

The background is a vibrant, abstract graphic. It features a central bright white light source from which numerous colorful rays emanate, creating a sunburst or starburst effect. The rays transition through a spectrum of colors including yellow, orange, red, and various shades of blue and green. Overlaid on this are several large, semi-transparent, wavy shapes in similar color tones, giving the overall image a sense of motion and energy.

cisco *Live!*

Let's go

#CiscoLive



The bridge to possible

Ansible Network Automation, GitOps for NetOps

Sean Cavanaugh - Tech Marketing - Red Hat
@IPvSean

Adrian Iliesiu - Senior Technical Leader - Cisco
@aidevnet

BRKATO-2106

CISCO *Live!*

#CiscoLive

Cisco Webex App

Questions?

Use Cisco Webex App to chat with the speaker after the session

How

- 1 Find this session in the Cisco Live Mobile App
- 2 Click “Join the Discussion”
- 3 Install the Webex App or go directly to the Webex space
- 4 Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until June 9, 2023.



<https://ciscolive.ciscoevents.com/ciscolivebot/#BRKATO-2106>

Who am I?

Sean Cavanaugh



Work History:

Red Hat	2017-Now
Cumulus Networks	2014-2017
Cisco Systems	2010-2014

Personal

Live with my wife, three kids, one border collie and two cats in Chapel Hill, North Carolina

Hobbies

Home automation, electric bikes, soccer, Y-Guides, piano, 5Ks

Who am I?

Adrian Iliesiu



Work History:

Cisco System	2014-Now
Radialpoint	2012-2014
Ajilon Consulting	2010-2012

Personal

Worked and lived in 5 countries over the past 15 years.
Currently residing in San Jose, California

Hobbies

Fast cars, tennis, movies, reading

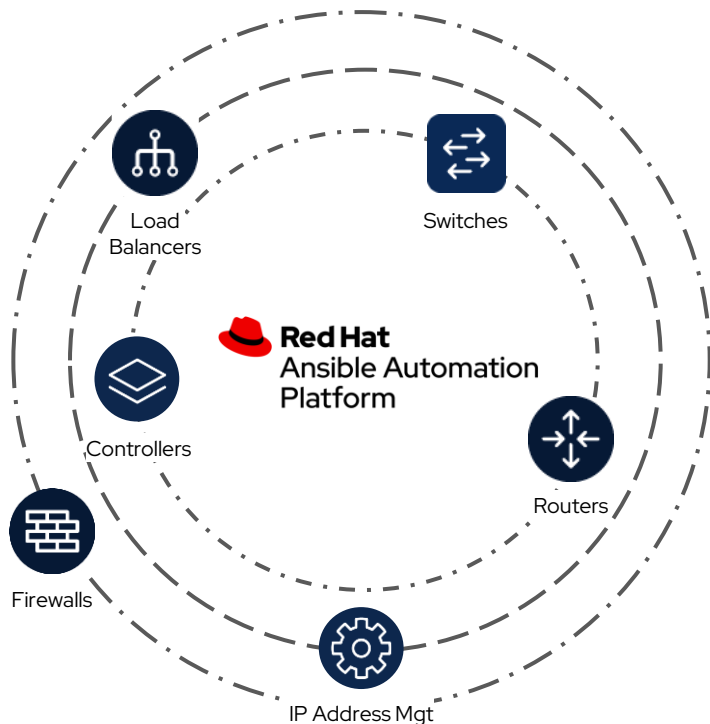
Agenda

- What is Ansible Network Automation?
- Automated NetOps
- Expanding on GitOps
- ansible.scm
- GitLab
- Demo
- Next Steps

What is Ansible Network Automation?



What is Ansible Network Automation?

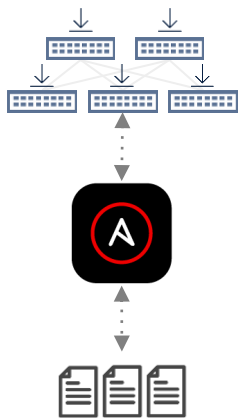


Ansible network automation provides network teams with the tools and an operational framework to implement **next-generation network operations**, manage network **infrastructure-as-code**, and better support digital transformation by connecting teams across the IT organization.

Ansible network automation is a set of Certified Content Collections designed to streamline and operationalize network operations across multiple platforms and vendors.

Start Small

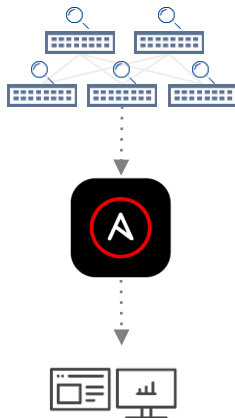
Quick automation victories for network engineers



Config Backup and Restore

Ubiquitous first touch use case

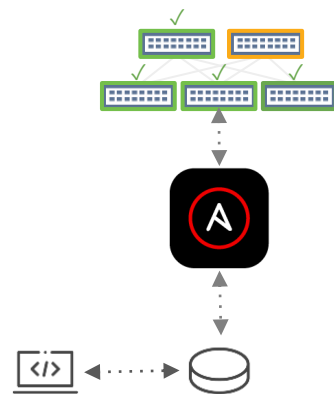
- Gain confidence in automation quickly
- First steps towards network as code
- Quickly recover network steady state



Dynamic Documentation

Use Ansible facts to gain information

- Read-only, no production config change
- Dynamic Documentation and reporting
- Understand your network



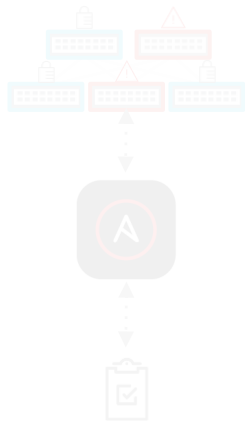
Scoped Config Management

Focus on high yield victories

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

Think Big

Institutionalizing automation into your organization



Network Compliance

Respond quickly and consistently

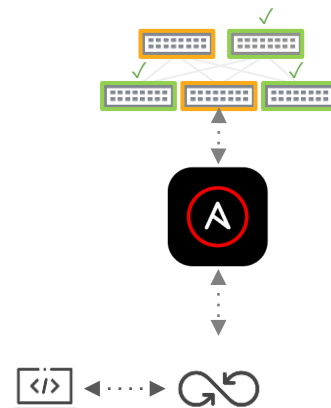
- Security and config compliance for network
- Remove human error from security responses
- Enforce Configuration policies and hardening



Operational State Validation

Going beyond config management

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

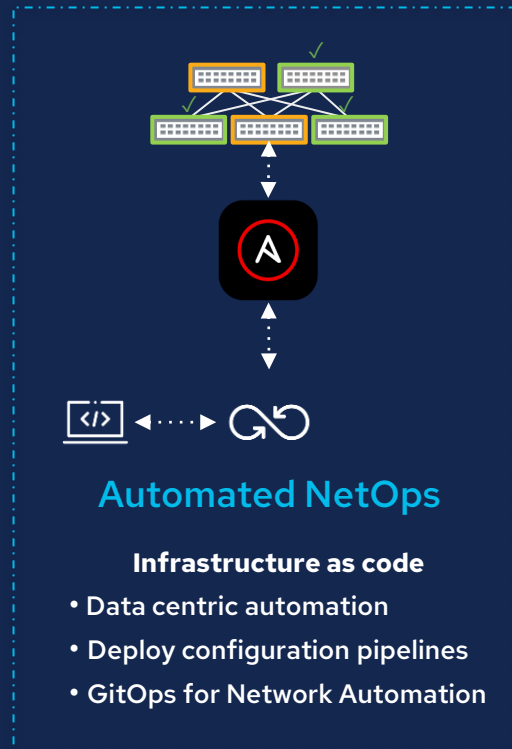


Automated NetOps

Infrastructure as code

- Data centric automation
- Deploy configuration pipelines
- GitOps for Network Automation

Automated NetOps



But first

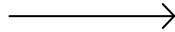
some terminology breakdowns

- Infrastructure as code?
- Source of Truth?
- Data centric automation?
- Pipelines?
- DevOps?
- CI/CD?
- GitOps?
- NetDevOps?

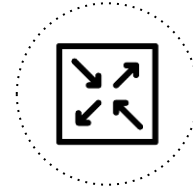
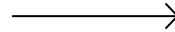
Infrastructure as code

manage IT infrastructure through definition files rather than physical hardware configuration or interactive configuration tools

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



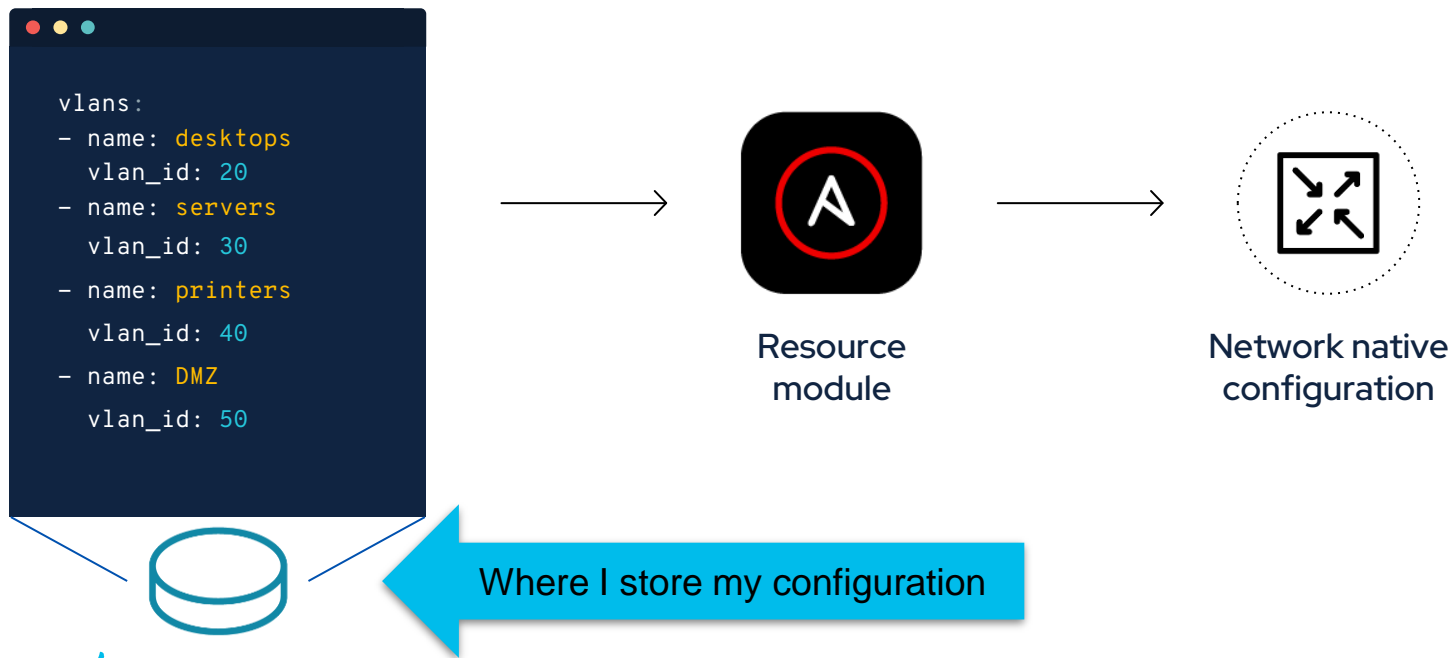
Resource
module



Network native
configuration

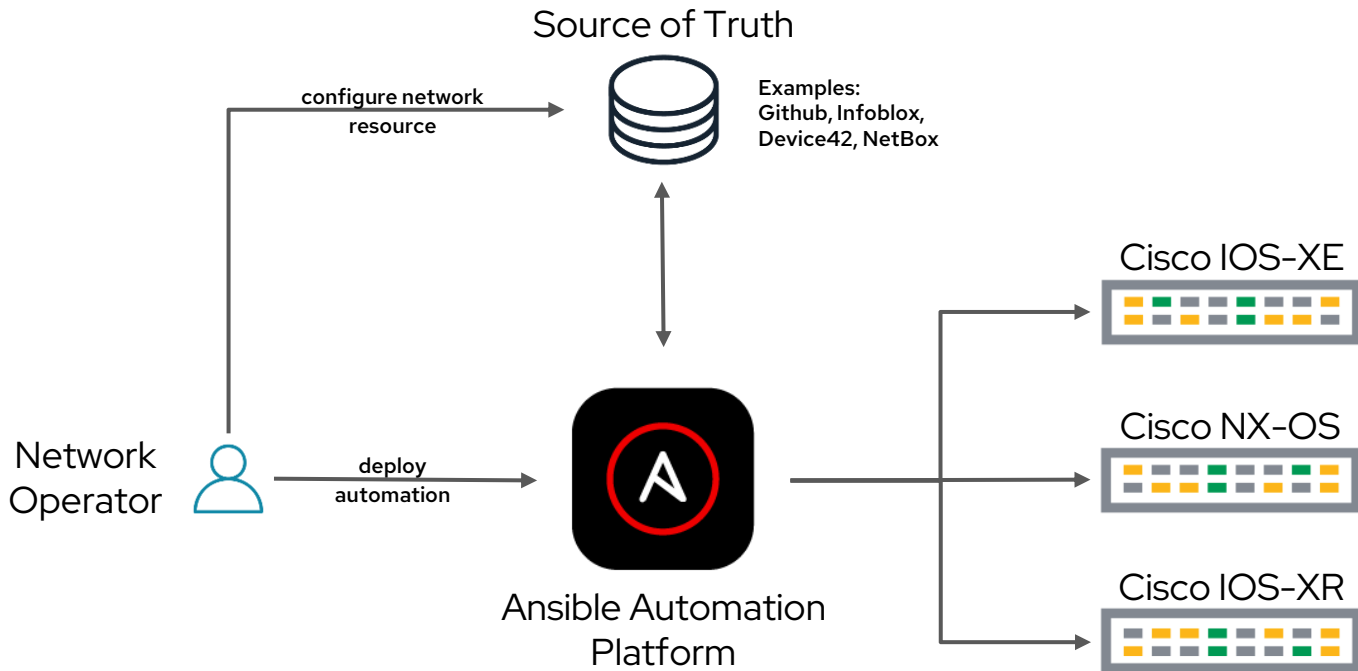
Source of Truth (SoT) 1/2

approach that ensures every piece of information is stored and updated in only one location



Source of Truth (SoT) 2/2

approach that ensures every piece of information is stored and updated in only one location



Data centric automation

focus on important data, not the implementation

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

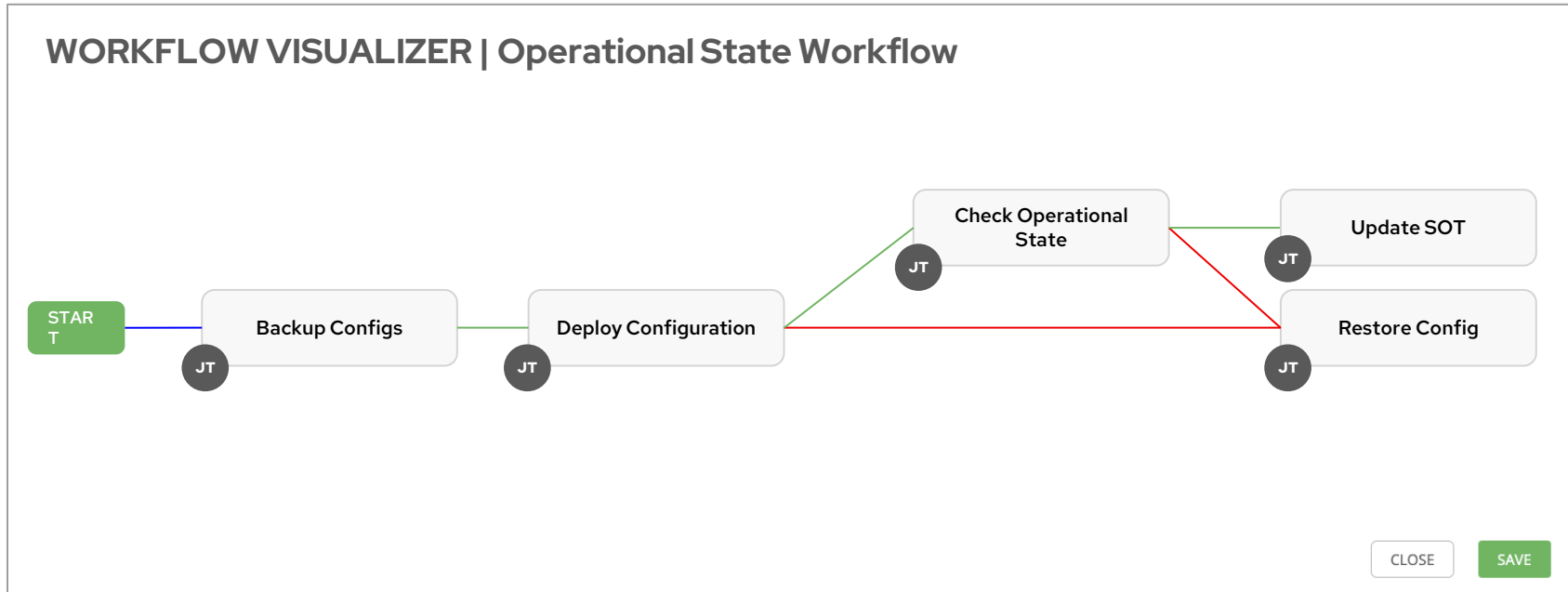
data

```
switch(config)# vlan 20  
switch(config-vlan-20)#  
switch(config-vlan-20)# vlan 30  
switch(config-vlan-30)# name servers  
switch(config-vlan-20)# vlan 40  
switch(config-vlan-40)# name printers  
switch(config-vlan-40)# vlan 50  
switch(config-vlan-50)# name DMZ
```

implementation

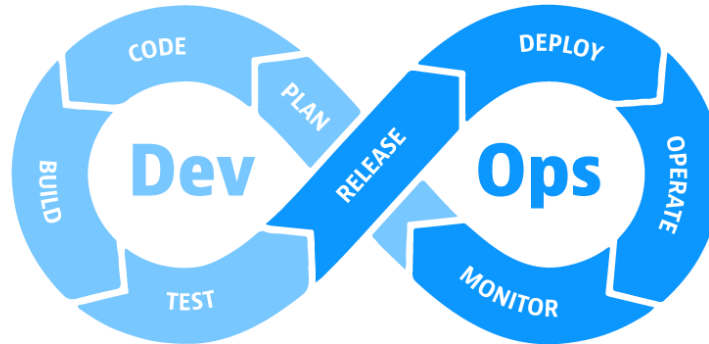
Pipelines

sequence of automated tasks or processes that work together to achieve a specific outcome or goal.



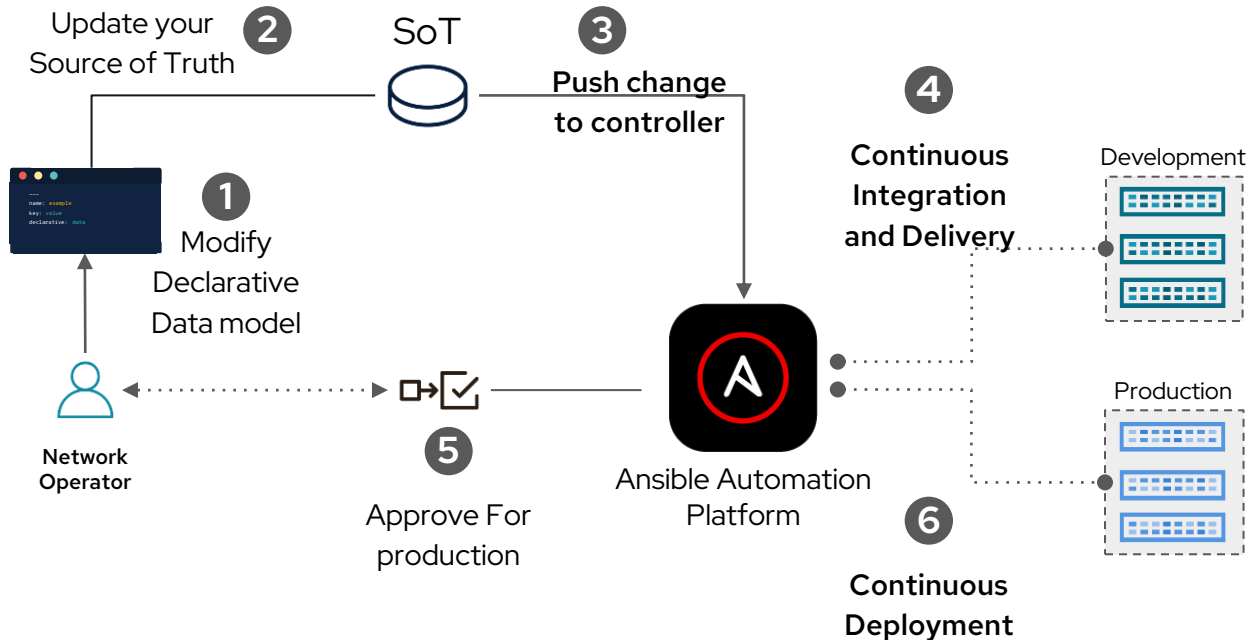
DevOps

breaking down the traditional silos between development and operations teams and promoting a culture of collaboration, shared responsibility, and continuous improvement.



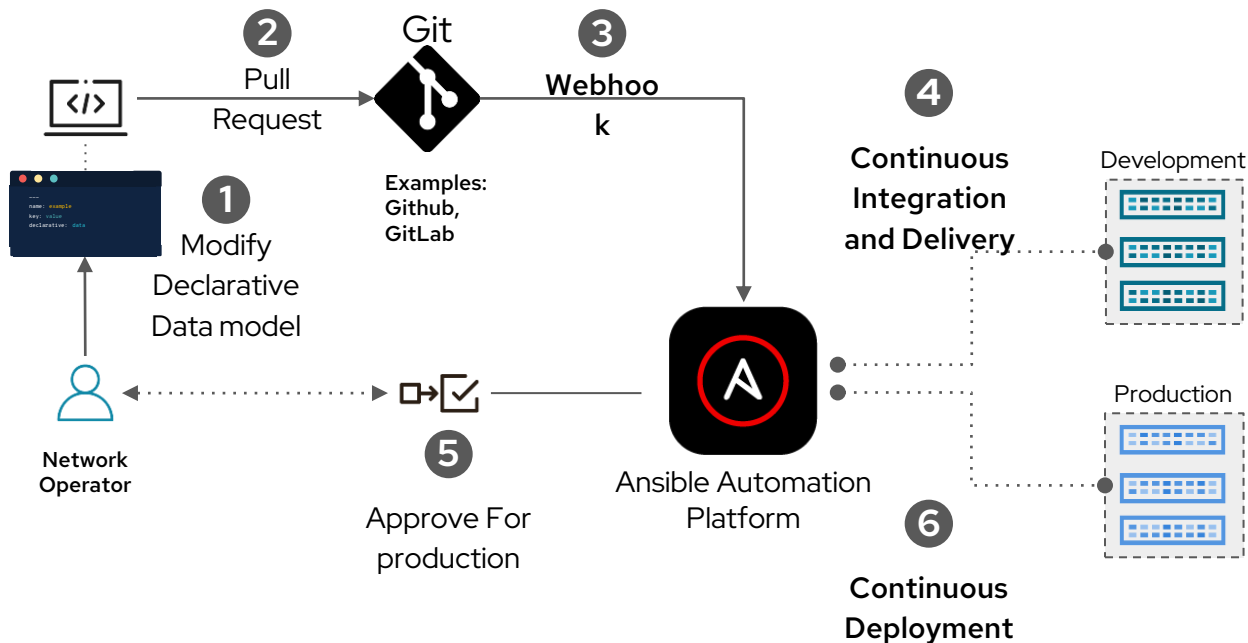
CI/CD

Continuous Integration and Continuous Delivery/Deployment. It is a set of practices and principles that aim to automate and streamline the process of building, testing, and deploying software applications.



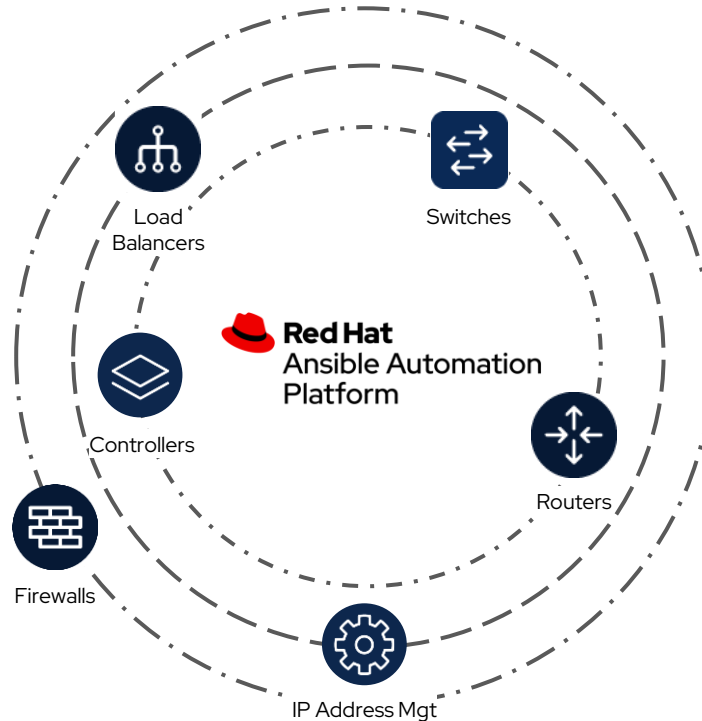
GitOps

GitOps enables a declarative, version-controlled approach to managing infrastructure and applications



NetDevOps

practices and principles that applies DevOps methodologies to network infrastructure operations

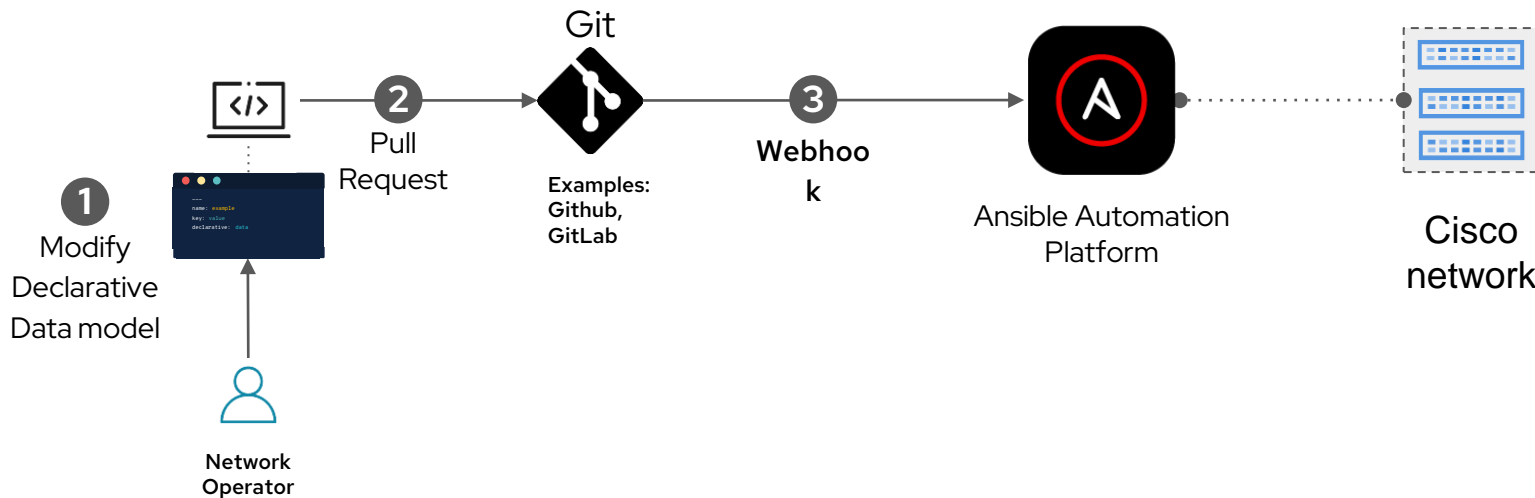


Expanding on GitOps



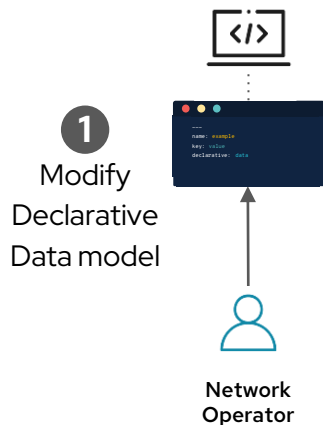
GitOps

What does an Ansible Automation solution look like?



GitOps

What does an Ansible Automation solution look like?

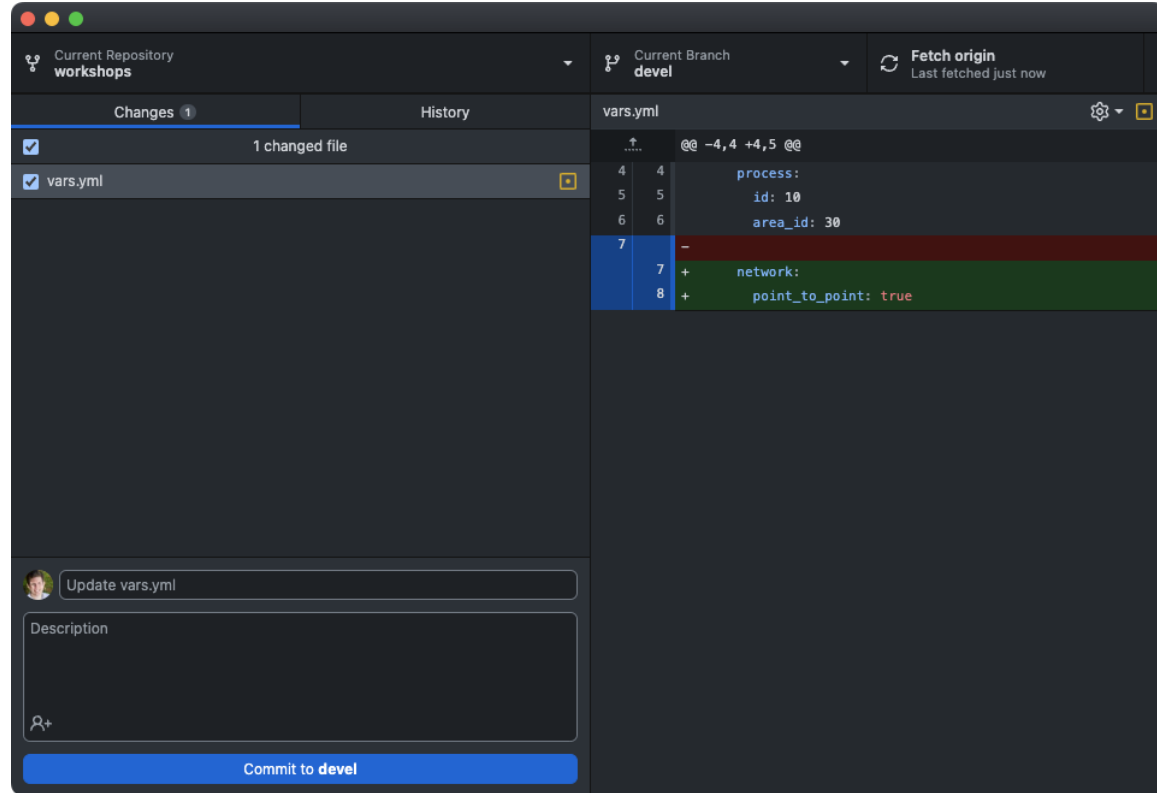
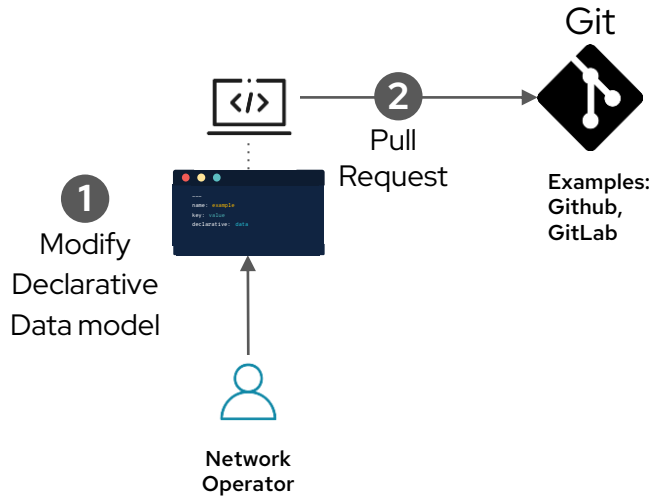


```
- name: GigabitEthernet0/1
  address_family:
    - afi: ipv4
      process:
        id: 10
        area_id: 30
      network:
        point_to_point: true
```

Added by user to the
data model

GitOps

What does an Ansible Automation solution look like?



GitOps

What does an Ansible Automation solution look like?

Webhook details

Choose a Webhook Service

- ✓ GitHub
- GitLab

Webhook URL [?]

<https://ansible.demoredhat.com/api/v2/...>

Webhook Key [?]

A NEW WEBHOOK KEY WILL BE...

Webhook Credential [?]

Ansible Workshop Testing PAT Pe...

[Webhooks](#) / Add webhook

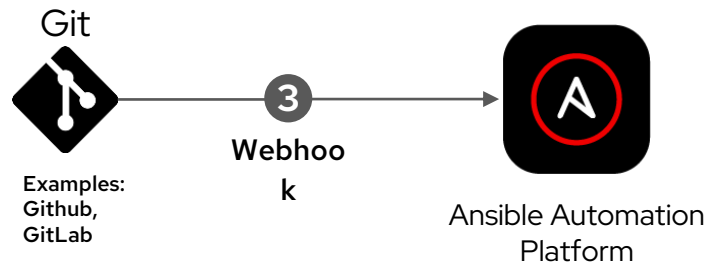
We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in [our developer documentation](#).

Payload URL *

https://ansible.demoredhat.com/api/v2/job_templates/70/github/

Content type

[application/json](#)



Overview: As soon as you push changes to git, Ansible Automation Platform will automatically kick off specified automation

GitOps

What does an Ansible Automation solution look like?

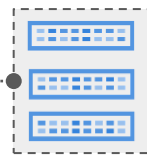
cisco.aci
cisco.mso
cisco.iosxr
cisco.ios
cisco.nxos
cisco.dcnm
cisco.asa
cisco.ise
cisco.dnac
cisco.nae

```
- name: OSPF Interfaces configuration
  cisco.ios.ios_ospf_interfaces:
    config: "{{ config }}"

- name: OSPF Interfaces configuration
  cisco.iosxr.ios_ospf_interfaces:
    config: "{{ config }}"
```



Ansible Automation
Platform



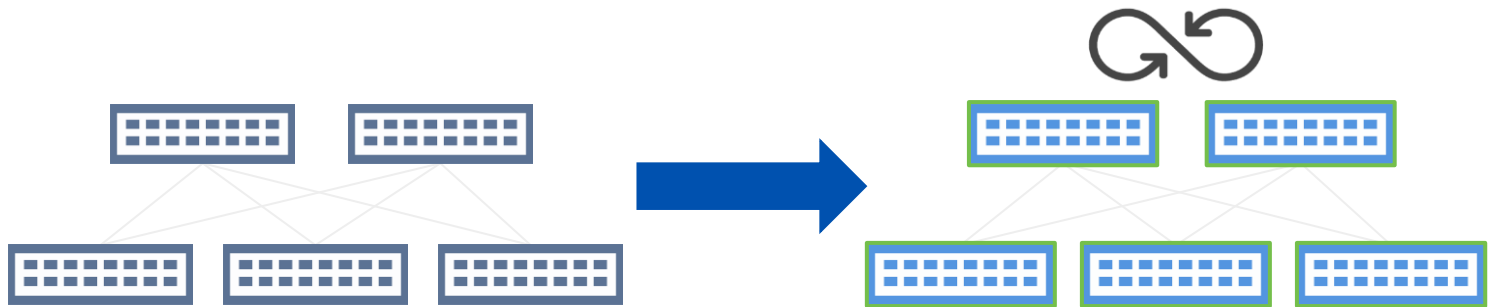
Cisco
network

Certified Collections found on:

<https://console.redhat.com/ansible/automation-hub/repo/published/cisco/>

What about brown field?

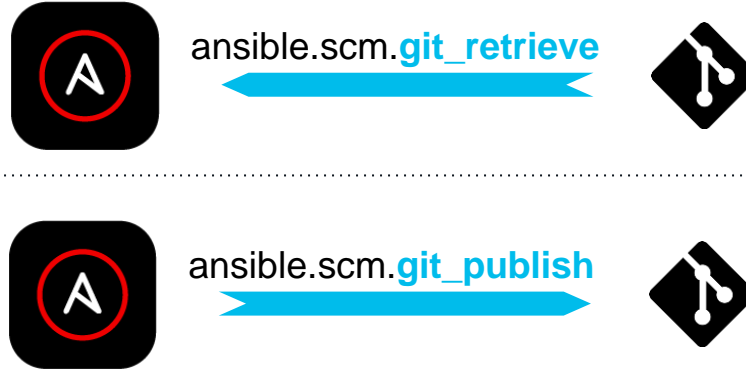
How do I convert an existing network into a GitOps methodology?



Introducing ansible.scm

ansible.scm

Ansible content collection for retrieving and publishing to Git repositories



Retrieve and Publish with any Git repository

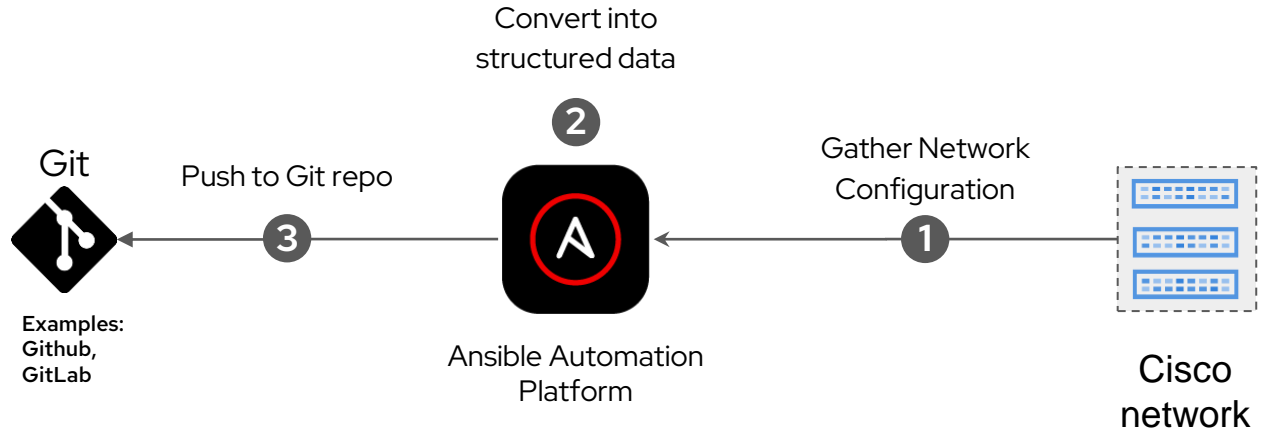
Easy way to work with Git repositories including Github and Gitlab

```
---
- hosts: localhost
  tasks:
    - name: Retrieve a repository
      ansible.scm.git_retrieve:
        origin:
          url: git@github.com:network/repo.git
        register: repository
```

```
---
- hosts: localhost
  tasks:
    - name: Publish a repo
      ansible.scm.git_publish:
        path: "/my/local/directory"
```

Brownfield Strategy

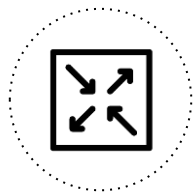
How to quickly make a GitOps solution in your existing network infrastructure



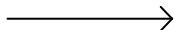
From existing network to Git

Ansible Automation Platform facts

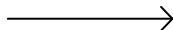
Network automation begins and ends with facts



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>>  
}
```

State Gathered

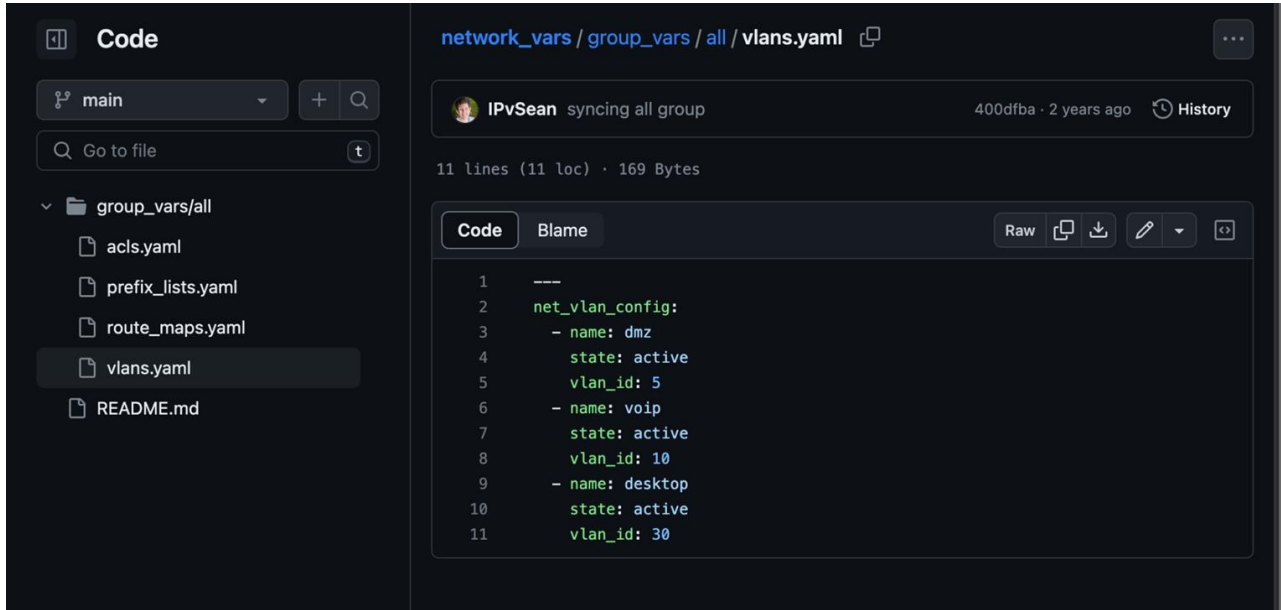
Retrieve any resource (or all of them)

```
- name: Gather OSPF Interfaces configurations
  cisco.ios.ios_ospf_interfaces:
    state: gathered

- name: Gather VLANs configurations
  cisco.ios.ios_vlans:
    state: gathered
```

Infrastructure as code

Brownfield are now variables (vars) in your Git repository



The screenshot displays a Git repository interface. On the left, a sidebar shows the file structure with a folder named `group_vars/all` containing files `acls.yaml`, `prefix_lists.yaml`, `route_maps.yaml`, `vlans.yaml` (selected), and `README.md`. The main area shows the content of `vlans.yaml`, which is a 11-line file (11 loc) of 169 Bytes. The file content is as follows:

```
1 ---
2 net_vlan_config:
3   - name: dmz
4     state: active
5     vlan_id: 5
6   - name: voip
7     state: active
8     vlan_id: 10
9   - name: desktop
10    state: active
11    vlan_id: 30
```



GitLab

CISCO *Live!*

GitLab

Overview

- Complete DevOps platform
- Project Management
- Hosted Git repositories
- Built-in CI/CD

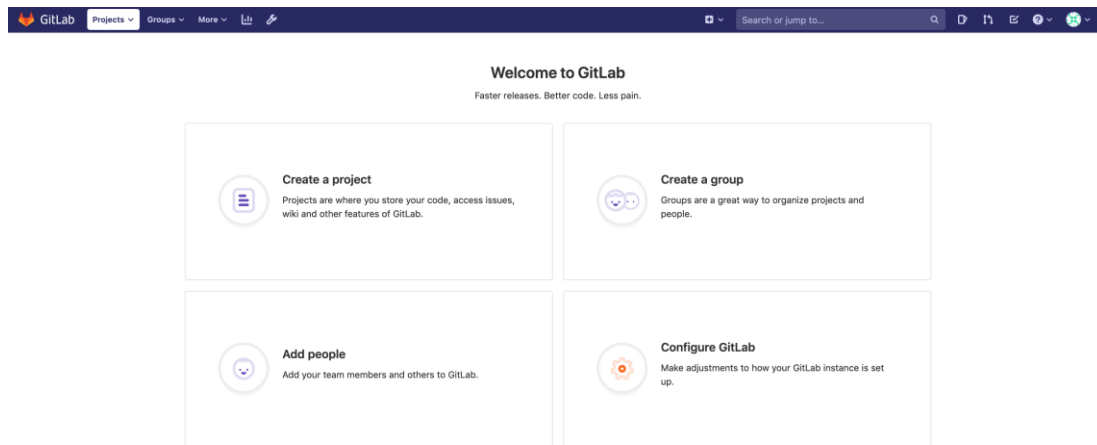
Stages of the DevOps lifecycle

Manage Plan Create Verify Package Secure Release Configure Monitor Defend

GitLab

Projects

- Code, issues, wikis, etc. are organized into projects



GitLab

Pipelines

- Fundamental building blocks for CI/CD

GitLab Pipelines page for the 'gitlab-runner' project. The page displays a list of pipelines with the following columns: Status, Pipeline, Triggerer, Commit, and Stages.

Status	Pipeline	Triggerer	Commit	Stages
running	#177869707	[Avatar]	12346 -> 8effc580 Merge branch 'k8sexectorclust...	[Success] [Warning] [Success] [Success] [Success] [Success] [Success]
running	#177869404	[Avatar]	12346 -> d3b5331f Merge branch 'k8sexectorclust...	[Success] [Warning] [Success] [Success] [Success] [Success] [Success]
running	#177606098	[Avatar]	12325 -> cfa77e77 Merge branch 'patch-4' into 'm...	[Success] [Warning] [Success] [Success] [Success] [Success] [Success]
running	#177604493	[Avatar]	12325 -> 3b94d486 Updated Windows steps	[Success] [Warning] [Success] [Success] [Success] [Success] [Success]
canceled	#177590399	[Avatar]	12325 -> 2846f438 Merge branch 'patch-4' into 'm...	[Success] [Warning] [Success] [Success] [Success] [Success] [Success]
canceled	#177535883	[Avatar]	12325 -> 2846f438 Merge branch 'patch-4' into 'm...	[Success] [Warning] [Success] [Success] [Success] [Success] [Success]
passed	#177412421	[Avatar]	26632-gitlab -> 9a6585be Add helper binary	[Success]
passed	#177408271	[Avatar]	26632-gitlab -> 825f9495 Add helper binary	[Success]

GitLab

Pipelines

- Composed of:
 - Jobs – What to do
 - Stages - When to run jobs
- A typical pipeline would contain four stages
 - Build (compile)
 - Test (unit-tests)
 - Staging (deploy-to-stage, integration-tests)
 - Production (deploy-to-prod)

GitLab

Runners

- Used to run the jobs and send results back to GitLab
- When a pipeline is triggered, a Runner will `git clone` the repo and then execute the instructions inside of `.gitlab-ci.yml`
- Implements a variety of executors
 - Jobs can run in different scenarios
 - SSH, Shell, Parallels, VirtualBox, Docker, Kubernetes, Custom

GitLab

.gitlab-ci.yml

```
1 stages:
2   - build
3   - test
4   - deploy
5
6 image: alpine
7
8 compile_a:
9   stage: build
10  script:
11    - echo "This job builds something."
12
13 compile_b:
14   stage: build
15   script:
16     - echo "This job builds something else."
17
18 unit_test_a:
19   stage: test
20   script:
21     - echo "This job tests something. It will only run when all jobs in the"
22     - echo "build stage are complete."
23
24 unit_test_b:
25   stage: test
26   script:
27     - echo "This job tests something else. It will only run when all jobs in the"
28     - echo "build stage are complete too. It will start at about the same time as test_a."
29
30 deploy_a:
31   stage: deploy
32   script:
33     - echo "This job deploys something. It will only run when all jobs in the"
34     - echo "test stage complete."
35
36 deploy_b:
37   stage: deploy
38   script:
39     - echo "This job deploys something else. It will only run when all jobs in the"
40     - echo "test stage complete. It will start at about the same time as deploy_a."
```



Demo

CISCO *Live!*

Test environment



Dev
Workstation



Hosting
Server



Network
Simulation



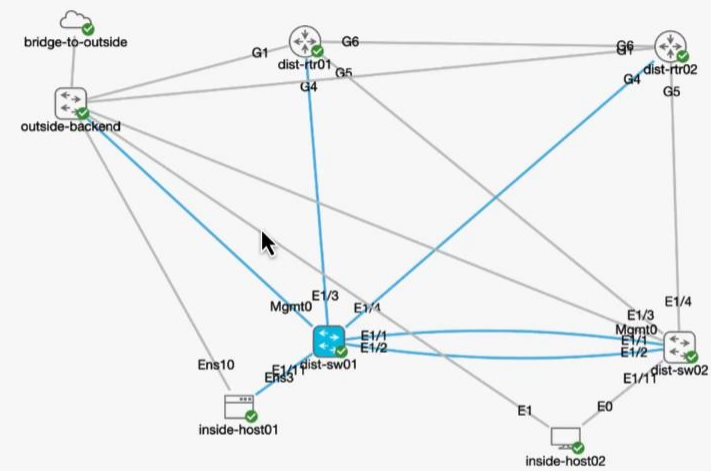
Infrastructure
as Code



Source
Control



Testing
Framework



NODE INFO SIMULATE CONNECTIVITY CONSOLE VNC EDIT CONFIG INTERFACES

CLONE NODE DELETE NODE

NX-OS 9000
Node Name:
dist-sw01

CPU 28.56% MEMORY 18.82% DISK 8.38% Notifications 6 Status OK



Next Steps

Learning resources

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



Red Hat
Ansible Automation
Platform



Networking Labs

<https://red.ht/ansible-labs>



E-Books

1. Modernize Your Network with Red Hat
red.ht/network-book1
2. Network Automation Guide
red.ht/Net_Auto_Guide
1. Automate Your Network with Red Hat
red.ht/network-book2



Red Hat Certification
Ansible for Network Automation
(DO457)

red.ht/network-training

Fill out your session surveys!



Attendees who fill out a minimum of four session surveys and the overall event survey will get **Cisco Live-branded socks** (while supplies last)!



Attendees will also earn 100 points in the **Cisco Live Challenge** for every survey completed.



These points help you get on the leaderboard and increase your chances of winning daily and grand prizes

Continue your education

- Visit the Cisco Showcase for related demos
- Book your one-on-one Meet the Engineer meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand



The bridge to possible

Thank you

CISCO *Live!*

#CiscoLive

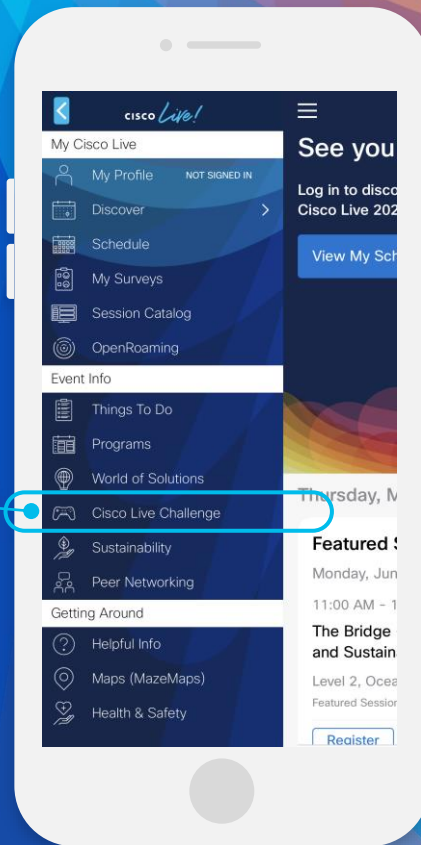
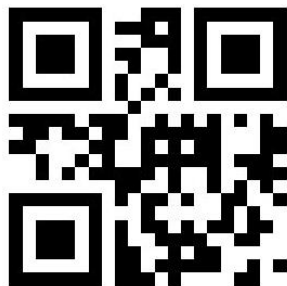


Cisco Live Challenge

Gamify your Cisco Live experience!
Get points for attending this session!

How:

- 1 Open the Cisco Events App.
- 2 Click on 'Cisco Live Challenge' in the side menu.
- 3 Click on View Your Badges at the top.
- 4 Click the + at the bottom of the screen and scan the QR code:



The background is a vibrant, abstract graphic. It features a central bright white light source from which numerous colorful rays emanate, creating a sunburst or starburst effect. The rays transition through a spectrum of colors including yellow, orange, red, and various shades of blue and green. Overlaid on this are several large, semi-transparent, wavy shapes in similar color tones, giving the overall image a sense of motion and energy.

cisco *Live!*

Let's go

#CiscoLive