



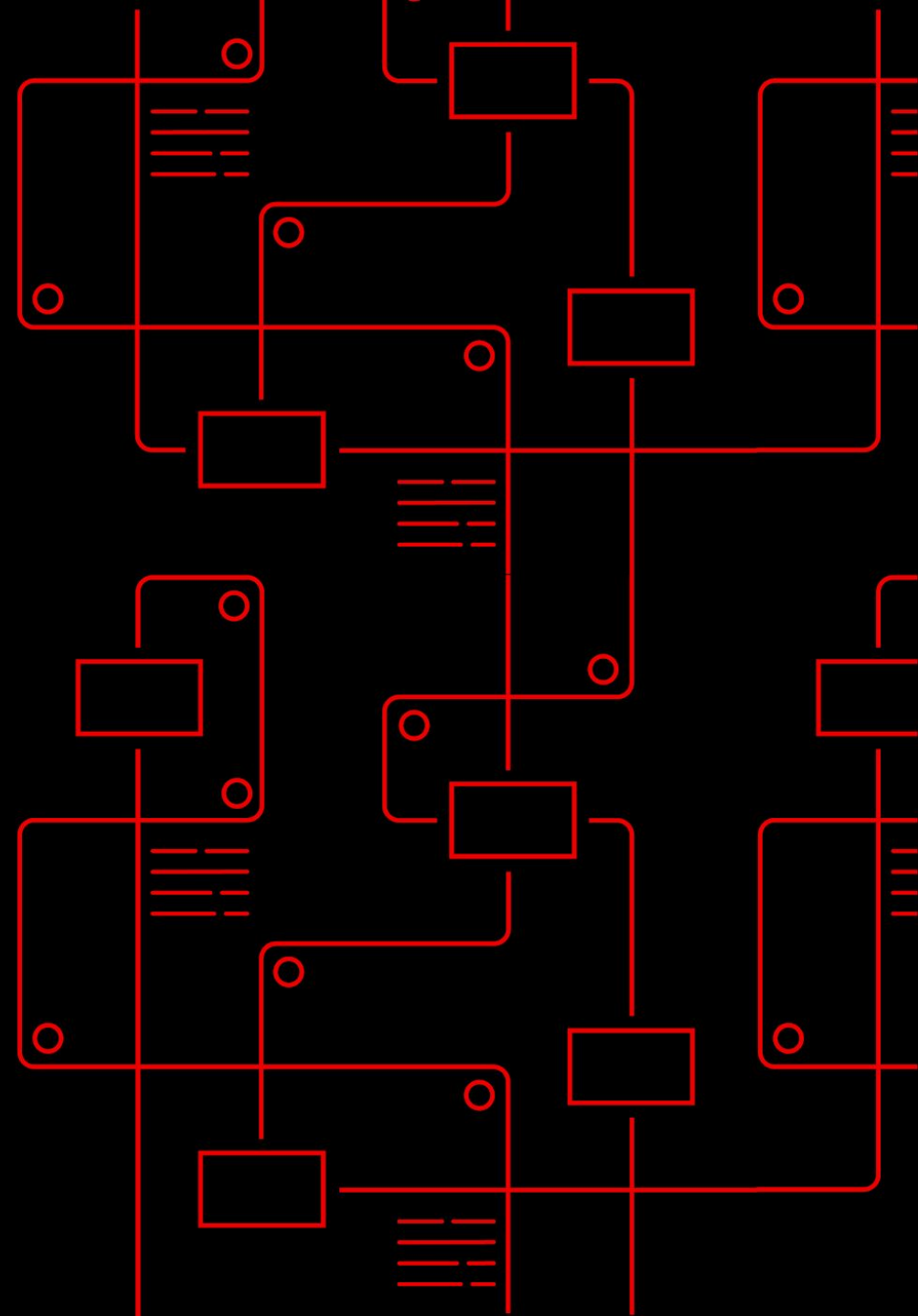
Red Hat Ansible Automation Platform

Developer Tools

Brad Krumme

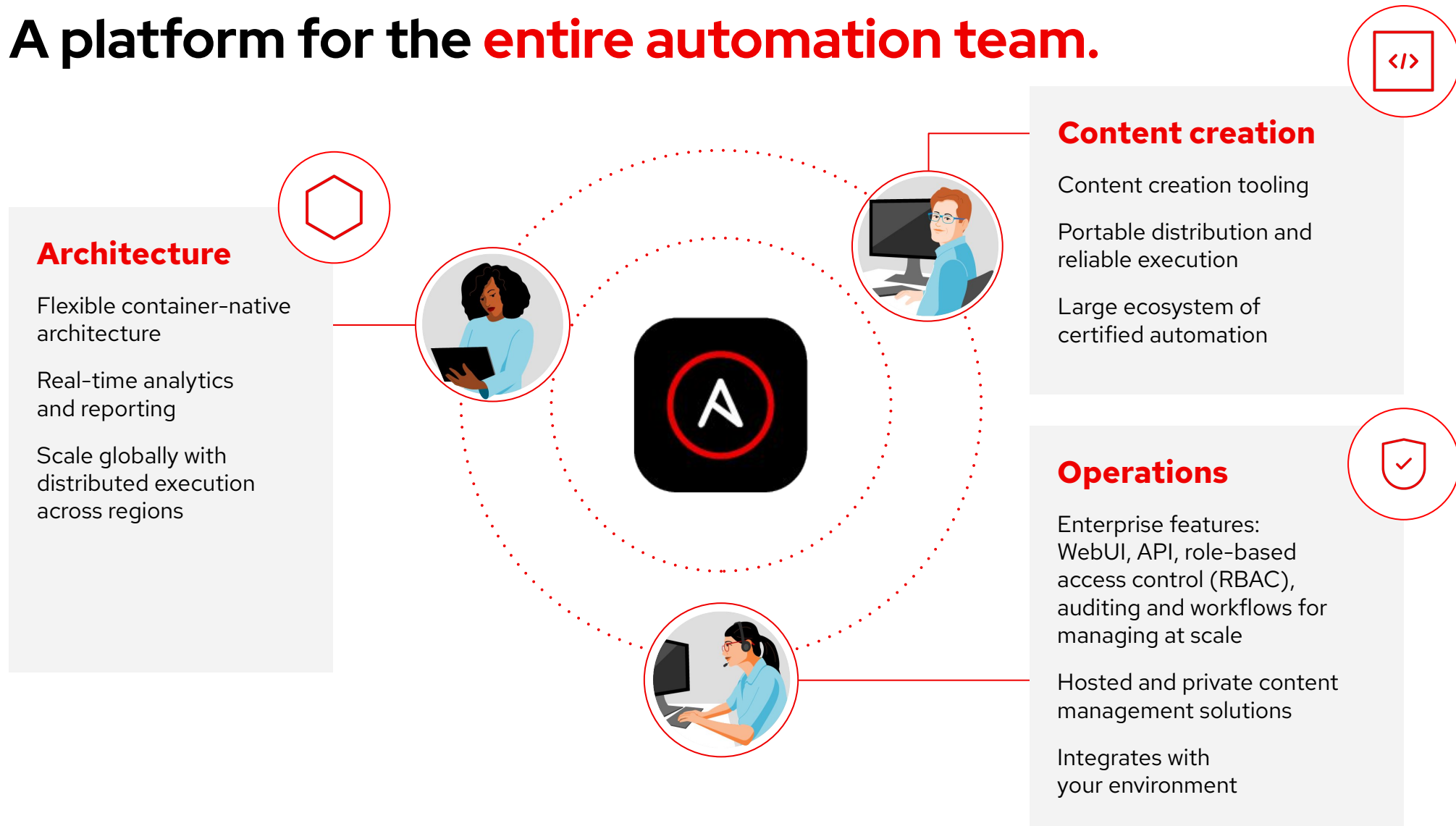
Senior Specialist Solution Architect - Automation



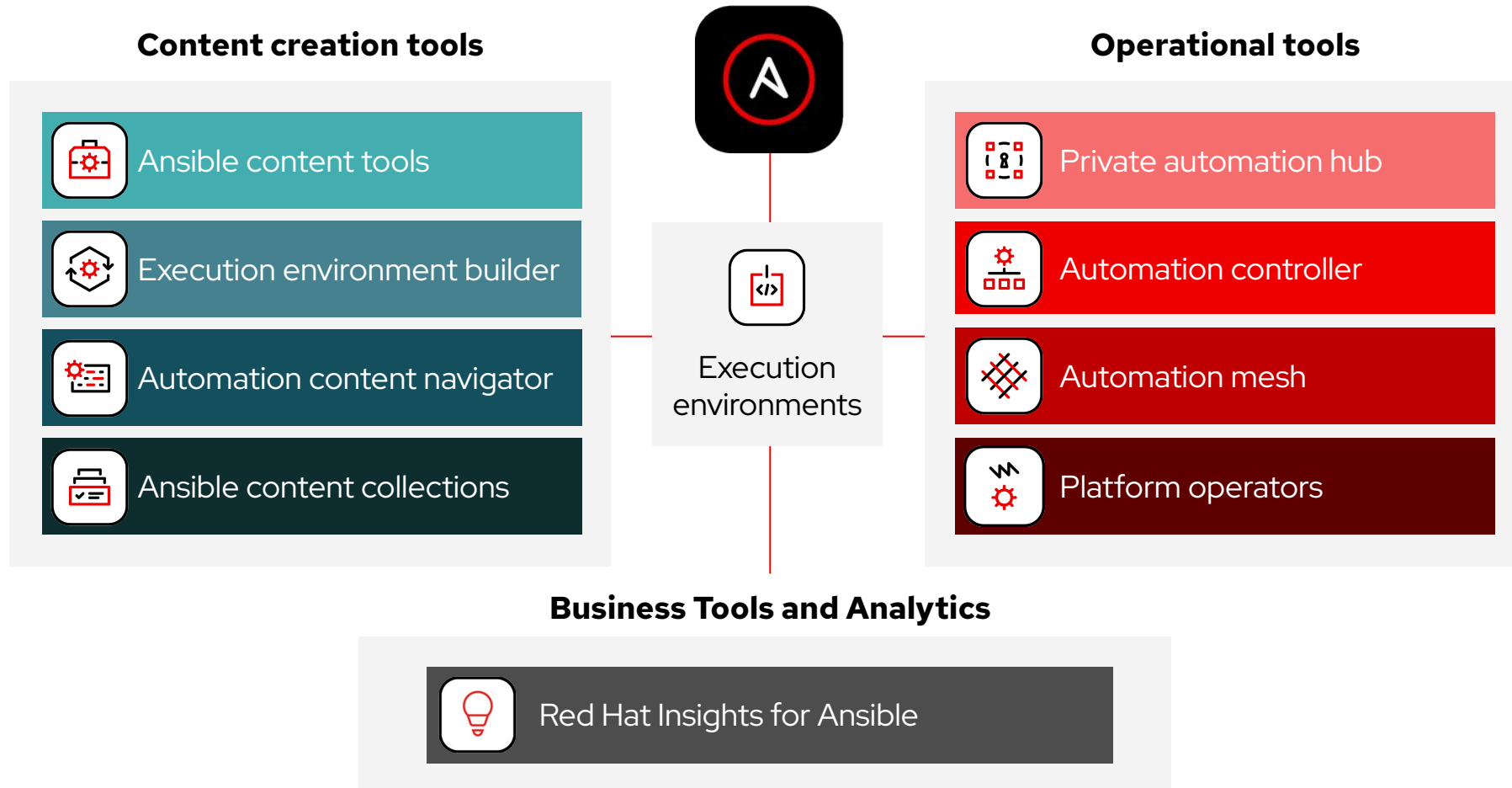


What is Red Hat[®] Ansible[®] Automation Platform?

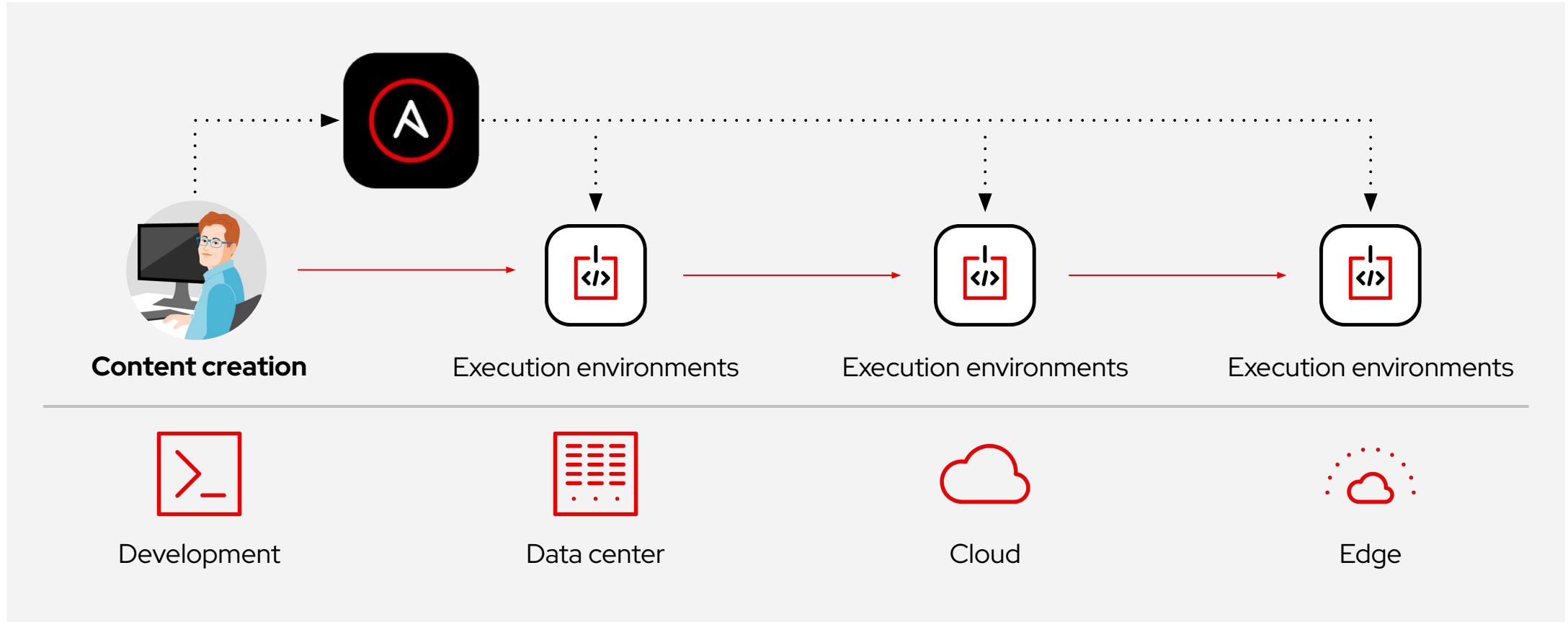
A platform for the **entire automation team.**



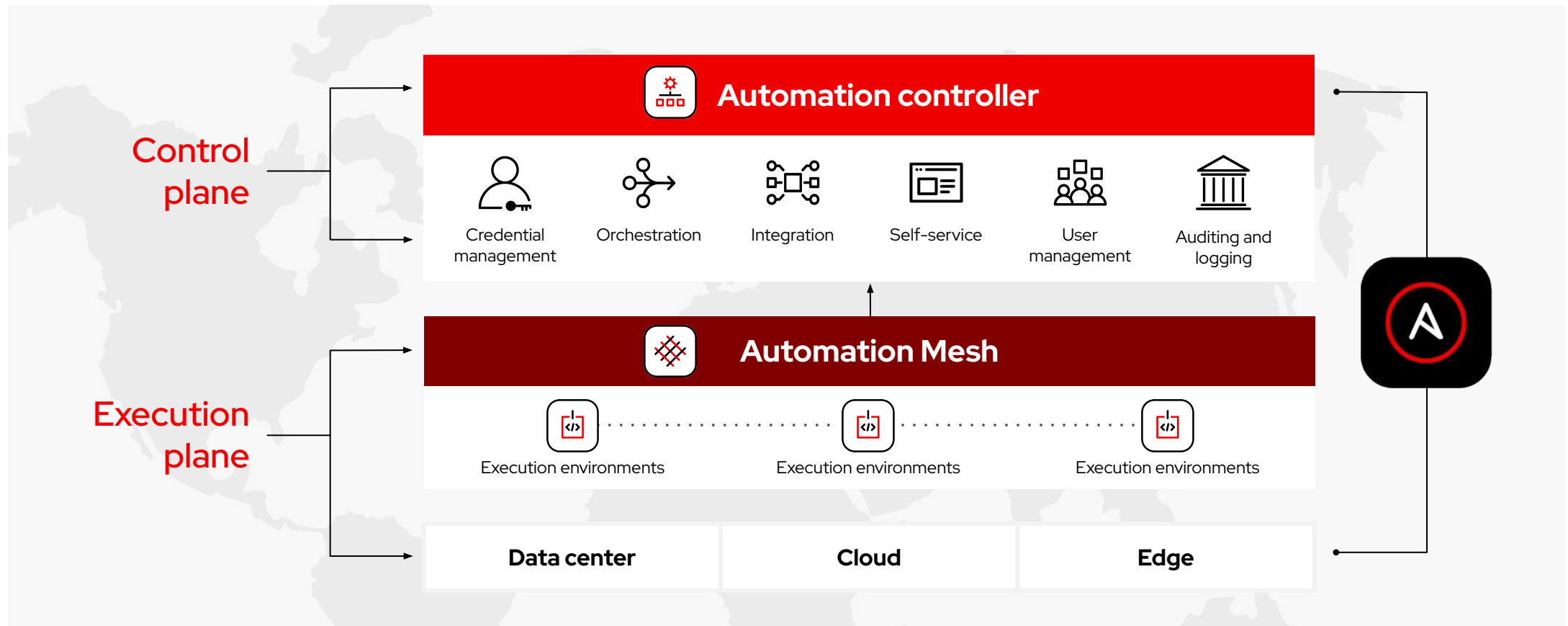
An integrated solution **for the enterprise.**



Built for consistency. **Portability is reliability.**



A distributed architecture **built for scale.**



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

Enabling your automation team **to consistently...**



Create

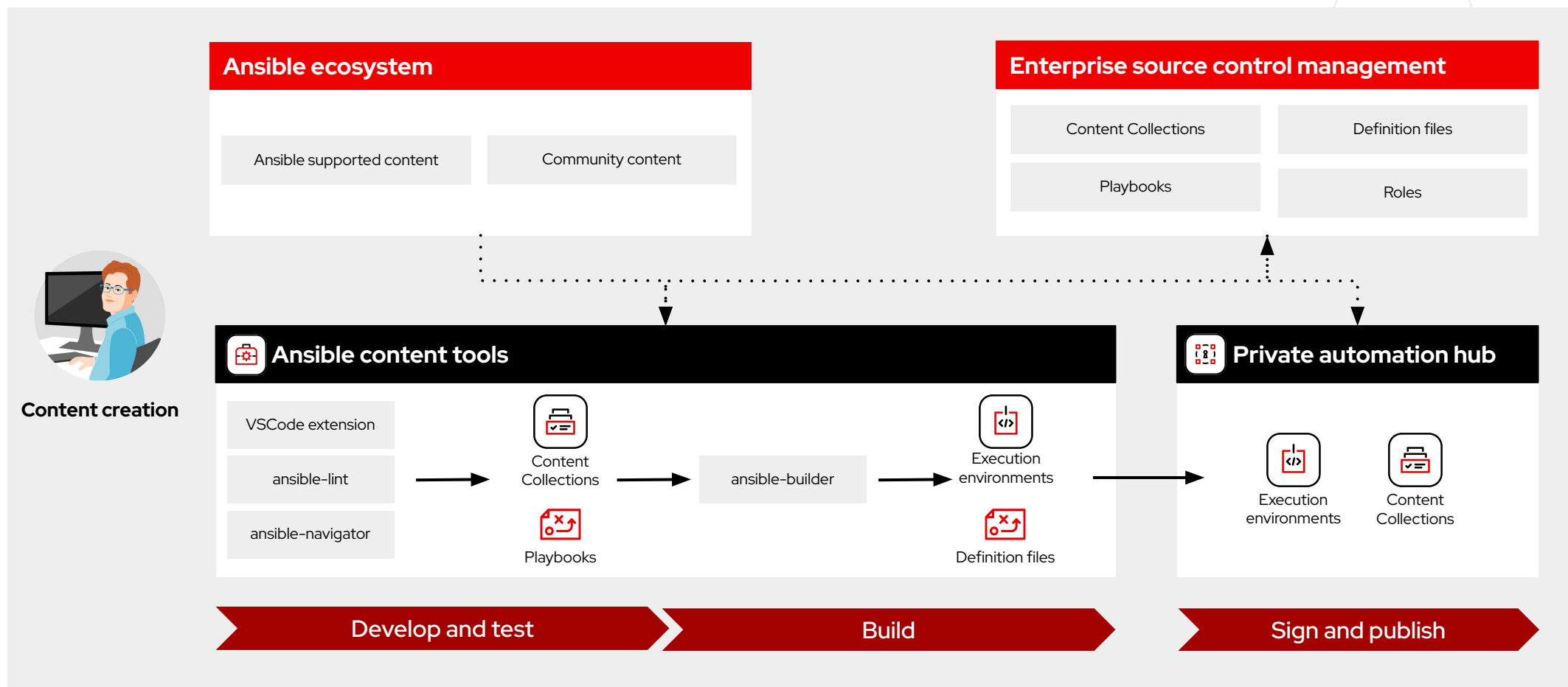
Manage

Scale

Create



The automation content life cycle. **Create.**



Ansible Core (**ansible-core**)



What is it?

- ▶ The main building block for Ansible
- ▶ Simple YAML syntax to develop Ansible Playbooks
- ▶ Provides CLI tools to develop, test and run playbooks
- ▶ Pluggable architecture that allows extensions through Content Collections

```
---
- name: Shutdown VM guest
  hosts: localhost
  gather_facts: false
  tasks:
    - name: Turn off specified VM guest
      vmware.vmware_rest.vcenter_vm_guest_power:
        state: shutdown
        vm: 1021343
        vcenter_hostname: vcenter.demoredhat.com
        vcenter_username: admin
        vcenter_password: tedlasso
```

Ansible playbooks

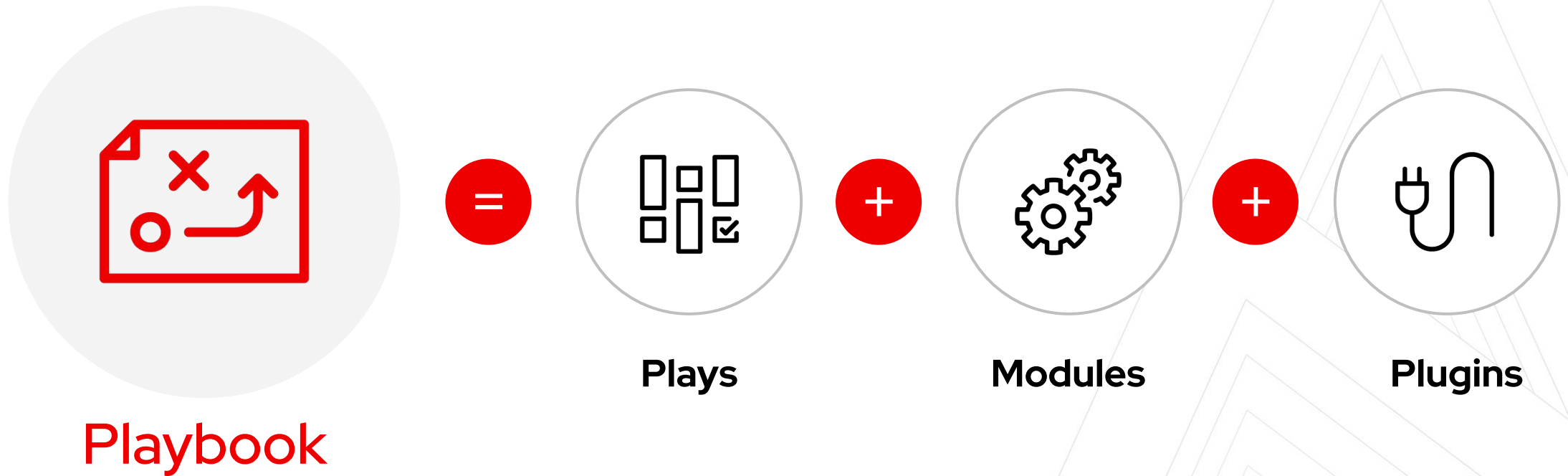
```
---
- name: Install and start apache
  hosts: web
  become: true

  tasks:
    - name: Ensure the httpd package is installed
      ansible.builtin.yum:
        name: httpd
        state: present

    - name: Create the index.html file
      ansible.builtin.template:
        src: files/index.html
        dest: /var/www/html/

    - name: Start the httpd service if needed
      ansible.builtin.service:
        name: httpd
        state: started
```

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: Ensure the httpd package is installed  
  hosts: web  
  become: true
```



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 created in Python, or Powershell for Windows setups, but can be developed in any language

```
- name: Create the index.html file
  ansible.builtin.template:
    src: files/index.html
    dest: /var/www/html/log
```



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: true  
  
Example filter plugins:  
{ { some_variable | to_nice_json } }  
{ { some_variable | to_nice_yaml } }
```


Ansible Roles. Reusable automation actions.



What are they?

- ▶ Group 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

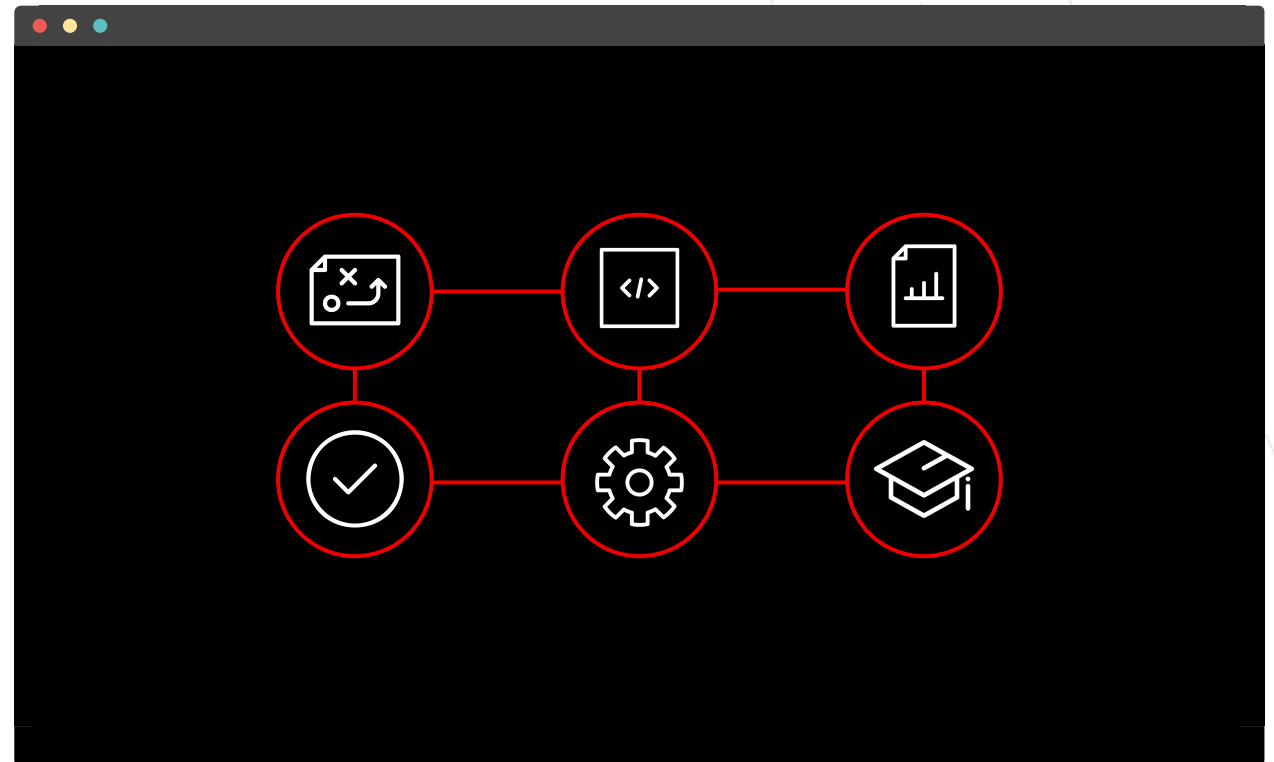
```
---
- name: Install and start apache
  hosts: web
  ansible.builtin.roles:
    - common
    - webservers
```

Content Collections.

Simplified, consistent content delivery.

What are they?

- ▶ Group 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



Ansible content tools: **Workflow**



Ansible content tools

VSCode extension

ansible-lint

ansible-navigator

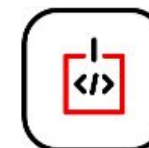


Content
Collections



Playbooks

ansible-builder



Execution
environments



Definition files

Develop and test

Build



Ansible VS Code extension

Simplifying content creation

What is it?

- ▶ Syntax highlighting of keywords such as module names
- ▶ Live validation of your code while you type
- ▶ Integration with ansible-lint*
- ▶ Autocompletion on play, block or task contents etc
- ▶ Documentation references as you code

```
- name: query incident number and creation time
  set_fact:
    incident_list: '{{ incident_list + [{"number": item.number, "opened_at": item.opened_at}] }}'
  loop: "{{ incidents.records }}"
  when: incidents

- name: Create a problem from incident
  proble
```

servicenow.itsm.problem

- servicenow.itsm.problem_info
- servicenow.itsm.problem_task
- servicenow.itsm.problem_task_info
- ansible.builtin.proxmox
- ansible.builtin.portage
- ansible.builtin.proxmox_kvm
- ansible.builtin.profitbricks
- ansible.builtin.portinstall
- ansible.builtin.proxysql_backend_servers
- ansible.builtin.profitbricks_nic
- ansible.builtin.proxmox_template

servicenow.itsm

Manage ServiceNow problems

Description

- Create, delete or update a ServiceNow problem.
- For more information, refer to the ServiceNow problem management documentation at https://docs.servicenow.com/bundle/paris-it-service-management/page/product/problem-management/concept/c_ProblemManagement.html.

Ansible VS Code extension

Install from VS Code Extensions Menu



The screenshot shows the Visual Studio Code interface with the Extensions Marketplace open. The search results for 'Ansible' are displayed on the left, and the details for the 'Ansible' extension by Red Hat are shown on the right.

EXTENSIONS: MARKETPLACE

Search: Ansible

- Ansible** (Red Hat) - 285K stars, 4 stars, Install
- [DEPRECATED] Ansible (Tomasz Maciążek) - 57K stars, 5 stars, Install
- ansible-vault (Eric Ho) - 51K stars, 3.5 stars, Install
- Azure REST for Ansible (Zim Kalinowski) - 12K stars, 5 stars, Install
- VSCode snippets for Ansible (Mattias Baake) - 11K stars, 5 stars, Install
- ansible-vault-inline (wolfmah) - 15K stars, 4 stars, Install
- Ansible Studio (Virtusa Corp) - 801 stars, Install
- NeoSoft-Ansible-Pack (Thomas Sanson) - 361 stars, 5 stars, Install
- ansible task container executor (daniloprevides) - 1K stars, Install
- ansible-server-sites (Stanislav Popov) - 3K stars, Install
- C2Platform Ansible Snippets (C2Platform) - 96 stars, Install
- auto-completion shell-ansible - 855 stars, Install

Extension: Ansible - Visual Studio Code

Ansible v1.2.44
Red Hat | 285,894 | 4 stars (23)
Ansible language support
Install

DETAILS | FEATURE CONTRIBUTIONS | CHANGELOG | DEPENDENCIES

Ansible VS Code Extension by Red Hat

This extension adds language support for Ansible to Visual Studio Code and OpenVSX compatible editors by leveraging ansible-language-server.

Language association to yaml files

The extension works only when a document is assigned ansible language. The following method is used to assign ansible language to the document opened by the extension:

Without file inspection

- yaml files under /p/Playbooks dir.
- files with the following double extension: .ansible.yaml or .ansible.yml.
- notable yaml names recognized by ansible like site.yaml or site.yml
- yaml files having playbook in their filename: *Playbook*.yaml or *Playbook*.yml

Additionally, in vscode, you can add persistent file association for language to settings.json file like this:

```
{  
  ...  
  "files.associations": {  
    "**plays.yaml": "ansible",  
    "**init.yaml": "yaml",  
  }  
}
```

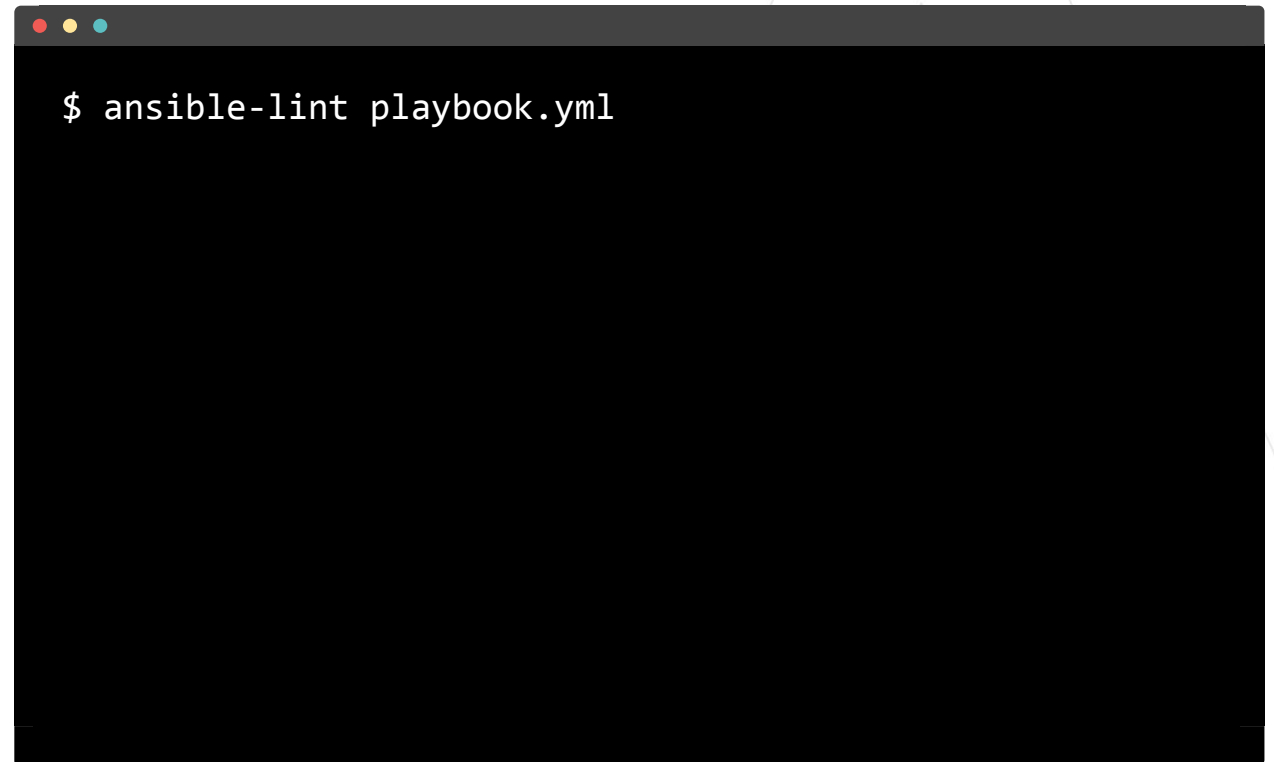
Ansible lint (`ansible-lint`)

Fully supported (AAP 2.3)

What is it?

Command-line tool for linting playbooks, roles and collections aimed towards any Ansible users.

- ▶ Promote best practices and and patterns.
- ▶ Develop consistent code across teams and scale using an opinionated strategy.
- ▶ Integrate into larger development workflows and CI tools.
- ▶ Helps upgrade playbooks to later Ansible Core versions.



```
$ ansible-lint playbook.yml
```



Ansible lint (`ansible-lint`).

Installation paradigms



```
# For installation on Red Hat Enterprise Linux  
(Requires Ansible Automation Platform Subscription)
```

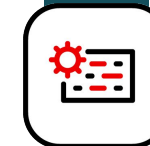
```
$ sudo dnf -y install ansible-lint
```

OR

```
# For installation on other linux systems  
(Installation from upstream)
```

```
$ python3 -m pip install ansible-lint
```

Ansible content navigator (`ansible-navigator`).



What is it?

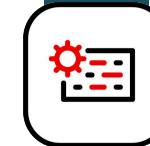
It is a command line utility and text-based user interface (TUI) for running, testing and developing Ansible automation content

- ▶ Review EEs
- ▶ Develop collections
- ▶ Develop playbooks
- ▶ Troubleshoot problems

```
$ ansible-navigator run playbook.yml -i inventory.ini
```


Ansible content navigator (ansible-navigator).

Installation paradigms



```
# For installation on Red Hat Enterprise Linux
(Requires Ansible Automation Platform Subscription)

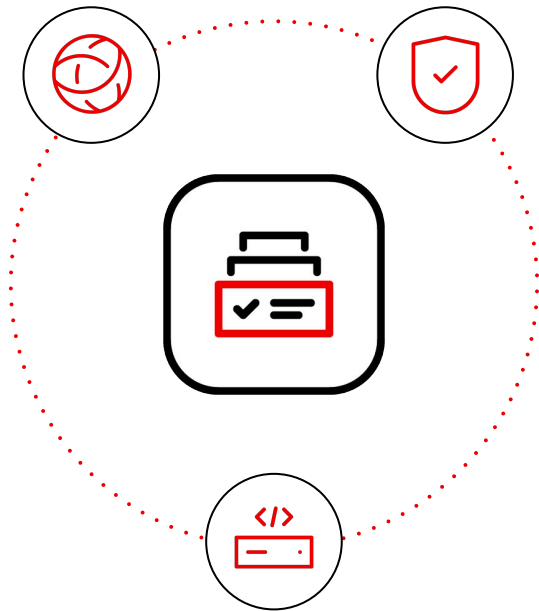
$ sudo dnf -y install ansible-navigator

OR

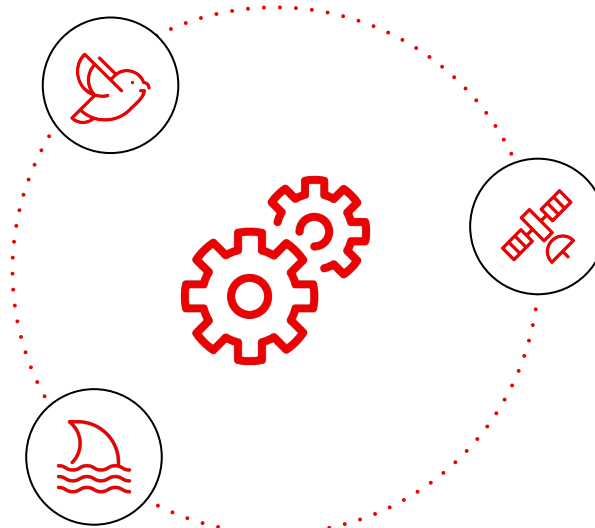
# For installation on other linux systems
(Installation from upstream)

$ python3 -m pip install ansible-navigator
```

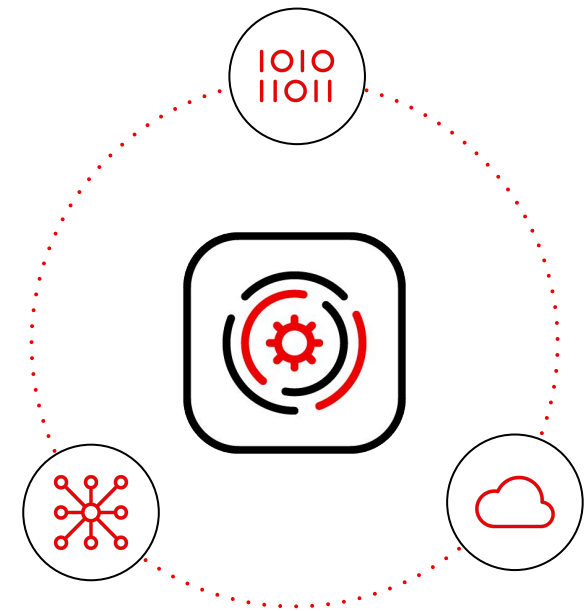
Many technologies, different life cycles. How do I keep it all aligned?



Collections

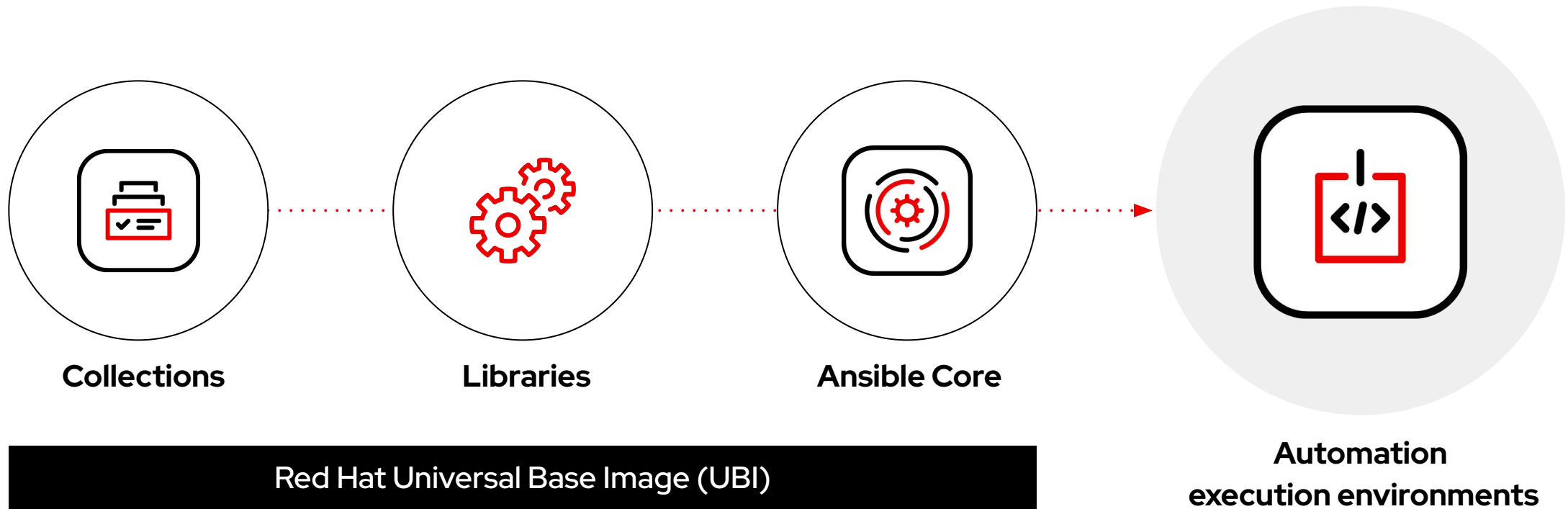


Dependencies



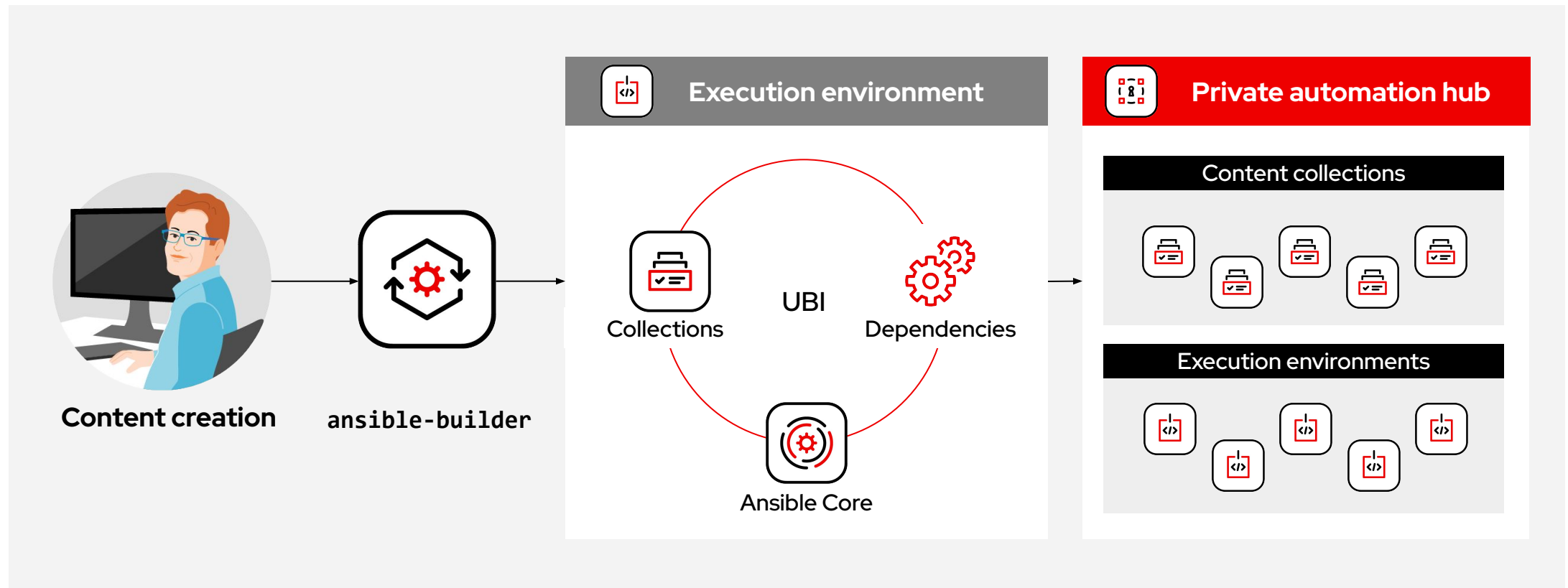
Runtime

Automation execution environments. Reuse and scale automation content.



Execution environment development.

Build, collaborate, sign, publish.



Execution environment builder. **Build.**

ansible-builder



What is it?

- ▶ Easily build custom execution environments with the exact Ansible content needed
- ▶ Manage, track and version execution environments
- ▶ Share execution environment build artifacts with other teams

```
$ ansible-builder build --tag repo/custom_ee:latest
```

Execution environment builder. **Build.**

Installation paradigms



```
# For installation on Red Hat Enterprise Linux
(Requires Ansible Automation Platform Subscription)

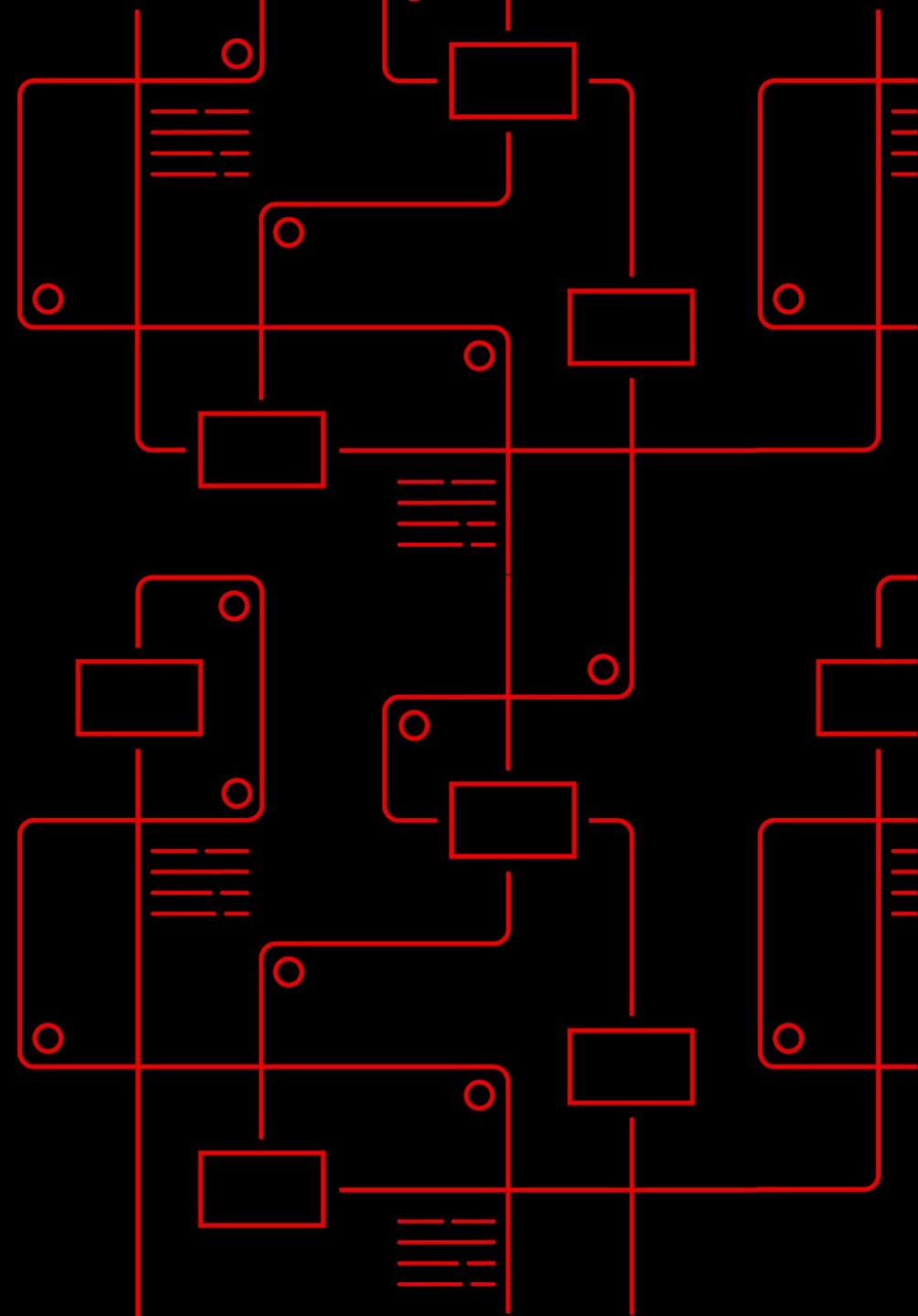
$ sudo dnf -y install ansible-builder

OR

# For installation on other linux systems
(Installation from upstream)

$ python3 -m pip install ansible-builder
```

Demo Time





Ansible Lightspeed

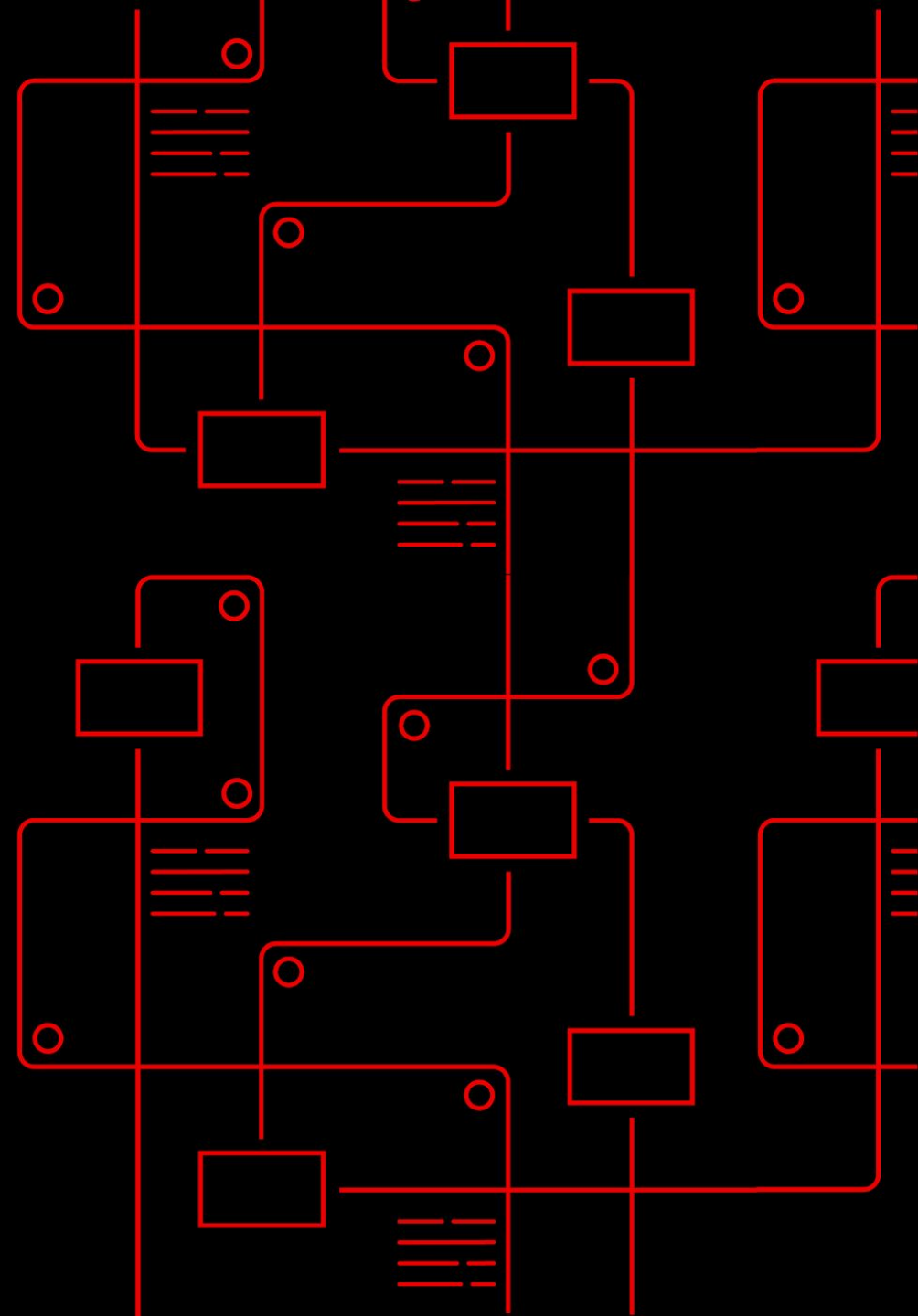
With IBM Watson Code Assistant



The experience

Ansible Lightspeed

with IBM Watson Code Assistant



It starts with Ansible Playbooks

Ansible Lightspeed will eventually impact the Ansible experience in a **number of profound ways**.

But at the start, the experience will be focused on the very foundation of it all: **Ansible Playbooks**.

```
---
- name: Apache server installed
  hosts: web
  become: true
  tasks:

  - name: latest Apache version installed
    yum:
      name: httpd
      state: latest

  - name: Apache enabled and running
    service:
      name: httpd
      enabled: true
      state: started

  - name: copy index.html
    copy:
      src: web.html
      dest: /var/www/html/index.html
```

Ansible Lightspeed with IBM Watson Code Assistant

Ansible Lightspeed with IBM Watson Code Assistant is a generative AI service accessed via the Ansible VSCode extension, allowing users to accept and run recommended code directly in their code editing environment while creating Ansible Playbooks.

A *Tech Preview* for the service will be available for all Ansible users in late June, with a commercial offering to follow this fall.

The **IBM Watson Code Assistant** integration is infused with IBM's Ansible foundation model. This foundation model combines Ansible Galaxy data and Red Hat subject matter expertise to deliver highly relevant code automation recommendations that adhere to Ansible best practices.

IBM Watson Code Assistant is built on the **Red Hat OpenShift Data Science** platform.

Ansible Lightspeed
with IBM Watson Code Assistant



IBM Watson Code Assistant



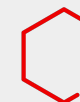
IBM Watsonx



Red Hat
OpenShift
Data Science



Red Hat
OpenShift



```
File Edit Selection View Go Run Terminal Help
cloudops.yml 1, M x
playbooks > cloudops.yml
1
2 - name: AWS EC2 Cloud operations
3 hosts: localhost
4 connection: local
5
6
7 tasks:
8   - name: Create a virtual network
9
10  - name: Create a cloud instance
11
```

The Ansible Lightspeed experience

Enhancing Playbook creation

```
File Edit Selection View Go Run Terminal Help
cloudops.yml 1, M X
playbooks > cloudops.yml
1 ---
2 - name: AWS EC2 Cloud operations
3   hosts: localhost
4   connection: local
5
6 tasks:
7   - name: Create a virtual network
8
9   - name: Create a cloud instance
10
11
12
```

1. Ansible Lightspeed with IBM Watson Code Assistant is **accessible via VSCode extension**

2. Type in a task directly into the VSCode editor. Ansible Lightspeed takes over.

```
File Edit Selection View Go Run Terminal Help
cloudops.yml 1, M X
playbooks > cloudops.yml
1 ---
2 - name: AWS EC2 Cloud operations
3   hosts: localhost
4   connection: local
5
6 tasks:
7   - name: Create a virtual network
8     ec2_vpc_net:
9       name: "{{ vpc_name }}"
10      cidr_block: "{{ vpc_cidr }}"
11      region: "{{ region }}"
12      state: present
13      aws_access_key: "{{ aws_access_key }}"
14      aws_secret_key: "{{ aws_secret_key }}"
15      register: vpc
```

3. Ansible Lightspeed **will make a code recommendation** for the developer to consider

4. User has option to **Accept, Ignore, or Modify** recommended code snippet

```
File Edit Selection View Go Run Terminal Help
cloudops.yml M X
playbooks > cloudops.yml
1 ---
2 - name: AWS EC2 Cloud operations
3   hosts: localhost
4   connection: local
5
6 tasks:
7   - name: Create a virtual network
8     ec2_vpc_net:
9       name: "{{ vpc_name }}"
10      cidr_block: "{{ vpc_cidr }}"
11      region: "{{ region }}"
12      state: present
13      aws_access_key: "{{ aws_access_key }}"
14      aws_secret_key: "{{ aws_secret_key }}"
15      register: vpc
16
```

5. If "accepted," playbook is automatically populated and user can move on to the next task

6. User prompted to provide feedback; this is important for **helping to train the model.**

Key differentiator: Content source matching

The screenshot shows a code editor window titled 'test.yml' with the following content:

```
1 ---
2 - name: Create a VM with nginx for a website
3   hosts: localhost
4   become: false
5   tasks:
6     - name: Install nginx on rhel
7       when: ansible_os_family == "RedHat"
8         ansible.builtin.package:
9           name: nginx
10          state: present
11
12
```

Below the code editor is a terminal window titled 'ANSIBLE: LIGHTSPEED TRAINING MATCHES' showing search results for 'puwanut.ansible_nginx':

- URL: https://galaxy.ansible.com/puwanut/ansible_nginx
- Path: tasks/nginx_install.yml
- Data Source: Ansible Galaxy roles
- License: MIT
- Ansible type: Playbook
- Score: 1.9092153

Other search results include 'hypery2k.ansible_oracle_java' and 'dockpack.base_nodejs'.

Specific URL

Data source description

Open source license info

Code score

Ansible Lightspeed with IBM Watson Code Assistant: Custom Models

```

- name: Add user to z/OS system
  hosts: all
  gather_facts: false
  environment: "{{ environment_vars }}"

  tasks:
    - name: Generate random password
  
```

```

- name: Add user to z/OS system
  hosts: all
  gather_facts: false
  environment: "{{ environment_vars }}"

  tasks:
    - name: Generate random password
      ansible.builtin.set_fact:
        password: "{{ lookup('password', '/dev/null length=15 chars=ascii_letters') }}"
  
```

```

- name: Add new user
  ansible.builtin.user:
    name: "{{ name }}"
    shell: /bin/bash
    password: "{{ password | password_hash('sha512') }}"
    update_password: "{{ update_password }}"

- name: Permit new user access to resources
  ansible.posix.authorized_key:
    user: "{{ user }}"
    key: "{{ lookup('file', item) }}"
    loop: "{{ copy_user_public_keys | flatten(levels=1) }}"
  
```

1. Task description written in natural language in VSCode editor

2. Hit enter, get code recommendation for the task (from base foundation model)

3. Developer fills in unique variables; watsonx foundation model delivers predictive suggestions

Model to customize

watsonx/granite.350m.ansible

The base model that you're going to be customizing.

Customization approach

Multitask Prefix Tuning

Select the approach to use. Your results will vary based on the dataset and model selected.

List of URLs

<https://github.ibm.com/ansible-collections-cio>
<https://github.ibm.com/ansible-collections-cirrus>
<https://github.ibm.com/ansible-projects-cio>

Enter a list of URLs to GitHub organizations, delimited by new lines.

```

name: Add user to z/OS system
hosts: all
gather_facts: false
environment: "{{ environment_vars }}"

tasks:
  - name: Generate random password
    ansible.builtin.set_fact:
      password: "{{ '' | generate_password }}"
    no_log: true

  - name: Add new user
    ibm_ibm_zos_core.zos_tso_command:
      commands:
        - ADDUSER({{ userid | upper }})
        - DFLTGRP({{ default_group | upper }})
        - AUTHORITY({{ default_group_authority | upper }})
        - OWNER({{ owner | upper }})
        - NAME({{ name }})
        - PASSWORD({{ password | upper }})
        - PHRASE({{ passphrase }})
        - SECLABEL({{ security_label | upper }})
        - SECLEVEL({{ security_level | upper }})
        - ADDCATEGORY({{ category | upper }})
        - TSO(ACCTNUM('{{ tso_account_number | upper }}') PROC('{{ tso_login_procedure | upper }}))
        - DFP(DATAAPPL('{{ dfp_data_application }})DATACLAS('{{ data_class }})MOMTCLAS('{{ management_class }})STORCLAS('{{ storage_class }})
        - OWNS(UID('{{ omvs_uid }})
        - HOME('{{ omvs_home_directory }})
  
```

4. Watson Code Assistant customizable with private data sets (IBM CIO team data in this example)

5. Tune base model with private data, which is then pulled in and processed

6. Ansible Lightspeed now shows new results specific to the custom data set, providing more prescriptive reco base

Ansible Lightspeed with IBM Watson Code Assistant

Features and capabilities found in the *Tech Preview* service



Task generation with NLP

Users can generate automation tasks to create Ansible Playbooks using natural language prompts.

Ansible Foundation Model

Includes access to IBM's Ansible "Granite" foundation model, which is trained on Ansible Galaxy content, and infused with additional Red Hat Ansible subject matter expertise.

Pre- and post- data processing

Feature that amplifies quality and relevance of automation code recommendations from Ansible specific Foundation Model. Code recommendation outputs processed to align with modern Ansible best practices.

Content source matching

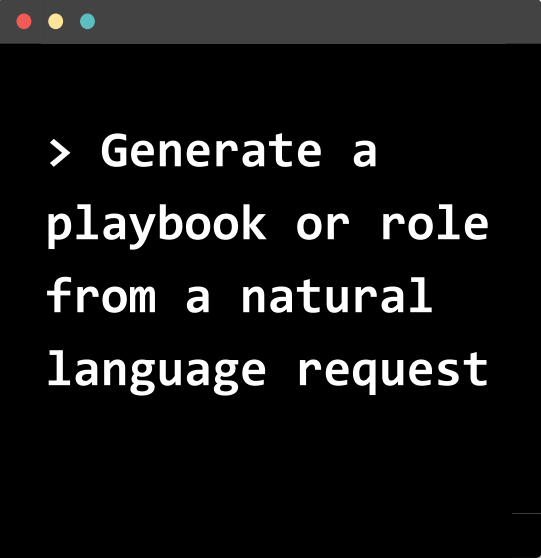
The service will always attempt to match a content recommendation to Ansible Galaxy data sources, in order to show the potential provenance of recommendation content from Ansible contributors.

Ansible content developers get recognition for their potential contributions to content recommendations.

Ansible experience enhancement

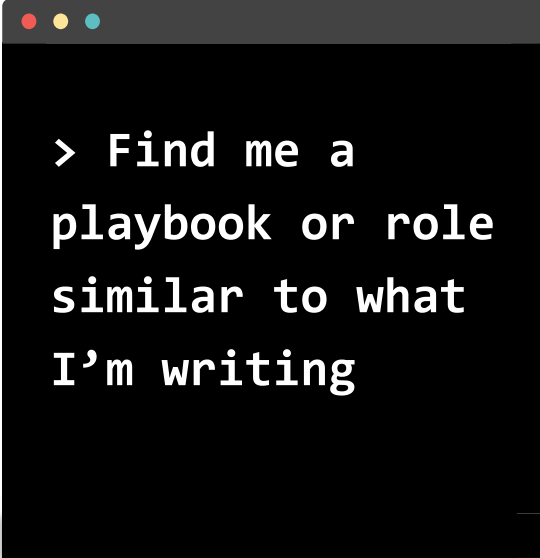
Delivered through Ansible's existing VS Code extension, and works in concert with other Ansible content tools (ansible-navigator, ansible-builder, ansible-lint)

Ansible Lightspeed will transform the Playbook creation process



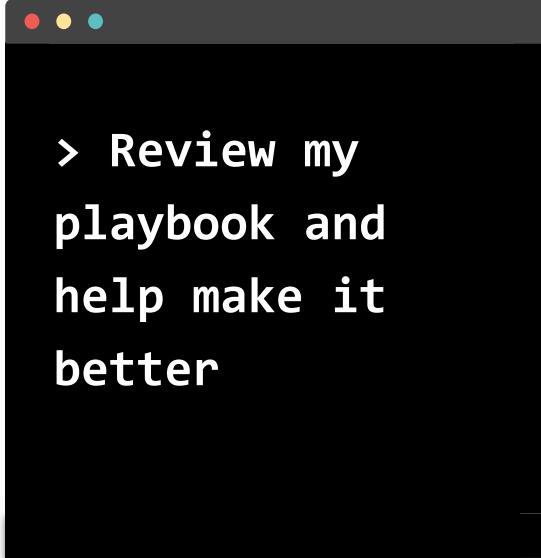
```
> Generate a
playbook or role
from a natural
language request
```

**Content
Generation**



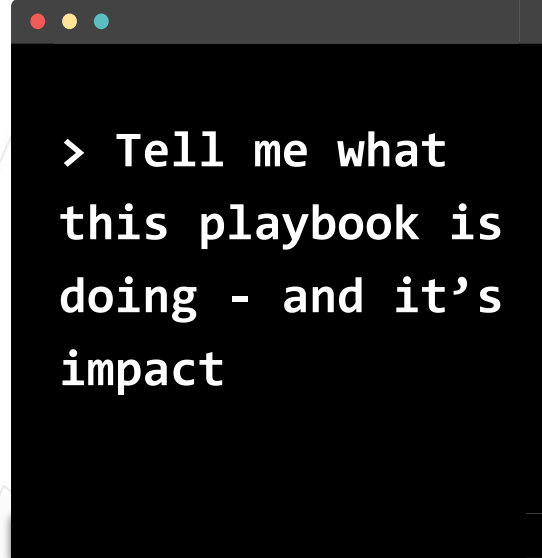
```
> Find me a
playbook or role
similar to what
I'm writing
```

**Content
Discovery**



```
> Review my
playbook and
help make it
better
```

**Content
Optimization**



```
> Tell me what
this playbook is
doing - and it's
impact
```

**Content
Explanation**

Features on roadmap for upcoming commercial offering

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