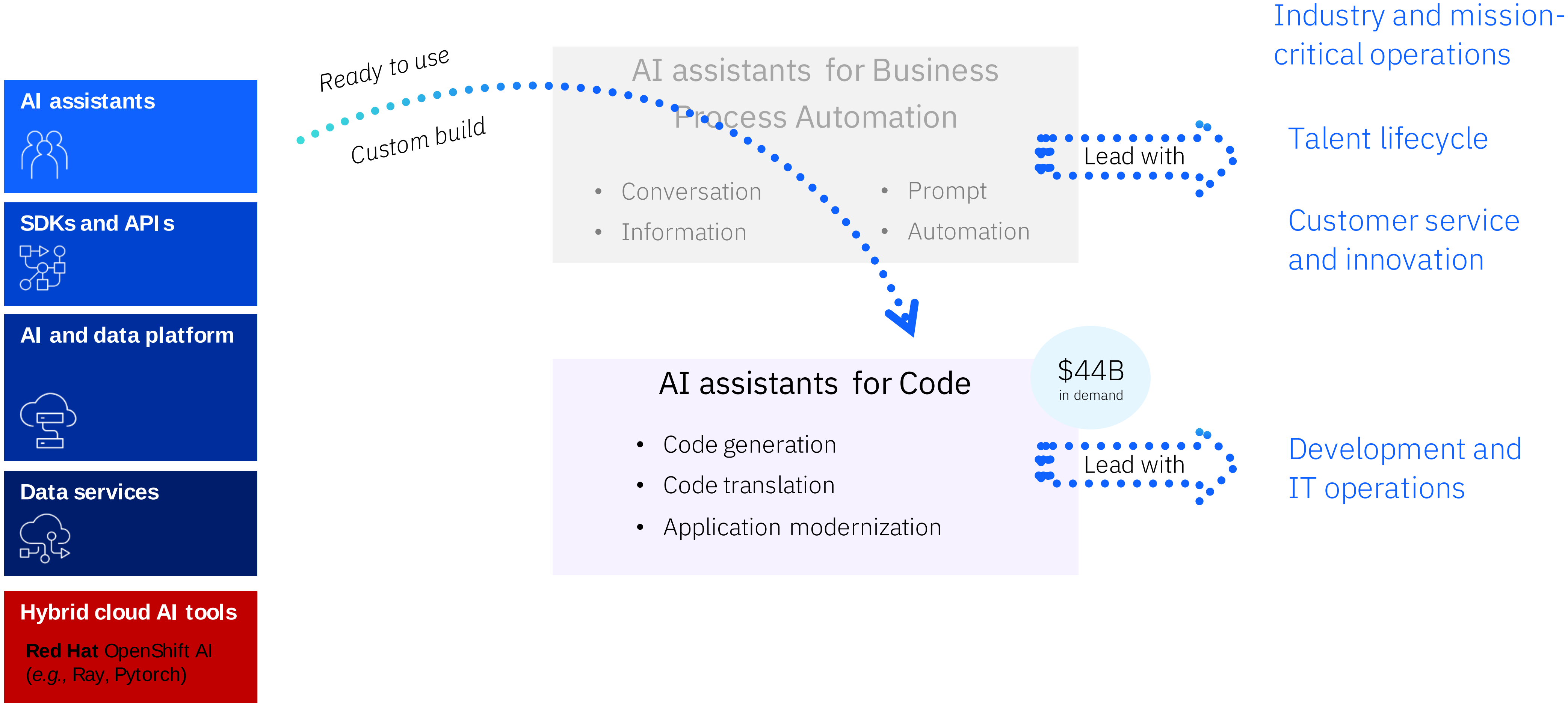
IBM watsonx Code Assistant for Red Hat Ansible Lightspeed

Data and AI
February 2024

Grace Williamson
watsonx Code Assistant
GTM Product Manager

Chetan Hireholi
watsonx Code Assistant for Red Hat Ansible Lightspeed
Product Manager

IBM's AI assistants are built to satisfy enterprise productivity needs



AI assistants

SDKs and APIs

AI and data platform

Data services

Hybrid cloud AI tools

Red Hat OpenShift AI
(*e.g.*, Ray, Pytorch)

Ready to use

Custom build

AI assistants for Business Process Automation

- Conversation
- Information
- Prompt
- Automation

Lead with

Industry and mission-critical operations

Talent lifecycle

Customer service and innovation

AI assistants for Code

$44B
in demand

- Code generation
- Code translation
- Application modernization

Lead with

Development and IT operations

IBM watsonx
Code Assistant
for Red Hat Ansible
Lightspeed

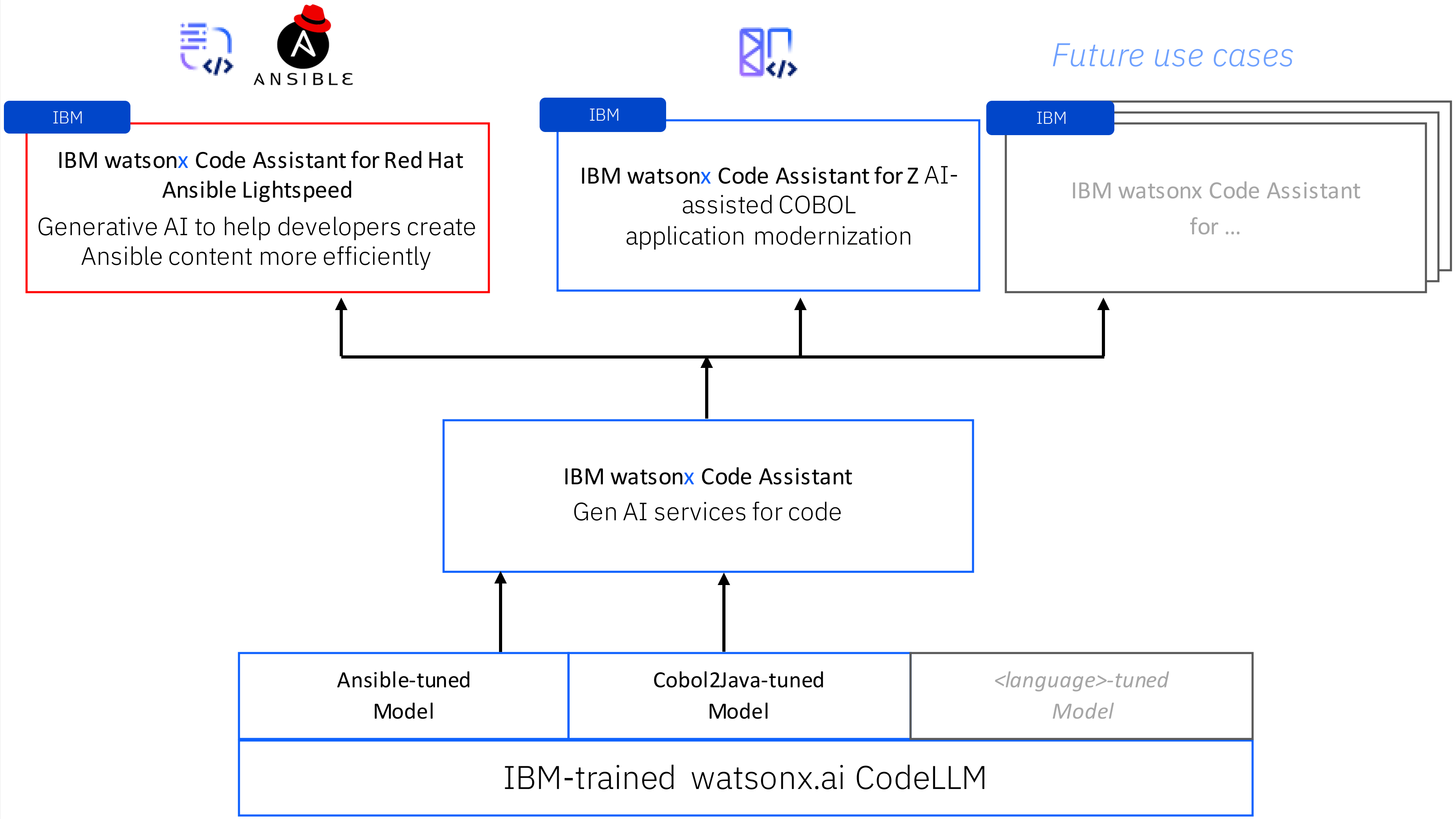
80% of the product development lifecycle will make use of generative AI code generation by 2025*

IBM watsonx Code Assistant

Product Family

Enterprise-grade AI code and content generation solutions

General Availability Oct 25, 2023



Future use cases

IBM

IBM watsonx Code Assistant for Red Hat Ansible Lightspeed

Generative AI to help developers create Ansible content more efficiently

IBM

IBM watsonx Code Assistant for Z AI-assisted COBOL application modernization

IBM

IBM watsonx Code Assistant for …

IBM watsonx Code Assistant

Gen AI services for code

Ansible-tuned Model

Cobol2Java-tuned Model

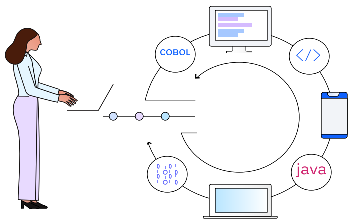
<language>-tuned Model

IBM-trained watsonx.ai CodeLLM

Join IBM **watsonx Code Assistant for Z** office hours for the latest updates

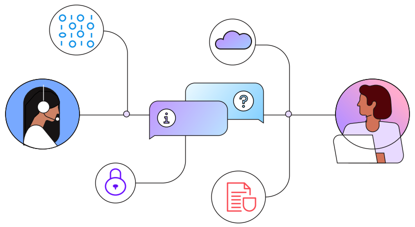
watsonx Code Assistant Product Family

Enterprise ready AI for Code solutions to address skills gaps and increase developer productivity



Powered by IBM Granite models

- watsonx Code Assistant uses IBM built Granite models
- Models trained on open-source code repositories like GitHub
- IBM’s Granite 20-billion parameter model for code was trained on 1.6 trillion code tokens



Trustworthy and targeted

- Filtered for sensitive and toxic information, as well as copyright-protected code
- Fine-tuned for targeted use case
  - Mainframe Application Modernization
  - IT Automation

Purpose built for the developer

- Accessible directly in Visual Studio Code
- Consistent, high-quality code and content recommendations
- True developer experience enhancement

IBM watsonx Code Assistant for Red Hat Ansible Lightspeed

watsonx Code Assistant for Red Hat Ansible Lightspeed leverages generative AI to accelerate development while maintaining the principles of trust, security and compliance at its core. Developers and IT Operators can speed up application modernization efforts and generate automation to rapidly scale IT environments.

Objectives

Faster Ansible Playbook creation

Generation of high-quality code

Adhere to best practices and standards

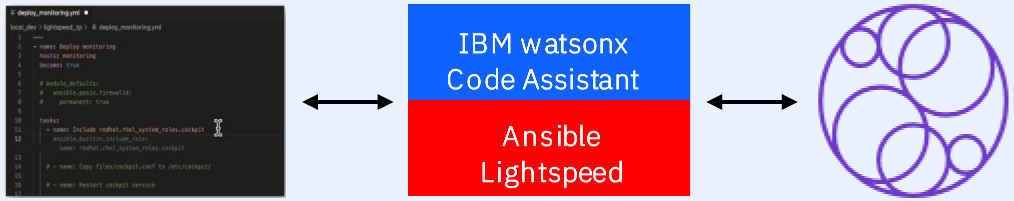
Benefits

Accelerate content development

Accuracy from AI-generated recommendations

Model customization and tuning for specificity

Accelerate Ansible Playbook creation using Generative AI



```
deploy_monitoring.yml
local_dev > lightspeed_tp > deploy_monitoring.yml
---
- name: Deploy monitoring
  hosts: monitoring
  become: true

  # module_defaults:
  #   ansible.posix.firewalld:
  #     permanent: true

  tasks:
    - name: Include redhat.rhel_system_roles.cockpit
      ansible.builtin.include_role:
        name: redhat.rhel_system_roles.cockpit

    # - name: Copy files/cockpit.conf to /etc/cockpit/

    # - name: Restart cockpit service
```

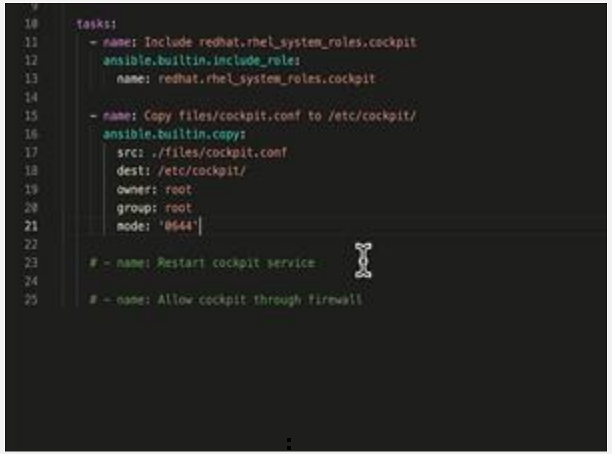
Ansible content creation

IBM Granite code model for Ansible

30%

Faster time to value in IT Automation leveraging Ansible

Ansible Content Creation



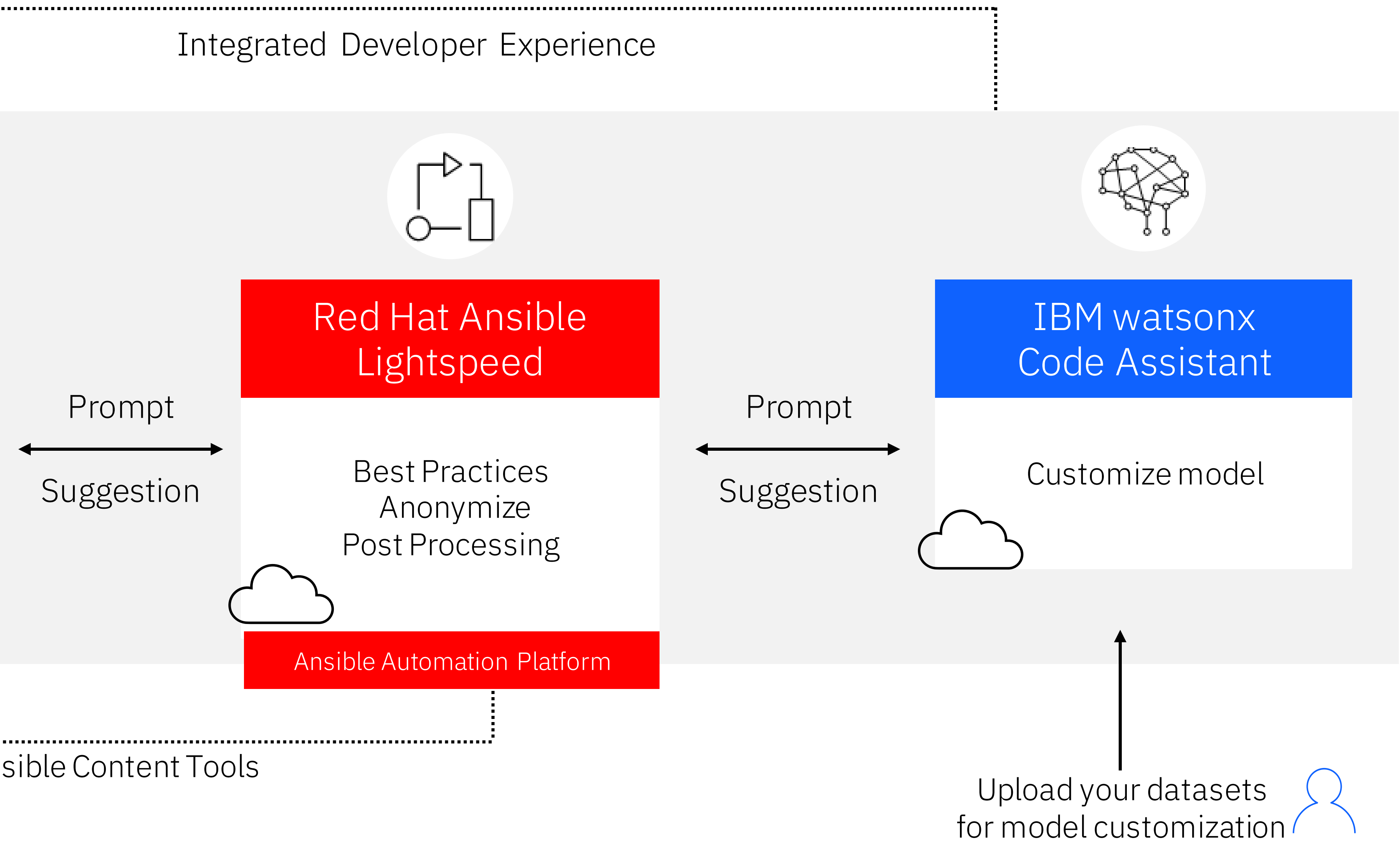
```
tasks:
  - name: Include redhat.rhel_system_roles.cockpit
    ansible.builtin.include_role:
      name: redhat.rhel_system_roles.cockpit

  - name: Copy files/cockpit.conf to /etc/cockpit/
    ansible.builtin.copy:
      src: ./files/cockpit.conf
      dest: /etc/cockpit/
      owner: root
      group: root
      mode: '0644'

  # - name: Restart cockpit service

  # - name: Allow cockpit through firewall
```

VS Code
Extension


Integrated Developer Experience
Prompt
Suggestion
Red Hat Ansible Lightspeed
Best Practices
Anonymize
Post Processing
Ansible Automation Platform
Prompt
Suggestion
IBM watsonx Code Assistant
Customize model
Upload your datasets for model customization

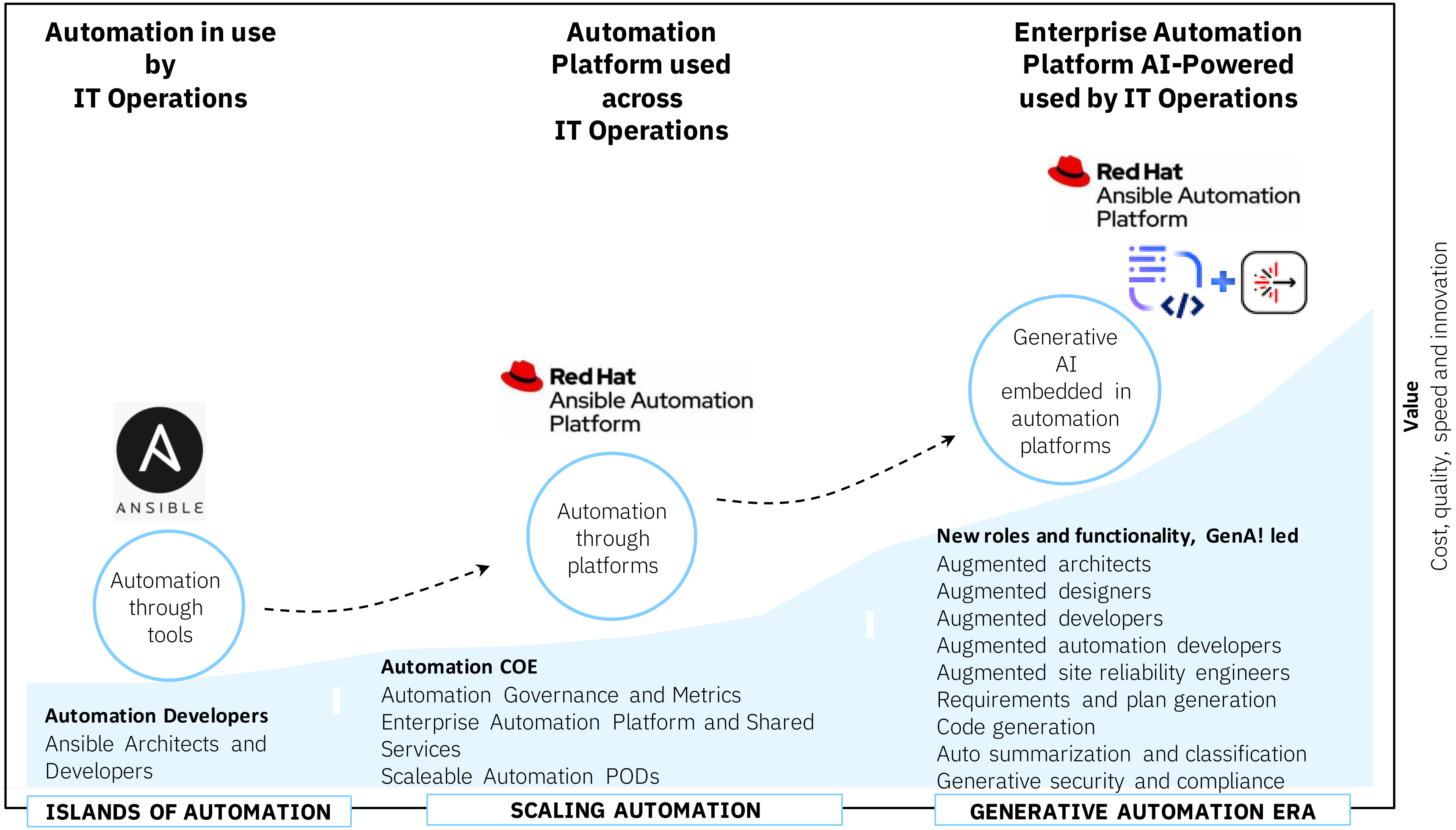
Ansible Content Tools

Accelerate business value with an approach enabled by generative AI automation for migration and modernization

Challenges and Opportunities

- Shortage of Ansible Skills & Capacity
- Legacy Ansible Code
- Heterogeneous Automation Tools
- Islands of Automation
- Inability to Scale Automation
- Governance, Tracking & Reporting
- Others, ...

IT Transformation Journey



Automation in use by IT Operations

Automation through tools

Automation Developers
Ansible Architects and Developers

ISLANDS OF AUTOMATION

Automation Platform used across IT Operations

Automation through platforms

Automation COE
Automation Governance and Metrics
Enterprise Automation Platform and Shared Services
Scaleable Automation PODs

SCALING AUTOMATION

Enterprise Automation Platform AI-Powered used by IT Operations

Generative AI embedded in automation platforms

New roles and functionality, GenA! led
Augmented architects
Augmented designers
Augmented developers
Augmented automation developers
Augmented site reliability engineers
Requirements and plan generation
Code generation
Auto summarization and classification
Generative security and compliance

GENERATIVE AUTOMATION ERA

ANSIBLE
Red Hat Ansible Automation Platform
Red Hat Ansible Automation Platform
Value
Cost, quality, speed and innovation

Based on rules

Collect, organize and grow data to build automation workflows

Based on intent

Establish Governance models and tooling to support enterprise automation at scale. Infuse AI in Automation platforms and enterprise applications.

Based on intuitive prompts

Have generative AI do the work augmenting specialized skills and manual effort Prompt, tune, generate, automate, and deploy your workflows.

Benefits for the entire IT Automation Team

Automation Novices

- Reduced Ansible + YAML learning curve
- Help ensure that automation content created adheres to best practices
- Boost skill development and confidence

Automation Developers

- Enhanced productivity for Ansible Playbook creation
- Integrates into existing workflows and tools for efficient adoption
- Ansible code bot tools streamline ongoing code maintenance

Automation SMEs

- Bridge gap between automation ideas and Ansible content generation
- Tap into a data model trained across automation domains, expanding expertise

Automation Operations

- Generated code adheres to Ansible best practice; can be trusted to run at scale
- Integrated enhancement to familiar Ansible Automation Platform experience

Citi Pilot

watsonx Code Assistant for Red Hat Ansible Lightspeed

IBM, RedHat and Citi partnered in a 4-week pilot to showcase the value of watsonx.

Two primary use cases:

1. **Enablement use case:**
Evaluated novice Ansible developer productivity with and without the code generation tool through a series of tasks to measure quality and efficiency.

2. **Quality use case:**
Ran the AAP CodeBot on existing Ansible Playbooks currently in production to assess quality and review recommendations.

60%
Reduction in time spent creating Ansible Playbooks

Internal Use Only

The Objective

- Evaluate enterprise-wide impact of integrating watsonx Code Assistant for Ansible Lightspeed (WCA) and advanced Ansible Automation Platform features (AAP) into the Citi software development pipeline

POC Execution

- Citi Scope: Developers and playbooks in the Global Platform team within Citi's Technology Infrastructure organization
- **Enablement use case**: Provided hands-on WCA training to 10 Citi developers and evaluated their speed, code quality and error rate based on a series of tasks with and without access to WCA
- **Quality use case**: Ran the AAP CodeBot on playbooks currently in production, conducting an initial code review and generating recommendations for enhancement

Results

- **60% reduction in time spend to create playbooks**
- No developer required external support while using WCA (i.e. documentation, Stack Overflow), as compared to 24X external outreach without WCA
- 2X reduction in critical failures

Submit a Deal Support Request

What's changed and what's new?

What is changing?	Why?
Adding a Lite Plan	To support a trial motion for WCA4RHAL we are introducing a capacity limited Lite plan that allows for evaluation of the service at no cost. A lite plan is also referred to a trial. Trial requires subscription to AAP.
Adding a Standard Plan	We are delivering model tuning capabilities that will allow a customer to customize the base model using their existing Ansible content. This capability will be available as part of a new Standard plan
Moving to consumption-based pricing (Essentials and Standard Plans)	1. We have recognized significant friction in our early customer engagements around counting seats. Removing the need for seats will greatly simplify the customer onboarding process 2. We are seeking to better align our pricing model with watsonx.

IBM **watsonx** Code Assistant for Red Hat Ansible Lightspeed Plans

New pricing plans are directly correlated to a customer's consumption.
*Key Metric: **Resource Units** are comprised of tokens and helps to includes estimations for Ansible tasks.*

○ Lite Plan

Use to evaluate and try WCA

Charge metric:

Free, limited consumption

Features:

Task Prompt only

Limit of **10 Resource Units** *(enough for about one developer making 200 task prompts using ~500 tokens each)*

For higher usage or production usage, use Essentials or Standard plan.

Developer tries the Lite Plan and uses about 200 Ansible tasks. After 200 tasks, trial will end.

● Essentials Plan

Use for customers with up to 60 Ansible developers

Charge metric:

Usage fee: $2 per Resource Unit

One Resource Units is 20 task prompts using 500 tokens each; one developer will use about ten Resource Units per month)

For clients with more than 60 developers or if model tuning is needed, use Standard Plan.

60 WCA developers
10 Resource Units per developer
= 600 RUs @
$2 each = $1,200/mo

○ Standard Plan

For customers with more than 60 Ansible developers and desire customized models

Charge metrics:

Instance fee: $1,200 per month (includes 660 Resource Units)

Usage fee: $2 per Add'l. Resource Units

Tuning hour: $500 per hour

Instance fee includes 660 Resource Units (Approx. 13,200 task prompts using 500 tokens each—enough for about 66 developers per month)

Include ability to prompt tune the model with tuning studio.

116 WCA developers
$1,200 instance fee (incl. 660 RUs)
$1,000 for 500 add'l. RUs
$2,000 for 4 tuning hours
$4,200/mo

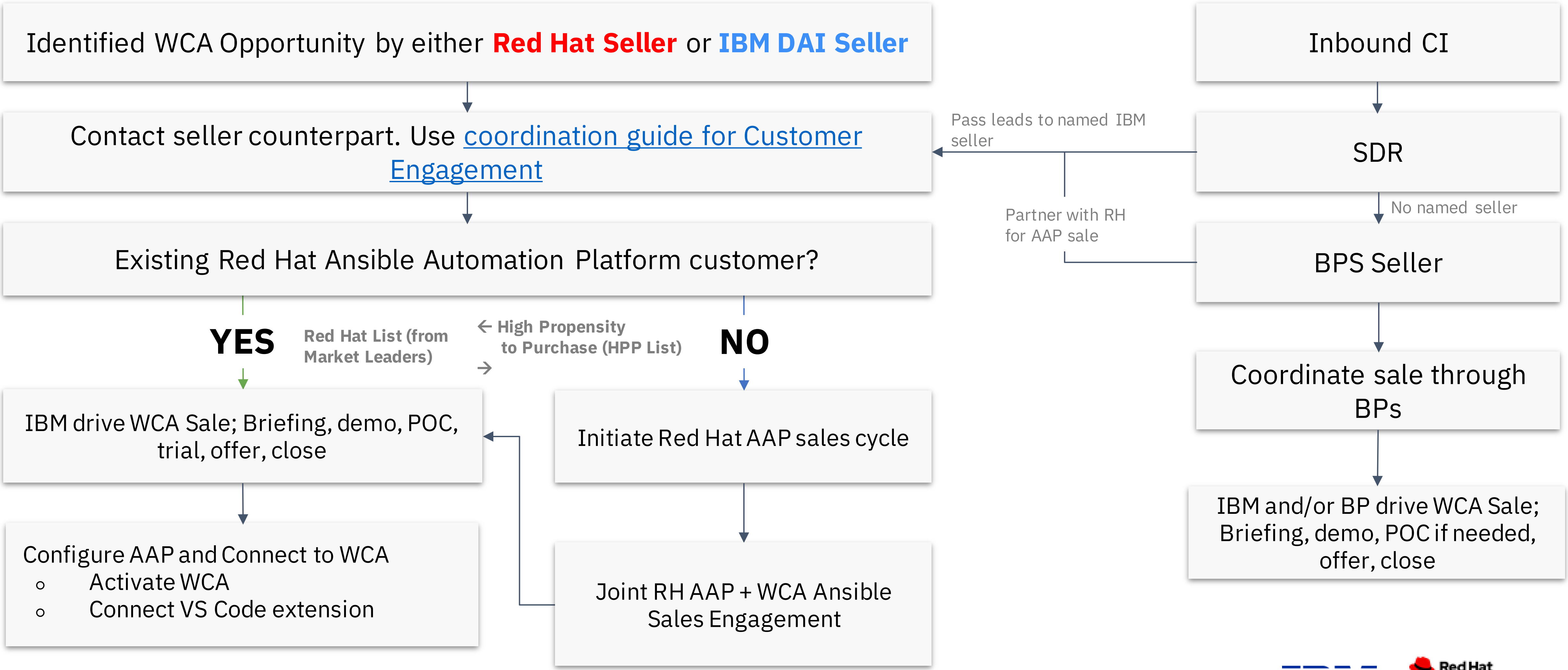
watsonx Code Assistant for Red Hat Ansible Lightspeed

T-Shirt Sizes

	Small/Medium Essentials Plan	Medium/Large Standard Plan	Enterprise Standard Plan
Instance Fee	$0	$1200/month	$1200/month
Resource Units	$1200/month 60 developers @ 10 RUs = **600 RUs @ $2 each**	$1080/ month 120 developers @ 10 RUs each=1200 RUs. 660 are included in instance fee. **540 add'l. RUs @ $2 each**	$4680/ month 300 developers @ 10 RUs each= 3000 RUs. 660 are included in instance fee. **2340 add'l. RUs @ $2 each**
Tuning hours	Not Available in Essentials	*4 hours of tuning @500= $2000 per year or* $167/mo	*6 hours of tuning @500= $3000 per year or* $250/mo
Total	**$14,400 / year** $1,200 / month	**$29,364 / year** $2,447/ month	**$73,560/ year** $6,130/ month
Notes / Sample Use-case	• 60 developers using task prompts to WCA • 200 task prompts/month/developer • 500 tokens per task prompt • Total 6M tokens = 600 RUs • Using Essentials Plan, no Tuning	• 200 prompts/month/developer • 4 hours of tuning per year • Large development team using prompting and tuning functionality (~120 Ansible developers) • Using Standard Tuning capability to iteratively improve coding recommendations	• 200 prompts/month/developer • 6 hours of tuning per year • Enterprise development team using prompting and tuning functionality (~300 Ansible developers) • Using Standard Tuning capability to iteratively improve coding recommendations

Use the pricing calculator on Seismic

Joint Sales Engagement: Q1 2024



Identified WCA Opportunity by either **Red Hat Seller** or **IBM DAI Seller**

Contact seller counterpart. Use coordination guide for Customer Engagement

Existing Red Hat Ansible Automation Platform customer?

YES

Red Hat List (from Market Leaders)

← High Propensity to Purchase (HPP List)

→

NO

IBM drive WCA Sale; Briefing, demo, POC, trial, offer, close

Configure AAP and Connect to WCA
- Activate WCA
- Connect VS Code extension

Initiate Red Hat AAP sales cycle

Joint RH AAP + WCA Ansible Sales Engagement

Inbound CI

SDR

Pass leads to named IBM seller

No named seller

Partner with RH for AAP sale

BPS Seller

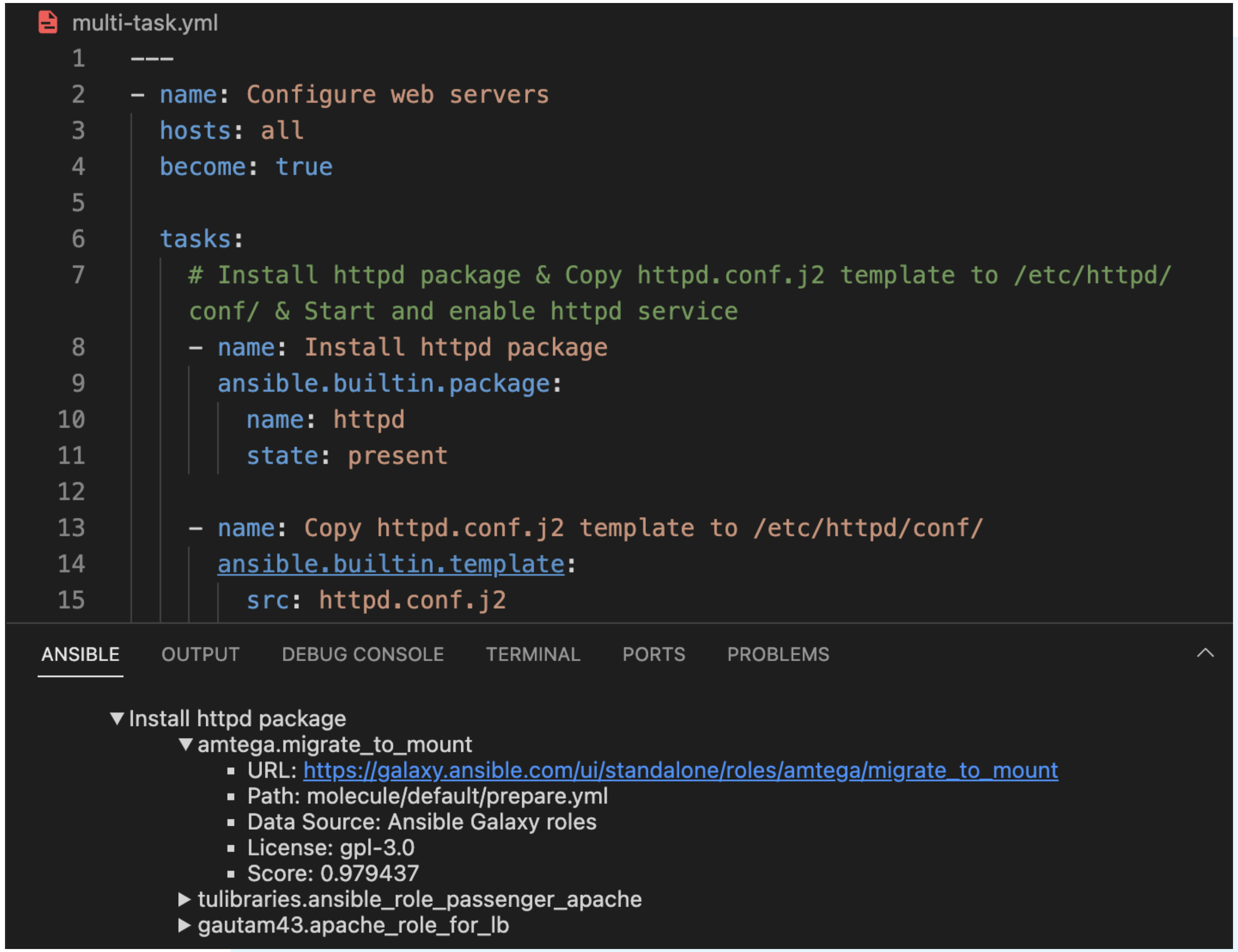
Coordinate sale through BPs

IBM and/or BP drive WCA Sale; Briefing, demo, POC if needed, offer, close

watsonx Code Assistant for Red Hat Ansible Lightspeed

New Feature Spotlight: Model Customization Available in Standard Plan

Key features: Task Generation and Content Similarity



```yaml
---
- name: Configure web servers
  hosts: all
  become: true

  tasks:
    # Install httpd package & Copy httpd.conf.j2 template to /etc/httpd/conf/ & Start and enable httpd service
    - name: Install httpd package
      ansible.builtin.package:
        name: httpd
        state: present

    - name: Copy httpd.conf.j2 template to /etc/httpd/conf/
      ansible.builtin.template:
        src: httpd.conf.j2
```

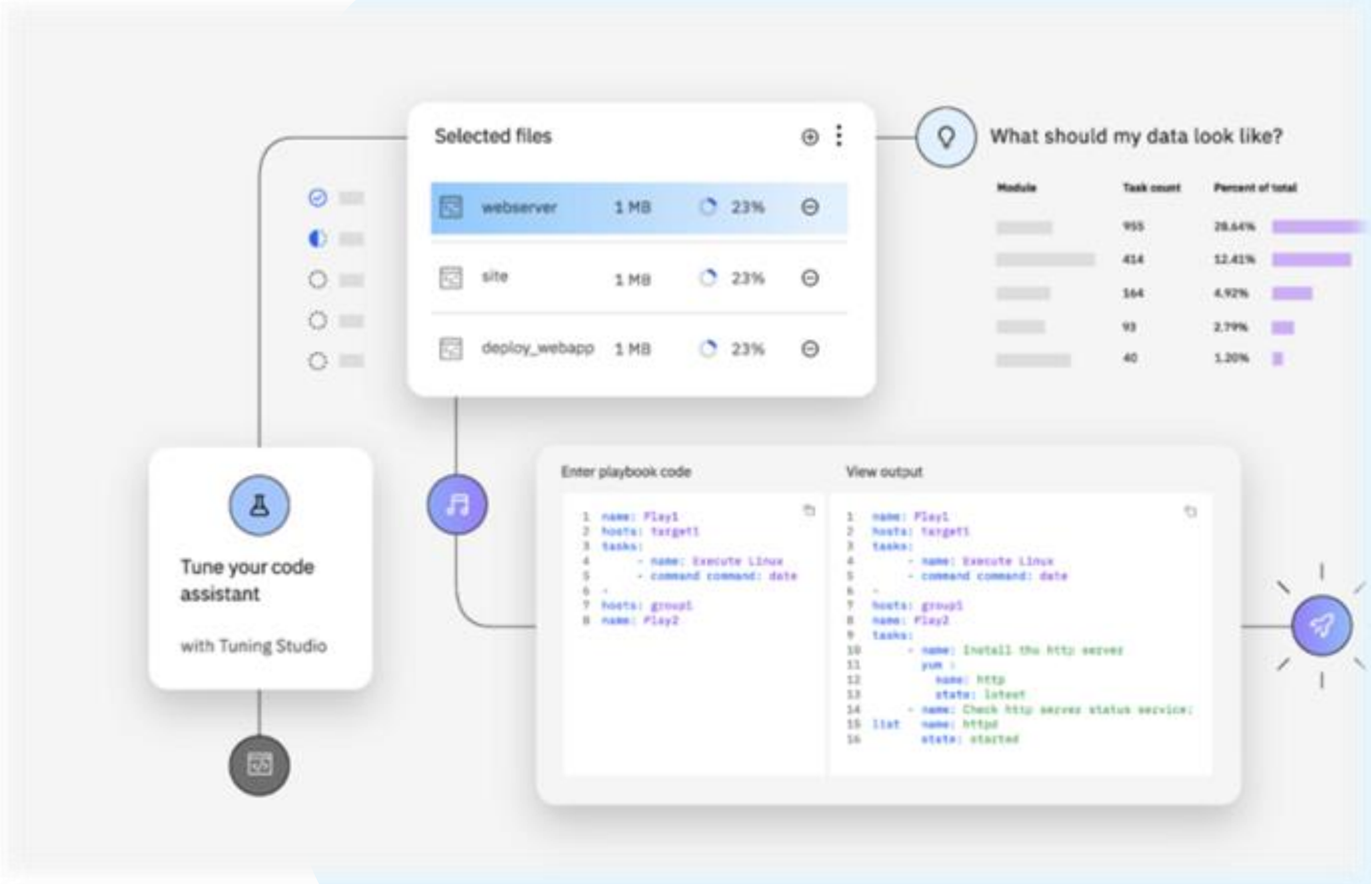
```
▼Install httpd package
    ▼amtega.migrate_to_mount
        ▪ URL: https://galaxy.ansible.com/ui/standalone/roles/amtega/migrate_to_mount
        ▪ Path: molecule/default/prepare.yml
        ▪ Data Source: Ansible Galaxy roles
        ▪ License: gpl-3.0
        ▪ Score: 0.979437
    ▶tulibraries.ansible_role_passenger_apache
    ▶gautam43.apache_role_for_lb
```

- Generate tasks for a playbook or role from natural language task description in single- or multi-task
- Improved accuracy of content from AI-generated recommendations trained on Ansible datasets
- Each suggestion will include a potential source, its author, and its license
- Data Security: Prompt/Suggestion data is encrypted in transit and is ephemeral

30%

Reduction in time spent creating Ansible Playbooks for Water Corporation

Key feature: Model Customization



What should my data look like?

Module	Task count	Percent of total
	955	28.64%
	414	12.41%
	164	4.92%
	93	2.79%
	40	1.20%

Enter playbook code

```
name: Play1
hosts: target1
tasks:
      - name: Execute Linux
      - command command: date
-
hosts: group1
name: Play2
```

View output

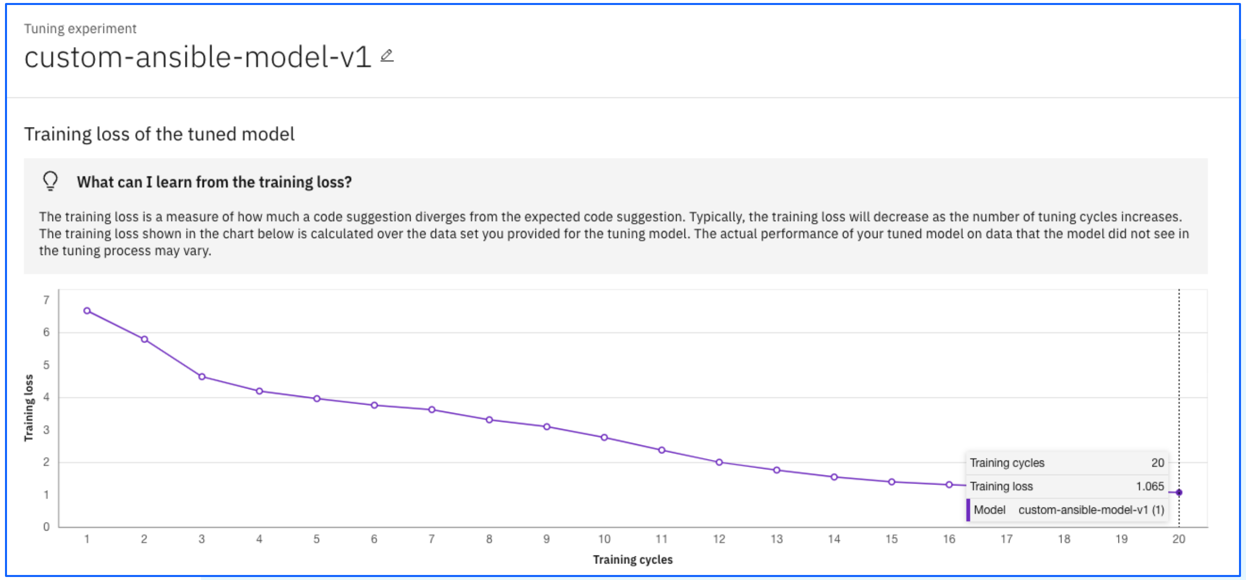
```
name: Play1
hosts: target1
tasks:
      - name: Execute Linux
      - command command: date
-
hosts: group1
name: Play2
tasks:
      - name: Install the http server
        yum :
          name: http
          state: latest
      - name: Check http server status service:
list    name: httpd
        state: started
```

- Model customization empowers organizations to tune its domain-specific LLM with their own Ansible data

- With Customization, customers can tailor the content recommendations to their organization’s preferences.

Key feature: **Model Customization**

Model Customization is done using **Prompt Tuning**



Tuning experiment

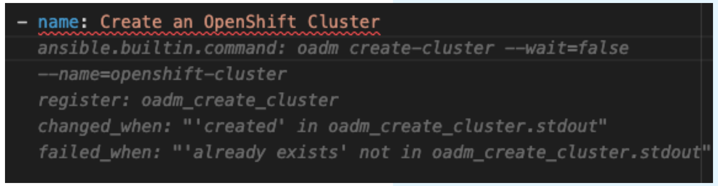
custom-ansible-model-v1

Training loss of the tuned model

What can I learn from the training loss?

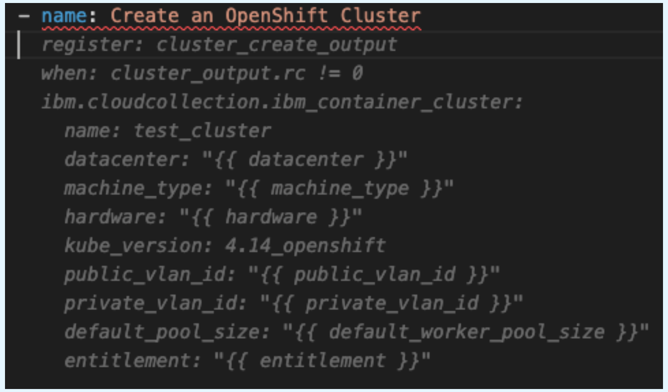
The training loss is a measure of how much a code suggestion diverges from the expected code suggestion. Typically, the training loss will decrease as the number of tuning cycles increases. The training loss shown in the chart below is calculated over the data set you provided for the tuning model. The actual performance of your tuned model on data that the model did not see in the tuning process may vary.

- **Prompt-tuning** is an efficient, low-cost way of adapting an AI foundation model to new downstream tasks
- **Content recommendations** are more similar to an enterprise's existing style for playbooks
- **Data security:** Customer's Ansible playbooks and customized models are stored in customer-owned Cloud Object storage and are not shared with IBM,
Red Hat, or any other customers



```
- name: Create an OpenShift Cluster
  ansible.builtin.command: oadm create-cluster --wait=false
  --name=openshift-cluster
  register: oadm_create_cluster
  changed_when: "'created' in oadm_create_cluster.stdout"
  failed_when: "'already exists' not in oadm_create_cluster.stdout"
```

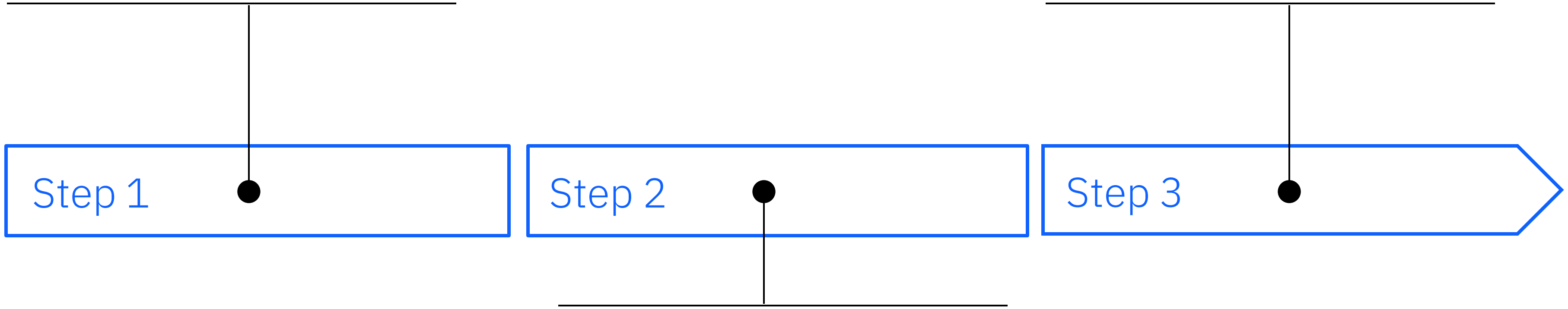
Before tuning, `ansible.builtin.command` module is being recommended



```
- name: Create an OpenShift Cluster
  register: cluster_create_output
  when: cluster_output.rc != 0
  ibm.cloudcollection.ibm_container_cluster:
    name: test_cluster
    datacenter: "{{ datacenter }}"
    machine_type: "{{ machine_type }}"
    hardware: "{{ hardware }}"
    kube_version: 4.14_openshift
    public_vlan_id: "{{ public_vlan_id }}"
    private_vlan_id: "{{ private_vlan_id }}"
    default_pool_size: "{{ default_worker_pool_size }}"
    entitlement: "{{ entitlement }}"
```

After tuning, `ibm.cloudcollection.ibm_container_cluster` module is recommended

Next steps



Step 1

See IBM watsonx Code Assistant for Red Hat Ansible Lightspeed in action

Visit Seismic for demos, badges, and more

Step 2

Explore TechZone

Ansible Lightspeed resources are available for sellers to explore and showcase client demos

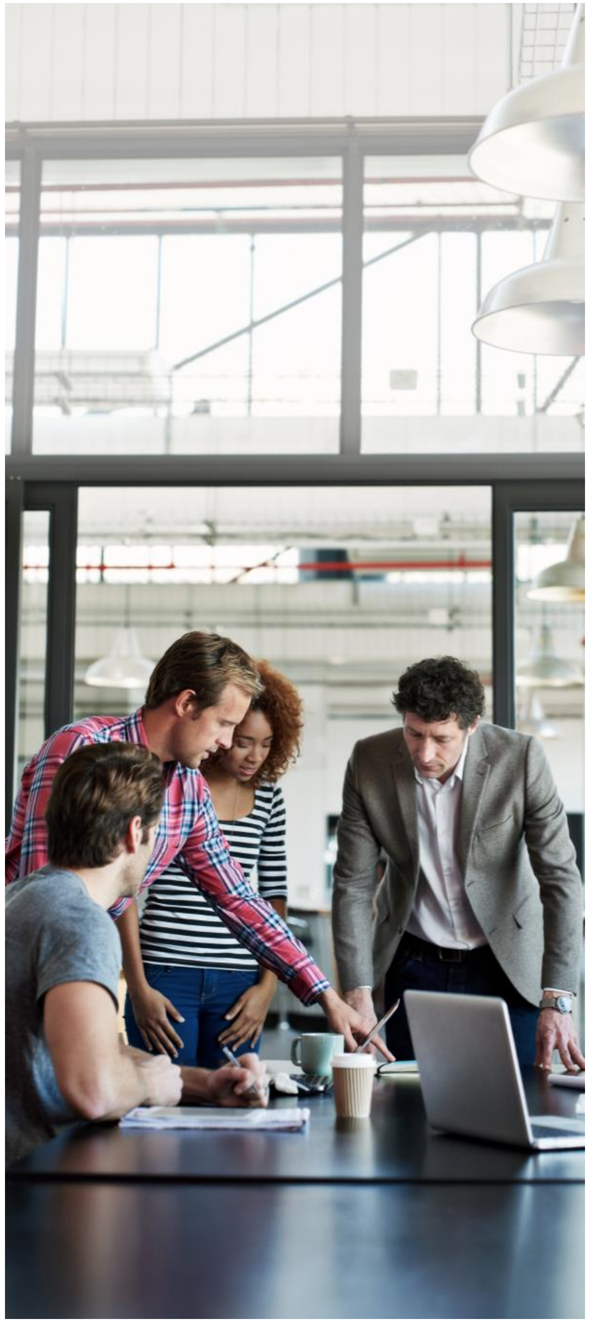
Try it for yourself→

Step 3

Conduct a Pilot

Work with Client Engineering to deliver results

Submit a Deal Support Request



IBM watsonx Code Assistant for Red Hat Ansible Lightspeed

2024 anticipated roadmap highlights

1Q’24	
Intended Capability	Outcome
IBM Cloud Trial Experience	Users can experience the capabilities of watsonx Code Assistant for Red Hat Ansible Lightspeed at no charge based on usage limits
Model Tuning & Customization	Create custom Ansible recommendation models
Productivity Dashboard	Organizations can view usage efficiency metrics for their Ansible developers
Expand WCA service to Frankfurt MZR	Data Center expansion to Frankfurt

2Q’24	
Intended Capability	Outcome
Playbook Generation (Phase 1)	Chat style experience to generate Ansible content from single prompt tasks.
Ansible Content Generation Improvements	Improved Ansible content recommendations
On-Premises Deployment	Make IBM watsonx Code Assistant for Red Hat Ansible Lightspeed available in an on-premises setup. Frankfurt Datacenter.
Model Lifecycle Management	Users will be able to manage their tunes, and understand when they need to re-tune as new models are published and old models are retired

2H’24	
Intended Capability	Outcome
Playbook Content Explanation	Generate descriptive explanation for content generated
Playbook Generation (Phase 2)	Expand on Ansible Playbook content recommendations
Tuning enhancements for Playbook generation	WCA admin should be able to tune the WCA model to improve Playbook generation inferences
Ansible content description and documentation	Find existing Ansible content instead of writing from scratch
Content Controls	Specific controls around data sent to Ansible Lightspeed
Custom Post-Processing	Parse unstructured data into structures
Runbook Recommendations	Support recommendations for Event-Driven Ansible
Expand WCA service to Japan and London MZR	Data Center expansion to Japan and London

Roadmap is subject to change

watsonx Code Assistant for Red Hat Ansible Lightspeed

2024 Plan Packages

Metric Name	What does it mean?	How is it calculated?	Additional Information
Resource Unit	Consumption metric directly tied to how much the customer uses. Is comprised of tokens and helps to estimate task prompts.	One Resource Unit = 10,000 tokens. Tokens are the usual charge metric for prompting a Foundation Model like watsonx Ansible Code Assistant for Z	• Tokens are consumed when developers make a task prompt the system, and WCA generates the code requested. • For example, developer asks "write a script to create a new service that has …." and WCA generates the Ansible code. • On average, one developer will task prompt WCA 200 times per month. • On average, one task prompt will use ~500 tokens for a total of ~100K tokens per developer per month. **On average, one developer will use 10 RUs per month.**
Instance	A flat monthly fee to access WCA Standard. The instance fee includes some Resource Units.	One Instance per Organization per Month	• The number of Resource Units included in this plan (660 RUs) provides a plentiful baseline for Enterprise size customers to populate tasks
Tuning Hour	A fee per hour of using the Tuning Studio to prompt tune the model. Tuning is available only in the Standard Plan.	Priced per hour of usage within the WCA Tuning Studio	• Approximation for tuning: Customers will tune the model 4 – 6 times per year

Detailed deck on how to Price Quote and Order watsonx Code Assistant for Red Hat Ansible in Seismic at:
https://ibm.seismic.com/Link/Content/DCfpXGXjQ3fWp82BPf6gC4VJCTc3
Sizing Calculator: https://ibm.seismic.com/Link/Content/DC4fjFh2m78dm8FQqpJ4HRWhTpgG