

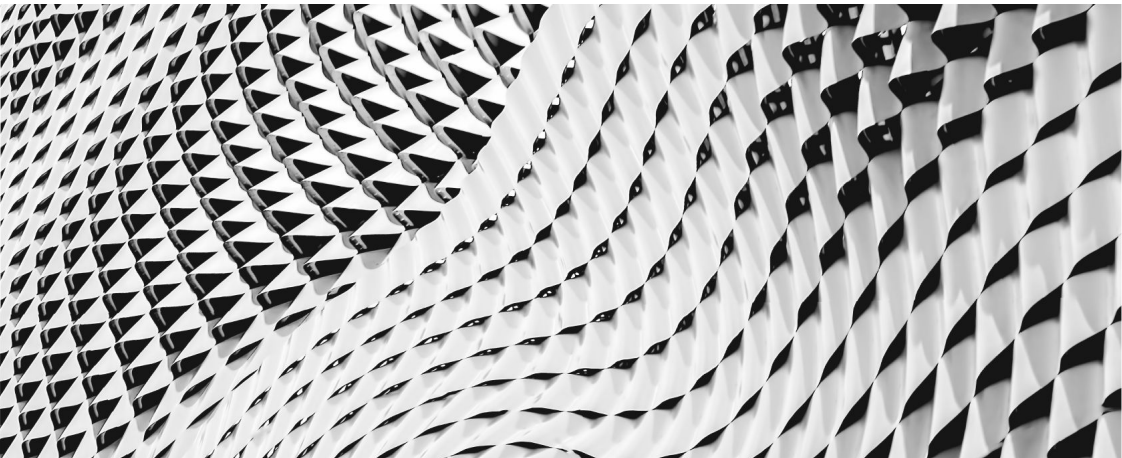
A N S I B L E

What's new in Ansible 2.10

Patrick Toal
Sr. Solutions Architect
Red Hat Canada

Background

The challenges that come
with success.



The Problem

ansible / ansible

Used by 9.8k Watch 2k Star 41.4k Fork 18k

Code Issues 4,294 Pull requests 2,032 Actions Projects 26 Security Insights

Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts or custom code to deploy and update your applications — automate in a language that approaches plain English, using SSH, with no agents to install on remote systems. <https://docs.ansible.com/ansible/> <https://www.ansible.com/>

python ansible

48,915 commits 43 branches 0 packages 299 releases 4,865 contributors GPL-3.0

The Problem

Customers	Developers
<ul style="list-style-type: none">● Support claim confusion, and who does what and where for issues/PRs?● Need for stability (longer life cycle) for foundation components while...● Need new module enablement / platform updates (shorter life cycle)	<ul style="list-style-type: none">● 4,300+ open issues, 2,000 open pull requests● “One size fits all” doesn’t work for all content sets● Some things go in quickly, some things don’t

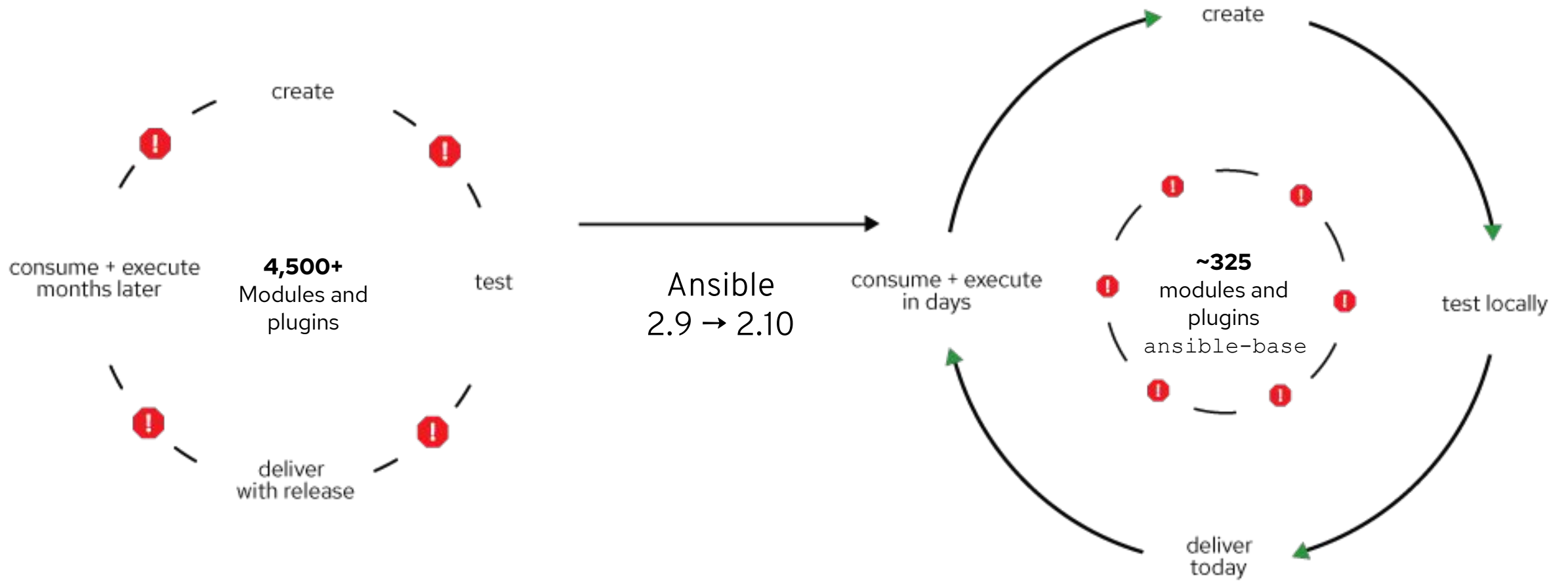
Different Goals and Needs

The needs of Community and Enterprise users are not always aligned.

A new structure is required to better meet the needs.

Community	Developer	Enterprise
Easy to Access	Easy to Develop	Easy to get Support
Simple to use / get started	Simple to commit, change, release	Simple to consume and deploy at scale
Standalone Tools	Ability to integrate	Integrated Platform
Rapid Innovation	Independent Innovation	Stable Integrations and Platform
Community Project	Community and/or Enterprise	Enterprise Product

The Plan



Ansible 2.10 will still install with all the plugins

Ansible Base

- Ansible “engine”
- Core Plugins
- Stability

Ansible Community Collection

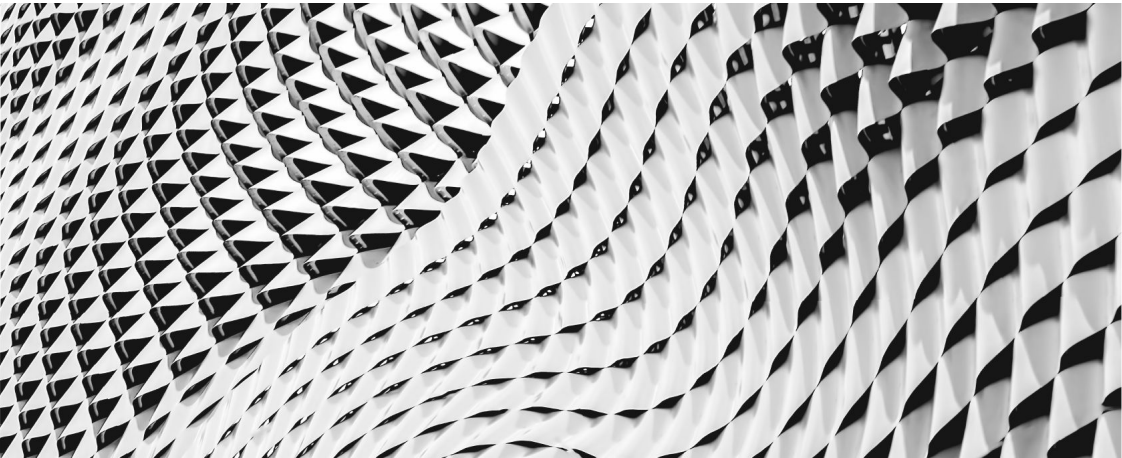
- All existing plugins /
collections
- “Slow and steady”

Ansible Repositories

Maintaining Ease of Consumption

Ansible Base	Ansible Community Collection	Ansible Community Distribution
Core Engine	A package of all current modules	Ansible Base + Ansible Community Collection
Core Plugins / Modules	Will be curated	Can be installed with pip install ansible
Main Goal: Stability	Main Goal: Maintain all existing modules in a single package	Main Goal: Maintain ease of use and "Batteries Included"

Collections



The collections framework gives us flexibility.

Start using collections now.

INTRODUCING THE ANSIBLE CONTENT COLLECTION

Simplified and consistent content schema



- A standardized way to organize and package Ansible content
- Include roles, modules, module utilities, plugins, documentation
- Semantic versioning
- Portable and flexible delivery

Creating a Collection

```
ptoal% ansible-galaxy collection init ptoal.example
- Collection ptoal.example was created successfully
ptoal% tree ptoal
ptoal
├── example
│   ├── README.md
│   ├── docs
│   ├── galaxy.yml
│   ├── plugins
│   │   └── README.md
│   └── roles
4 directories, 3 files
```

Developer docs: https://docs.ansible.com/ansible/devel/dev_guide/developing_collections.html

Folder Structure

```
collection/
├── docs/
├── galaxy.yml
├── plugins/
├── Deps
│   ├── modules/
│   │   └── module1.py
│   ├── inventory/
│   └── .../
├── README.md
├── roles/
│   ├── role1/
│   ├── role2/
│   └── .../
├── playbooks/
├── files/
├── vars/
├── templates/
├── tasks/
└── tests/
Write Tests!
```

<-- Name of collection
<-- Not optional. :)
<-- Metadata about collection
<-- Inventory / Ansible Modules +
<-- Also not optional. :)
<-- Ansible Roles
<-- Ansible Playbooks
<-- Also not optional. Really.

Step 2: Allow distribution of content more openly and freely, with support claims much more clear

Ansible Galaxy

galaxy.ansible.com

- Community supported
- Extended to leverage Collections framework
- “Latest and greatest”

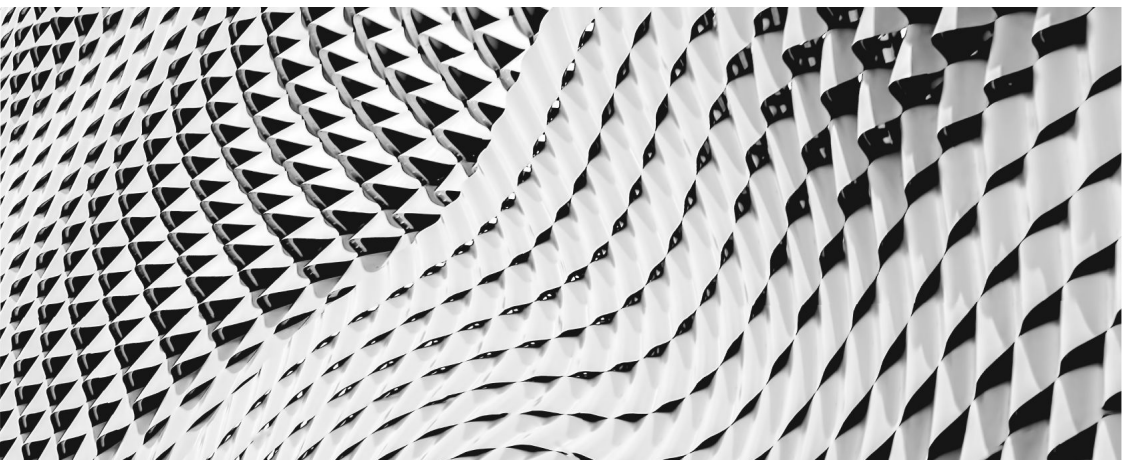
Ansible Automation Hub

cloud.redhat.com

- Certified, jointly supported by Red Hat and Partner
- Access to advanced analytics
- “Slow and steady”

Collections in Practice

Practical examples of
collections



Accessing the right module (Namespaces)

```
hosts: somehosts
collections:
  - tima.pinger
  - redhat.open_ping

tasks:
  - tima.pinger.ping:

  - ansible.builtin.ping: # use only the ping packaged in core

  - ansible.legacy.ping: # use core or library(etc)/ping.py
    when: thing | tima.pinger.filter == 42

  - ping: # searches collections "path" otherwise...
          # still works, == ansible.legacy.ping:
```

Accessing the right module (Namespaces)

```
hosts: somehosts
collections:
  - tima.pinger
  - redhat.open_ping

tasks:
  - tima.pinger.ping:

  - ansible.builtin.ping: # use only the ping packaged in core

  - ansible.legacy.ping: # use core or library(etc)/ping.py
    when: thing | tima.pinger.filter == 42

  - ping: # searches collections "path" otherwise...
          # still works, == ansible.legacy.ping:
```


Managing multiple collections (requirements.yml)

```
---  
collections:  
# With just the collection name  
- netbox_community.ansible_modules  
  
# With the collection name, version, and source options  
- name: azure.azcollection  
  version: 0.0.2  
  source: automation_hub
```

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