nodes feature used?**
  - Automation mesh feature in AAP 2.1
- ▶ **Using ansible-playbook?**
  - We are replacing this with new tooling (builder and navigator)
  - Still available in ansible-core
- ▶ **There is more: Python venvs, collections...**
  - Must migrate all virtual environments to execution environments

# What's Next? Get Started!

## The Inside Playbook

New reference architecture: Deploying Red Hat Ansible Automation Platform 2.1

What's new: an introduction to automation mesh

Introducing Red Hat Ansible Automation Platform 2.1

Why 2022 will be the year for edge automation

Automation for the cloud: Cloud Field Day 12 recap

Automation controller workflow deployment as code

Automating execution environment image builds with GitHub Actions

The Power of AI and the Science of Operations (Part 1)

AnsibleFest 2021 - What it means for Partners

Boost your cloud-native deployments with Red Hat OpenShift

Ansible for AWS: Introduction to Spot Instance Automation

What's New: The Ansible AWS Collection 2.0 Release

Red Hat Ansible Automation Platform 2: Migration strategy considerations

VMware resource MOID lookup filter

What's new in Ansible Automation Platform 2: automation content navigator

What's new in Ansible Automation Platform 2: private automation hub

What's new in Ansible Automation Platform 2: automation controller

What's new in Ansible Automation Platform 2: automation execution environments

Introducing Ansible Automation Platform 2

- ▶ **The Inside Playbook:**

<https://www.ansible.com/blog/all>

- ▶ **Self-paced interactive hands-on labs with Ansible Automation Platform:**

<https://www.redhat.com/en/engage/redhat-ansible-automation-202108061218>

- ▶ **Ansible Workshops on RHPDS**

- ▶ **Sessions from AnsibleFest 2021!**

- ▶ **The docs, for sure...**

- ▶ **Ask your friendly Red Hat Solution Architect!**



# 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)