

Red Hat
Summit

Connect

Network Automation Unleashed

Summit Connect Dublin
Workshop Track

Ahmed Gaber
Snr. Solutions Architect
Red Hat

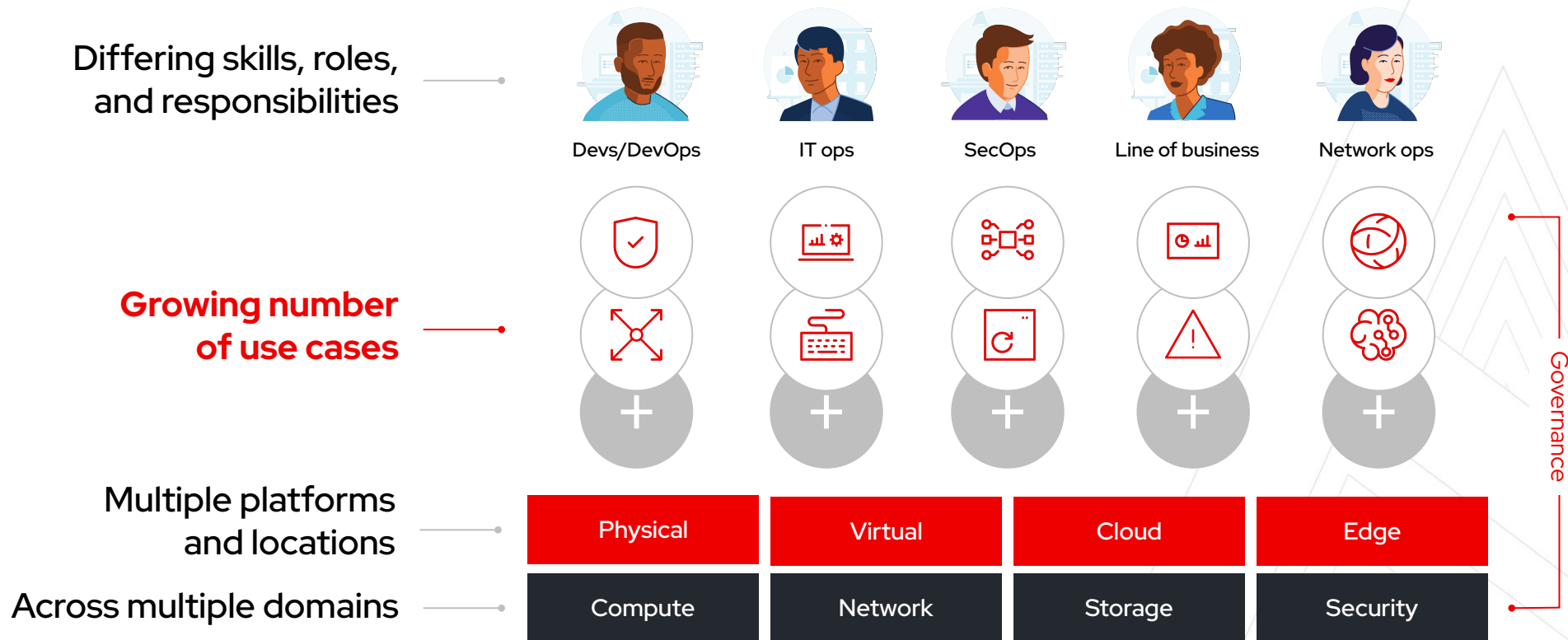
Michal Krzysztofowicz
Network Architect
Agile Networks



What we'll cover today...

- ▶ What customers face today
- ▶ How can Ansible Automation Platform help?
- ▶ What are the use cases covered by Ansible ?
- ▶ Introduction to Network Automation
- ▶ Workshop time
- ▶ Conclusion

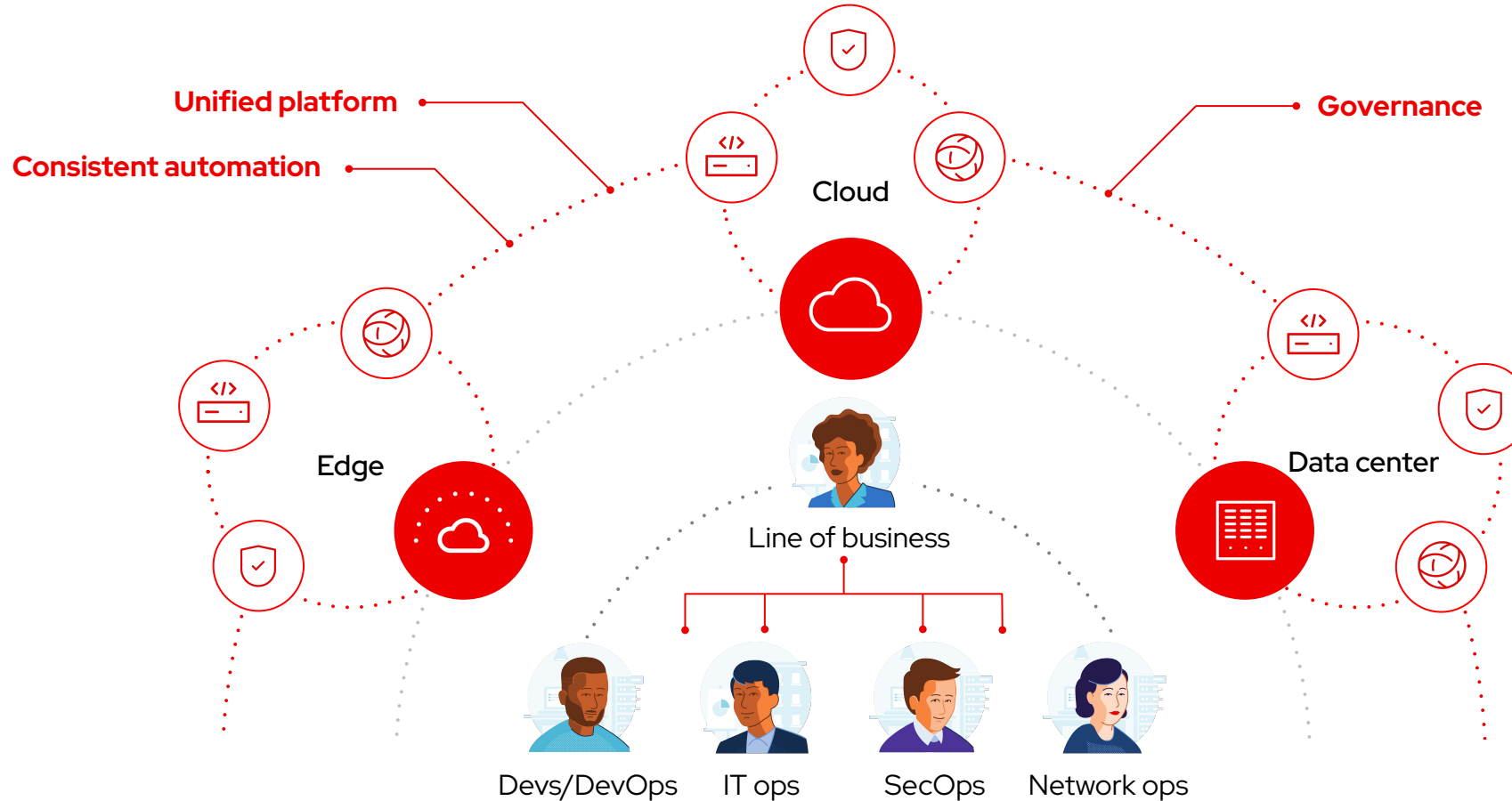
Many organizations share the same challenge.



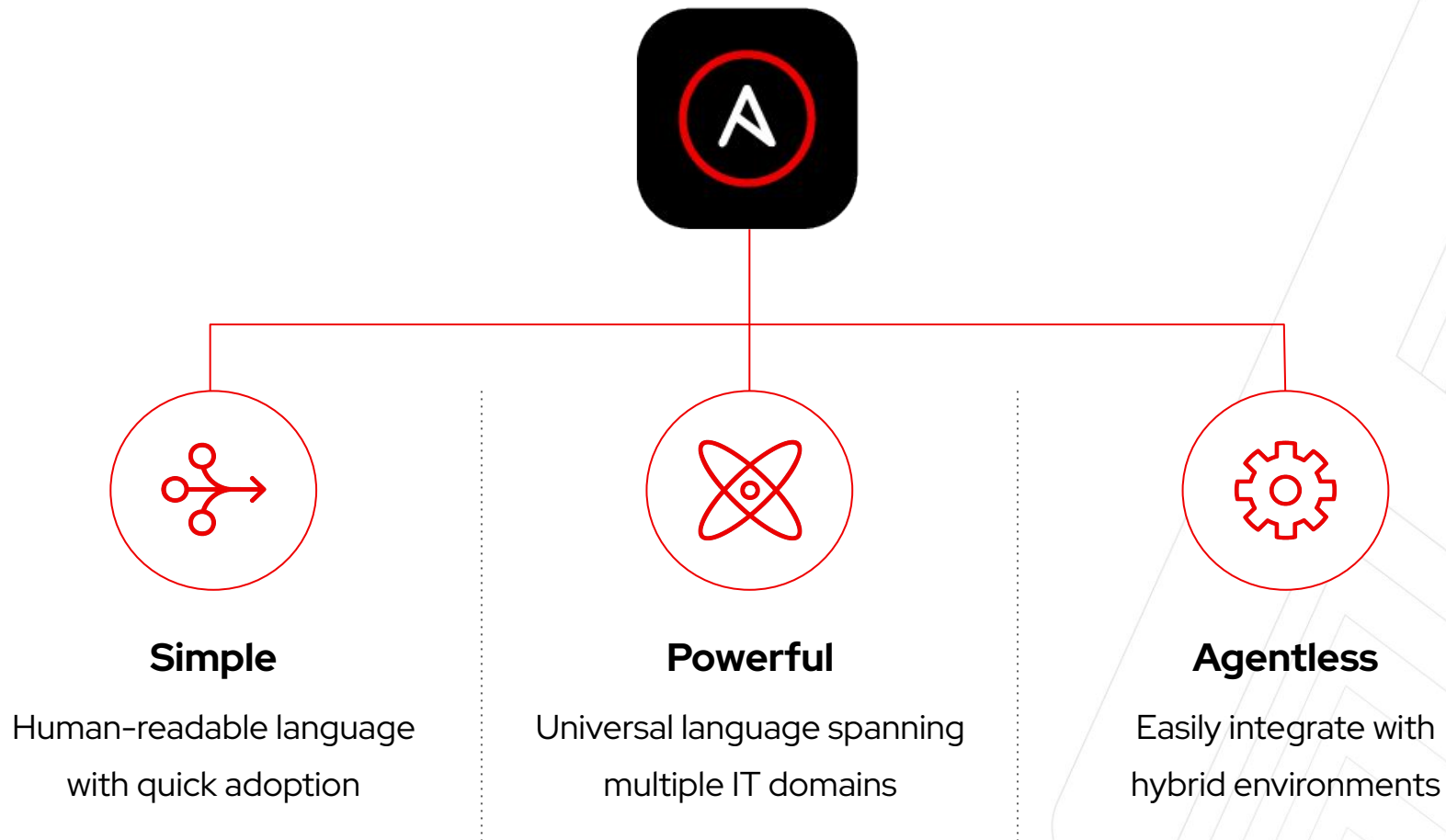


Anyone can automate...
but an enterprise needs to coordinate
and scale

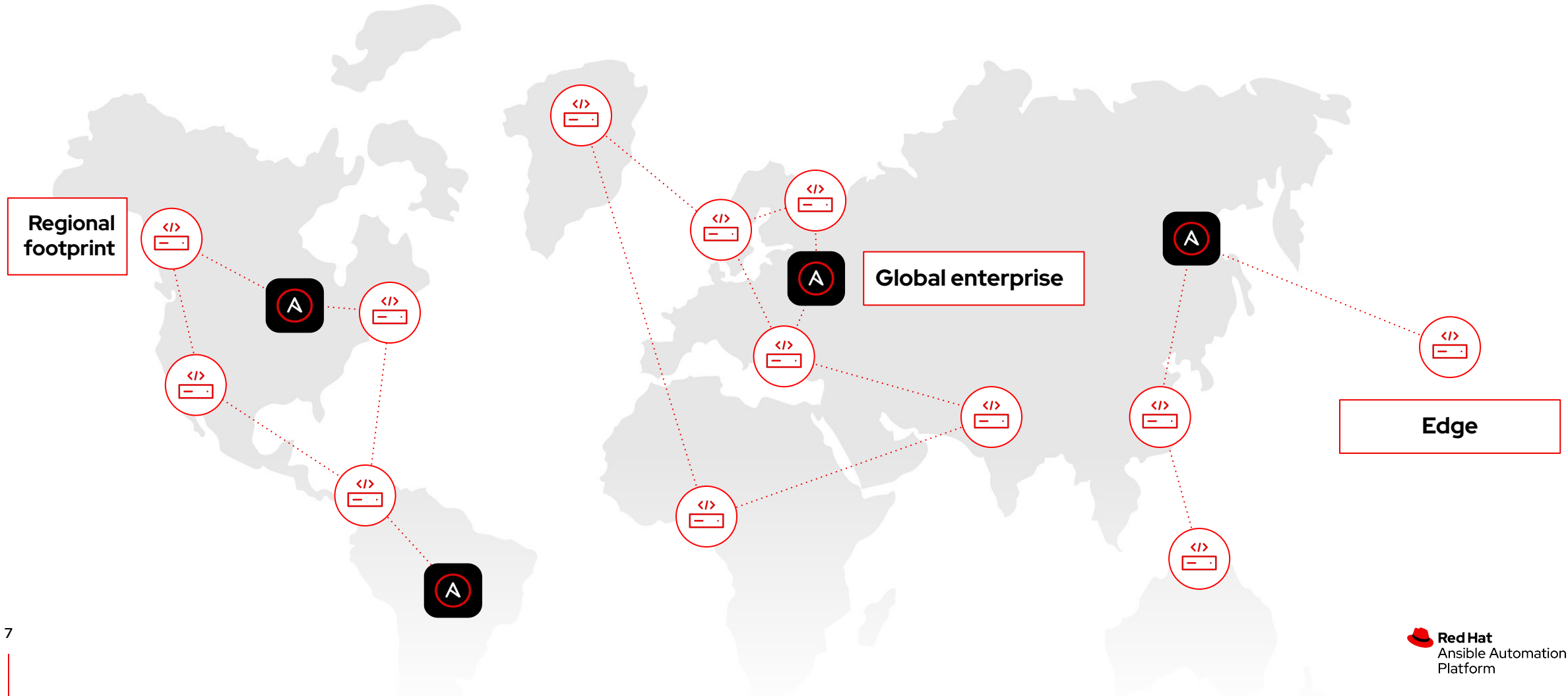
The solution? **Break down the silos.**



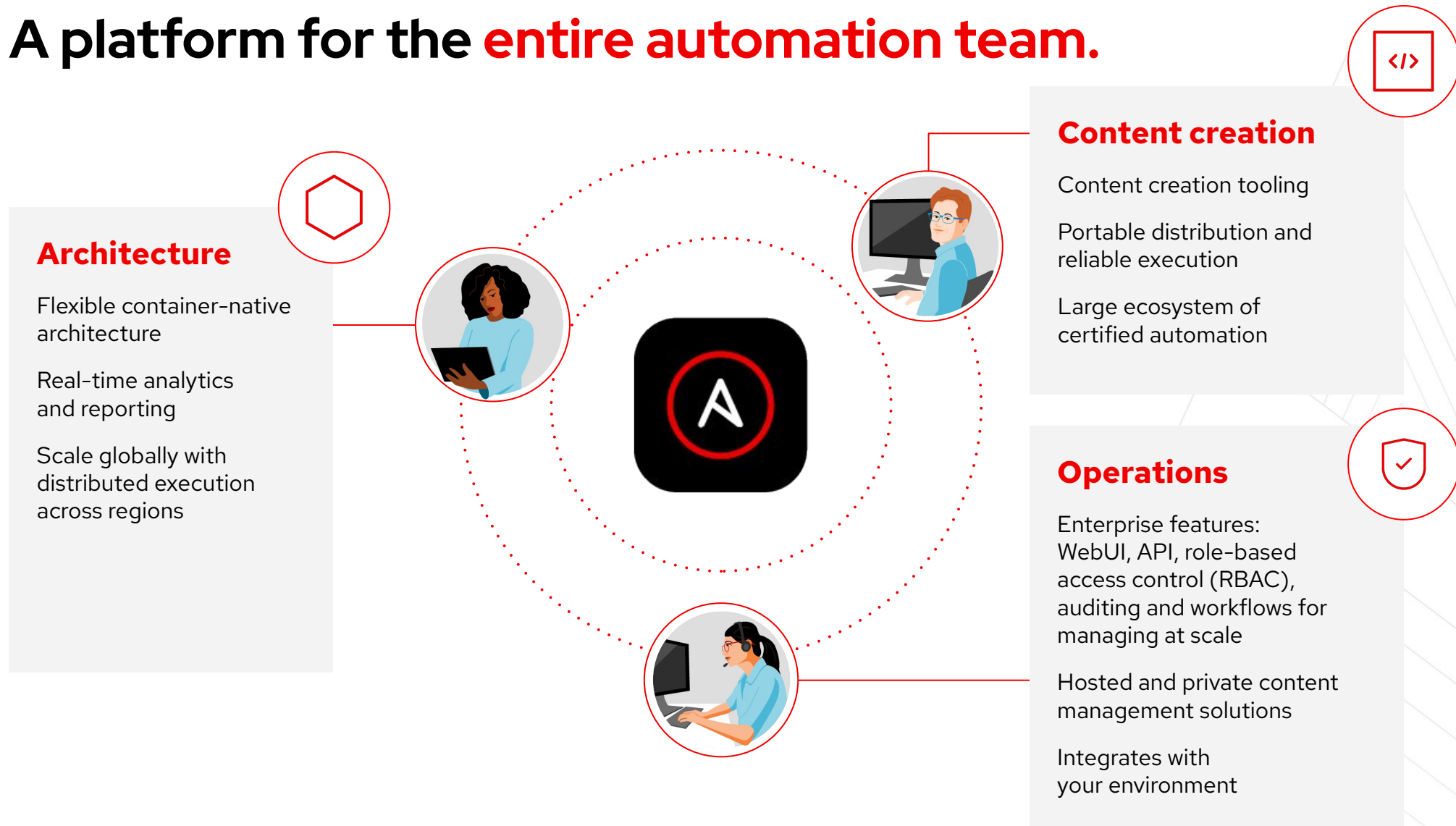
Ansible is the **de facto automation language.**



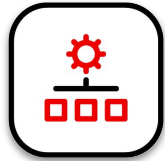
The flexibility to scale, **wherever that may be.**



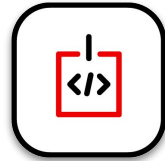
A platform for the **entire automation team.**



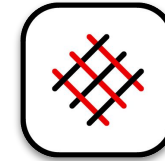
One subscription. **One integrated platform.**



Automation controller
Automation control plane

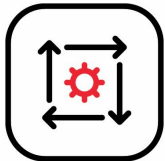


Automation execution environments
Scalable packaging and runtime execution plane

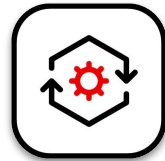


Automation mesh
Connectivity across diverse enterprise automation environments

NEW



Event-Driven Ansible
Automatic response to environment changes based on environment intelligence



Ansible-builder
Ansible containerized execution environment builder



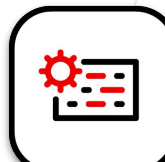
Automation analytics & Red Hat Insights
Visibility, predictive analytics, and more



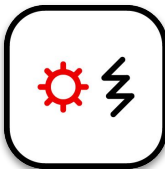
Ansible Content Collections
100+ certified content collections



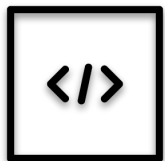
Automation hub
Hosted certified content repository.



Ansible-navigator
Execution environment orchestration tooling



Ansible Platform Operator
Package, deploy and manage this platform on Red Hat OpenShift



NEW

Ansible Lightspeed
Write and manage Ansible code using generative AI



Red Hat
Ansible Automation Platform

The capabilities you need **across your IT footprint.**



Applications

- ▶ DevOps
- ▶ CI/CD
- ▶ GitOps



Network

- ▶ Configuration management
- ▶ Infrastructure awareness
- ▶ Network validation



Cloud

- ▶ Orchestration
- ▶ Operationalisation
- ▶ Governance



Security

- ▶ Investigation enrichment
- ▶ Threat hunting
- ▶ Incident response



Infrastructure

- ▶ Deployment
- ▶ Provisioning
- ▶ Management



Edge

- ▶ Extend security
- ▶ Scalability
- ▶ Interoperability

Why and How to Automate?

- ▶ Automation happens when one person meets a problem they never want to solve again
- ▶ Many organisations share the same challenge:
too many unintegrated, domain-specific tools
- ▶ **Ansible Automation Platform**
 - **Powerful** - orchestrate complex processes at an enterprise scale
 - **Simple** - simplify automation creation and management across multiple domains
 - **Agentless** - easily integrate with hybrid environments
 - Fuelled by **Open Source Community**

Ansible Automation Platform

- ▶ **More than Ansible** - provides the whole ecosystem and a suite of products
- ▶ Uses **familiar Ansible playbooks, tasks, roles, collections** and other constructs for easy creation of automation
- ▶ Unified, common **Web GUI**, which can be used by unrelated teams (sysadmins, network engineers, developers, operators, service provisioning engineers)
- ▶ **Role-Based Access Control**
- ▶ **Certified Vendor Collections** (Cisco, Juniper, etc)
- ▶ Stable **REST API**
- ▶ **Event-Driven Automation**
- ▶ **Workflows** - run automation, and depending on its result run another one (e.g. to roll back or continue)
- ▶ Red Hat and Partner **Support**

Anatomy of Ansible Playbook

- ▶ A **Playbook** contains a list of **Plays**
- ▶ Each **Play** comprises one or more **Tasks** and/or references to **Roles** and **Collections**. A Play is a top-level specification for group of tasks.
- ▶ **Plugins** are pieces of code that augment Ansible's core functionality and are used to enable a rich, flexible, expandable feature set
- ▶ **Modules** are parametrised components with internal logic representing a single step to be done; they "do" things in Ansible; usually written in Python
- ▶ **Inventory** is a list of systems in your infrastructure that automation is executed against. It supports grouping of systems in a hierarchy of groups and hosts.

Anatomy of Ansible Playbook

- ▶ **Tasks** describe individual units of work, e.g. create directory, change a line in a file, copy a file over to the remote. Tasks use **Modules** to achieve their goals
- ▶ **Roles** are reusable automation actions logically grouped together to perform a more complex job. They make use of a series of tasks to accomplish this. Write roles once and share them with others who have similar challenges in front of them.
- ▶ **Collections** are a data structure containing automation content:
 - Modules, Playbooks, Roles, Plugins, Docs, Tests

Anatomy of Ansible Playbook

- ▶ **Variables** can be defined at multiple levels and are a way of parametrising Ansible actions by allowing substitution. Often these are stored in **group_vars** and **host_vars** files. Ansible is smart about variable overriding.
- ▶ **Templates** are a way of creating text files with some content parametrised to make use of variables, as well as supporting some processing (e.g. conditionally generating a part of the text file). Jinja2 is the templating engine used in Ansible
- ▶ **YAML** is the file format used for vars files, playbooks, roles, etc.

Anatomy of Ansible Playbook

```
# vim:ft=ansible:
#
# Inventory File

[lab:vars]
ansible_network_os=junipernetworks.junos.junos

[lab:children]
upstream
ispnet

# Upstream Carriers
[upstream]
big-isp-1

# Core Network
[ispnet:children]
route_reflectors
core_routers
edge_routers
cpe

# Route Reflectors
[route_reflectors]
rr1
[core_routers]
p1
```

```
# vim:ft=ansible:
#
# Group Vars
---

generated_configs_dir: generated-configs

domainname: lab.agile.intra
system_time_zone: "Europe/Dublin"

as_number: 65535

ipv4_allocations:
- 10.0.0.0/16
ipv6_allocations:
- "fd8e:8d8d::/32"

prefix_lists:
- name: LOOPBACKS-v4
  prefixes:
  - 10.0.0.0/24
- name: LINKS-v4
  prefixes:
  - 10.0.1.0/24
- name: MANAGEMENT-ACCESS-v4
  prefixes:
  - 10.255.255.0/24
```

```
# vim:ft=ansible:
#
# Host Vars file
---

host:
  net: 49.0001.0000.0003.00

  management:
    interface: fxp0
    description: OOB Management
    ipv4: dhcp

  loopback:
    lo0:
      description: Management Loopback
      ipv4: "{{ loopback_pfx_v4 }}.3"
      ipv6: "{{ loopback_pfx_v6 }}::3"

  interfaces:
    - interface: ge-0/0/0
      description: "p1:ge-0/0/1"
      ipv4: 10.0.1.3/31
      ipv6: "fd8e:8d8d:1:2::2/64"
      mpls: true
      iso: p2p
```


Anatomy of Ansible Playbook

```
# vim:ft=ansible:  
#  
# The Playbook  
#  
# This playbook generates the configs, and optionally pushes them to the network  
---
```

```
- name: generate the configs locally
```

```
hosts: lab  
connection: local  
gather_facts: no  
vars_files:  
  - vault.yaml
```

```
roles:
```

```
- make-directories  
- delete-config  
- generate-junos-configs  
- assemble-junos-configs
```

```
- name: push configs to network
```

```
hosts: lab  
gather_facts: no  
connection: netconf  
remote_user: agile
```

```
tags:
```

```
- never  
- push
```

```
roles:
```

```
- push-configs
```

Modes of Operation

Module code is executed locally on the control node



Network Devices /
API Endpoints

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

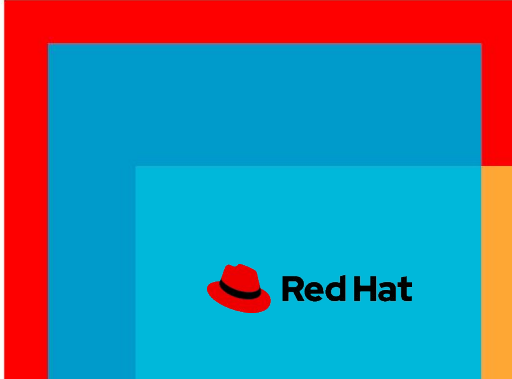
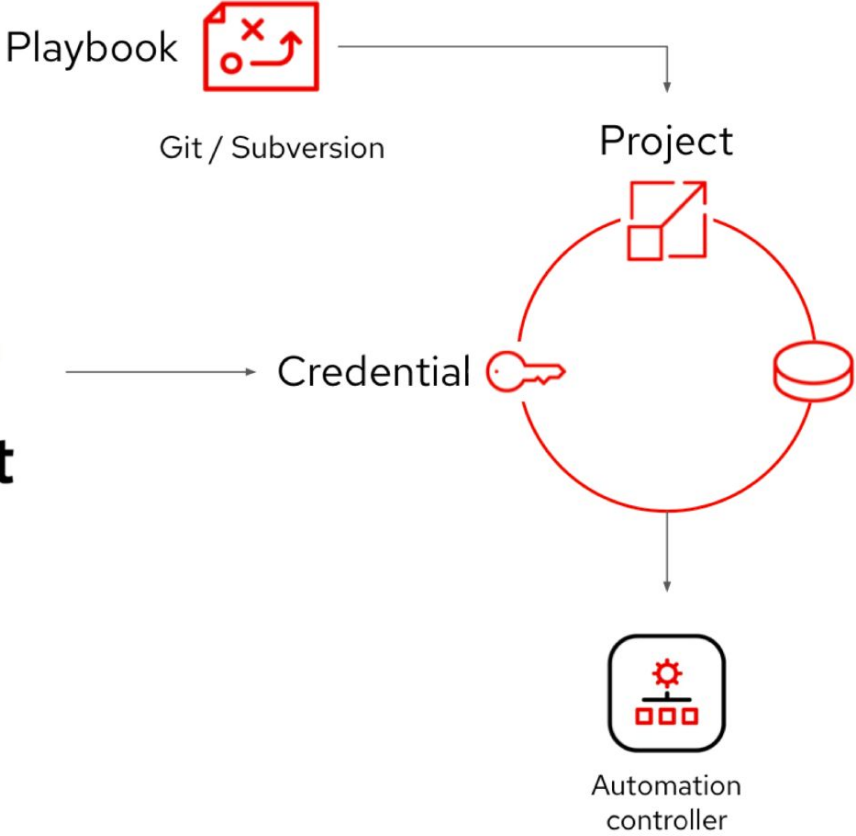


Linux / Windows
Hosts

Ansible Automation Controller

- ▶ **Push Button** - Intuitive, graphical user interface which makes it easy for novice users to execute playbooks you allow them access to
- ▶ **RESTful API** - API first mentality; every feature and function of the controller can be API driven
- ▶ **RBAC** - Restrict playbook access to authorised users. Supports organising users into teams and assigning different access rights (read-only, admin)
- ▶ **Enterprise Integrations** - integrate with enterprise authentication like TACACS+, RADIUS, Azure AD; Supports OAuth 2; Supports notifications with PagerDuty, Slack or Twilio
- ▶ **Centralised Logging** - securely log all automation activity (who, what, when). All viewable later or exported through API
- ▶ **Workflows** - chain any number of playbooks together to build complex deployments, including conditional execution, clean up jobs, etc.

Ansible Automation Controller



Network Automation Workshop



<https://www.redhat.com/en/interactive-labs/ansible>

Red Hat
Summit

Connect

Thank you



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



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



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



twitter.com/RedHat