ANSIBLE

Montreal Ansible Meetup

Michael Lessard,
Senior Solutions Architect
mlessard@redhat.com

22 Novembre, 2017

redhat

AGENDA

18:00 - 18:30

Bienvenue/nouvelles (Par Michael Lessard/ Red Hat) 18:00 - 18:30

Mot du commanditaire de la soirée (Cloudops)

Ansible 2.4, Tower 3.2 , Ansible Engine + Networking, Ansible AWX

18:30 - 19:00

Ansible au LanETS (par Laurent Dumont/LAN ETS)

19:00 - 19:10

Mise à jour sur ARA 1.0 (par David Moreau Simard/ Red Hat)

:::::::::: **PAUSE ::::::::::**

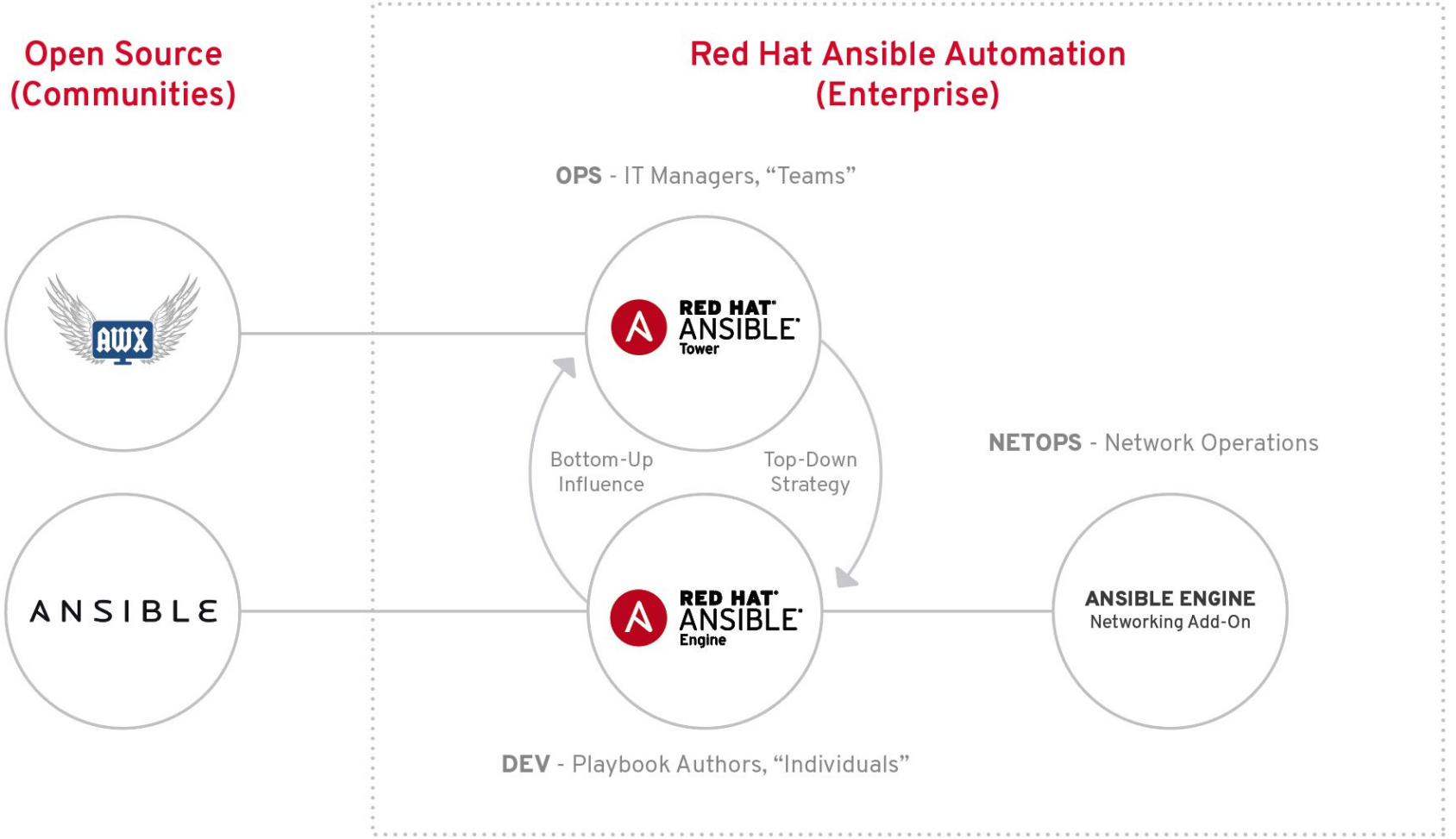
19:30 -20:00

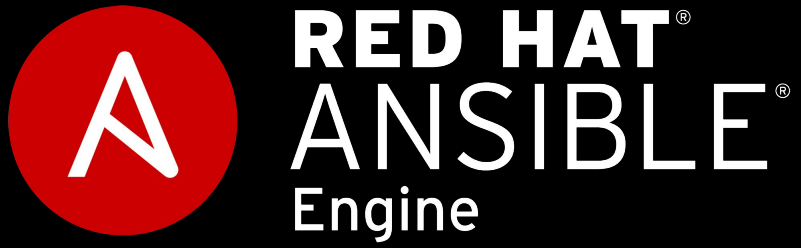
Molecule - Isolating role development (par Alain Chiasson)

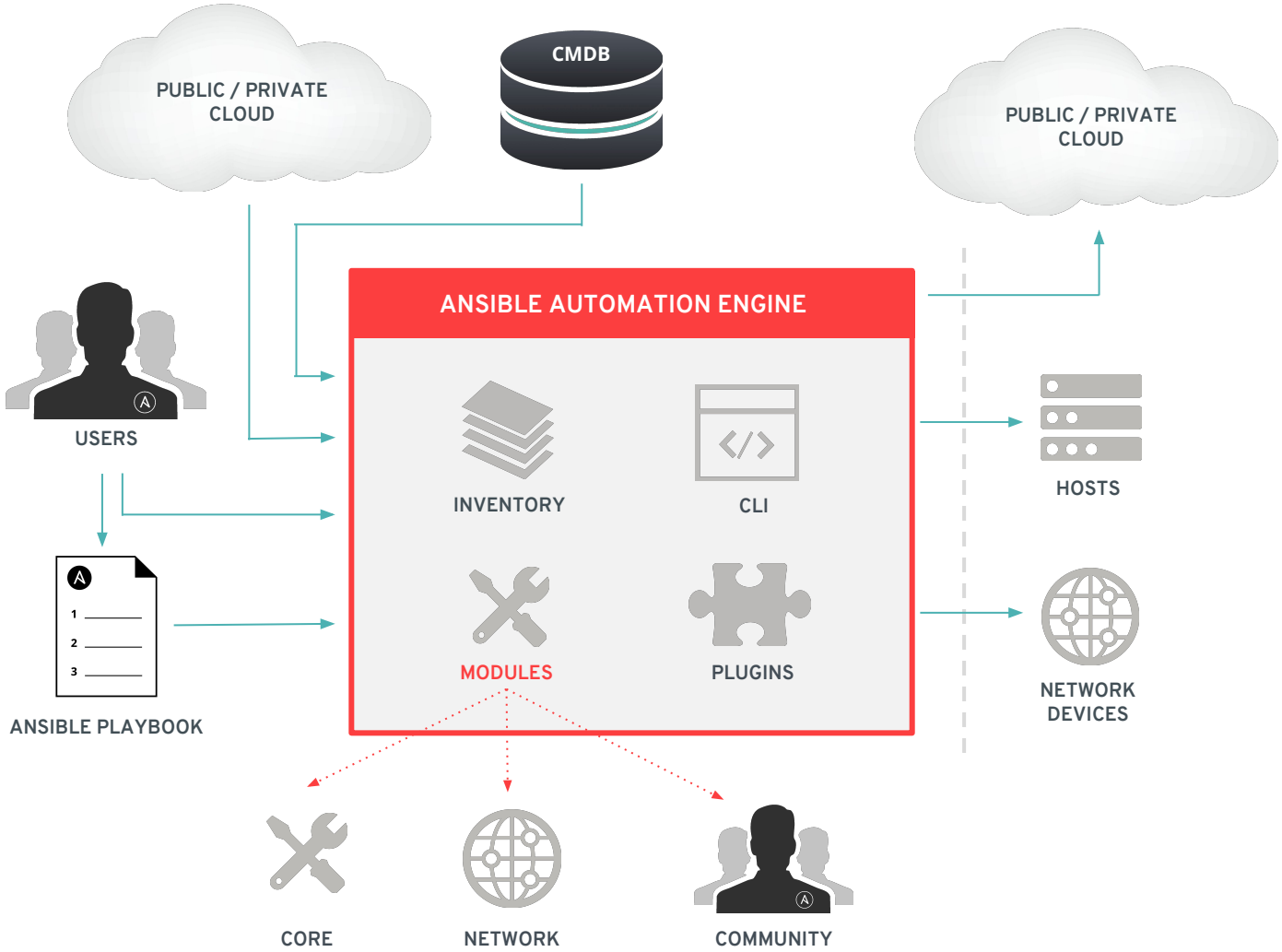
20:00 - 20:30

Software Factory - Développer, Tester et Valider vos rôles et playbooks Ansible (Par Nicolas Hicher/ Red Hat)

redhat.


Open Source
(Communities)
Red Hat Ansible Automation
(Enterprise)
OPS - IT Managers, "Teams"
AWX
RED HAT ANSIBLE Tower
Bottom-Up
Influence
Top-Down
Strategy
NETOPS - Network Operations
ANSIBLE
RED HAT ANSIBLE Engine
ANSIBLE ENGINE
Networking Add-On
DEV - Playbook Authors, "Individuals"


RED HAT®
ANSIBLE®
Engine


PUBLIC / PRIVATE CLOUD
CMDB
PUBLIC / PRIVATE CLOUD
ANSIBLE AUTOMATION ENGINE
USERS
INVENTORY
CLI
HOSTS
1
2
3
ANSIBLE PLAYBOOK
MODULES
PLUGINS
NETWORK DEVICES
CORE
NETWORK
COMMUNITY

RED HAT ANSIBLE ENGINE CORE MODULES

CORE MODULES

- **Developed, tested, and maintained** by Red Hat
- **~100 modules** and growing
- Fully supported: we'll **report and fix problems**
- Side note: we're actively working with other Red Hat product teams to get their modules into Engine as core modules:
  E.g. oVirt, redhat_subscription, IPA, etc.

http://docs.ansible.com/ansible/latest/modules_support.html
https://access.redhat.com/support/policy/updates/ansible-engine

INCLUDED:

commands

files

inventory actions

basic network actions

Packaging (i.e. yum)

source control

systems

Utilities

AWS

Windows

RED HAT®
ANSIBLE®
Engine

+

NETWORK AUTOMATION
ADD-ON

RED HAT ANSIBLE ENGINE NETWORKING ADD-ON

NETWORK MODULES

- **Developed, maintained, tested, and supported** by Red Hat
- **140+ supported modules** and growing*
- Red Hat **reports and fixes problems**
- **Networking modules included** with Ansible Engine offering, but **the Ansible Engine Networking Add-On** SKU purchase is **required** for full support

*take special note of the specific supported platforms

NETWORKING ADD-ON INCLUDED SUPPORT:

Arista EOS

Cisco IOS

Cisco IOS XR

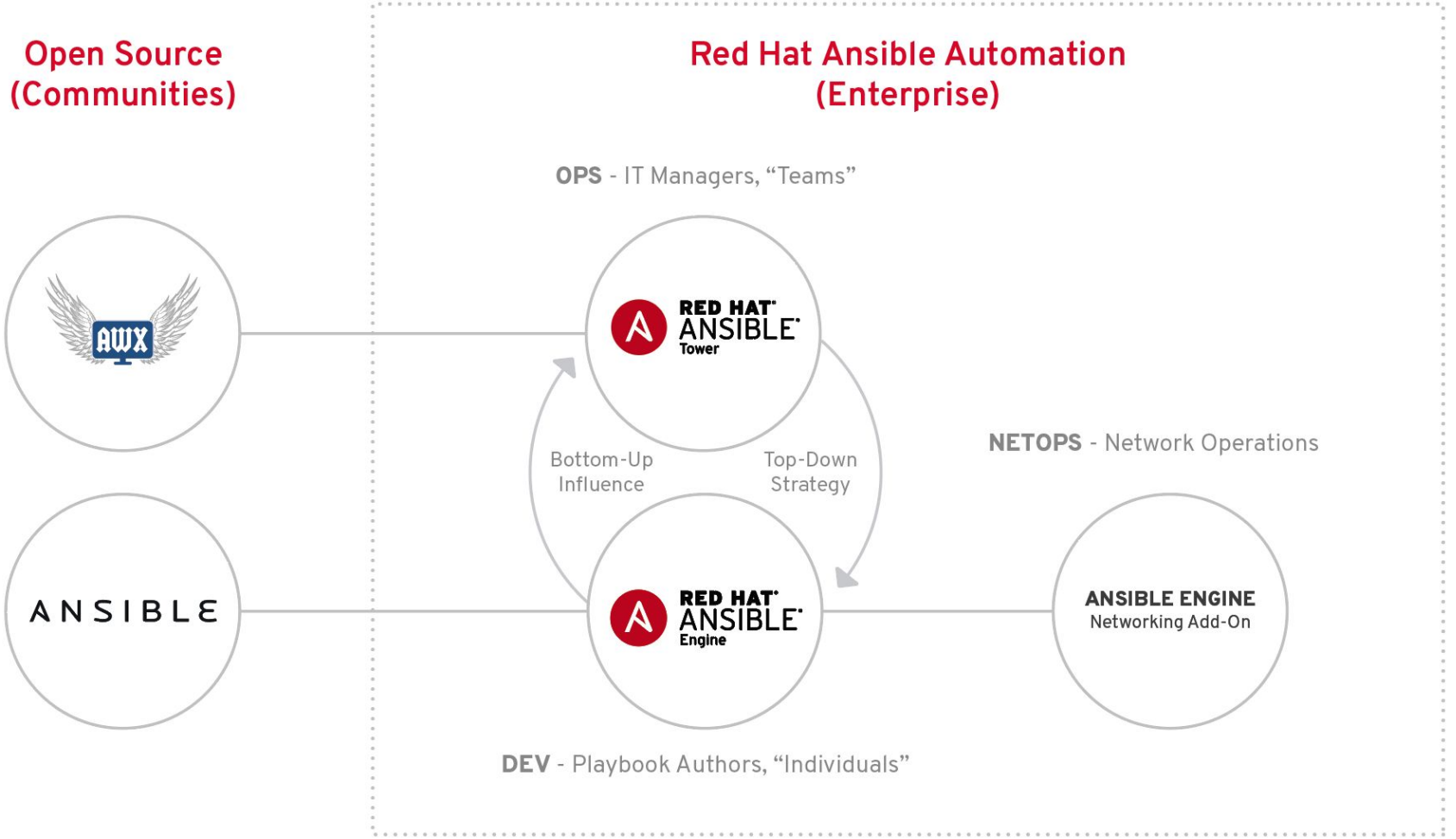
Cisco NX-OS

Juniper Junos

Open vSwitch

VyOS

redhat.


Open Source
(Communities)
Red Hat Ansible Automation
(Enterprise)
OPS - IT Managers, "Teams"
AWX
RED HAT ANSIBLE Tower
Bottom-Up
Influence
Top-Down
Strategy
NETOPS - Network Operations
ANSIBLE
RED HAT ANSIBLE Engine
ANSIBLE ENGINE
Networking Add-On
DEV - Playbook Authors, "Individuals"

2.4

GA :: SEPTEMBER 19th 2017

ANSIBLE 2.4

MODERNIZE. ORGANIZE. EXTEND.

MODERNIZE
Python 2.6 (targeted hosts)
Python 3 support

ORGANIZE
Inventory is now fully pluggable (with backwards compatibility)
Better variable handling in inventories
Significant updates to fact gathering and fact management
Vault: Playbooks that use multiple roles can use a different vault password per team/role

EXTEND
Updated and added modules

ANSIBLE 2.4

MODULE ADDITIONS AND UPDATES

CLOUD

AWS (more ELB, S3, direct connect, iam, lambda, redshift, ECS, Lightsail)
Azure (ACS, availability sets, dns, functions, loadbalancer, managed disks, scalesets, vm extensions, Azure stack private clouds)

WINDOWS

Powershell DSC support
plugin_loader support in powershell
New modules for hotfix, defrag, security policy, and power plan allow finer-grained control for initial setup and ongoing maintenance

TOWER

Ansible now ships with modules to control Tower

ANSIBLE NETWORK AUTOMATION

33
Networking platforms

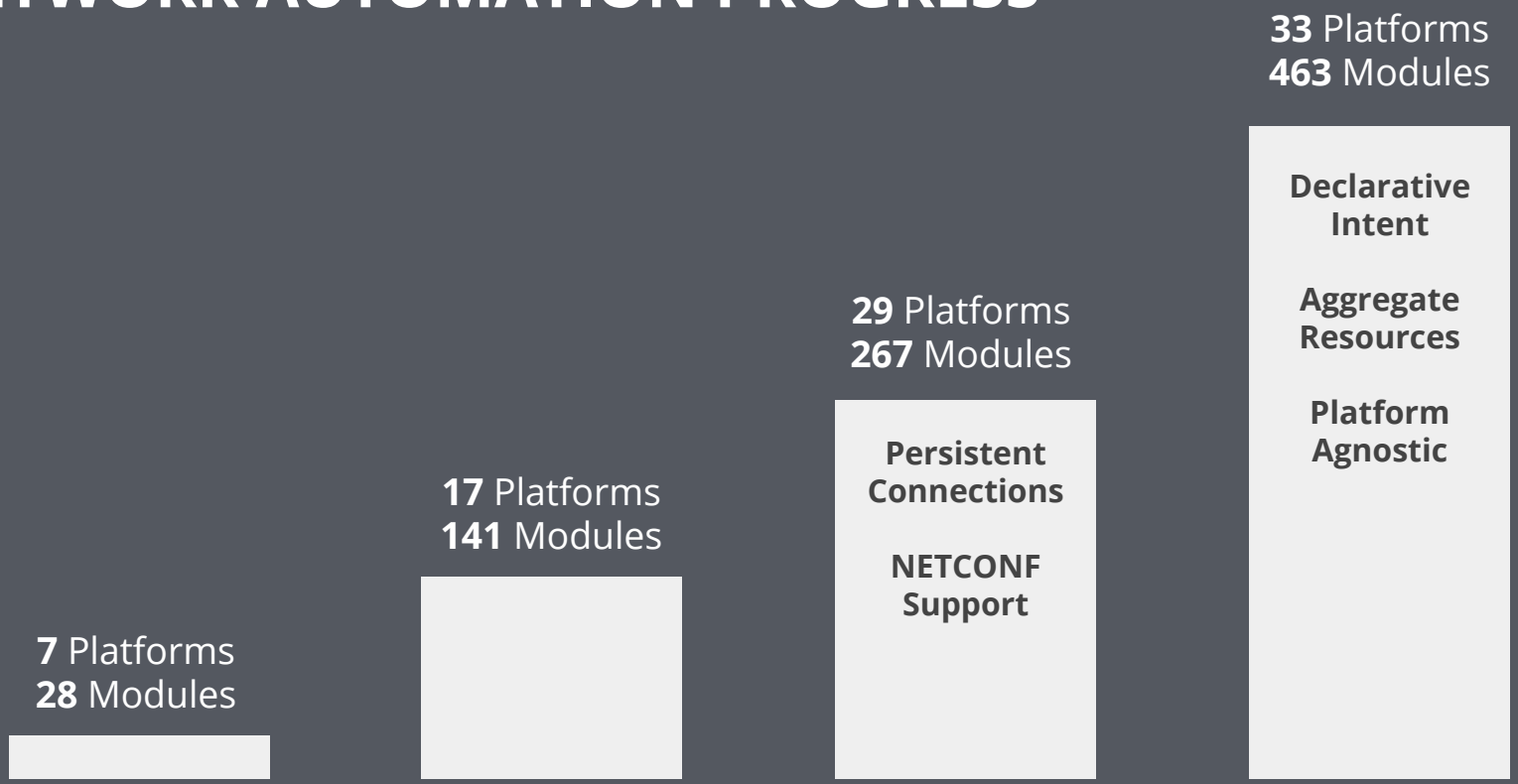
460+
Networking modules

ansible.com/networking

NETWORK MODULES: DEVICE ENABLEMENT INCLUDED

- A10
- Apstra
- Arista EOS (cli, eAPI), CVP
- Aruba Networks
- AVI Networks
- Big Switch Networks
- Cisco ACI, AireOS, ASA, IOS, IOS-XR, NX-OS
- Citrix Netscaler
- Cumulus Linux
- Dell OS6, OS9, OS10
- Exoscale
- F5 BIG-IP
- Fortinet FortIOS
- Huawei
- Illumos
- Juniper Junos
- Lenovo
- Ordnance
- NETCONF
- Netvisor
- Openswitch
- Open vSwitch (OVS)
- Palo Alto PAN-OS
- Nokia SR OS
- VyOS

NETWORK AUTOMATION PROGRESS



AGGREGATE RESOURCES

```
- name: configure vlans neighbor
  net_vlan:
    vlan_id: "{{ item.vlan_id }}"
    name: "{{ item.name }}"
    state: "{{ item.state | default('active')
}}"
  with_items:
    - { vlan_id: 1, name: default }
    - { vlan_id: 2, name: Vl2 }
    - { vlan_id: 3, state: suspend }
```

Ansible Engine 2.3

```
- name: configure vlans neighbor
  net_vlan:
    aggregate:
      - { vlan_id: 1, name: default }
      - { vlan_id: 2, name: Vl2 }
      - { vlan_id: 3, state: suspend }
    state: active
    purge: yes
```

Ansible Engine 2.4

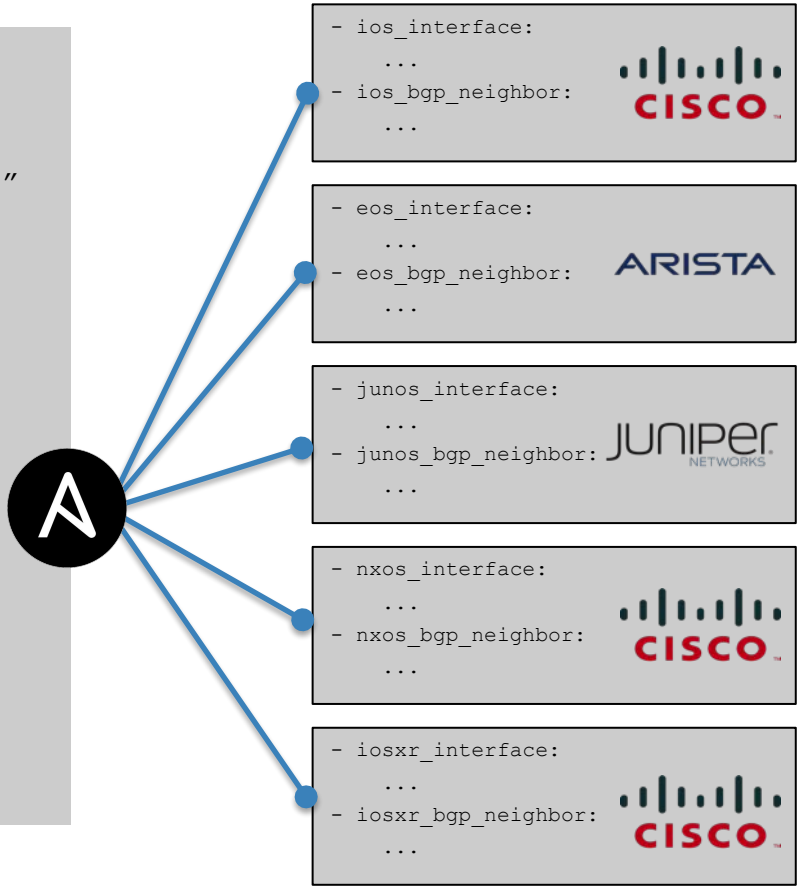
RESOURCE MODULES

```
---
- name: system node properties
  hosts: all

    tasks:
      - name: configure eos system properties
        eos_system:
          domain_name: ansible.com
          vrf: management
        when: network_os == 'eos'

      - name: configure nxos system properties
        nxos_system:
          domain_name: ansible.com
          vrf: management
        when: network_os == 'nxos'

      - name: configure ios system properties
        ios_system:
          domain_name: ansible.com
          lookup_enabled: yes
        when: network_os == 'ios'
```

- Per Platform Implementation
- Declarative by design
- Abstracted over the connection
- Violates DRY principals
- Makes platforms happy
- … Not so much for operators

MINIMUM VIABLE PLATFORM AGNOSTIC (MVPA)

```
- name: configure network interface
  net_interface:
    name: "{{ interface_name }}"
    description: "{{ interface_description }}"
    enabled: yes
    mtu: 9000
    state: up

- name: configure bgp neighbors
  net_bgp_neighbor:
    peers: "{{ item.peer }}"
    remote_as: "{{ item.remote_as }}"
    update_source: Loopback0
    send_community: both
    enabled: yes
    state: present
```



redhat.

DECLARATIVE INTENT

Declared Configuration

Intended State

```
- name: configure interface
  net_interface:
    aggregate:
      name: GigabitEthernet0/2
      description: public interface configuration
      enabled: yes
      state: present
      status:
        state: connected
        tx_rate: ge(7Gbps)
        rx_rate: ge(2Gbps)
        delay: 30
        neighbors:
          - host: core-01
            port: Ethernet5/2/6
```

DECLARATIVE INTENT (CONT.)

CONFIGURATION

```
- name: configure bgp neighbor
  net_bgp_neighbor:
    peer: 1.1.1.1
    remote_as: 65000
    enabled: yes
```

Only perform configuration

Ignore resource state on the device

VALIDATE STATE

```
- name: validate bgp neighbor
  net_bgp_neighbor:
    peer: 1.1.1.1
    nbr_state: established
    pfx_rx: 16593
    pfx_tx: 132
```

Only perform state validation

Ignore configuration of the resource

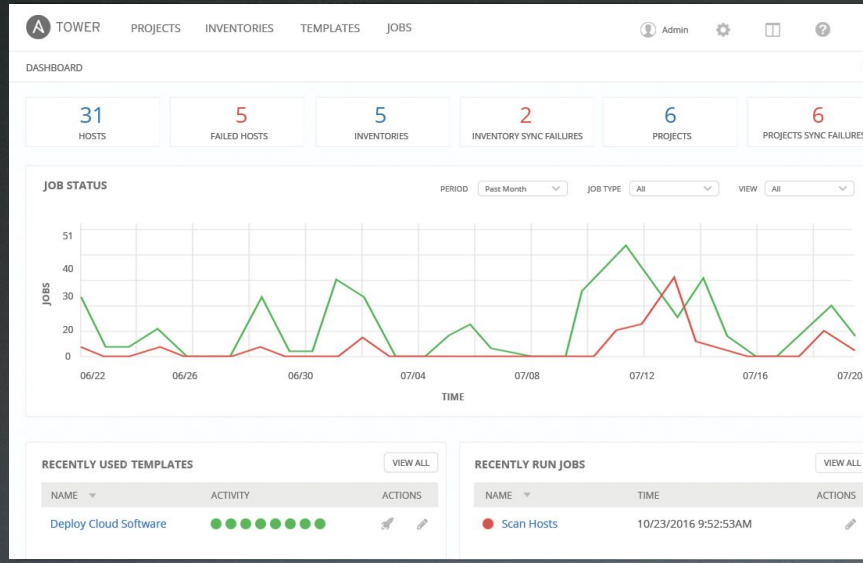
RED HAT® ANSIBLE® Tower 3.2

GA:: OCTOBER 2nd 2017

SCALE

MANAGE

CONNECT



ANSIBLE TOWER 3.2

SCALE

ISOLATED NODES
Move execution to remote and/or partially-disconnected networks

INSTANCE GROUPS
Guarantee automation capacity for user organizations

FACT UPDATES
Re-vamped fact scanning and caching
Reference cached facts directly from Playbook runs
Automatically update fact caches without separate Job Template runs

IMPROVED UX
Tower is Automation for Teams, and we've made it even easier to understand and use

ANSIBLE TOWER 3.2

MANAGE

SMART INVENTORY GROUPS

Create inventory-spanning host groups from any number of cached system or custom facts

Groups are automatically updated to include current host that match rules

DEEPER RED HAT INSIGHTS INTEGRATION

Directly run Playbooks provided by Insights

View, select, and apply specific Insights findings to desired hosts/inventories

SCM-CONTROLLED INVENTORY

Import and directly use a project-stored list as a dynamic inventory source

ANSIBLE TOWER 3.2

CONNECT

PLUGGABLE CREDENTIALS

Tower credentials can source secrets from other systems (i.e. CyberArk, etc.)
Custom inventory scripts can now access Tower credentials

NAMED URL ACCESS

Access Jobs via names in API, rather than numbers
Unifies management of numerous Tower systems

GA :: 19 SEPTEMBRE 2017

Installation of Ansible AWX - Using Minishift

Setup Minishift : https://docs.openshift.org/latest/minishift/getting-started/installing.html
Clone the repo
$ mkdir awx ; cd awx
$ git clone https://github.com/ansible/awx

Configure the installation
$ cd installer ; vim inventory
Add these line under the openshift install section

openshift_host=192.168.42.172:8443 (ip of your minishift environment)
awx_openshift_project=awx
openshift_user=developer
awx_node_port=30083

Launch the installation
$ eval $(minishift docker-env)
$ ansible-playbook -i inventory install.yml -e openshift_password=developer -e docker_registry_password=$(oc whoami -t)

Installation Minishift on Fedora 27

Download Minishift : https://github.com/minishift/minishift/releases

```
$ tar zcvf minishift-1.9.0-linux-amd64
$ cd minishift-1.9.0-linux-amd64
$ cp minishift /usr/bin
$ sudo dnf install libvirt
$ sudo curl -L \
https://github.com/dhiltgen/docker-machine-kvm/releases/download/v0.7.0/docker-machine-driver-kvm -o \
/usr/local/bin/docker-machine-driver-kvm

$ sudo chmod +x /usr/local/bin/docker-machine-driver-kvm
$ sudo usermod -a -G libvirt <username>
```