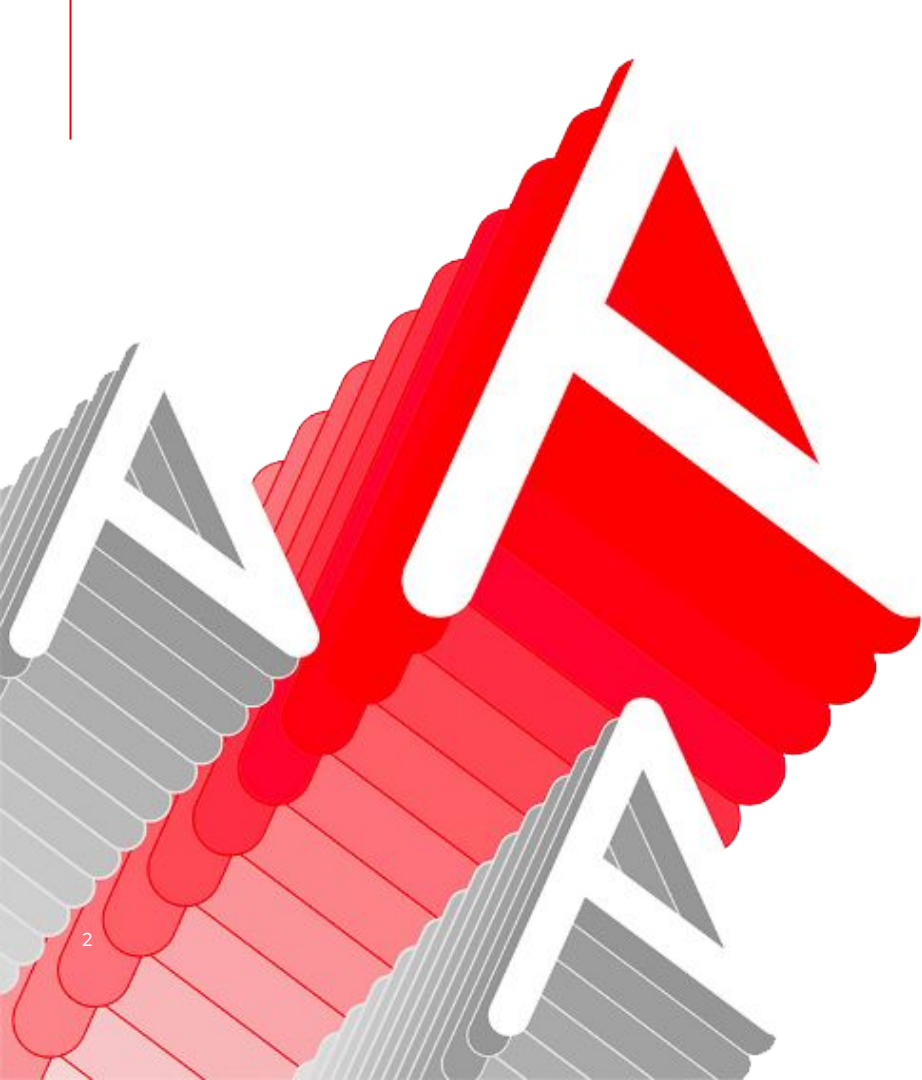


Getting started with Network Automation

Sean Cavanaugh
Technical Marketing Engineer
CCIE #40514
 @IPvSean

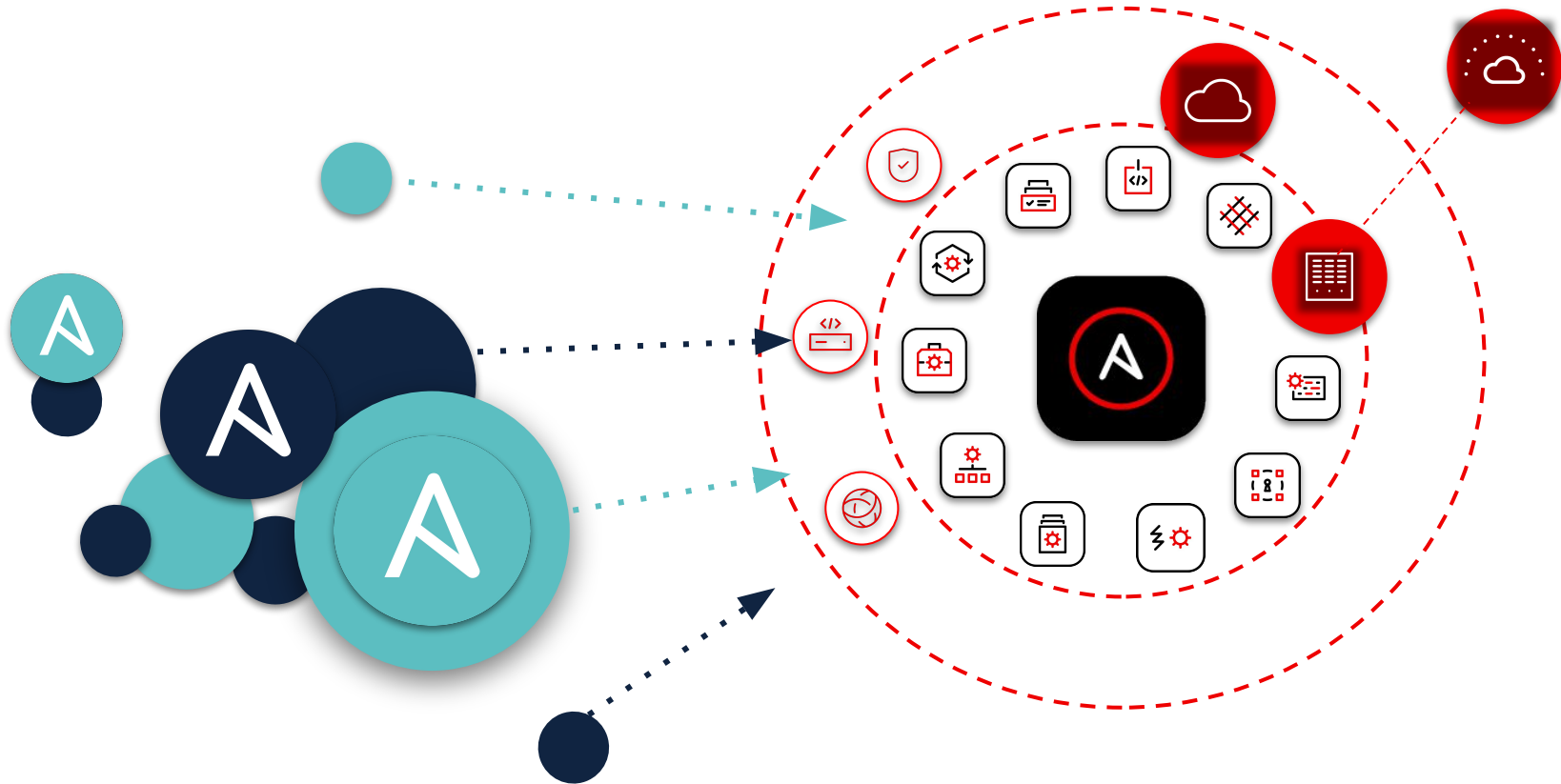


Agenda

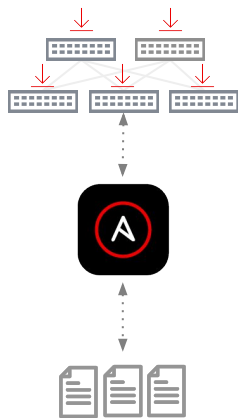
What are we even talking about?

- What is Red Hat Ansible Automation Platform?
- Automation basics
- Network automation

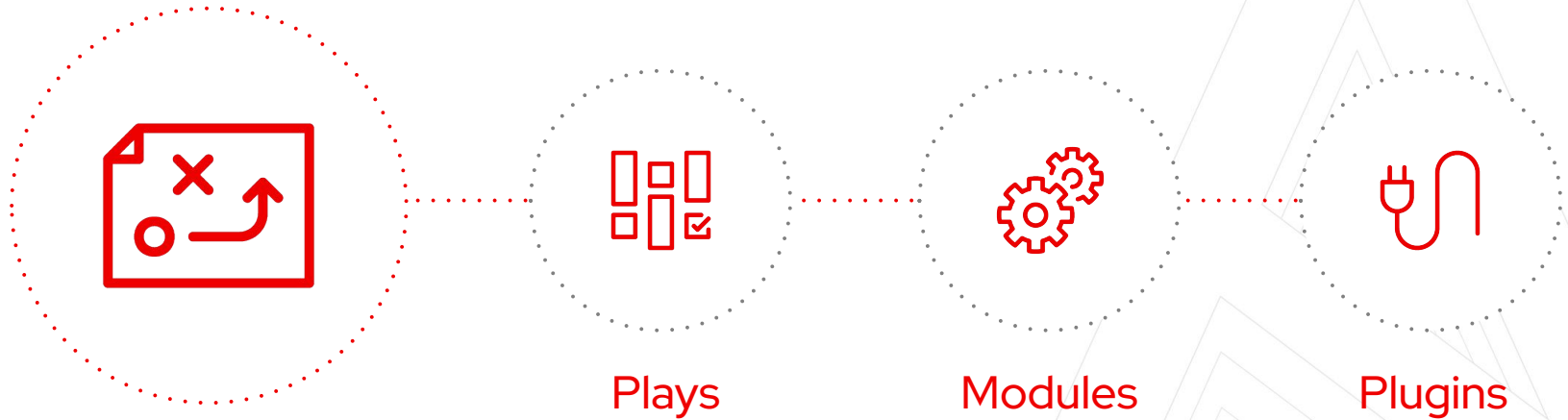




Ansible basics



What makes up an Ansible playbook?



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

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

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.

```
- name: latest index.html file ...  
  template:  
    src: files/index.html  
    dest: /var/www/html/
```

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.

Example become plugin:

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

Example filter plugins:

```
{{ some_variable | to_nice_json }}  
{{ some_variable | to_nice_yaml }}
```

Ansible Inventory

The systems that a playbook runs against



What are they?

List of systems in your infrastructure that automation is executed against

```
[web]
webserver1.example.com
webserver2.example.com

[db]
dbserver1.example.com

[switches]
leaf01.internal.com
leaf02.internal.com
```

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.

```
---  
- name: install and start apache  
  hosts: web  
  roles:  
    - common  
    - webservers
```

Collections

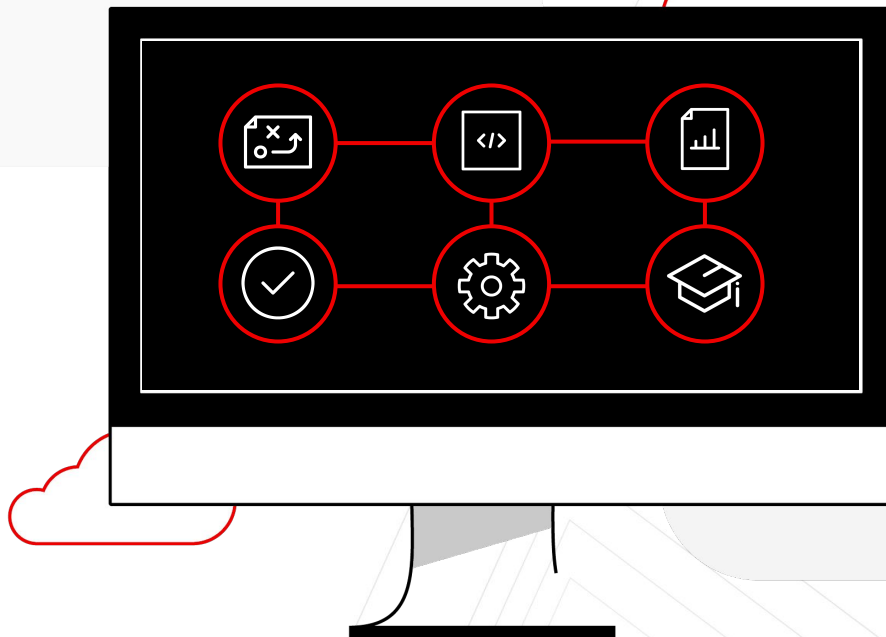
Simplified and consistent content delivery



What are they?

Collections are a directory structure containing automation content:

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





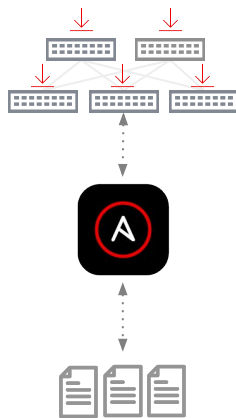
```
nginx_core
├── MANIFEST.json
├── playbooks
│   └── deploy-nginx.yml
├── ...
├── plugins
├── README.md
├── roles
│   └── nginx
│       ├── defaults
│       ├── files
│       │   └── ...
│       ├── tasks
│       ├── templates
│       └── ...
├── nginx_app_protect
└── nginx_config
```

deploy-nginx.yml

```
---
- name: Install NGINX Plus
  hosts: all
  tasks:
    - name: Install NGINX
      include_role:
        name: nginxinc.nginx
      vars:
        nginx_type: plus

    - name: Install NGINX App Protect
      include_role:
        name: nginxinc.nginx_app_protect
      vars:
        nginx_app_protect_setup_license: false
        nginx_app_protect_remove_license: false
        nginx_app_protect_install_signatures: false
```

Automating networks



- Networks as YAML

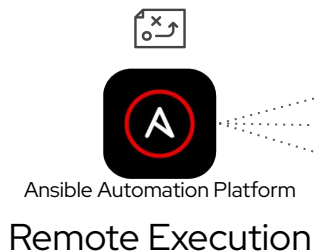
Network Automation compared to servers

Module code is
executed locally on the
control node



Network Devices /
API Endpoints

Module code is copied
to the managed node,
executed, then
removed



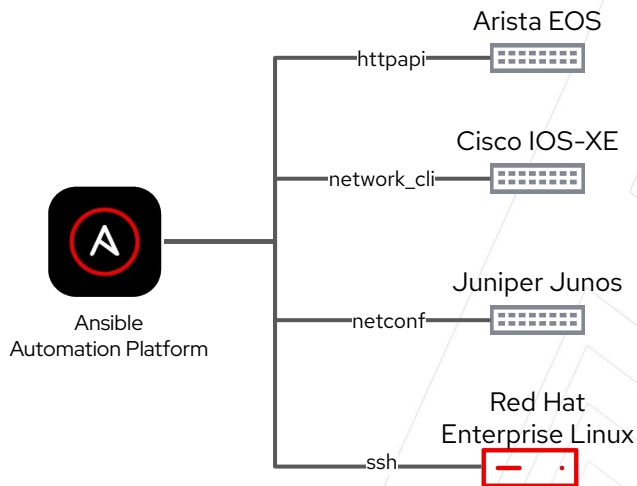
Linux / Windows
Hosts

Network Connection Plugins

Use your vendor connection of choice

ansible_connection

- › **netconf** - XML over netconf
example: Juniper Junos
- › **network_cli** - command line over SSH
example: Cisco IOS-XE, Arista EOS
- › **httpapi** - vendor API
example: Arista eAPI, Cisco NX-API



<https://docs.ansible.com/ansible/latest/plugins/connection.html>

Certified content collections for Cisco

✓ Certified

✓ Supported

✓ Jumpstart automation

Cisco ACI

Cisco ASA

Cisco DNA Center

Cisco IOS, IOS XR

Cisco Multisite Orchestrator

Cisco NX-OS

120+ Certified Content Collections for multivendor IT environments

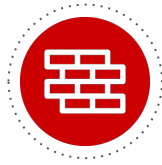
Ansible Network Ecosystem



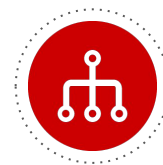
SWITCHES



ROUTERS



ENTERPRISE
FIREWALLS



LOAD
BALANCERS



CONTROLLERS



IP ADDRESS
MGMT



ARISTA



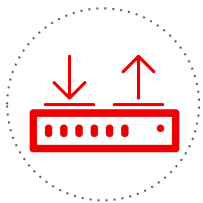
DELL EMC



VYOS

Which use cases?

Automate your network with a single tool



Configuration Accuracy

Platform agnostic configuration management to standardize and enforce best-practices.



Operational State Management

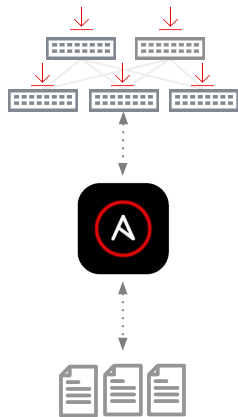
Track network state through facts gathering, to perform preventive maintenance, reducing outage risks and make data-driven decisions.



Compliance & Traceability

Enabling network governance through tracing & auditing configuration changes while keeping control of configuration & software versions through a single source of truth.

Config Backup and Restore



Why is it important?

- Read-only, no changing of production configs
- Ubiquitous use case
- Good intro level use-case for network engineers

Why Ansible Automation Platform?

- Push button via WebUI
- Easy scheduling
- Network Operating System (NOS) agnostic

Practical Example

```
---  
- name: network backup  
  hosts: all  
  gather_facts: false  
  tasks:  
    - name: multi platform backup  
      include_role:  
        name: network.toolkit.backup
```

Multiple network operating systems

Arista EOS



Cisco IOS-XE



Juniper Junos

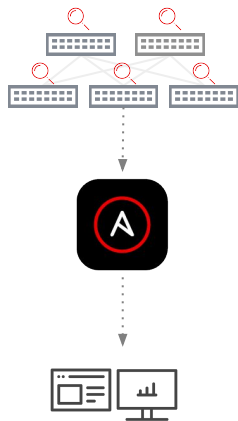


```
---
- name: backup arista configuration
  arista.eos.config:
    backup: true
  register: config_output
```

```
---
- name: backup cisco configuration
  cisco.ios.config:
    backup: true
  register: config_output
```

```
---
- name: backup juniper config
  junipernetworks.junos.config:
    backup: true
  register: config_output
```

Infrastructure Awareness



Why is it important?

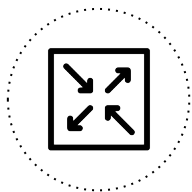
- Read-only, no changing of production configs
- Normalizes configs into structured data
- Dynamic Documentation and reporting

Why Ansible Automation Platform?

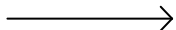
- Huge ecosystem of tools
- Standard data models across multiple platforms
- constantly expanding fact capabilities

Ansible Automation Platform facts

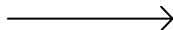
Network automation begins and ends with **facts**



Network native
configuration

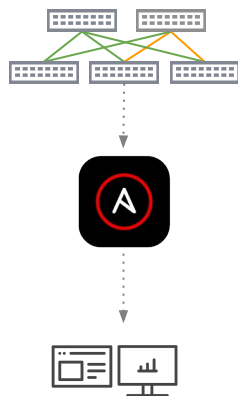


Convert to
structured data



```
"ansible_facts": {  
  "ansible_net_iostype": "IOS-XE",  
  "ansible_net_version": "16.09.02",  
  "ansible_net_serialnum": "9L8KQ482JFZ",  
  "ansible_net_model": "CSR1000V",  
  
  <<rest of output removed for brevity>>  
}
```

Operational State Validation



Why is it important?

- Parsing operational state to structured values
- Schema validation and verification
- Enhance operational workflows

Why Ansible Automation Platform?

- Rollback is built-in on a module by module basis
- Workflows allow flexible scenarios to be automated
- Operational State can be treated just like Ansible Facts

Operational state versus config

What is configured, versus what the network state is

```
# sh run int Ethernet1/1
interface Ethernet1/1
  no switchport
  ip address 10.101.101.2/24
  ip ospf network point-to-point
  ip ospf area 0.0.0.0
```

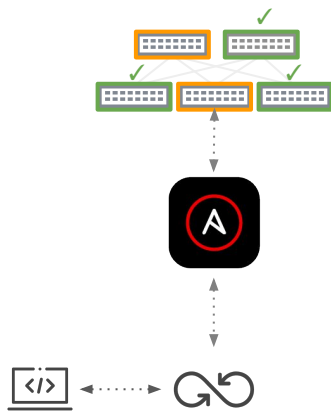
OSPF Interface Configuration

```
# show ip ospf neigh
```

Neighbor ID	Instance	VRF	Pri	State	Dead Time	Address	Interface
192.168.4.4	1	default	0	FULL	00:00:31	10.101.101.4	Ethernet1/1

OSPF Operational State

Automated NetOps



Why is it important?

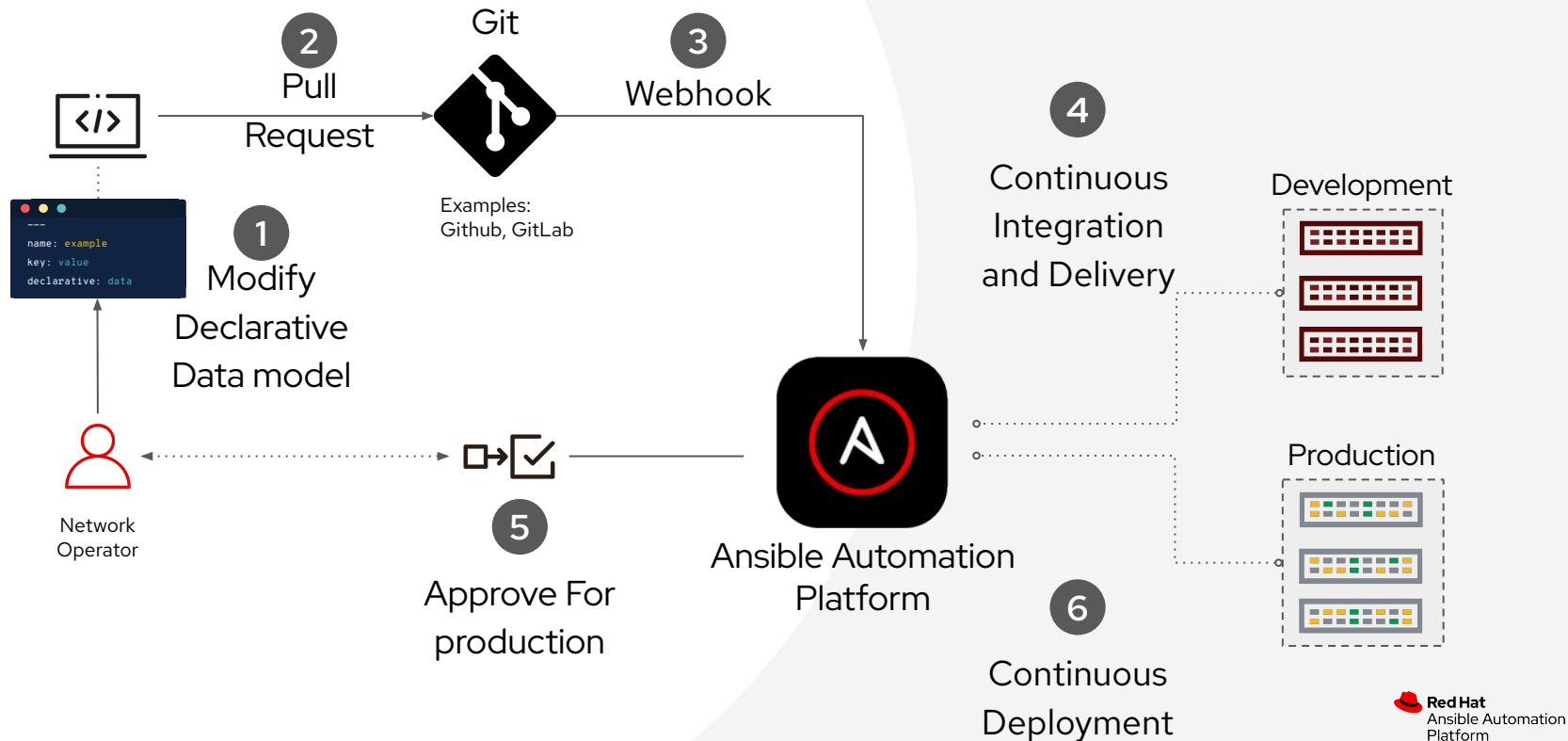
- Data centric automation , infrastructure as code
- Deploy configuration pipelines
- GitOps for Network Automation

Why Ansible Automation Platform?

- Native Webhooks for Github/Gitlab
- Brownfield is easy with network resource modules
- Not all or nothing, can be adopted in incremental steps

Automated NetOps

Tying everything together



Learning resources

Continue your automation journey with Red Hat® Ansible® Network Automation



Red Hat
Ansible Automation
Platform

ansible.com/network-automation



Video: Five use cases for network automation

http://red.ht/5_use_cases



Networking workshop

<https://aap2.demoredhat.com>



E-Books

Part 1: Modernize Your Network with Red Hat

red.ht/network-book1

Part 2: Automate Your Network with Red Hat

red.ht/network-book2



Red Hat Certification

Ansible for Network Automation (DO457)

red.ht/network-training

Ansible Fest 2022

October 18-19th

For more information visit ansible.com/ansiblefest

Q+A

Learn more:

red.ht/Ansible_Cisco

Thank you!



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



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

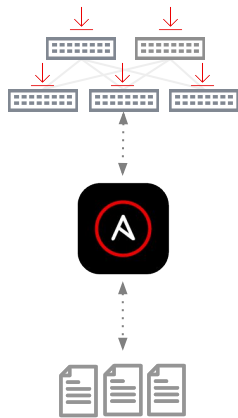


[facebook.com/redhatinc](https://www.facebook.com/redhatinc)

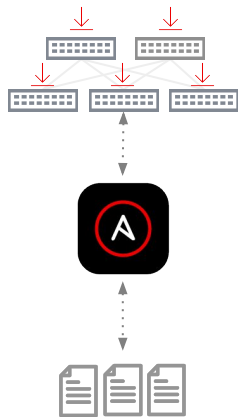


twitter.com/RedHat

Backup slides



Config Backup and Restore



Why is it important?

- Read-only, no changing of production configs
- Ubiquitous use case
- Good intro level use-case for network engineers

Why Ansible Automation Platform?

- Push button via WebUI
- Easy scheduling
- Network Operating System (NOS) agnostic

Practical Example

```
---  
- name: network backup  
  hosts: all  
  gather_facts: false  
  tasks:  
    - name: multi platform backup  
      include_role:  
        name: network.toolkit.backup
```

Multiple network operating systems

Arista EOS



Cisco IOS-XE



Juniper Junos

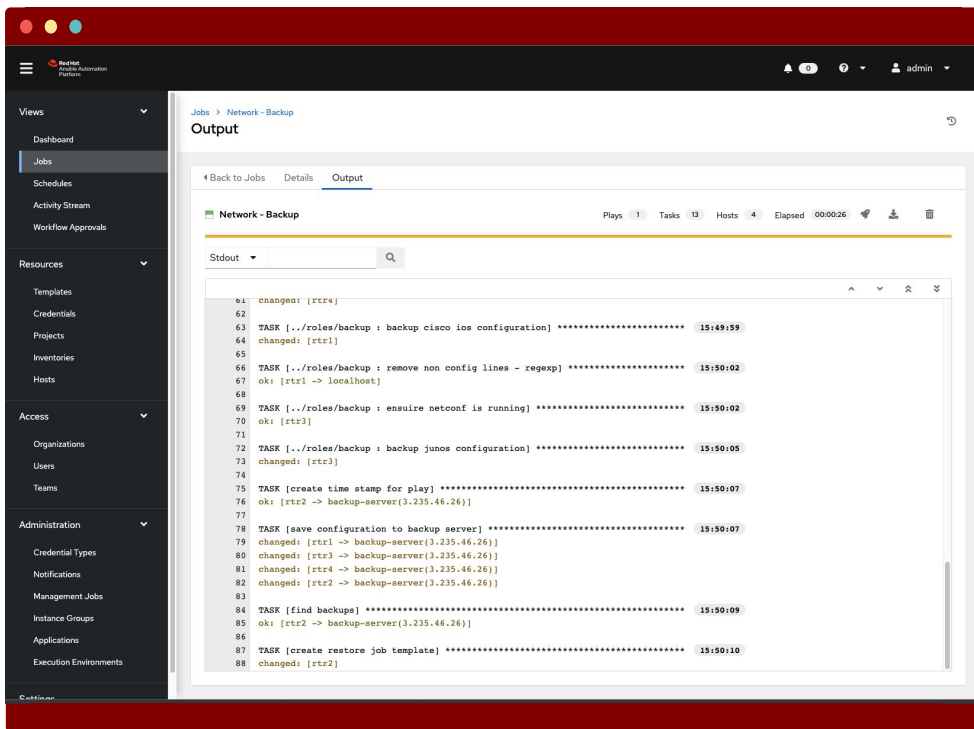


```
---
- name: backup arista configuration
  arista.eos.config:
    backup: true
    register: config_output
```

```
---
- name: backup cisco configuration
  cisco.ios.config:
    backup: true
    register: config_output
```

```
---
- name: backup juniper config
  junipernetworks.junos.config:
    backup: true
    register: config_output
```

Push button Network-IT



- Kick off automation jobs with Automation controller
- Programmatic API allows other tools to call network automation
- Sync with enterprise authentication TACACS+, RADIUS, SAML

Elevate Tasks via Surveys

- No network platform specific knowledge required!
- Automate the routine and boring tasks away
- Empower novices to take on more activities with guardrails

Launch | Network Automation - Restore

1 Survey

2 Preview

Rollback to which date? ⓘ

2022-01-27-20-50

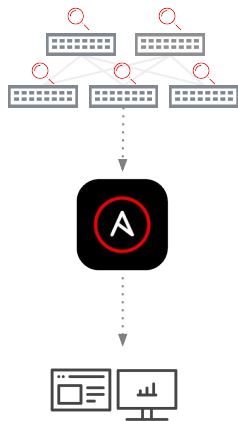
2022-01-27-20-50

2022-01-27-20-53

Next Back Cancel

Network-User Job Template

Infrastructure Awareness



Why is it important?

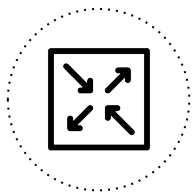
- Read-only, no changing of production configs
- Normalizes configs into structured data
- Dynamic Documentation and reporting

Why Ansible Automation Platform?

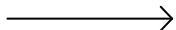
- Huge ecosystem of tools
- Standard data models across multiple platforms
- constantly expanding fact capabilities

Ansible Automation Platform facts

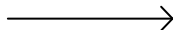
Network automation begins and ends with **facts**



Network native
configuration



Convert to
structured data



```
"ansible_facts": {  
  "ansible_net_iostype": "IOS-XE",  
  "ansible_net_version": "16.09.02",  
  "ansible_net_serialnum": "9L8KQ482JFZ",  
  "ansible_net_model": "CSR1000V",
```

```
<<rest of output removed for brevity>>
```

Data output is flexible

Create customized network reports

```
ansible_facts:
  ansible_net_api: cliconf
  ansible_net_fqdn: rtr2
  ansible_net_gather_network_resources:
    - interfaces
  ansible_net_gather_subset:
    - default
  ansible_net_hostname: rtr2
  ansible_net_image: flash:EOS.swi
  ansible_net_model: vEOS
  ansible_net_python_version: 2.7.5
  ansible_net_serialnum:
D00E130991A37B49F970714D8CCF7FCB
  ansible_net_system: eos
  ansible_net_version: 4.22.0F
  ansible_network_resources:
    interfaces:
      - enabled: true
        name: Ethernet1
      - enabled: true
        name: Loopback0
<<rest of output removed for slide
```



Ansible Automation
Platform

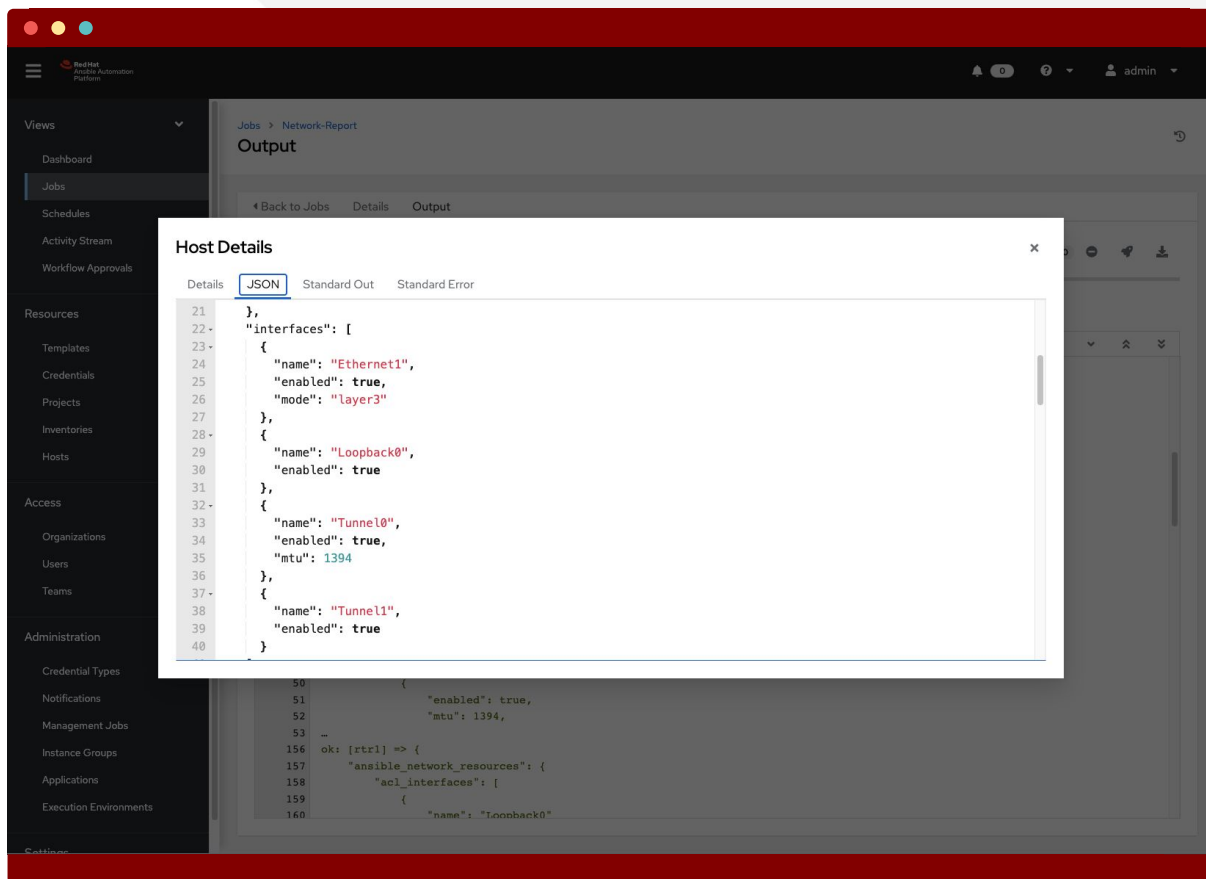


Customized
Report

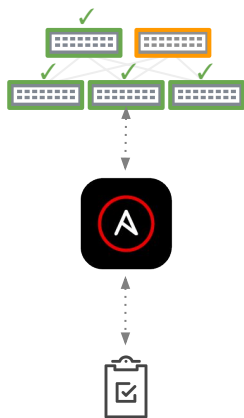
Automation Controller

Networks as code

- ▶ Network resource modules convert networks into JSON/YAML which is malleable
- ▶ Automation controller lets users see the data models, for easy troubleshooting.
- ▶ Send data to your tool of choice or simply create some simple dynamic documentation



Scoped Config Management



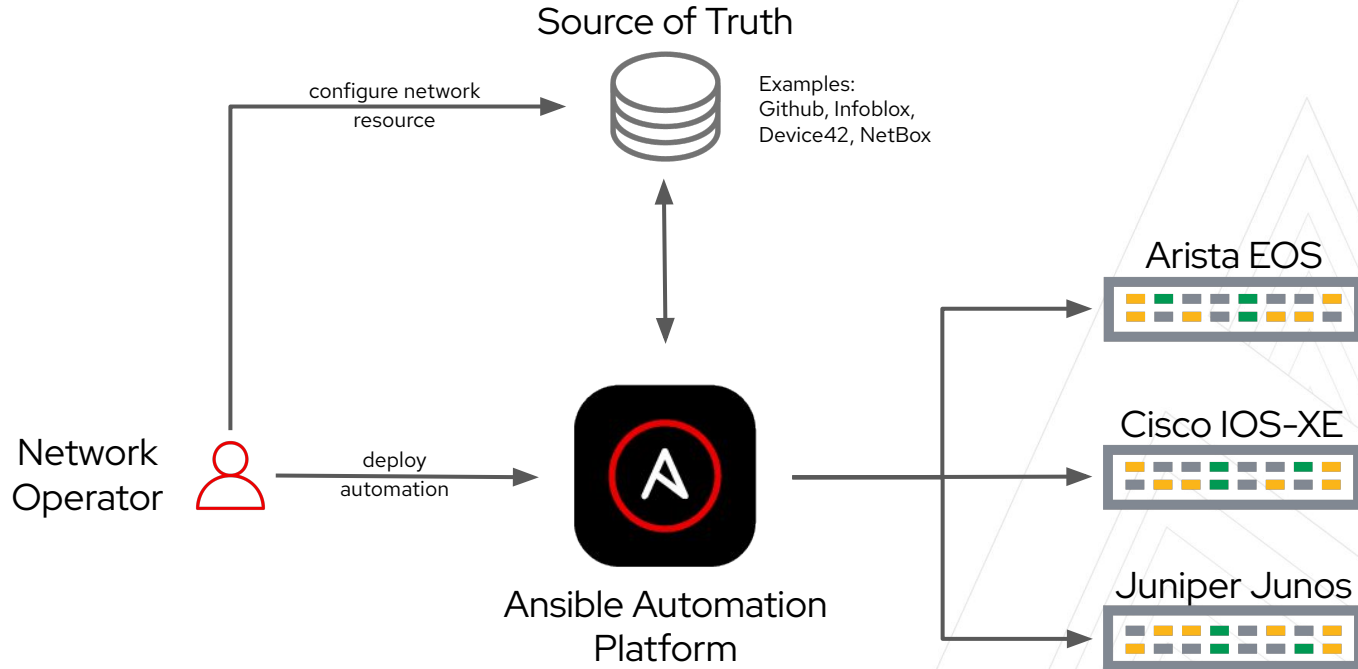
Why is it important?

- Automate VLANs, ACLs and SNMP config
- Introduce source of truth concepts
- Enforce Configuration policy

Why Ansible Automation Platform?

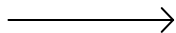
- Resource Modules provide easy automation journey
- No complex templating or coding requirements
- Simple native data models

Scoped Config Management

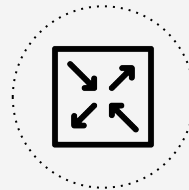
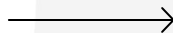


Manage specific network resources

```
vlan:  
- name: desktops  
  vlan_id: 20  
- name: servers  
  vlan_id: 30  
- name: printers  
  vlan_id: 40  
- name: DMZ  
  vlan_id: 50
```



Resource
module



Network native
configuration

Managing device state

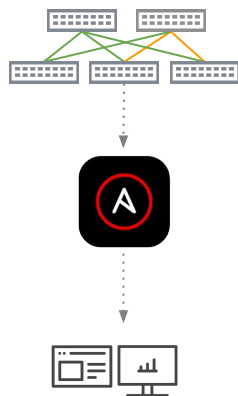
Practical examples of using network resource modules

```
vlan:  
- name: desktops  
  vlan_id: 20  
- name: servers  
  vlan_id: 30  
- name: printers  
  vlan_id: 40  
- name: DMZ  
  vlan_id: 50
```

```
- name: add VLAN configuration  
  arista.eos.vlan:  
    config: "{{ vls }}"  
    state: merged
```

State:	Merged	- add/increment
	Replaced	- template/diff
	Overridden	- force/policy
	Deleted	- destroy/remediate

Operational State Validation



Why is it important?

- Parsing operational state to structured values
- Schema validation and verification
- Enhance operational workflows

Why Ansible Automation Platform?

- Rollback is built-in on a module by module basis
- Workflows allow flexible scenarios to be automated
- Operational State can be treated just like Ansible Facts

Operational state versus config

What is configured, versus what the network state is

```
# sh run int Ethernet1/1
interface Ethernet1/1
  no switchport
  ip address 10.101.101.2/24
  ip ospf network point-to-point
  ip ospf area 0.0.0.0
```

OSPF Interface Configuration

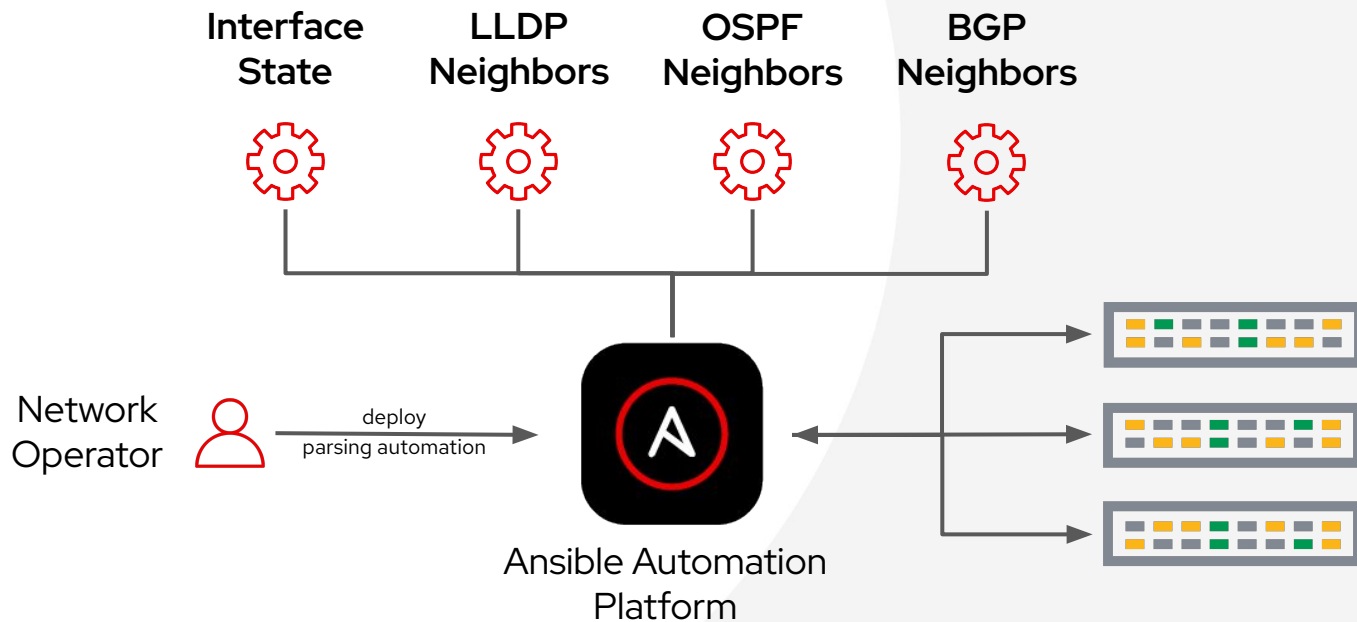
```
# show ip ospf neigh
```

Neighbor ID	Instance	VRF	Pri	State	Dead Time	Address	Interface
192.168.4.4	1	default	0	FULL	00:00:31	10.101.101.4	Ethernet1/1

OSPF Operational State

What should I parse?

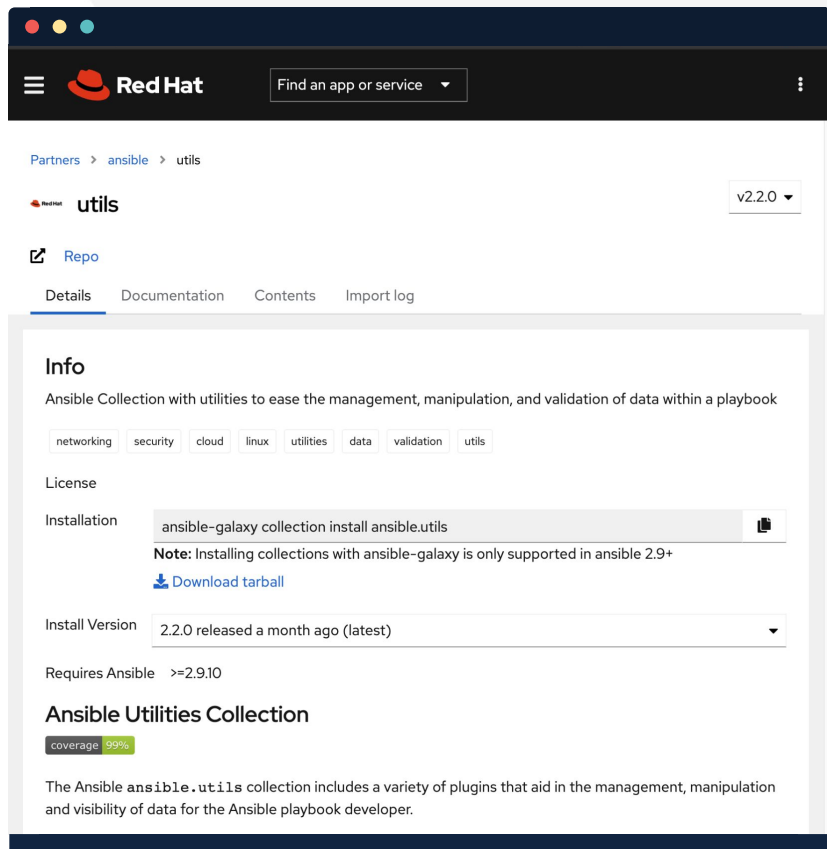
Quick automation victories for operational state



Ansible Utils

Tool chest for playbook authors

- ▶ An Ansible Collection with utilities to ease the management, manipulation, and validation of data within a playbook
- ▶ Fully supported in Automation Hub on <https://cloud.redhat.com>
- ▶ Included in Ansible Network Meta Collection `ansible.network`



Parsing Example

parsing using native Ansible parsing library

Network Device output

```
# show interfaces
Ethernet1 is up, line protocol is up (connected)
  Hardware is Ethernet, address is 022e.dbe8.1375 (bia
022e.dbe8.1375)
  Internet address is 172.18.104.95/16
  Broadcast address is 255.255.255.255
  Address determined by DHCP
  IP MTU 1500 bytes , BW 1000000 kbit
  Full-duplex, 1Gb/s, auto negotiation: on, uni-link: n/a
  Up 10 hours, 51 minutes, 55 seconds
  Loopback Mode : None
  3 link status changes since last clear
  Last clearing of "show interface" counters never
  5 minutes input rate 950 bps (0.0% with framing
overhead), 1 packets/sec
  5 minutes output rate 858 bps (0.0% with framing
overhead), 1 packets/sec
    19361 packets input, 2964452 bytes
    Received 0 broadcasts, 0 multicast
    0 runts, 0 giants
<rest of output removed for brevity>
```

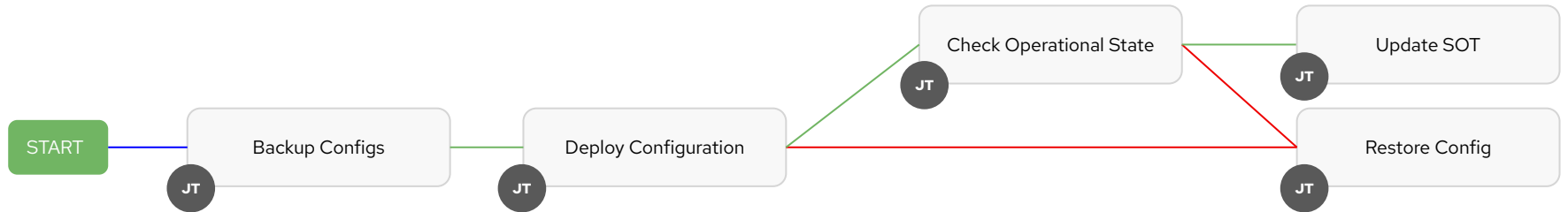
Parsed Data

```
result["parsed"]:
  Ethernet1:
    hardware: Ethernet
    mac_address: 022e.dbe8.1375
    state:
      operating: up
      protocol: up
  Loopback0:
    hardware: Loopback
    state:
      operating: up
      protocol: up
  Tunnel10:
    hardware: Tunnel
    mac_address: ac12.685f.0800
    state:
      operating: up
      protocol: up
```

Ansible Automation Platform

Using workflows to enhance your automation

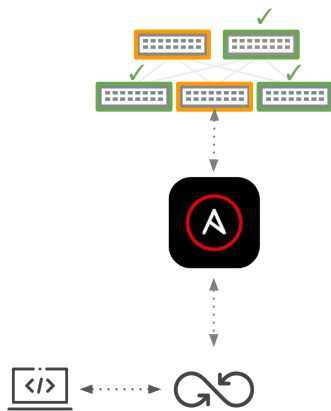
WORKFLOW VISUALIZER | Operational State Workflow



CLOSE

SAVE

Automated NetOps



Why is it important?

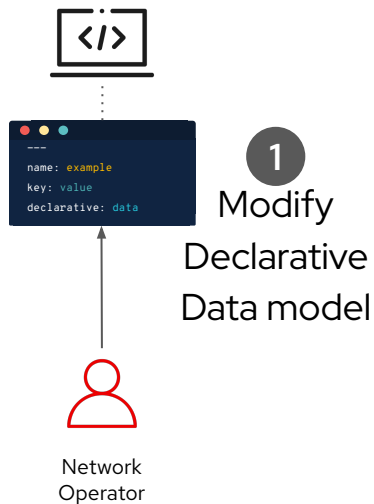
- Data centric automation , infrastructure as code
- Deploy configuration pipelines
- GitOps for Network Automation

Why Ansible Automation Platform?

- Native Webhooks for Github/Gitlab
- Brownfield is easy with network resource modules
- Not all or nothing, can be adopted in incremental steps

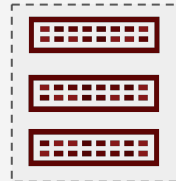
Automated NetOps

Tying everything together

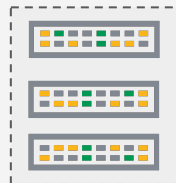


Ansible Automation
Platform

Development

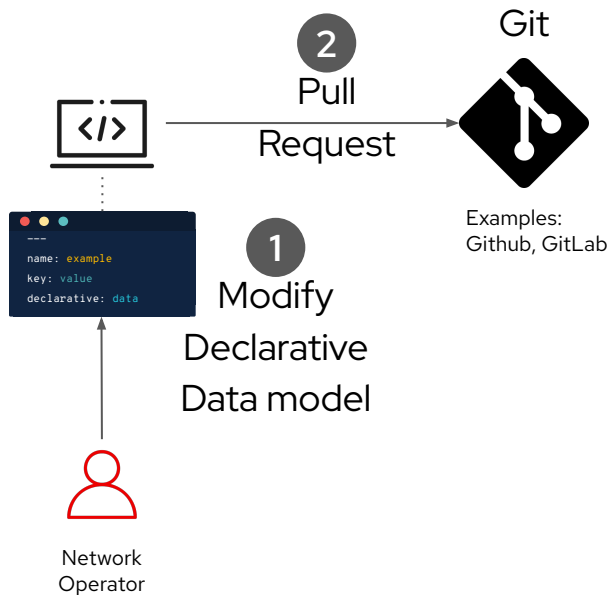


Production

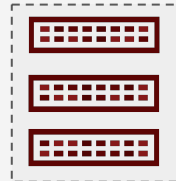


Automated NetOps

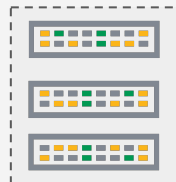
Tying everything together



Development

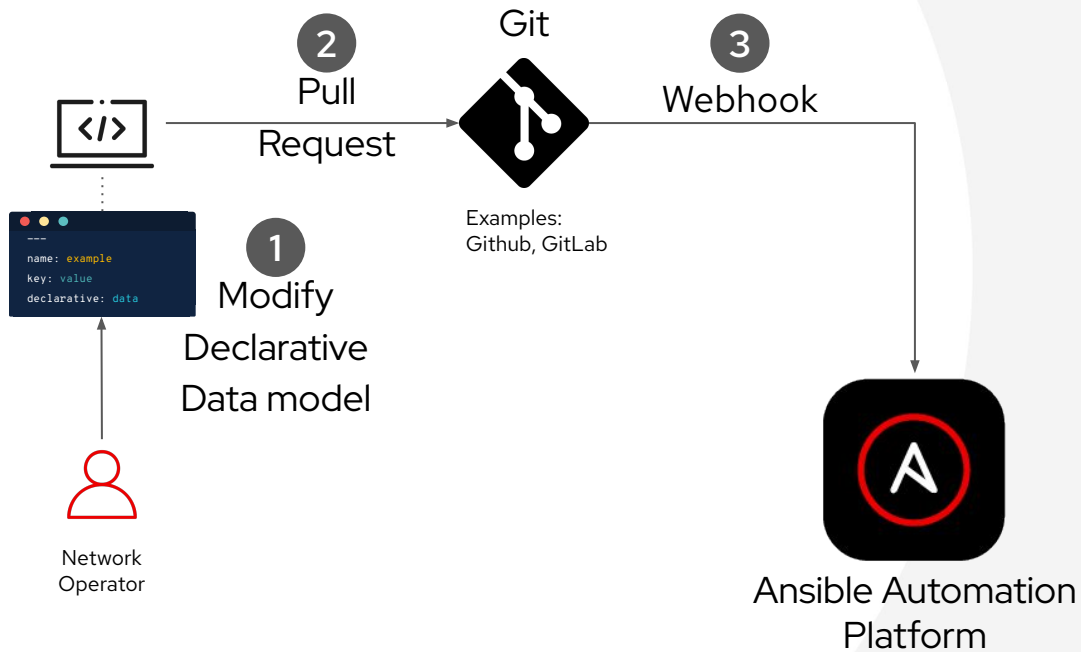


Production

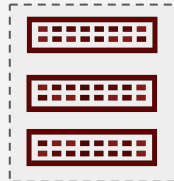


Automated NetOps

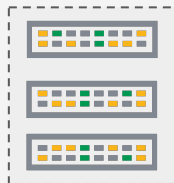
Tying everything together



Development

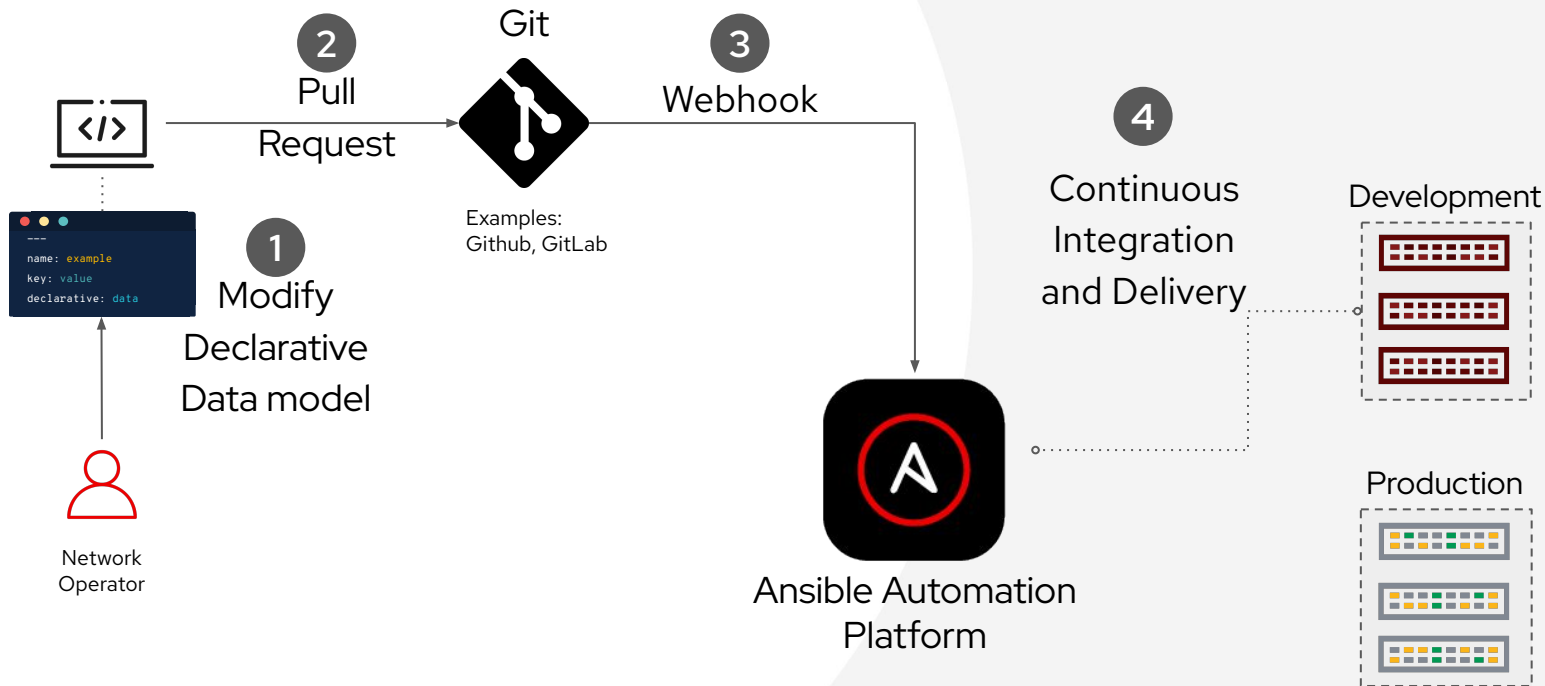


Production



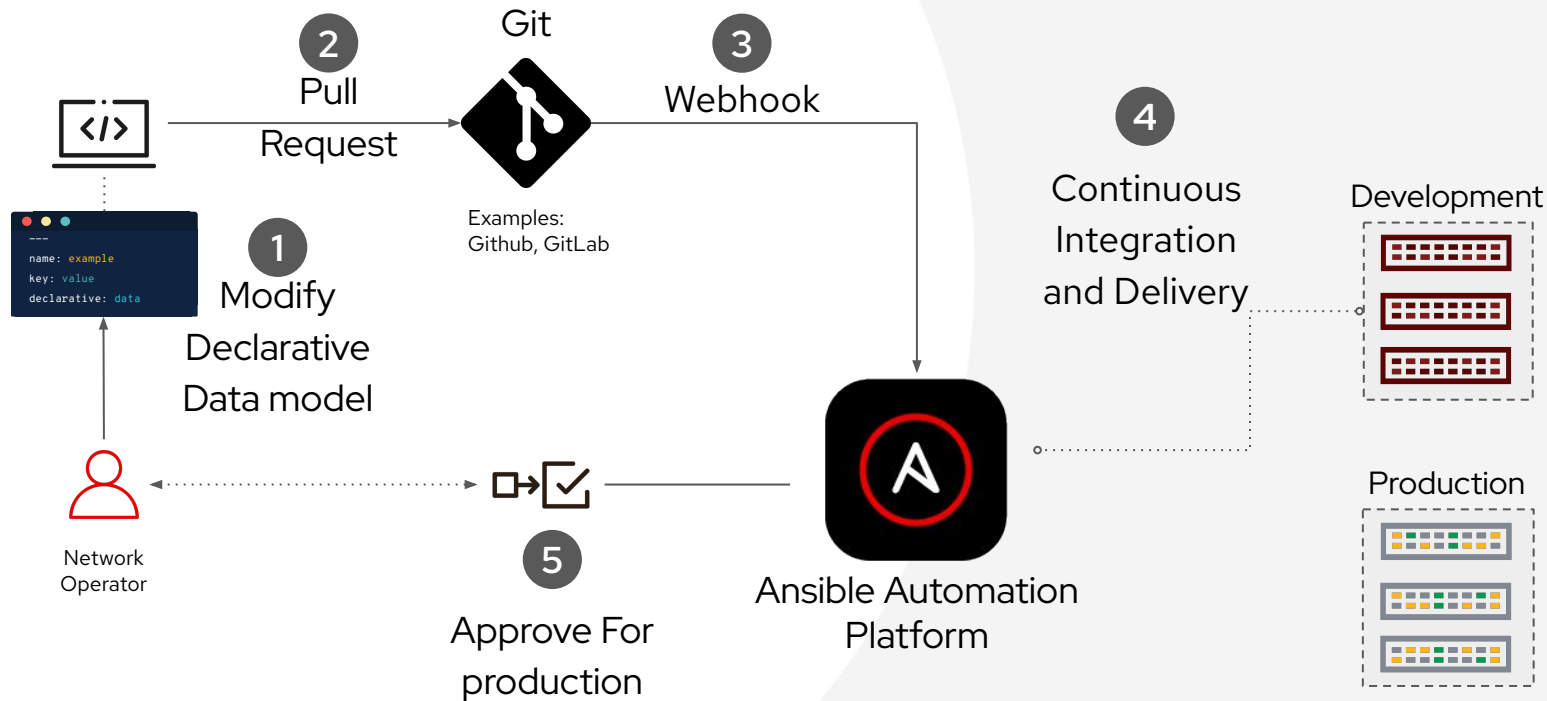
Automated NetOps

Tying everything together



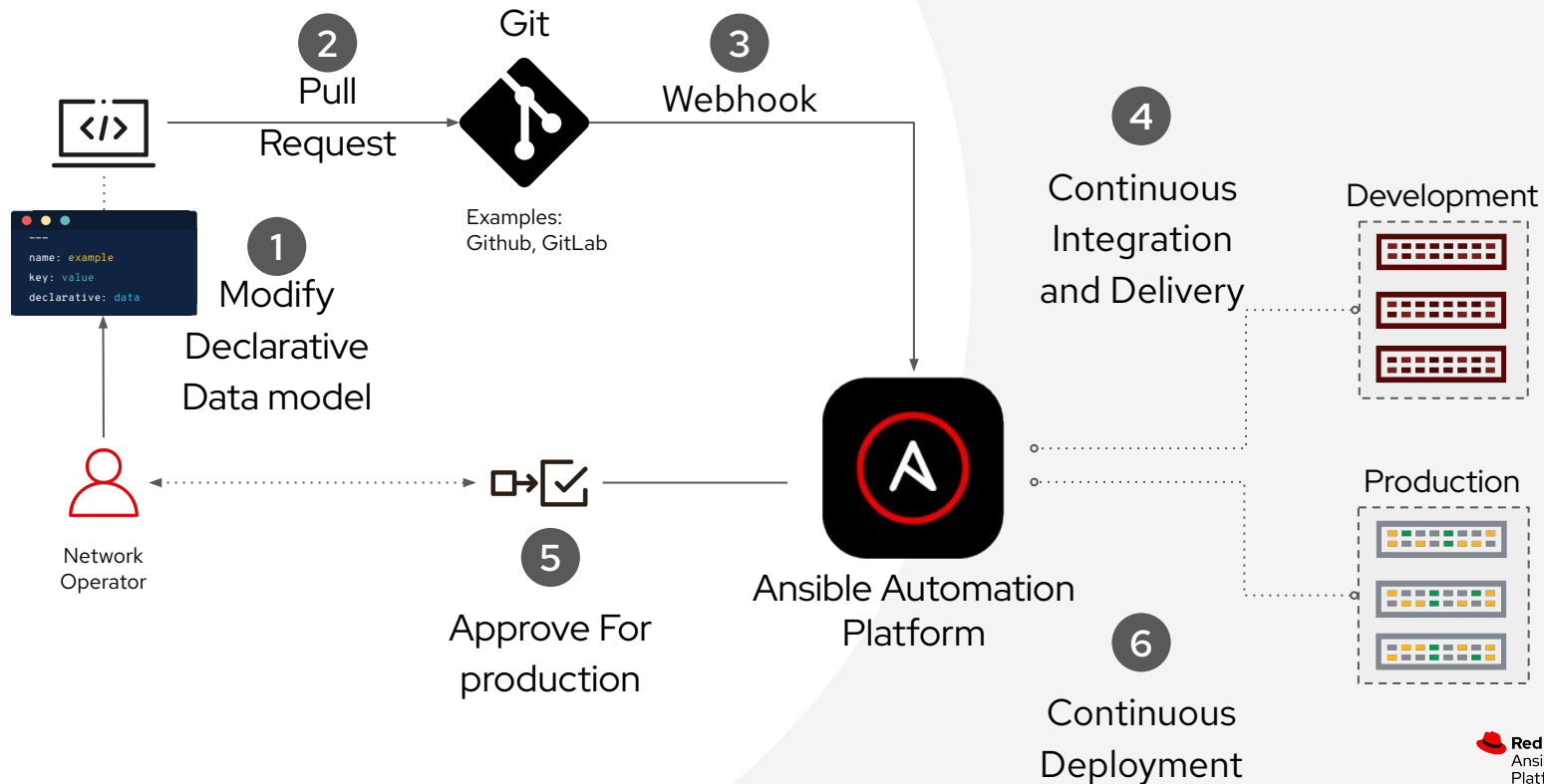
Automated NetOps

Tying everything together



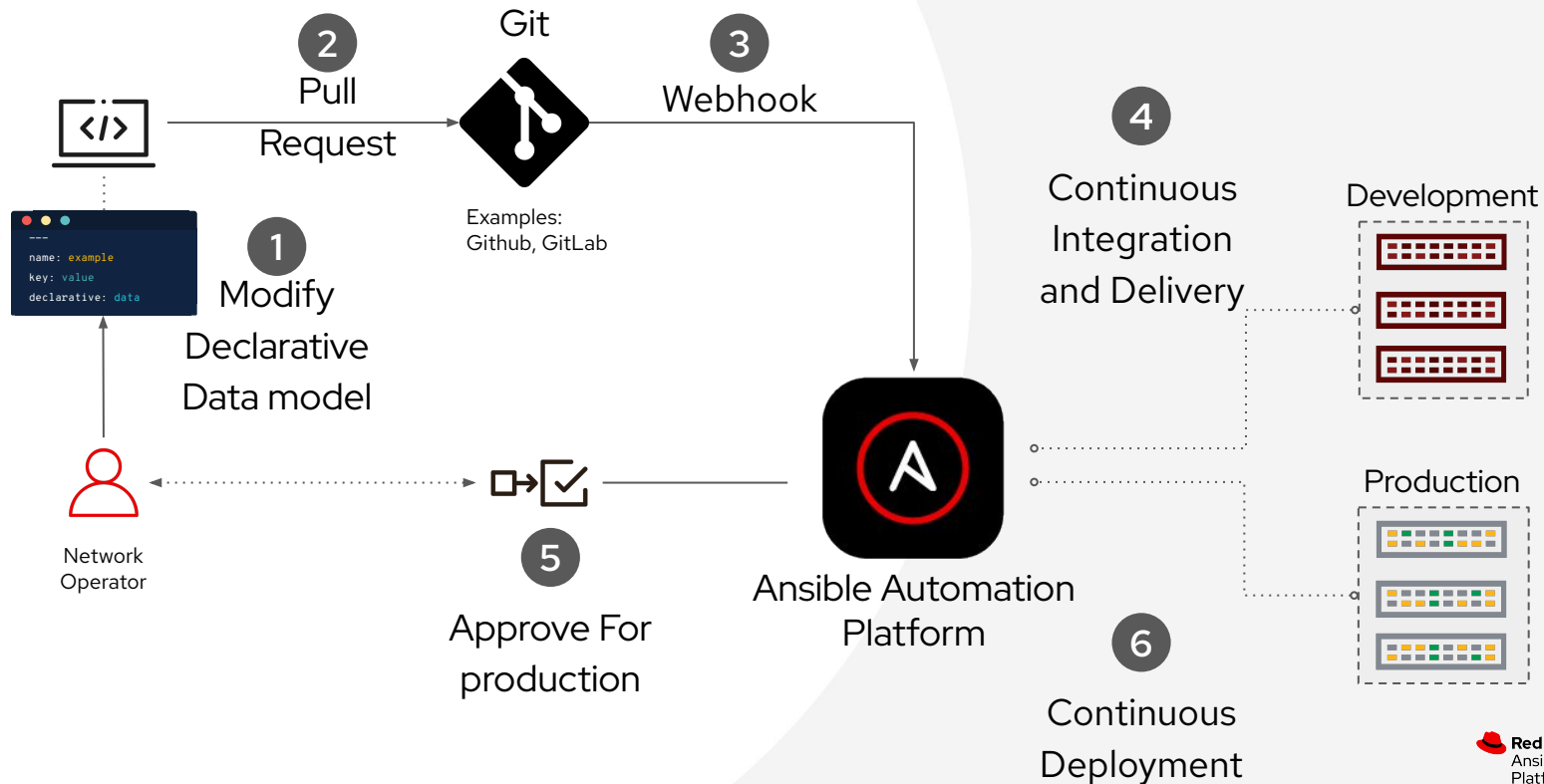
Automated NetOps

Tying everything together



Automated NetOps

Tying everything together



Approvals

Require user intervention in order to advance the workflow

- ▶ Can be configured to timeout and stop workflows in no human intervention within specified time limit
- ▶ Users can be assigned Approve permissions explicitly via RBAC
- ▶ Gives time to users to review automation status and testing before applying to production

