

IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide

Description	IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide
Author(s)	Andrew R. Jones (andrewj@us.ibm.com) and Max Weiss (maxwell.g.weiss@ibm.com)
Copyright	Copyright © 2024 IBM - V 2.0

Table of Contents

Welcome

- Support
- Using the demonstration guide

IBM Technology Zone environment

- Create a reservation request
- Extend the reservation
- Join the
- Accessing the environments
- Troubleshooting

Pilot setup

Creating an assistant and configuring conversational search

- Access the
- Create your Assistant
- Configure conversational search
- Complete the configuration
- Configure the base large language model
- Testing conversational search
- Troubleshooting

Ingesting client documents

Creating a stand-alone OpenSearch instance for document ingestion

- Install the Red Hat OpenShift command-line interface utility
- Prepare to ingest documents
 - Log in to the OpenShift cluster from your local terminal
 - Create a working directory
 - Install IBM Certificate Manager on Red Hat OpenShift
 - Install the watsonx Assistant for Z Operator (for OpenSearch)
 - Deploy required secrets and the custom bring-your-own-search (BYOSearch) resources
 - Verify all the required pods are running and get the network route to your
 - Update your assistant with the new
- Troubleshooting

Installing and using zassist to ingest client documents

- Install the zassist utility
- Ingest client documentation using zassist
- Verify the document that is ingested is now returned as a source file for a query

- Adjusting the search behavior

Adding skills to the assistant

Getting started with skills and actions

- Environments
 - Watsonx Orchestrate
 - Ansible Automation Platform and Wazi as a Service

Explore Ansible Automation Platform

- Access the

Importing skills from Ansible Automation Platform

- Import skills into your assistant

Connecting skills to your assistant

Creating actions for your assistant

Configure the number of input fields

- Create actions

Verify the job in the Ansible Automation Platform console

- Troubleshooting

Creating skill flows

- Add the utility skill
- Add the skills to your Personal skills
- Create the skill flow
- Enable the skill flow in your assistant

Creating custom-built actions

Importing pre-packaged z/

Publishing and deploying your assistant

- Publish the assistant
- Configure the live environment
- Connect the skills to the live environment
- Deploy the assistant

Adding other integrations

Next steps

- Earn the badge

Welcome

Welcome to the IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide (lab guide). The lab guide is part of the IBM watsonx Assistant for Z for Technical Sales Level 4 learning plan for IBM and Business Partner Technical Sales and related badge. The learning plan is intended to teach technical sellers and Business Partners how to conduct a proof of experience (PoX) for a client.



In a fast-paced world, things change

The products and services may appear differently than what is shown in the lab guide. This can occur if the product or service is updated with a new version.

Responses generated by IBM watsonx Assistant for Z are likely to change over time. The responses you see when you run the queries in this section may differ from the screen images captured in the lab guide.

This lab guide covers the setup, configuration, and usage of watsonx Assistant for Z. This lab guide uses the [IBM watsonx Assistant for Z Velocity collection](#) and the three Velocity Pilot lab environments in IBM Technology Zone (ITZ).

The lab guide also enables dedicated lab environments for customized client PoXs and demonstrations. If you are preparing for an actual pilot engagement, refer to the [Pilot Scoping Guide for watsonx Assistant for Z](#) for additional information.

The lab guide provides the following guidance to:

- Provisioning the lab environments
- Creating an assistant and configuring conversational search
- Configuring assistant settings
- Testing conversational search
- Deploying a dedicated instance of OpenSearch for client document ingestion (Optional)
- Importing skills for z/OS automations
- Connecting apps to assistants
- Creating assistant actions
- Creating skill flows
- Creating custom-built actions
- Importing pre-packaged z/OS skills
- Publishing and deploying your assistant

Support

Think something is down? Check the applicable status pages for any known issues such as a site or service not being available:

- [IBM Technology Zone status](#)

For issues with provisioning the ITZ environment for this lab (for example, a failed reservation request due to insufficient quota capacity), open a ticket with ITZ support:

- Web: [IBM Technology Zone ticket system](#)
- Email: techzone.help@ibm.com

For issues related to specific steps found in the demonstration guide after the ITZ environment is provisioned, contact the authors:

- Slack:
 - [#watsonx-assistant-z-technical](#) - IBM only
 - [#wxo-practitioners](#) - IBM only - for questions that are related to the software as a service (SaaS) instance of watsonx Orchestrate
- Email: andrewj@us.ibm.com and maxwell.g.weiss@ibm.com

Business Partners, use the IBM Training live Chat Support service or other support methods that are found on the IBM Training portal [here](#).

Using the demonstration guide

Use these helpful tips to take full advantage of the IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide.



Printing the demonstration guide

⚠ Printed or saved copies can be out of date

The IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide changes regularly to match the IBM watsonx Assistant for Z offering and associated ITZ environment. Printed or saved copies of the demonstration guide can become out-of-date quickly and result in failed steps.

A ready-to-print PDF version of the IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide is [here](#).



Viewing images

Images in the demonstration guide can be enlarged by clicking on the image. Press the **Esc** key or click the **X** to dismiss the enlarged image.



Reserve the IBM Technology Zone environment



Search

IBM watsonx Assistant for Z for
Technical Sales Level 4 Lab
Guide

Welcome

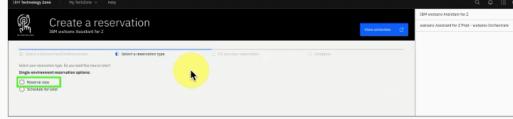
Reserve the IBM Technology
Zone environment

Pilot setup

Next steps

2. Click Reserve now.

The **Reserve now** option creates a reservation for immediate use. Optionally, schedule the reservation for a later date, like when you will be at your client's office.



3. Complete the reservation request and click Submit.

The first two reservations will be similar to the first image below and have fields a-e that will need to be completed.

a. Optionally, change the **Name** field for the reservation.

b. Select the **Education** purpose tile.

c. Enter a **Purpose description**.

d. Select the region nearest your physical location in the **Preferred Geography** drop-down.

e. The **End date and time** will be set to 2 days after the current date and time.

f. Accept the IBM Technology Zone's terms and conditions and security policies.

g. When satisfied with the parameters, click **Submit**.



Image highlighting

In some images, the following styles of highlighting are used:

- **Solid highlight box:** This style of box highlights where to click, enter, or select an item.



- **Dash highlight box:** This style of box highlights one of two things: the path to follow to get to a specific location in the user interface, or areas to explore on your own.



Copying commands and prompts

Copying and pasting commands and prompts from this demonstration guide is easy and can eliminate typographical errors.

Click the highlighted copy icon and then use your operating system's paste function. For example, $\text{^ Ctrl } + \text{ V }$ or right-click and select Paste.

Prompt 1

What is the APF list in z/OS? Provide a detailed explanation.





Acronyms and terminology



IBM employees and the tech industry in general, tend to use acronyms. In the demonstration guide, most acronyms will appear with a dashed underline. Hover over the acronym to learn its meaning. A question mark (?) icon will first appear and after a second the tool tip with the acronym's meaning is displayed. Try it here: LPAR.

Guidance for delivering a demonstration

IBM watsonx Assistant for Z

Welcome

Reserve the IBM Technology Zone environment

Guidance for delivering a demonstration

Demonstration scenarios >

Next steps

Specific guidance for IBM watsonx Assistant for Z and the ITZ environment

Following the scripts provided in the IBM watsonx Assistant for Z for Technical Sales Level 3 Demonstration Guide will help guarantee a successful demonstration. Use these tips to help insure success with IBM watsonx Assistant for Z and the ITZ environment:

- Follow the scripts in the IBM watsonx Assistant for Z for Technical Sales Level 3 Demonstration Guide for the automations and skills to execute as expected.

Table of contents

General demonstration guidance

Specific guidance for IBM watsonx Assistant for Z and the ITZ environment



The Lab Guide table of contents

This **Demonstration Guide** uses a responsive browser-based interface to ensure pages are usable on various devices with different screen sizes. The Demonstration Guide table of contents may be displayed as highlighted in the green dashed box in this image:

The screenshot shows a browser window for the "IBM watsonx Assistant for Z for Technical Sales Level 3 Demonstration Guide". The title bar includes the course name and a search bar. On the left, a green dashed box highlights the "Table of Contents" sidebar, which lists various sections like Welcome, Guidance for delivering a demonstration, Reserve the IBM Technology Zone environment, etc. The main content area displays the "Welcome" page, which introduces the course goal and provides information about its focus and the Velocity Pilot information.

However, if the browser window is sized smaller, the table of contents can be accessed by clicking the main menu icon (≡):

The screenshot shows the same browser window after the main menu icon was clicked. The sidebar is now collapsed, and the main content area displays the "Welcome" page, which includes the introductory text about the course goal and the Velocity Pilot information.

Click the main menu icon (≡) to expand the table of contents.

Continue to the [Reserve the IBM Technology Zone environments](#) section to begin the journey to obtain the IBM watsonx Assistant for Z Technical Sales Advanced badge.

IBM Technology Zone environment

To enable sellers to learn how to deliver client pilots of IBM Watsonx Assistant for Z, three environments are available in IBM Technology Zone (ITZ). The environments are part of the Watsonx Assistant for Z Velocity lab collection and can be found in the [IBM Watsonx Assistant for Z](#) collection.

- **Watsonx Assistant for Z lab – Watsonx Orchestrate:** a dedicated environment with Watsonx Orchestrate on IBM Cloud that is used for creating and configuring the assistant, configuring conversational search, importing skills, and configuring actions.
- **Ansible Automation Platform (AAP) & z/OS:** a pre-configured instance of AAP and Wazi z/OS. The environment includes preinstalled Ansible playbooks that can be imported as skills within Watsonx Orchestrate and connected to your assistant. The environment provides preinstalled templates for various use cases that are covered in a later section. Learn more about AAP [here](#). Learn more about Wazi, [here](#).
- **Single Node OpenShift with NFS storage:** used to provision a single-node Red Hat OpenShift cluster (SNO) on IBM Cloud. The cluster is used to install a dedicated instance of [OpenSearch](#) for Watsonx Assistant for Z. This environment enables ingestion of client-supplied documents.



All activities in this lab guide are required.

While all three environments may not be required for every client pilot, to complete the Level 4 learning plan and earn the IBM Watsonx Assistant for Z Technical Sales Advanced badge, you must provision all three ITZ environments and complete all sections in the lab guide. **Any statements in the ITZ collection regarding optional environments or tasks should be ignored.**

Follow the instructions to create new reservation requests, extend the reservations, and access the ITZ demonstration environments. Provisioning the SNO environment in ITZ can take several hours, while the other two environments typically provisioning in under 30 minutes.

Create a reservation request

1. Click each of the links that follow to open a browser to the reservation pages of the **IBM Watsonx Assistant for Z** ITZ environments.

You may be asked to authenticate to IBM Technology Zone.

The steps to authenticate to ITZ are not detailed here as they may vary between users.

[Watsonx Assistant for Z lab – Watsonx Orchestrate - reservation page](#)

[Ansible Automation Platform \(AAP\) & z/OS - reservation page](#)

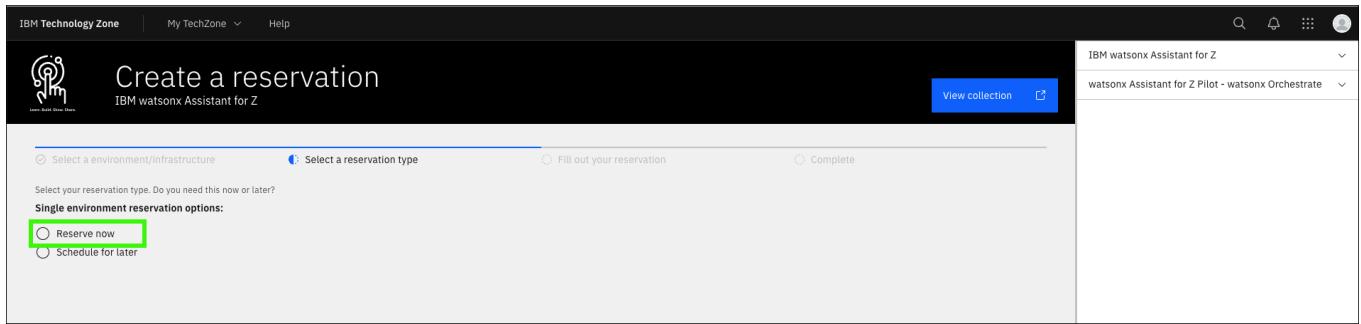
Single Node OpenShift with NFS storage - reservation page

 The next two steps are for one of the three environments. Repeat for all three environments.

Be sure to follow these steps to create a reservation in ITZ for all three environments.

2. Select Reserve now.

The **Reserve now** option creates a reservation for immediate use. Optionally, schedule the reservation for a later date, for example, when you are at your client's office to start a pilot.



The screenshot shows the 'Create a reservation' interface. At the top, there are tabs for 'Select a environment/infrastructure', 'Select a reservation type', 'Fill out your reservation', and 'Complete'. Below these tabs, there is a section titled 'Single environment reservation options:' with two radio buttons: 'Reserve now' (which is selected and highlighted with a green box) and 'Schedule for later'.

3. Complete the reservation request form and then click Submit.

The first two reservations are similar to the first image and have fields **a-e** that need to be completed.

- a.** Name: specify a name for the reservation (optional).
- b.** Purpose: select the **Education** purpose tile.



For client pilots...

For client pilots, set the **Purpose** field in the reservation to **Pilot** and provide an opportunity number to receive a longer reservation.

- c.** Purpose description: enter a description, for example: Level 4 education.
- d.** Preferred geography: select the region nearest to your physical location for improved performance and reduced network latency.
- e.** End date and time: the initial maximum will be set to two days after the current date and time. Instructions follow to extend the reservation end date.
- f.** Accept the IBM Technology Zone's terms and conditions and security policies.
- g.** Click **Submit**.

Create a reservation

IBM Watsonx Assistant for Z

Name a

Select a reservation type b

Purpose c

Sales Opportunity number d

Preferred Geography e

End date and time f

Notes

I agree to IBM Technology Zone's [Terms & Conditions](#) and [End User Security Policies](#) g

Submit

In addition to the preceding fields, the reservation for the **Single Node OpenShift with NFS storage** has these additional fields:

- h. OCP/Kubernetes cluster network: leave the default setting of **10.128.0.0/14**.
- i. Enable FIPS security: leave the default setting of **No**. Learn more about the Federal Information Processing Standards (FIPS) [here](#).
- j. Master single node flavor: select **16 vCPU x 64 GB - 300 GB ephemeral storage**.
- k. OpenShift version: select **4.14**.
- l. OCP/Kubernetes service network: leave the default setting of **172.30.0.0/16**.
- m. Accept the IBM Technology Zone's terms and conditions and security policies.
- n. Click **Submit**.

IBM Technology Zone | My TechZone | Help

Enter date and time Select a date 11/01/2024 Select a time 11:51 AM America/Chicago

Reservation policy: Recommended 2 days, but can be reserved up to 2 days on this reservation form. Extend later for 2 days increments up to 4 days total. Max time 6 days total.

OCP/Kubernetes Cluster Network 10.128.0.0/14 h

Enable FIPS Security No i

Master Single Node Flavor 16 vCPU x 64 GB - 300 GB ephemeral storage j

OpenShift Version 4.14 k

OCP/Kubernetes Service Network 172.30.0.0/16 l

Notes Enter any notes you would like to attach to this reservation

m I agree to IBM Technology Zone's [Terms & Conditions](#) and [End User Security Policies](#)

n Submit

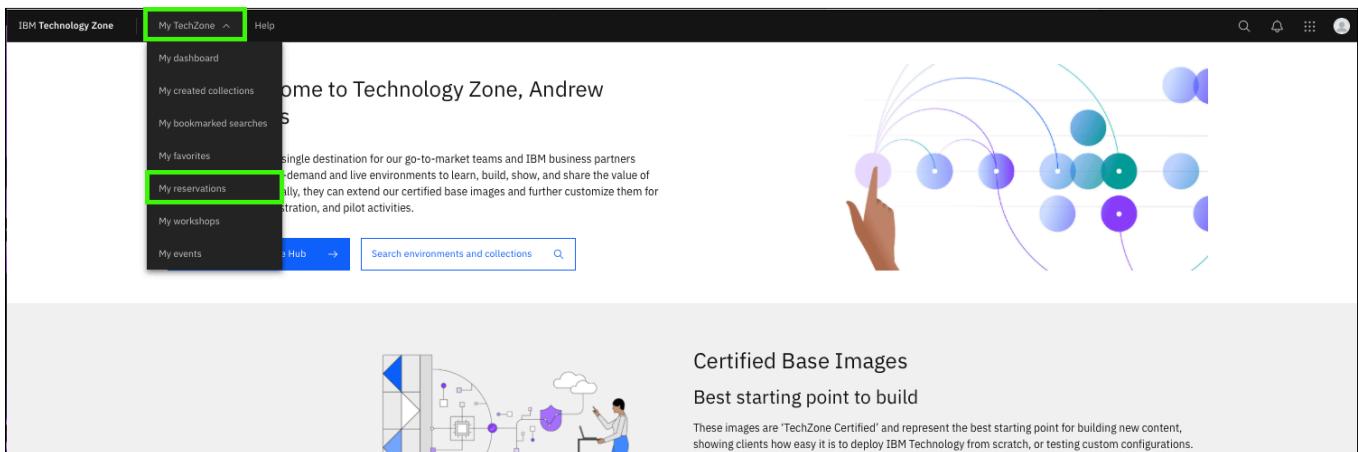
During the provisioning process, multiple emails are sent to you from ITZ as the provisioning process runs. One email states the reservation is provisioning and the other email states that the environment is **Ready**.

In rare cases, the provisioning process may fail. If you receive an email stating the reservation failed, try again by repeating Steps 1-3 for the environment that failed to provision. In addition, review the [Troubleshooting](#) section that follows. If issues continue, open an [ITZ support ticket](#) by using the methods that are mentioned in the [Support](#) section.

Extend the reservation

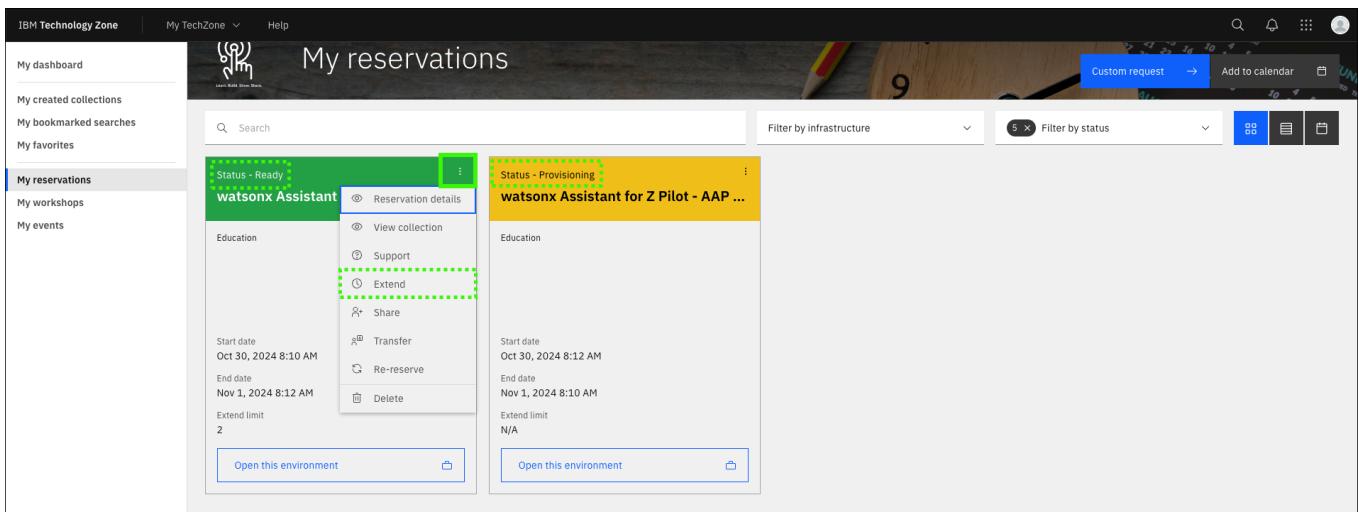
When the reservations are in the **Ready** state, you can extend each reservation to a total of six days.

1. In the IBM Technology Zone portal, expand **My TechZone** and select **My Reservations**.



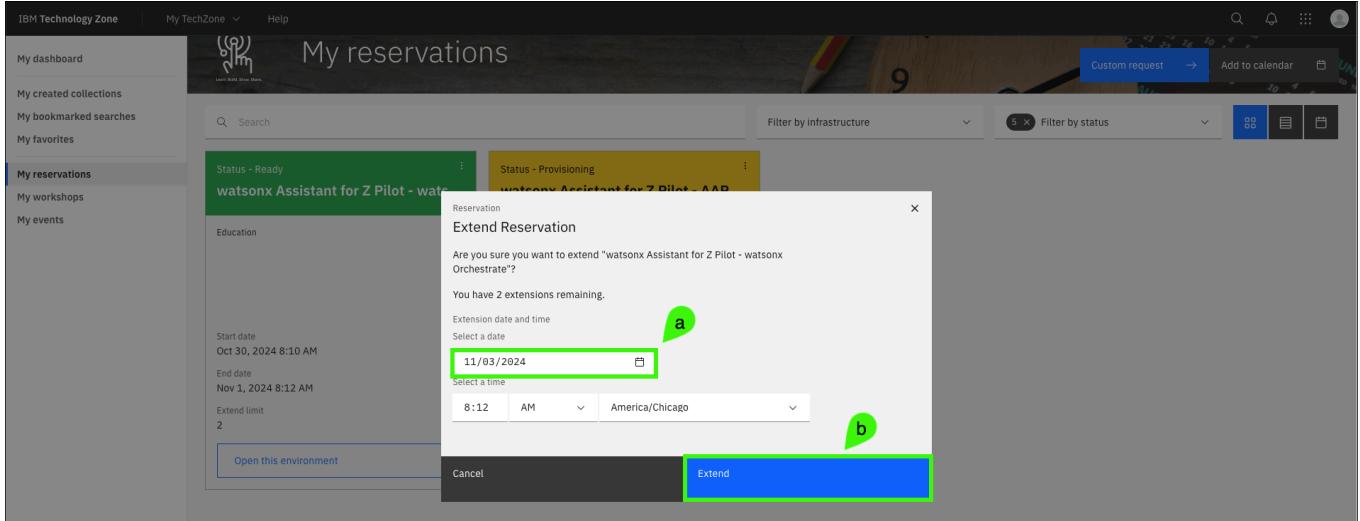
The screenshot shows the IBM Technology Zone portal. At the top, there's a navigation bar with 'IBM Technology Zone', 'My TechZone ▾', and 'Help'. Below the navigation is a sidebar with links: 'My dashboard', 'My created collections', 'My bookmarked searches', 'My favorites', 'My reservations' (which is highlighted with a green box), 'My workshops', and 'My events'. The main content area has a heading 'Welcome to Technology Zone, Andrew' and a graphic of a hand pointing at a network of nodes. Below this is a section titled 'Certified Base Images' with a sub-section 'Best starting point to build' and a note about TechZone Certified images.

2. Click the overflow icon (≡) on the reservation tile and select **Extend**.



The screenshot shows the 'My reservations' page. The left sidebar is identical to the previous screenshot. The main area displays two reservation cards. The first card, for 'watsonx Assistant', shows a status of 'Status - Ready'. A context menu is open over this card, with the 'Extend' option highlighted. The second card, for 'watsonx Assistant for Z Pilot - AAP ...', shows a status of 'Status - Provisioning'. Both cards have 'Open this environment' buttons at the bottom. The top right of the screen shows various navigation and search tools.

3. Click the **Select a date** option, (a) specify the date to extend to, and then (b) click **Extend**.



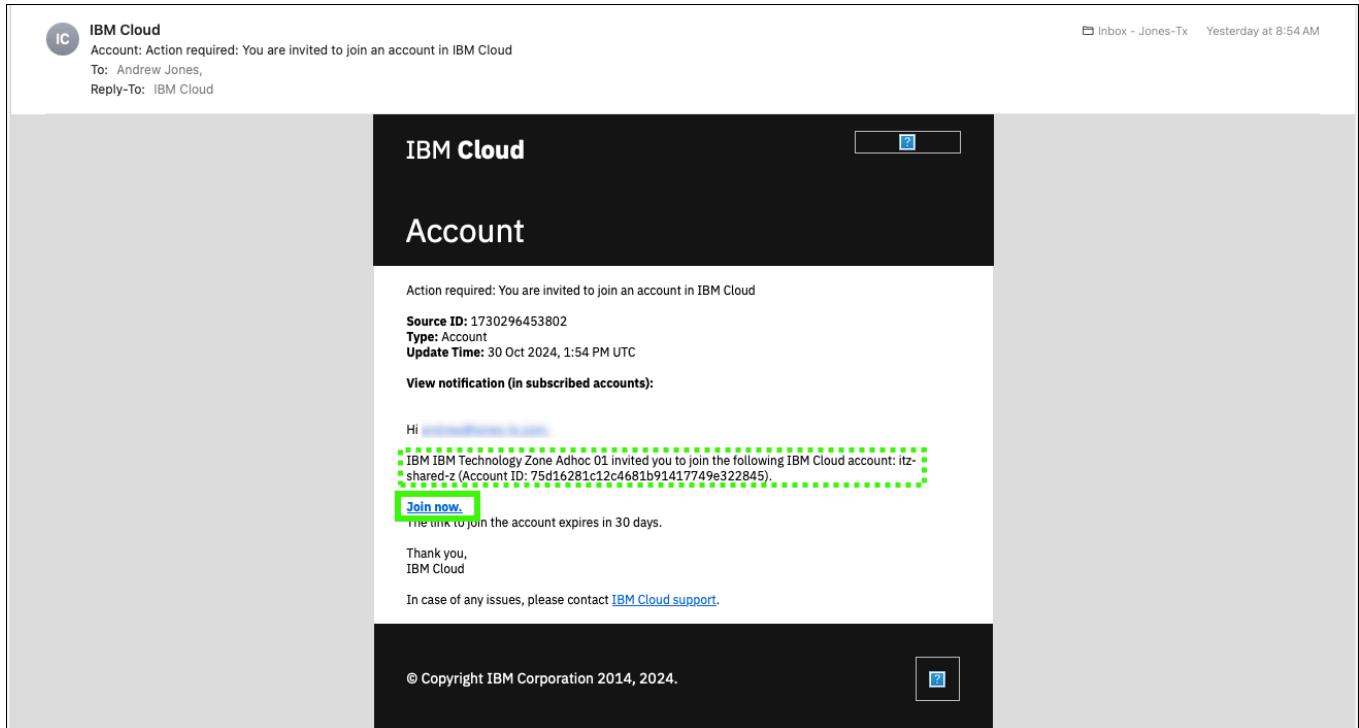
If you anticipate needing more time, repeat Steps 5-6 to extend the reservation to the maximum of six days. Repeat these steps for the other two reservations.

Join the ITZ IBM Cloud account

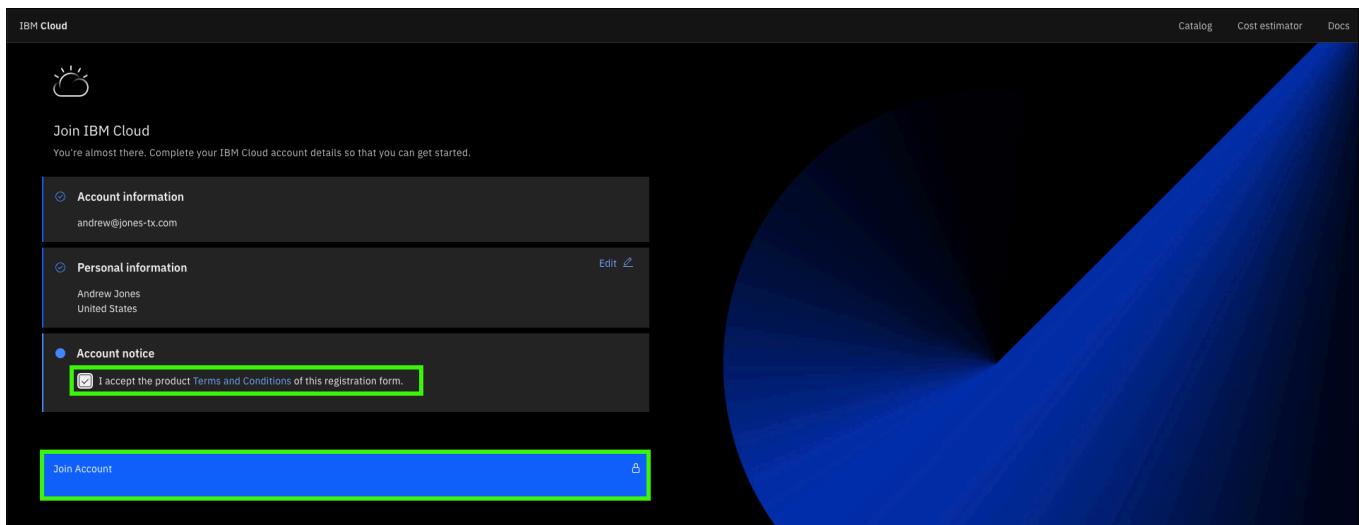
Both the **watsonx Assistant for Z lab – watsonx Orchestrate** and the **Ansible Automation Platform (AAP) & z/OS** environments add you to an IBM Cloud account while your reservation is active. During the provisioning process of these **ITZ** environments, you receive two emails from IBM Cloud.

You only need to accept the invitation to the **watsonx Assistant for Z lab – watsonx Orchestrate** environment.

1. Open the email from **IBM Cloud** and click the **Join now** links.



2. In the **Join IBM Cloud** browser window that opens, select the **I accept the product Terms and Conditions** of the registration form, and then click **Join Account**.

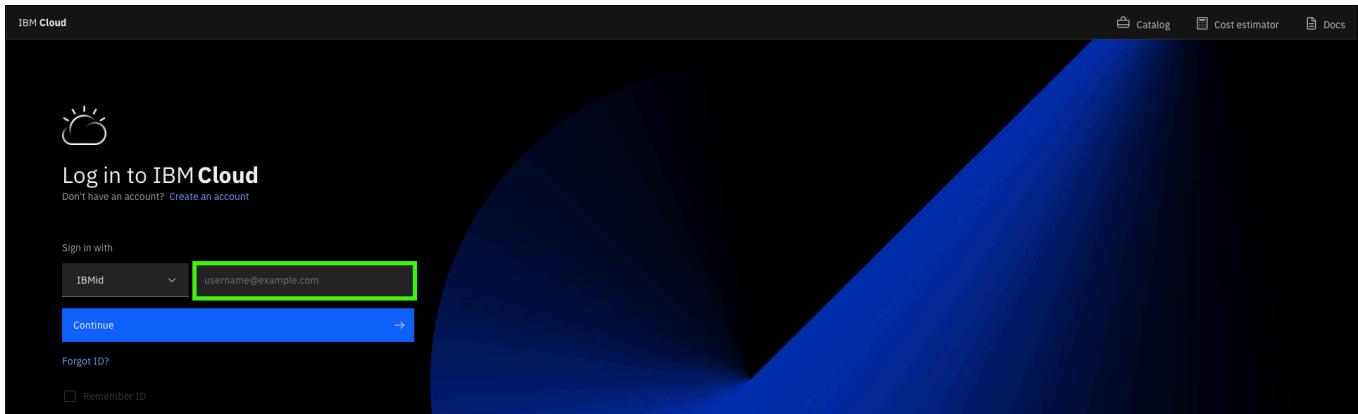


After joining the account, verify that the account appears in your available account list in the IBM Cloud portal.

- Click the following link to open a browser to the IBM Cloud portal.

IBM Cloud portal

- Follow the directions to complete the authentication to IBM Cloud using the same email address you used to login to ITZ. The login steps vary depending on any two-factor authentication methods enabled.



- Click the **account** menu and verify access to the IBM Cloud account listed in your ITZ reservation.

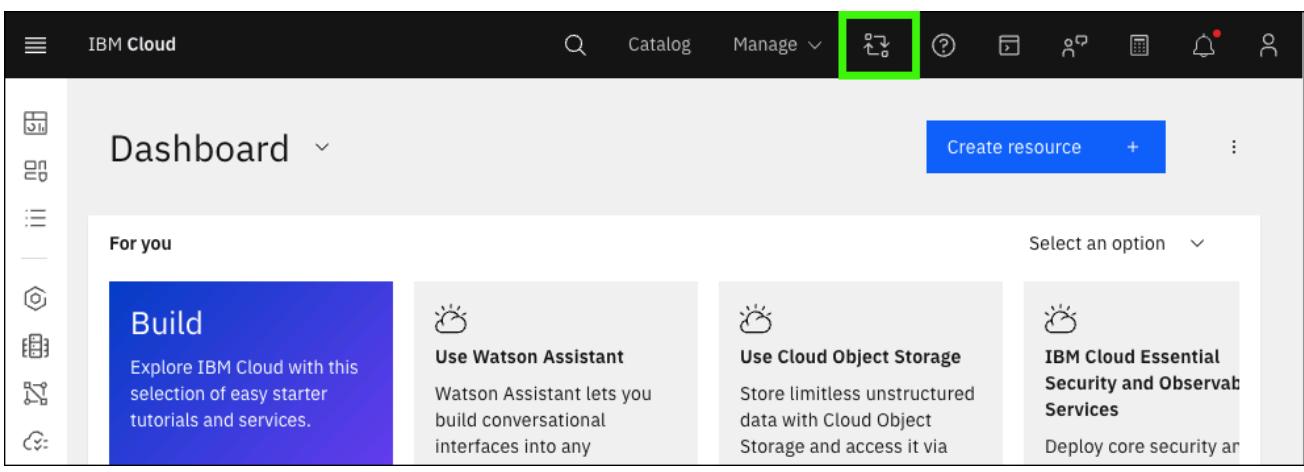


The account may change within ITZ.

Over time, the account may change for the environments. The account names should align with the account named in the invitation email you received.

 Does your IBM Cloud portal view look different?

If your IBM Cloud portal looks different from the images above, it could be because the IBM Cloud portal has gone through a design change, or your browser window is set to smaller size. Instead of the current selected account appearing in the top menu, you may see this **change account** icon:  Click this icon to view the list of accounts you can access.



Accessing the environments

Each reservation provides access to its respective environment. Details for accessing each environment are provided in the **Pilot setup** sections that follow in the lab guide.

After all three reservations are in the **Ready** state and you accept the invitations to the IBM Cloud accounts, proceed to the next section to complete the pilot setup.

Troubleshooting

 If your reservation for the Single Node OpenShift environment fails...

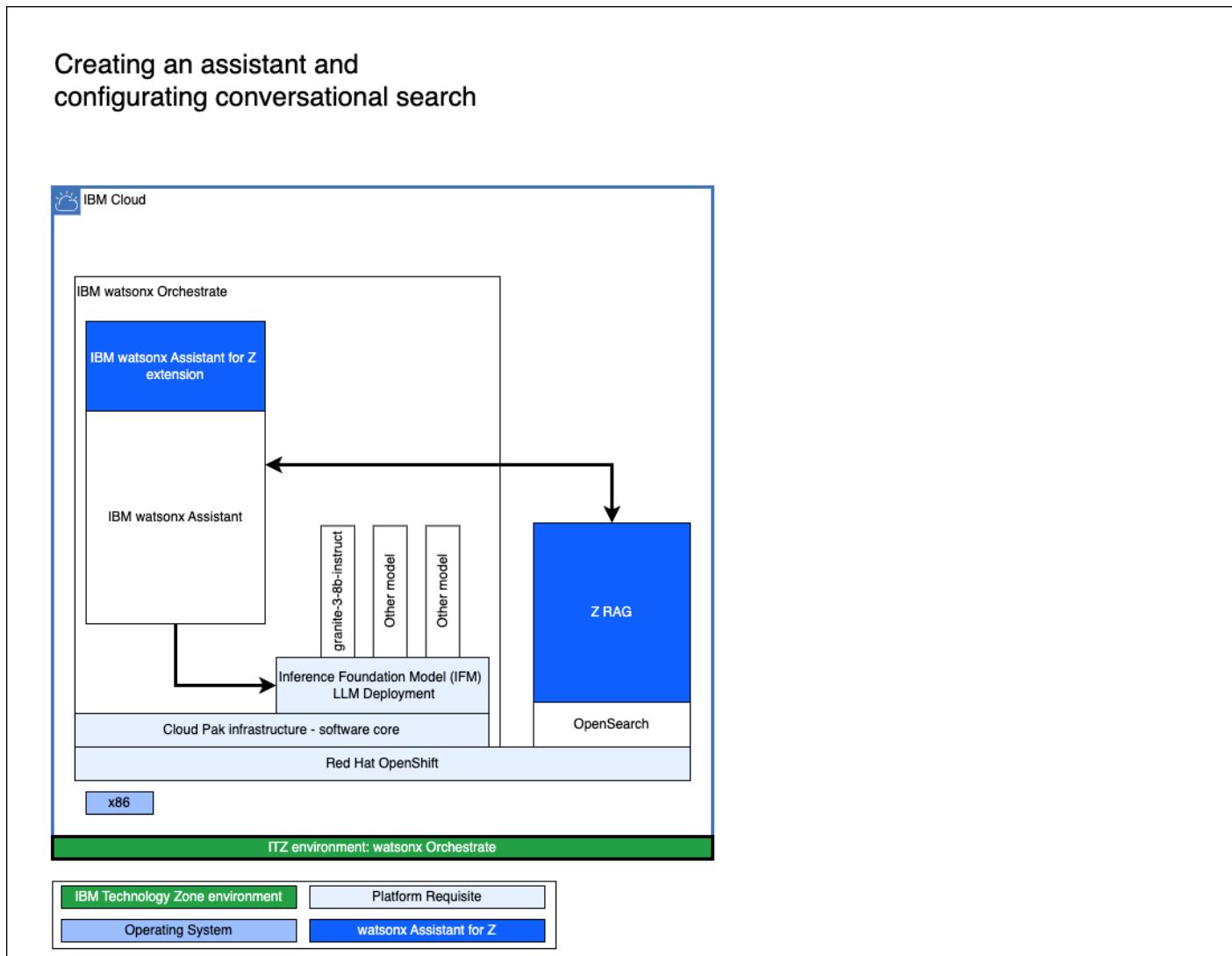
If your reservation for the Single Node OpenShift environment fails, try selecting one of the **eu-gb region** options as the **Preferred Geography**.

Pilot setup

Creating an assistant and configuring conversational search

You use [watsonx Orchestrate](#) to create the assistant and configure conversational search. You can configure your assistant to use conversational search by using a hosted [OpenSearch](#) instance. The pre-configured instance in IBM Technology Zone (ITZ) has over 220 knowledge sources and supports Retrieval Augmented Generation (RAG). The large language model (LLM) providing the conversational AI augments this knowledge based on IBM Z documentation. All of these elements create IBM Z context-aware responses to queries with its content-grounded knowledge.

Below is a high-level, logical architecture of the environment you will deploy in this section.



Access the [ITZ IBM Cloud account](#) for the Watsonx Assistant for Z Pilot environment

1. In the IBM Technology Zone portal, expand **My TechZone** and select **My Reservations**, or click the following link.

[ITZ My reservations](#)

Welcome to Technology Zone, Andrew

Certified Base Images

Best starting point to build

These images are 'TechZone Certified' and represent the best starting point for building new content, showing clients how easy it is to deploy IBM Technology from scratch, or testing custom configurations.

2. Click the **watsonx Assistant for Z Pilot - watsonx Orchestrate** tile.

My reservations

watsonx Assistant for Z Pilot - AAP & z/OS

watsonx Assistant for Z Pilot - watsonx O...

Single Node OpenShift (VMware on IBM C...

3. Record the ITZ IBM Cloud account name associated with the reservation.

watsonx Assistant for Z Pilot - watsonx Orchestrate

L4 training

Date: Nov 5, 2024 6:03 AM Nov 11, 2024 6:08 AM Expires in: 5 days, 22 hours, 45 minutes End limit: 0

Status: Ready

Purpose

Purpose: Education Opportunity Product(s): Opportunity ID(s):

Customer(s): Opportunity description: L4 training

Environment

Reservation ID: 672a09a1a8f85062f891e081 Transaction ID: 115897c9-58a1-4f17-af9c-b16dc2a97590

Request method: watsonx-orchestrate Geo: americas

Cloud Account: ITZ-WATSONX-036 Datacenter: dat10

Region: us-south Customer data: false Environment: watsonx-orchestrate-wusdf

Idle runtime limit: 10800 Timeout action:

Reservation Details

IBM Cloud Login: <https://cloud.ibm.com/resources>

4. Click the **IBM Cloud Login** link.

Purpose

- Purpose
- Education
- Opportunity Product(s)
- Customer(s)

Environment

- Reservation ID: 672a091a8f85062f891e081
- Type: IBM Cloud
- Request method: watsonx-orchestrate
- Region: us-south
- Customer data: false
- Idle runtime limit: 10800
- Transaction ID: 115897c9-58a1-4f17-af9c-b16dc2a97590
- Geo: americas
- Datcenter: dal10
- Environment: watsonx-orchestrate-wusdf
- Timeout action:

Reservation Details

IBM Cloud Login
<https://cloud.ibm.com/resources>

Steps to authenticate to IBM Cloud are not illustrated here.

You may need to authenticate to IBM Cloud after clicking the link. These steps are not shown here as they may vary by individual.

- Verify that the current IBM Cloud account is the same as the account name recorded in step 3. If the account is not the same, switch to the proper account.

Note: The formatting of the name can appear differently than what is shown in the ITZ reservation.

Resource list

Name	Group	Location	Product	Status	Tags
Filter by name or IP address...	Filter by group or org...	Filter...	Filter...	Filter...	Filter...
Compute (0)					
Containers (0)					
Networking (0)					
Storage (0)					
Converged infrastructure (0)					
Enterprise applications (0)					
AI / Machine Learning (1+)					
Analytics (0)					

If the proper account is not listed, click the account drop down and select the proper account.

Note: If your browser window is narrow, the account drop down can be depicted with the Switch Account icon ().

The screenshot shows the IBM Cloud Resource list interface. At the top, there's a search bar labeled "Search resources and products...". Below it is a navigation bar with "Catalog" and "Manage" tabs. On the left, there's a sidebar with various icons and a "Resource list" heading. The main area is a table with columns: "Name", "Group", "Location", "Product", and "Status". One row in the table is highlighted with a green box, showing the identifier "2953593 - itz-watsonx-036".

Create your Assistant

1. Click the **Resources** icon (☰).

The screenshot shows the IBM Cloud Dashboard. On the left, there's a sidebar with a "Resources" icon highlighted with a green box. The main area is a grid of service cards under the heading "For you". Each card has a title, a brief description, and some status indicators like "Popular" or "Recommended". A specific card for "Watson Assistant" is visible.

2. Expand the **AI / Machine Learning** section and click the **watsonx Orchestrate** instance listed (the instance name is different than shown in the following image).

The screenshot shows the IBM Cloud Resource list interface again. This time, the "Enterprise applications" section is expanded, and the "AI / Machine Learning" subsection is also expanded. A specific instance, "itwxo-270003nft-erspw", is highlighted with a green box. It shows details like "watsonx-orchestrate-erspw", "Dallas", "watsonx Orchestrate", and "Active".

3. Click **Launch watsonx Orchestrate**.

Resource list / itzwxo-2700039nft-erspw Active Add tags [Edit](#)

Manage

Service credentials

Start by launching the tool

[Launch Watson Orchestrate](#) [Getting started tutorial](#)

Credentials

API key: [Download](#) [Show credentials](#)

URL: <https://api.us-south.assistant-builder.watson.cloud.ibm.com/instances/a7675d8e-e885-4891-92f3-1a2a2a2a2a2a>

Plan

Essentials Plan

4. Click the AI assistant builder tile to start creating a new assistant.

Welcome, Andrew Jones!

Take productivity to the next level.

[Try Skills in Chat](#)

Build
Start building the skills, conversations, and automations your team needs here.

AI assistant builder
Boost productivity and customer care by creating conversational experiences.

Skill studio
Build the skills your team needs to get their work done more quickly.

Upgrade
Standard plan | Skill studio
Automate how your business works

5. Enter a name and optional description for your assistant and click **Next**.

Welcome to AI assistant builder

Create Personalize Customize Preview

Create your first assistant

Let's get your assistant up and running. Name your assistant, add a description, and choose a language. In following steps we'll gather more information, show you basic customizations, and give you a preview of what your assistant will look like.

Assistant name:

Your assistant name will be kept internally and not visible to your customers

Description (optional):

Assistant language:

This is the language your assistant will speak.

Next

6. Complete the **Personalize your assistant** form and click **Next**.

Explore the personalization options. In creating an assistant for a client pilot, consider specifying attributes that align with the client's business.

- a. Select **Web**.
- b. Select the industry of your choice.
- c. Select the role of your choice.
- d. Select the need of your choice.

Welcome to AI assistant builder

Create **Personalize** **Customize** **Preview**

Personalize your assistant

Tell us where your assistant will live
You may add multiple channels from your dashboard.

Where do you plan on deploying your assistant?
Web

Tell us about yourself
This information will be used to personalize your onboarding experience.

Which industry do you work in?
Software

What is your role on the team building the assistant?
Developer

Which statement describes your needs best?
I want to automate common tasks in a natural way

This is what your customers will experience

watsonx Assistant

Do you have the Speed Demons in stock?

The Speed Demons are in stock at our Downtown and Northgate locations, which are both within 5 miles of you.

What size and color do you need?

I'm looking for a size **9** in white

Great news! The Speed Demons are available in white in a size 9.

You can purchase them for curbside pickup or we can ship them to you. Which would you prefer?

I'll pick them up! **Ship them to me!**

Type something... ➤

7. Complete the **Customize your chat UI** form and click **Next**.

Explore the customization options. When creating an assistant for a client pilot, consider specifying attributes that align with the client (for example, colors and logos).

Welcome to AI assistant builder

Create **Personalize** **Customize** **Preview**

Customize your chat UI

Update the style to match your brand and your website. You can change these settings later. A developer can also add more advanced styling changes with code. [Learn more](#)

Assistant's name as known by customers
Zeeves

Intended purpose
 Standard: For virtual agents and customer support experiences.
 Carbon for AI: For use in internal IBM products.

Choose a theme

Primary color **#FFFFFF** Secondary color **#3D3D3D**

Chat header User message bubble

Accent color **#035AE9**

Significant and interactive objects

Size
 The size of the web chat on this page will not change by updating these fields.

Width **380px** Height **640px**

IBM Watermark
Enable IBM Watermark

Streaming
Enable Streaming
 Off

Restart conversation ➤

Zeeves

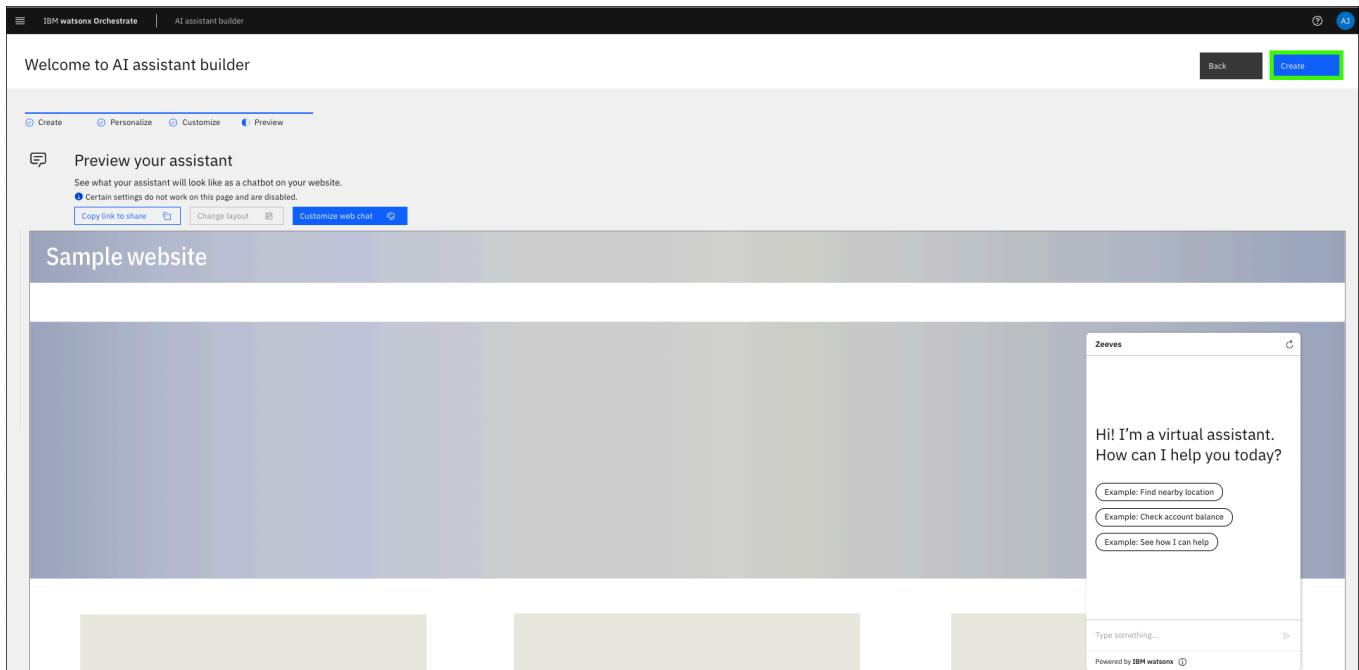
Hi! I'm a virtual assistant.
How can I help you today?

Example: Find nearby location
Example: Check account balance
Example: See how I can help

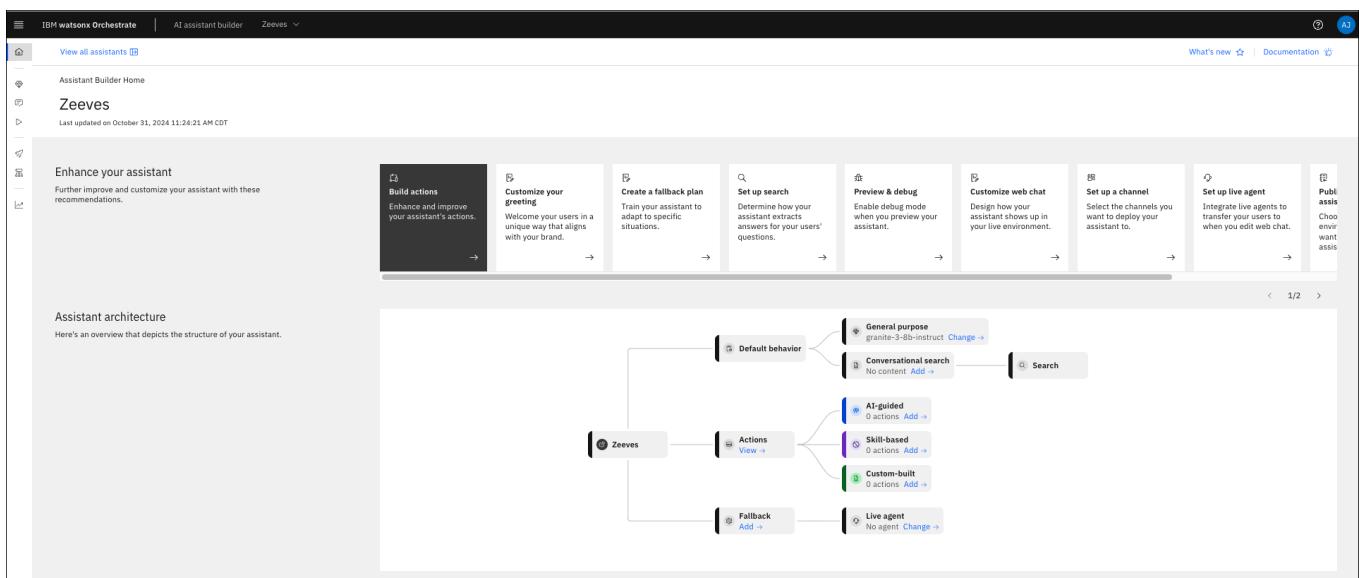
Type something... ➤

Powered by IBM watsonx

8. Preview your assistant and then click **Create**.



The assistant is now created.



Configure conversational search

In the next steps you will be to configure **conversational search** for your assistant that uses a hosted instance of OpenSearch.

1. Click **Generative AI** menu item (💡) in the left navigation.

The screenshot shows the AI assistant builder interface with the 'Generative AI' tab selected. A flowchart titled 'Structure' depicts the assistant's architecture. It starts with 'Default behavior' branching into 'General purpose' (granite-3-8b-instruct) and 'Conversational search' (No content). Below 'Default behavior' is 'Zeeves'. From 'Zeeves', an arrow points to 'Actions' (View →), which further branches into 'AI-guided' (0 actions) and 'Skill-based' (0 actions). To the right of the flowchart, there are several cards: 'Build actions' (Enhance and improve your assistant's actions), 'Personalize your greeting' (Welcome your users in a unique way that aligns with your brand), 'Create a fallback plan' (Train your assistant to adapt to specific situations), 'Set up search' (Determine how your assistant extracts answers for your users' questions), 'Preview & debug' (Enable debug mode when you preview your assistant), 'Customize web chat' (Design how your assistant shows up in your live environment), 'Set up a channel' (Select the channels you want to deploy your assistant to), 'Set up live agent' (Integrate live agents to transfer your users to when you edit web chat), and 'Choose envir want assis'.

2. Review the base large language model (LLM) settings.

Notice the other LLM models available. For most pilots, the **granite-3-8b-instruct** model is appropriate.

The screenshot shows the 'Base large language model (LLM)' configuration screen. It includes fields for 'Select a model' (with 'granite-3-8b-instruct' highlighted), 'Add prompt instructions' (with a note about default prompts), 'Answer behavior' (with sections for 'General-purpose answering' and 'Conversational search'), and a 'Preview' window showing a customer interaction. A green dashed box highlights the 'granite-3-8b-instruct' model selection.

3. Click Set up your Search Integration.

By default, conversational search is not enabled when an assistant is created. Conversational search takes priority over general-purpose answering if both are enabled. Learn more about conversational search in watsonx [here](#).

4. Click Custom service.

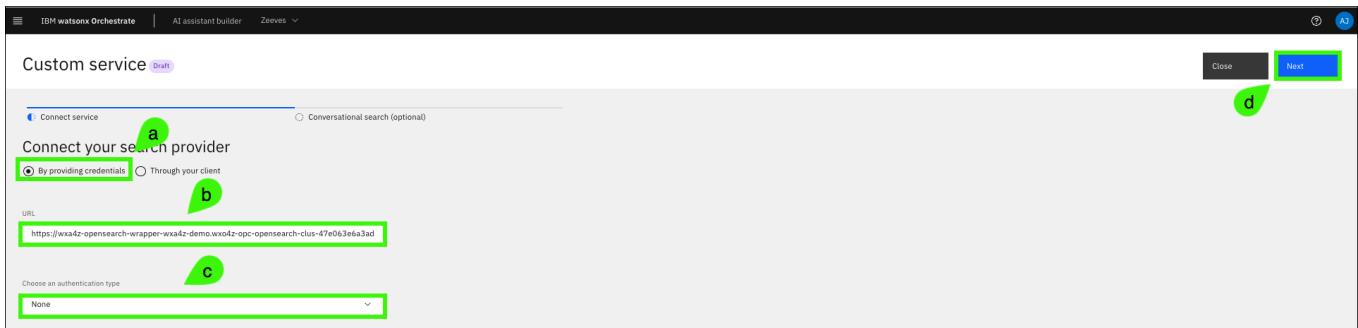
5. Complete the **Custom service (a-c)** form and then click **Next (d)**.

a. Select **By providing credentials**.

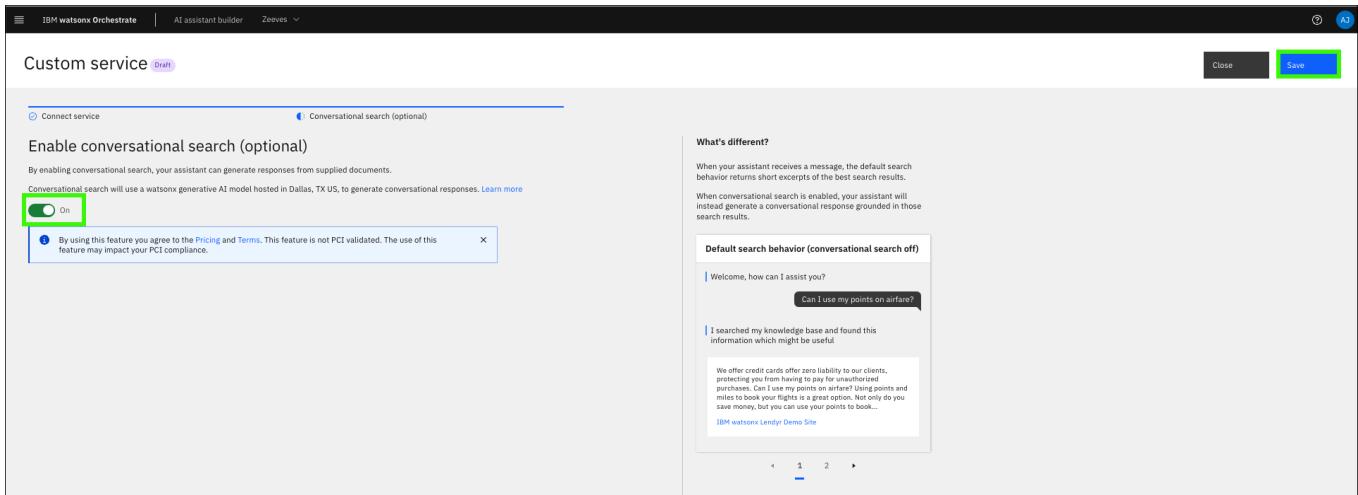
b. Enter the following value in the **URL** field (use the copy icon to avoid typographical errors). This is the **URL** for the a shared **OpenSearch** instance. In later sections you will created and customize a dedicated instance.

```
https://wxa4z-opensearch-wrapper-wxa4z-demo.wxo4z-opc-opensearch-clus-47e063e6a3ad1f71bf2e58f91c3b4c2e-0000.us-south.containers.appdomain.cloud/v1/query
```

c. Select **None** in the **Choose an authentication type** drop-down list.



6. Enable conversational search and then click Save.



7. Update the conversational search **custom service** settings based on your requirements.

Note: The **Settings** page is divided into two sections in the following images to enhance the visibility of the screen captures.

Learn more about these **custom service** settings [here](#).

The following settings are proven to work well. You can experiment with these settings to see how they affect queries for your client's pilot.

a. Enable **Conversational search**.

b. Select **Single turn**. Enabling **multi-turn conversation** (by selecting Entire conversation) is not yet supported for the solution on-premises. Be mindful in using this option and help ensure that the client understands what is supported in the solution.

c. Specify the text that appears to instruct the user to expand the list of citations in the assistant (except web chat client).

d. Select **Lowest** for the **retrieval confidence threshold** setting. This setting checks the confidence of the retrieved citations before a response is generated.

e. Select **Verbose** for the **generated response length**. This setting affects the average response length. Depending on user input, variations from the selected length can occur.

f. Select **Lowest** for the **response confidence threshold**. This setting checks the confidence of the generated citations after the response is generated.

Conversational search

Use a watsonx generative AI model hosted in Dallas, TX US, to generate conversational responses. [Learn more](#)

a **On**

b **Single turn**
The assistant uses only the current user input for retrieving search results and generating answers. This works well for clear, complete inputs but generally won't work with context-dependent questions such as, "Why is that?" after a previous answer.

c **Entire conversation**
The assistant uses the entire session history for retrieving search results and generating answers. This handles context-dependent questions well but may over-rely on past topics, even if the user has moved on.

Define the text for the citations title:

Search configuration

To test this, go to [evaluations page](#)

d **Lowest** Low High Highest

Retrieval confidence threshold:

Generated response length:
e **Verbose** **f** **Concise** **Moderate**

Response confidence threshold:
g **Lowest** Low High Highest

g. Keep the default setting of **All** for the listing of citations.

h. Keep the **Default filter** field empty.

i. The **Metadata** field provides a way to adjust your assistant's behavior during conversational search for your OpenSearch instance. This option is explored in detail in the [Installing and using zassist to ingest client documents](#). Leave the field empty for now.

j. The **Search display text** options specify the default text displayed when no results are found or when connectivity issues to the backend search service occur. You can keep the defaults or customize the service.

Response confidence threshold:

Citations
Citations will be displayed to the end user. This doesn't impact the number of citations used in the response.
g **All**

Default filter
Can be anything (plain text, JSON, etc)

Metadata
JSON example:
{
 "example_field": "example_value",
 "other_example_field": ?
}

Search display text
Define the text your search will display to the end user

No results found **j** I searched my knowledge base, but did not find anything related to your query
Connectivity issue

8. Click **Save** (a) and then click **Close** (b).

Custom service (draft)

Settings **Instance**

Conversational search

Use a watsonx generative AI model hosted in Dallas, TX US, to generate conversational responses.

[Learn more](#)

Conversational search **On**

Contextual awareness **Single turn**

The assistant uses only the current user input for retrieving search results and generating answers. This works well for clear, complete inputs but generally won't work with context-dependent queries such as, "Why is that?" after a previous answer.

Entire conversation Beta

The assistant uses the entire session history for retrieving search results and generating answers. This handles context-dependent questions well but may over-rely on past topics, even if the user has moved on.

Define the text for the citations title. (i)

How do we know?

Search configuration Restore default (i)

To test this, go to the [evaluations page](#)

Complete the configuration

After you save and close the **Conversational search** configuration page, a few more configurations are needed to get the best experience from your conversational chat. Details on these settings are available [here](#).

1. Hover over the **Generative AI** icon () in the left navigation and click **Actions**.

Home **Build** **Generative AI** **Actions**

Language model (LLM) Beta

model that your assistant uses for all base LLM functions.

Instructions

ded to produce the best general responses. Optionally, you may add instructions to refine the These instructions complement the default prompt, but are not integrated as part of it.

Preview

2:08 PM Greet customer [default]

Welcome, how can I assist you?

2. Click **Set by assistant** under the **All items** menu.

Actions

All items

Created by you

Set by assistant

Variables

Created by you

Set by assistant

Set by integration

Saved responses

Create your first action

With actions, you can help your customers accomplish their goals.

Create action +

3. Click **No matches**.

Name	Last edited	Examples Count	Status
Greet customer	3 hours ago	1	Green
Trigger word detected	3 hours ago	1	Green
No matches	3 hours ago	1	Green
Fallback	3 hours ago	1	Green

4. Click Step 1 under Conversation steps.

5. Select without conditions (a) in the Is taken drop-down menu and then click Clear conditions (b).

Note: the **Is taken** value does not change from **with conditions** after selecting **without conditions**.

6. Delete the default text in the Assistant says entry field.

7. Expand the And then drop-down menu and select Search for the answer.

The screenshot shows the AI assistant builder interface with the 'Editor' tab selected. In the 'Conversation steps' section, step 1 is highlighted with a green dashed border. The 'Assistant says' section contains a text input field with placeholder text: 'For example: Please select from the following options:'. Below it is a dropdown menu labeled 'Define customer response'. The 'And then' section is expanded, showing a list of actions. The 'End the action' option is selected and highlighted with a green box. A sub-menu for 'Search for the answer' is open, listing 'Continue to next step', 'Re-ask previous step(s)', 'Go to a subaction', 'Use an extension', 'Search for the answer' (which is also highlighted with a green box), 'Connect to agent', and 'End the action'.

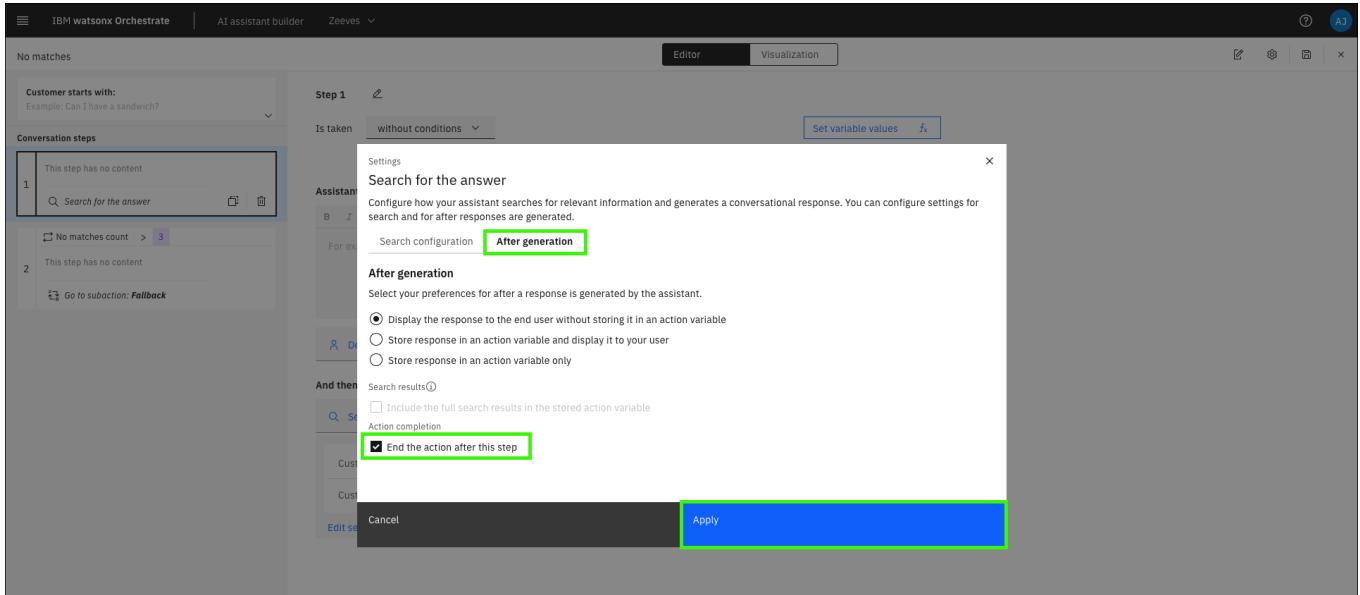
8. Click Edit settings.

This screenshot is similar to the previous one but shows the 'Edit settings' button in the 'And then' section's sub-menu for 'Search for the answer' highlighted with a green box.

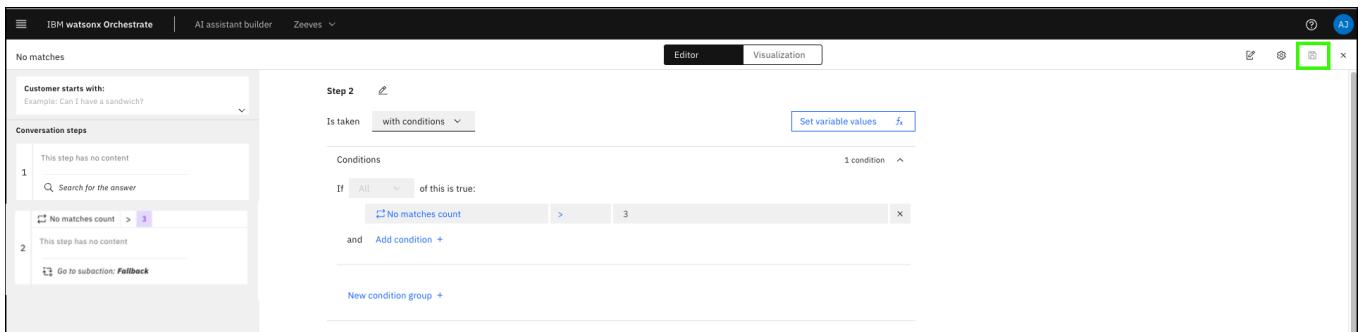
9. Click After generation.

A modal dialog box titled 'Search configuration' is displayed over the main interface. It has tabs for 'Search configuration' (which is selected) and 'After generation'. The 'Search configuration' tab contains fields for 'Custom search query (Optional)' and 'Custom results filter' (with 'Use default' selected). The 'After generation' tab contains a note about configuring settings for search and responses. At the bottom of the dialog are 'Cancel' and 'Apply' buttons, with 'Apply' being the active button.

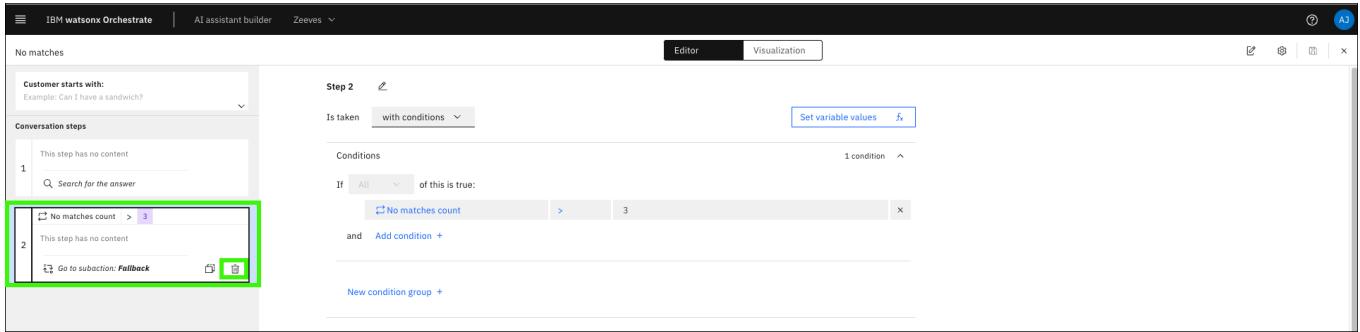
10. Select End the action after this step and then click Apply.



11. Click Save (💾).



12. Select Step 2 (No matches count) under Conversation steps and click delete (🗑).



13. Click Delete in the confirmation dialog to delete Conversation step 2.

The screenshot shows the IBM Watsonx Orchestrator interface. In the center, there's a configuration panel for 'Step 2'. It includes a condition section: 'If All of this is true:' followed by a condition 'No matches count > 3'. Below this, there's a 'New condition group' button and an 'Add condition +' button. At the bottom right of the configuration area, a confirmation dialog box is displayed: 'Are you sure you want to delete step 2? You will not be able to recover this data once deleted.' It has 'Cancel' and 'Delete' buttons, with 'Delete' being highlighted.

14. Click **Close** (the x icon) the **Editor** window.

This screenshot shows the same configuration panel for 'Step 1' as the previous one, but the confirmation dialog box from step 14 is still visible. The 'Delete' button in the dialog is highlighted with a green box.

15. Click **Fallback** in the **Actions** table.

This screenshot shows the 'Actions' table in the IBM Watsonx Orchestrator interface. The table has columns: Name, Last edited, Examples Count, and Status. There are five rows: 'Greet customer' (last edited 3 hours ago, 1 example, green status), 'Trigger word detected' (last edited 3 hours ago, 1 example, green status), 'No matches' (last edited a minute ago, 1 example, green status), and 'Fallback' (last edited 3 hours ago, 1 example, green status). The 'Fallback' row is highlighted with a green box.

Name	Last edited	Examples Count	Status
Greet customer	3 hours ago	1	Green
Trigger word detected	3 hours ago	1	Green
No matches	a minute ago	1	Green
Fallback	3 hours ago	1	Green

16. Delete all of the **Conversation steps**.

Note: The following image is edited. Only five steps are shown, but all six need to be deleted. You need to select each step individually. Click **delete** (trash icon) and confirm the deletion.

The screenshot shows the IBM Watsonx Assistant Editor window. On the left, under 'Customer starts with:', there is a single step: 'Call agent'. Below it, under 'Conversation steps', there are five steps, each with a green border around them. Step 1: 'Fallback reason is Failed to confirm topic.' Step 2: 'Fallback reason is Step validation failed.' Step 3: 'Fallback reason is Agent requested.' Step 4: 'Fallback reason is No matches.' Step 5: 'Fallback reason is Danger word detected.' To the right, the 'Step 1' configuration is shown, including conditions and responses. The 'Editor' tab is selected at the top.

17. Verify that all **Conversation steps** are deleted and then click the **x** to close the **Editor** window.

The screenshot shows the IBM Watsonx Assistant Editor window after the conversation steps have been deleted. The 'Conversation steps' section on the left is empty, indicated by a dashed green border. The configuration details on the right remain the same as in the previous screenshot.

18. Click the **Global settings** (⚙️).

The screenshot shows the 'Actions' section of the Global settings page. On the left, there is a sidebar with categories like 'All items', 'Created by you', 'Set by assistant', 'Variables', 'Created by you', 'Set by assistant', 'Set by integration', and 'Saved responses'. The 'Set by assistant' category is expanded, showing items: 'Greet customer', 'Trigger word detected', 'No matches', and 'Fallback'. On the right, there is a table with columns: Name, Last edited, Examples Count, and Status. Each item has a green status icon. The 'Status' column header has a green border.

19. Click **No matches** under the **Conversation routing** tab.

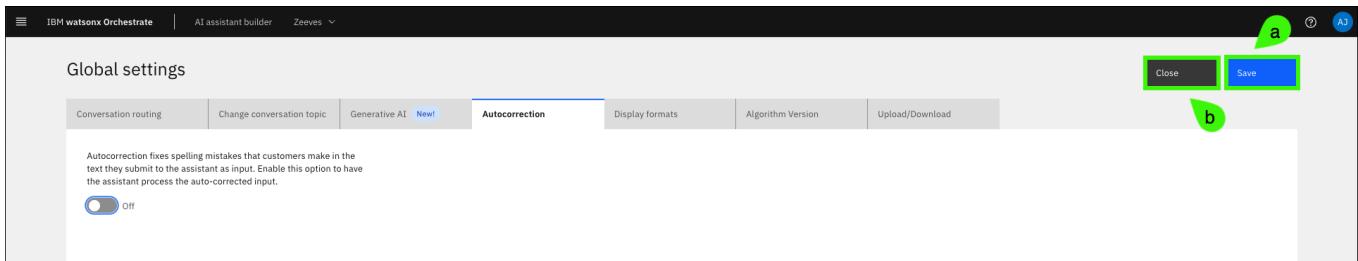
20. Move the slider to **More often** (or select **More often** in the drop-down).

The setting helps ensure that actions are triggered less often unless the user's query specifically matches the action's input.

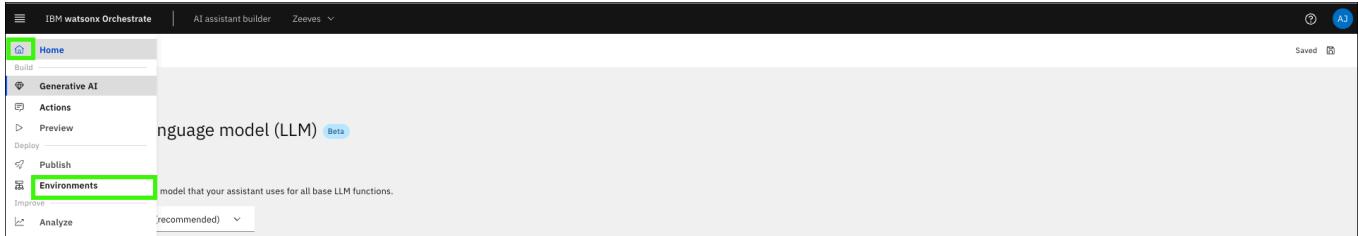
21. Click Autocorrection.

22. Click the autocorrection toggle to turn the feature **Off**.

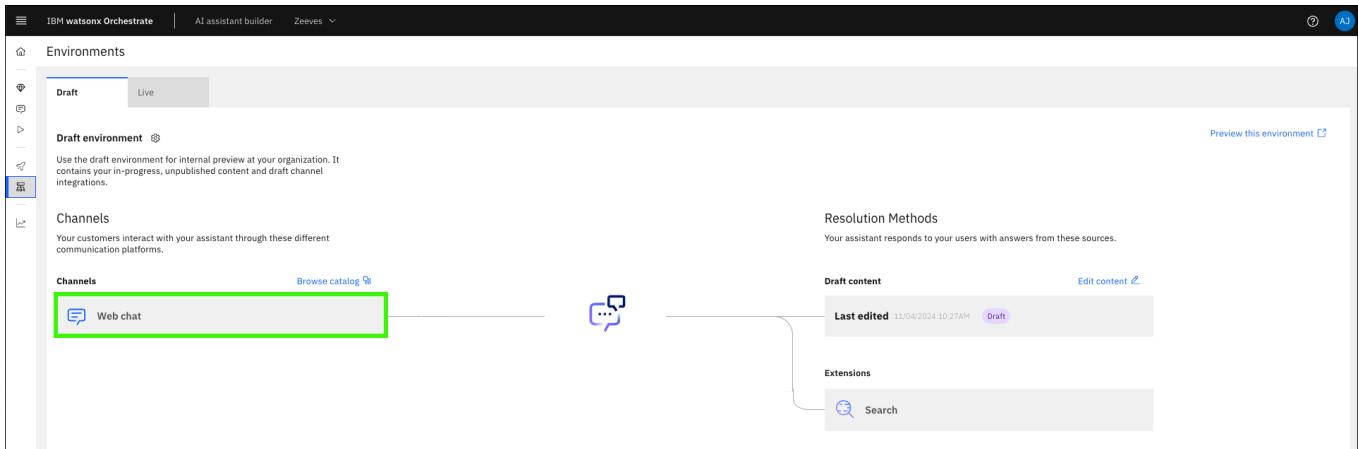
23. Click **Save** (a) and then **Close** (b).



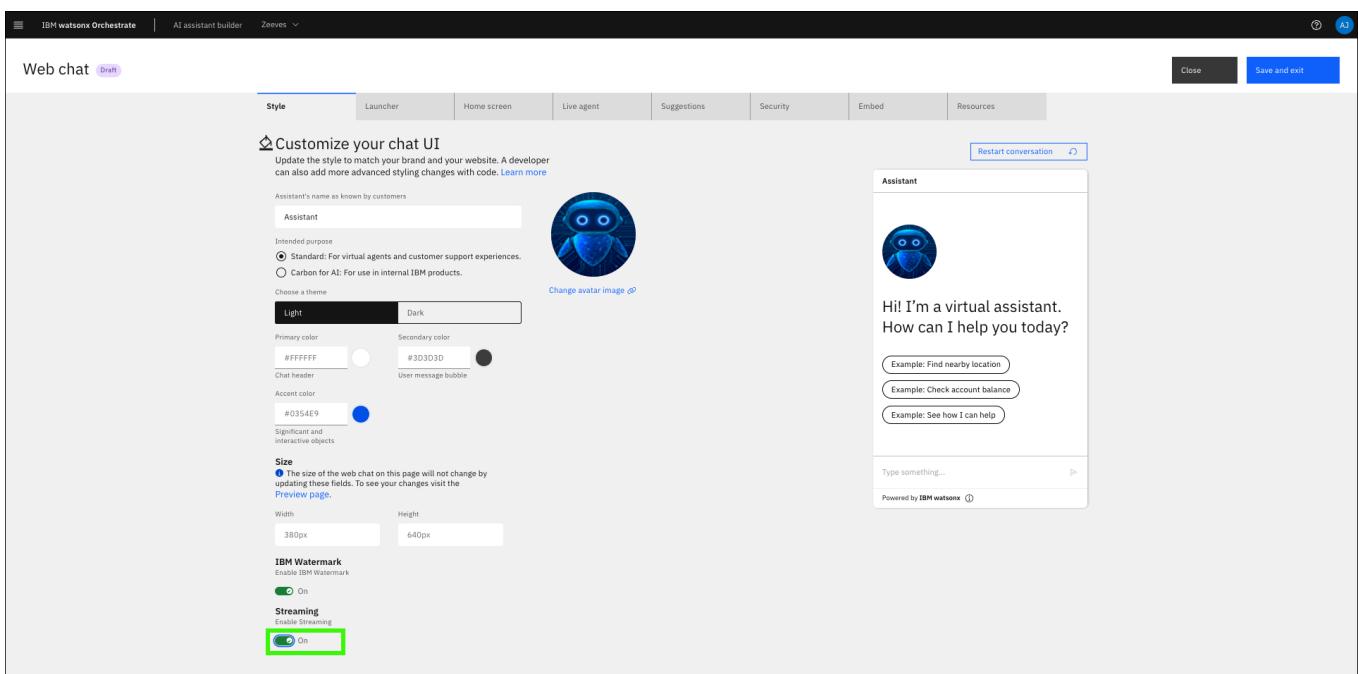
24. Hover over the Home (🏠) and click Environments.



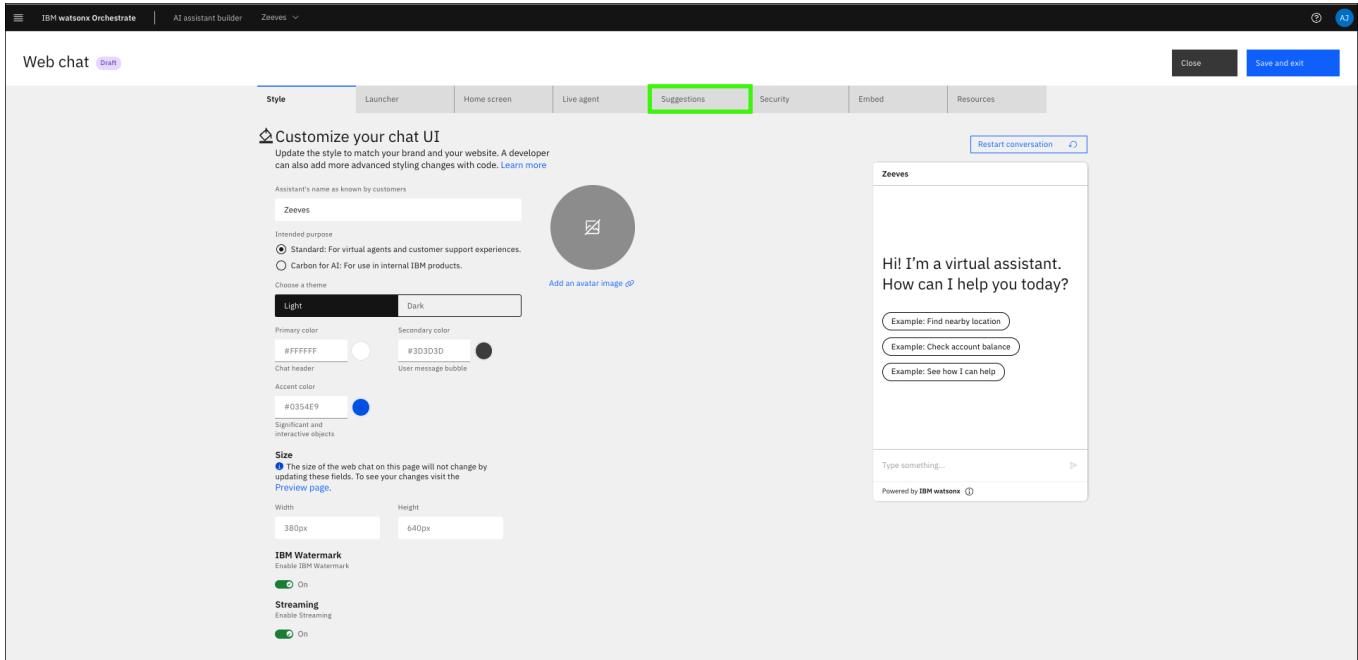
25. Click Web chat.



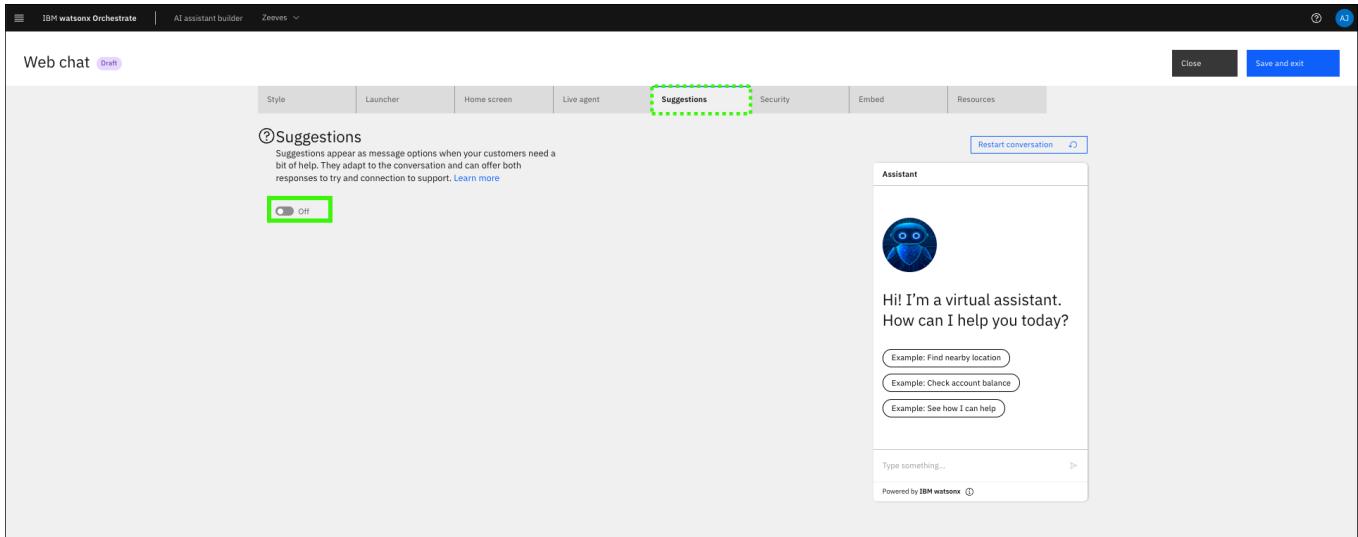
26. On the Style tab, click the Streaming toggle to enable streaming.



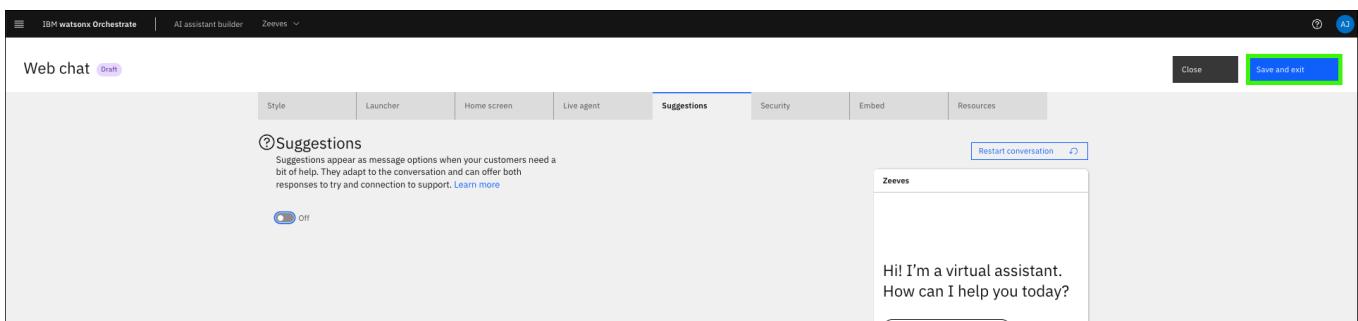
27. Click **Suggestions**.



28. Click the **Suggestions** toggle to turn this feature **Off**.



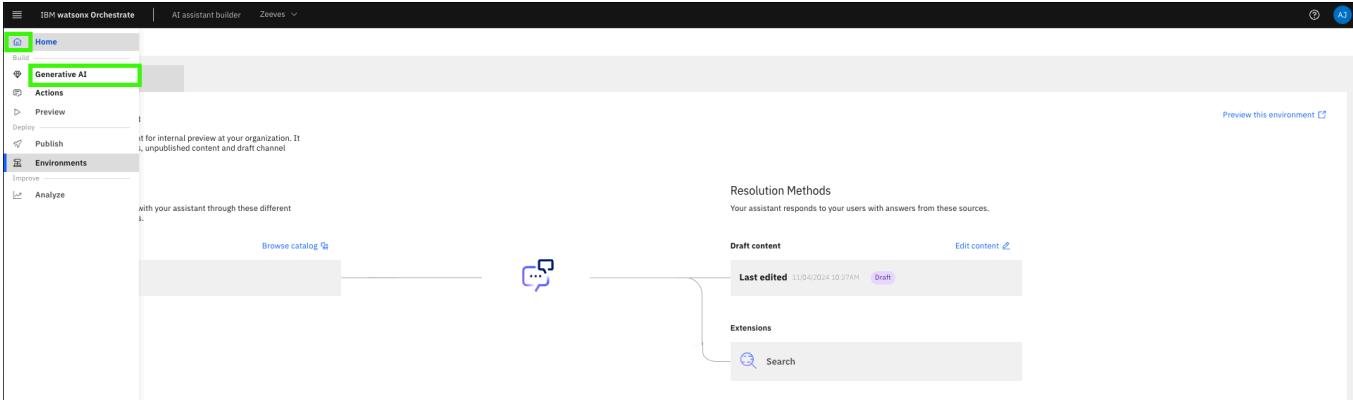
29. Click **Save and exit**.



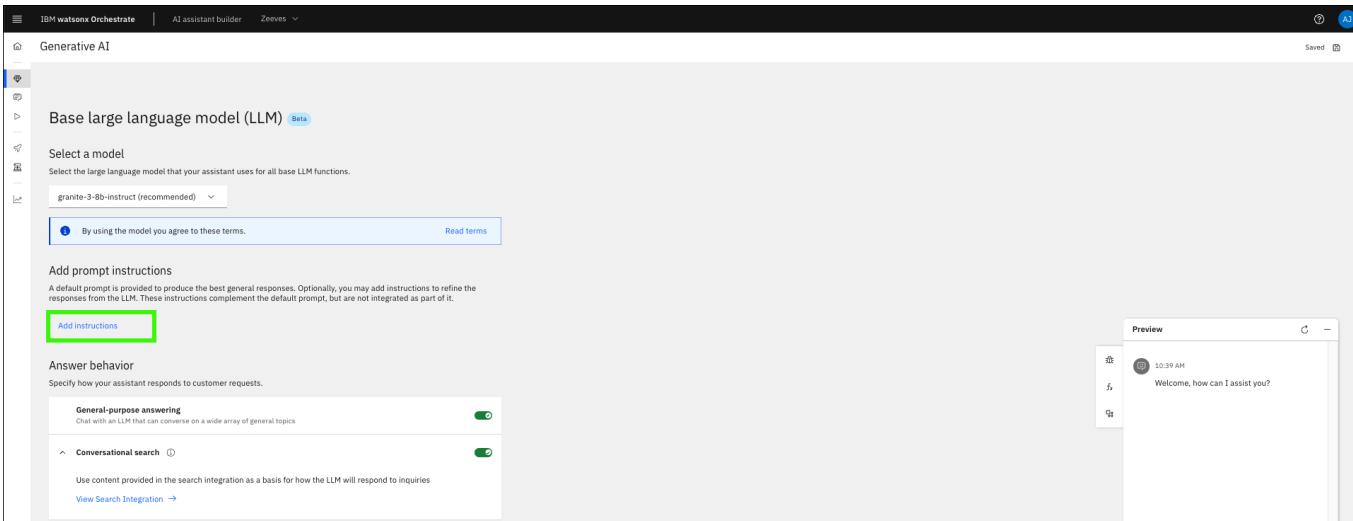
Configure the base large language model

After the preceding steps are completed, there are enhancements you can make to configure how the large language model (LLM) responds to your queries including adding prompt instructions and configuring the LLM's answer behavior. These options can be summarized [here](#).

1. Hover over the Home () and click Generative AI.



2. Click Add instructions.



3. Enter a prompt instruction.

This option instructs the LLM in your assistant to give refined responses by adding prompt instructions. The instructions help the LLM guide the conversations by clarifying how to achieve the end-goal of an action.

Enter prompt instructions in the field. The maximum number of characters you can enter in the prompt instruction field is 1,000.

The following is an example prompt instruction that works well. Experiment with different prompt instructions.

You are a subject matter expert on mainframe systems. Please respond to all prompts with truth and accuracy. Keep all answers short and concise, unless requested to provide details.

Note: When the instructions are typed in, they are automatically saved and the LLM is immediately trained on them.

The screenshot shows the 'Generative AI' configuration page. In the 'Answer behavior' section, the 'General-purpose answering' toggle switch is turned on (green). Below it, the 'Conversational search' toggle switch is also turned on (green). A preview window on the right shows a message from 'Greet customer [default]' at 10:47 AM: 'Welcome, how can I assist you?'. The 'Save' button is visible in the top right corner.

4. Toggle **General-purpose answering** to **Off** and then click **Save** (💾).

The ability exists to configure the answering behavior of your assistant to provide responses that are based on the preinstalled content or general content.

On the **Generative AI** page (under **Prompt Instructions**), you see the **Answer behavior** section. After you configure **Conversational search**, you see that it is enabled (toggled on) with the search integration added.

If you enable both general-purpose answering as well as conversational search, the conversational search answering takes precedence over General-purpose answering.

Recommendation: For purposes of retrieving Z-specific answers and responses, it is recommended that you turn off general-purpose answering and leave only conversational search turned on.

The screenshot shows the 'Generative AI' configuration page. In the 'Answer behavior' section, the 'General-purpose answering' toggle switch is turned off (gray). Below it, the 'Conversational search' toggle switch is turned on (green). A preview window on the right shows a message from 'Greet customer [default]' at 10:47 AM: 'Welcome, how can I assist you?'. The 'Save' button is visible in the top right corner.

Testing conversational search

Now you can begin issuing queries to test the assistant's responses.

Important: Many of the settings configured earlier can be iteratively modified based on your assessment of the quality of responses. The settings can be reviewed and changed at any time. For example, adding extra prompt instructions, changing the verbosity of the responses, and modifying the indexes used for OpenSearch.

1. Hover over the **Home** () and click **Preview**.

2. Experiment with different prompts and validate that the answers are reasonable and related to IBM Z.

Other prompts and responses follow.

Note: The responses that you receive can vary from the ones shown.

Prompt:

What is z/OS continuous delivery?

Example output:

Prompt:

What is the APF list in z/OS? Provide a detailed explanation?

Example output:

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with icons for preview, AI assistant builder, and Zeeves. The main area has a title "Sample website". On the right, there's a chat window titled "Zeeves" with a message from "You 11:02 AM" asking "What is the APF list in z/OS? Provide a detailed explanation?". A large green dashed box highlights the response message, which provides a detailed explanation of the APF list in z/OS.

Prompt:

Why is Db2 different than other database systems?

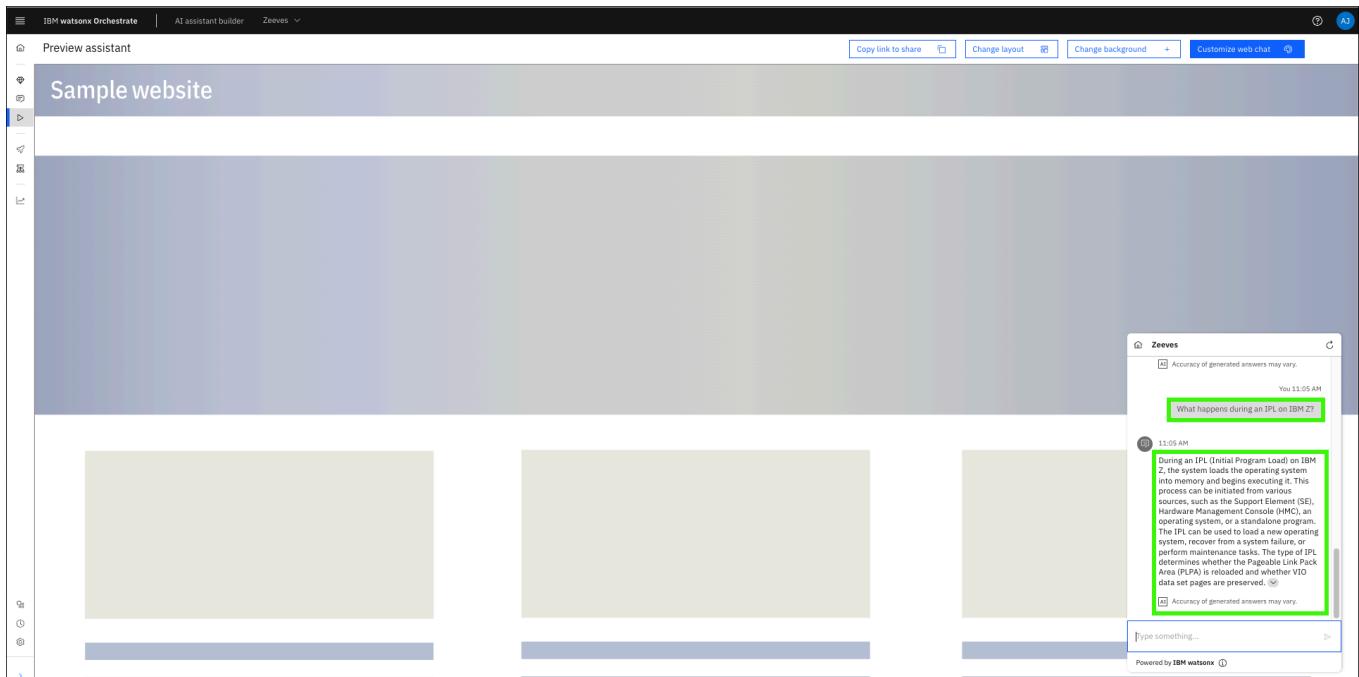
Example output:

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with icons for preview, AI assistant builder, and Zeeves. The main area has a title "Sample website". On the right, there's a chat window titled "Zeeves" with a message from "You 11:03 AM" asking "Why is Db2 different than other database systems?". A large green dashed box highlights the response message, which explains that Db2 is a database management system (DBMS) that runs under the z/OS operating system, making it specific to mainframe computers. It also notes that other database systems may run on various operating systems like Oracle, MySQL, and PostgreSQL. Additionally, Db2 provides a higher level of data management and security features compared to other database systems, as it manages data stored in databases and controls access to the data.

Prompt:

What happens during an IPL on IBM Z?

Example output:



You now have a working assistant that uses IBM watsonx Assistant for Z. Take time to explore different prompt instructions and settings. If you encounter any issues, the Troubleshooting section that follows can help resolve them.

Continue to the [Creating a stand-alone OpenSearch instance for document ingestion](#) to learn how to configure a dedicated OpenSearch instance for ingesting client-specific documentation into the RAG model.

Troubleshooting

The following are issues that you may encounter. If the provided resolutions do not work, contact support by using the methods that are mentioned in the [Support](#) section.

✖ Assistant responds to all prompts with, "I might have information related to your query to share, but am unable to connect to my knowledge base at the moment" ✖

This Assistant is unable to connect to the custom service URL specified. This could be a network issue, the service may be down, the service may be restarting, or the service is no longer running at that URL.

Before reaching out to [Support](#), try the following:

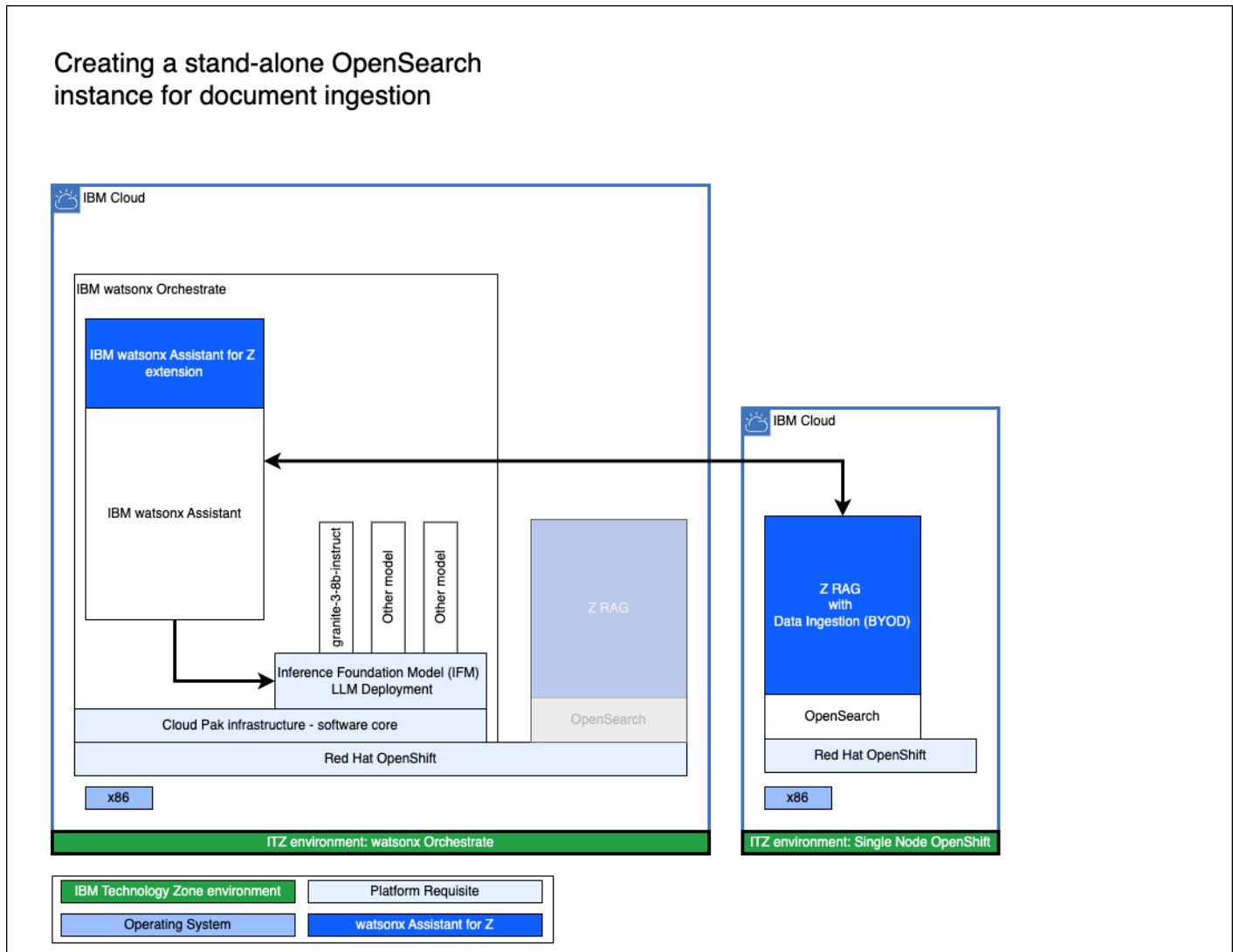
- Wait a few minutes and try again. It may be the service was in the process of restarting.
- If you printed this demonstration guide or saved a copy, verify you are using the most current version of the [lab guide](#) and the correct service URL (<https://wxa4z-opensearch-wrapper-wxa4z-demo.wxo4z-opc-opensearch-clus-47e063e6a3ad1f71bf2e58f91c3b4c2e-0000.us-south.containers.appdomain.cloud/v1/query>). The URL may have changed since you saved or printed the lab guide.

Ingesting client documents

Creating a stand-alone OpenSearch instance for document ingestion

Next, learn to enable a client to ingest their own documentation into the Retrieval Augmented Generation (RAG) used by IBM watsonx Assistant for Z. The first step is to deploy a dedicated [OpenSearch](#) instance. The dedicated search instance is referred to as bring-your-own-search (BYOS).

Below is a high-level, logical architecture of the environment you will deploy in this section.



Earlier, you provisioned three IBM Technology Zone (ITZ) environments. One of which was a single-node Red Hat OpenShift (SNO) cluster. If you have not reserved this environment, or it is not in the **Ready** state, return to the [IBM Technology Zone environment](#) section to complete the reservation.

Install the Red Hat OpenShift command-line interface utility

The Red Hat OpenShift command-line interface (CLI) utility, which is known as **oc**, must be installed on your local workstation. If you already installed the **oc** utility, you can proceed to [log in to the SNO cluster](#).

1. Click the following link to open a browser window to your ITZ reservations.

[ITZ My reservations](#)

2. Click the **Single Node OpenShift** tile.

My reservations

Status - Ready
Single Node OpenShift (VMware on IBM C...)

Education

Start date
Oct 31, 2024 7:02 AM

End date
Nov 6, 2024 6:49 AM

Extend limit
0

[Open this environment](#)

Status - Ready
watsonx Assistant for Z Pilot - AAP & z/OS

Education

Start date
Oct 30, 2024 8:12 AM

End date
Nov 5, 2024 7:53 AM

Extend limit
0

[Open this environment](#)

Status - Ready
watsonx Assistant for Z Pilot - watsonx O...

Education

Start date
Oct 30, 2024 8:10 AM

End date
Nov 5, 2024 7:12 AM

Extend limit
0

[Open this environment](#)

3. Scroll down and record the **Cluster Admin Username** and **Cluster Admin Password**.

Reservation Details

API URL
<https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443>

Bastion Password

Bastion RDP address
ap.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43389

Bastion SSH connection
ssh ituser@api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com -p 40222

Bastion Username
ituser

Cluster Admin Username
kubeadmin

Cluster Admin Password

OCP Console
<https://console.openshift-console.apps.672371d38376796fb96a6c4d.ocp.techzone.ibm.com>

OCP Version
4.14

vCenter
itzeu-vc.eu.cloud.techzone.ibm.com

[Download kubeconfig](#)

4. Click the **OCP Console** link.

Note: OCP stands for OpenShift Container Platform.

The screenshot shows the IBM Technology Zone portal. At the top, there are two cluster entries:

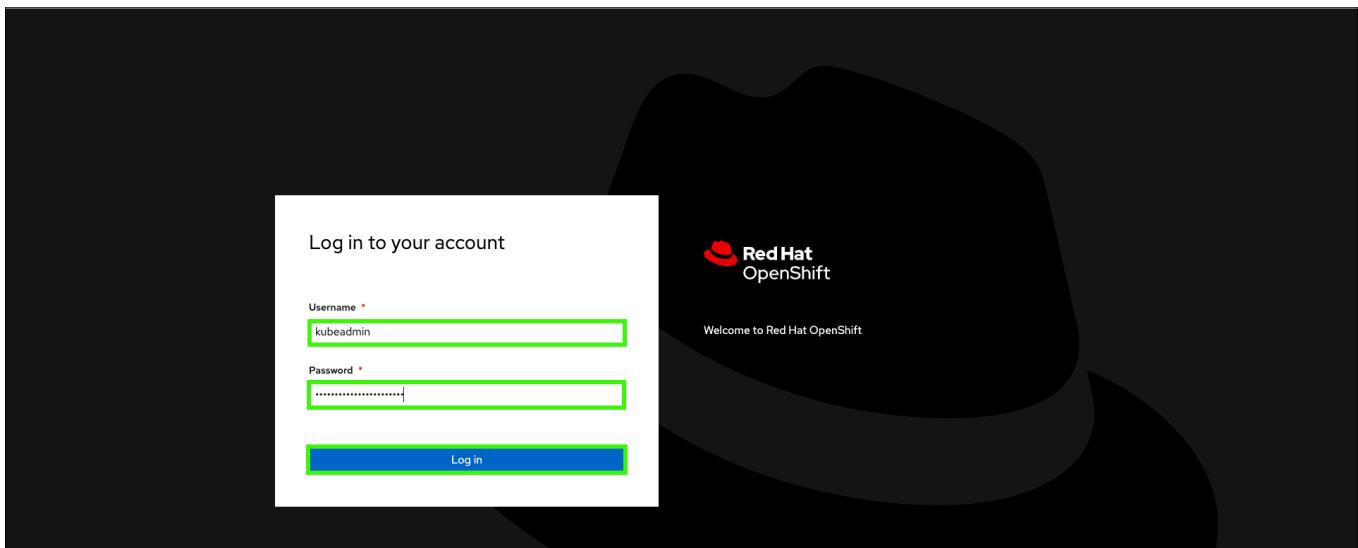
- 672371d38376796fb96a6c4d-master-1: Red Hat Enterprise Linux 8 (64-bit), IP 192.168.252.11, Running. A 'Console' button is shown.
- 672371d38376796fb96a6c4d-nfs: Red Hat Enterprise Linux 8 (64-bit), IP 192.168.252.17, Running. A 'Console' button is shown.

Below the clusters is a section titled 'Reservation Details' containing the following information:

- API URL: <https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443>
- Bastion Password: [REDACTED]
- Bastion RDP Address: api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43389
- Bastion SSH connection: ssh itzuser@api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com -p 40222
- Bastion Username: itzuser
- Cluster Admin Username: kubeadmin
- Cluster Admin Password: [REDACTED]
- OCP Console: <https://console-openshift-console.apps.672371d38376796fb96a6c4d.ocp.techzone.ibm.com> (highlighted with a green box)
- OCP Version: 4.14
- vCenter: itzeu-vc.eu.cloud.techzone.ibm.com

At the bottom, there is a 'Download kubeconfig' button.

5. Enter the **Cluster Admin Username** and **Cluster Admin Password** values from step 3 and click **Log in**.



6. Click Help (?) and then click Command Line Tools.

The screenshot shows the Red Hat OpenShift admin console. The left sidebar has sections like 'Administrator', 'Home', 'Operators', 'Workloads', 'Networking', and 'Storage'. The 'Overview' section is selected. On the right, there is a message about OAuth configuration. A 'Help' icon (a question mark) is visible in the top right corner. A context menu is open over the 'Help' icon, with 'Command Line Tools' highlighted by a green box. Other options in the menu include 'Documentation', 'Share Feedback', 'About', 'Learning Portal', 'OpenShift Blog', and 'API Explorer'.

7. Click the link under **oc - OpenShift Command Line Interface (CLI)** for the operating system of your local machine.

The screenshot shows the 'Command Line Tools' section of the Red Hat OpenShift interface. A green dashed box highlights the 'oc - OpenShift Command Line Interface (CLI)' section, which includes a brief description and a list of download links for different architectures.

Clicking the preceding link automatically downloads either a **.zip** or **.tar** file specific to your operating system. Unzip or untar the file. Place the **oc** binary for your operating system (**OS**) in a directory that is in your default PATH, or set the PATH environment variable to include the location of the **oc** binary.

8. Verify the installation by running the **oc** command on your local workstation.

```
oc --help
```

Sample output:

```
andrewjones@Andrews-MBP ~ % oc --help
OpenShift Client

This client helps you develop, build, deploy, and run your applications on any
OpenShift or Kubernetes cluster. It also includes the administrative
commands for managing a cluster under the 'adm' subcommand.

Basic Commands:
  login           Log in to a server
  new-project     Request a new project
  new-app          Create a new application
  status           Show an overview of the current project
  project          Switch to another project
  projects         Display existing projects
  explain          Get documentation for a resource

Build and Deploy Commands:
  rollout          Manage a Kubernetes deployment or OpenShift deployment
  config           Revert part of an application back to a previous deployment
```



Mac/OS users may need to adjust security settings.



The **oc** binary may cause a security exception. Adjust the security settings by opening the **System Settings** utility and clicking **Privacy & Security**. Under **Security** locate the message about the **oc** binary and click **Allow Anyway**. Return to the terminal window and try the **oc --help** command again and click **Allow Anyway** when prompted.

Prepare to ingest documents

Before ingesting documents, complete the following setup steps.

Log in to the OpenShift cluster from your local terminal

Note: If you just installed the **oc** utility, skip the next 5 steps.

1. Click the following link to open a browser window to your ITZ reservations.

[ITZ My reservations](#)

2. Click the **Single Node OpenShift** tile.

3. Scroll to the bottom of the reservation page and record the **Cluster Admin Username** and **Cluster Admin Password**.

4. Click the **OCP Console** link.

Reservation Details

API URL
<https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443>

Bastion Password [REDACTED]

Bastion RDP Address
api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43389

Bastion SSH connection
ssh itzuser@api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com -p 40222

Bastion Username
itzuser

Cluster Admin Username
kubeadmin

Cluster Admin Password [REDACTED]

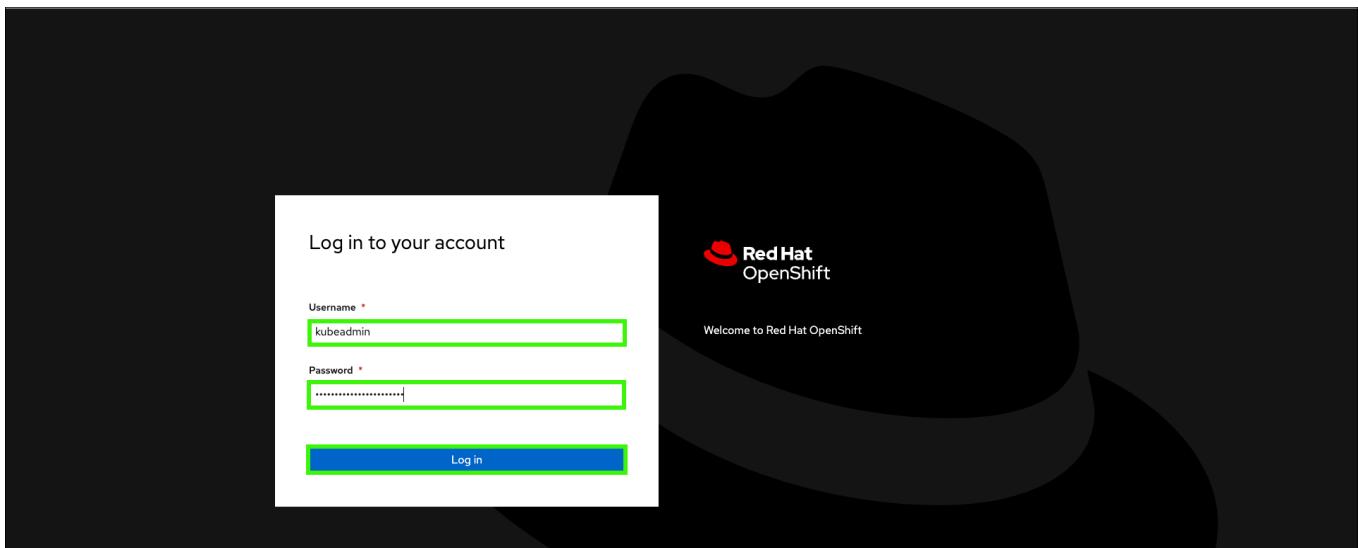
OCP Console
<https://console-openshift-console.apps.672371d38376796fb96a6c4d.ocp.techzone.ibm.com>

OCP Version
4.14

vCenter
itzeu-vc.eu.cloud.techzone.ibm.com

Download kubeconfig

- Enter the **Cluster Admin Username** and **Cluster Admin Password** values from step 3 and click **Log in**.



- Click the **kube:admin** profile drop-down and click **Copy login command**.

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Command Line Tools

[Copy login command](#)

oc - OpenShift Command Line Interface (CLI)

With the OpenShift command line interface, you can create applications and manage OpenShift projects from a terminal.

The oc binary offers the same capabilities as the kubectl binary, but it is further extended to natively support OpenShift Container Platform features.

- [Download oc for Linux for x86_64](#)

- Click **Display Token**.

- Select and copy the **Log in with this token** string.

For most operating systems, double-click the value, then right-click and select **Copy**.

Your API token is
sha256~zuWR0KDnkYniIY0m8g8iKoUXPdFFFmou~o4s5FsrDNA

Log in with this token

```
oc login --token=sha256~zuWR0KDnkYniIY0m8g8iKoUXPdFFFmou~o4s5FsrDNA --server=https://api.672b79320c7a71b728e523b4.ocp.techzone.ibm.com:6443
```

Use this token directly against the API

```
curl -H "Authorization: Bearer sha256~zuWR0KDnkYniIY0m8g8iKoUXPdFFFmou~o4s5FsrDNA" "https://api.672b79320c7a71b728e523b4.ocp.techzone.ibm.com:6443/v1/users/~"
```

Request another token

[Logout](#)

A context menu is open over the token value, with 'Copy' highlighted.

9. Open a command prompt or terminal window on your local workstation.

10. Paste the login command and press **enter**.

```
andrewjones@Andrews-MBP ~ % oc login --token=sha256~mJ4L8K6cUMyNyk2Z69KMm3vbP1sWc8SW0eeOdVqtA94 --server=https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443
Logged into "https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443" as "kube:admin" using the token provided.
You have access to 70 projects, the list has been suppressed. You can list all projects with 'oc projects'
Using project "default".
andrewjones@Andrews-MBP ~ %
```

Create a working directory

1. Create a directory to store the configuration files that you will create in the next steps.



Instructions vary by your local workstation's operating system.

The directions that follow may vary depending on your operating system. The examples provided are based upon MacOS.

```
mkdir watsonxAssistant
```

2. Change to the new directory.

```
cd watsonxAssistant
```

```
andrewjones@Andrews-MBP ~ % oc login --token=sha256~mJ4L8K6cUMyNyk2Z69KMm3vbP1sWc8SW0eeOdVqtA94 --server=https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443
Logged into "https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443" as "kube:admin" using the token provided.
You have access to 70 projects, the list has been suppressed. You can list all projects with 'oc projects'
Using project "default".
andrewjones@Andrews-MBP ~ % mkdir watsonxAssistant
andrewjones@Andrews-MBP ~ % cd watsonxAssistant
andrewjones@Andrews-MBP watsonxAssistant %
```

Install IBM Certificate Manager on Red Hat OpenShift

1. In a text editor, create a file named `catalogCertManager.yaml` and paste the following text in the file.



Formatting of the yaml file is critical!

The content of the YAML file must be formatted exactly as shown. Use the **Copy** icon to prevent typographical errors.

File name:

```
catalogCertManager.yaml
```

File contents:

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: ibm-cert-manager-catalog
  namespace: openshift-marketplace
spec:
  displayName: ibm-cert-manager-4.2.7
  grpcPodConfig:
    securityContextConfig: restricted
    image: icr.io/cpopen/ibm-cert-manager-operator-
catalog@sha256:4dcf4ace4b5f166f83b31063f7e6404dbf78d8e98a9d4fcf52fedf576a55ca6c
  publisher: IBM
  sourceType: grpc
  updateStrategy:
    registryPoll:
      interval: 30m0s
```

2. Install the IBM Certificate Manager operator in the Red Hat OpenShift cluster.

```
oc apply -f catalogCertManager.yaml
```

The preceding command returns a message that states the **ibm-cert-manager-catalog** was created.

3. In the OpenShift web console, click **Operators** and then select **OperatorHub**.

The screenshot shows the Red Hat OpenShift web console interface. The left sidebar has a navigation menu with items like 'Administrator', 'Home', 'Operators' (which is highlighted with a green border), 'OperatorHub' (which is also highlighted with a green border), and 'Installed Operators'. The main content area has a blue header bar with the text 'You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.' Below the header, there's a section titled 'Command Line Tools' with a 'Copy login command' button. Another section titled 'oc - OpenShift Command Line Interface (CLI)' provides instructions on how to use the 'oc' binary and links to download it for various platforms.

4. Click the **Project** to pull-down menu and click the **Show default projects** toggle.

The screenshot shows the Red Hat OpenShift interface on the OperatorHub page. The left sidebar has 'OperatorHub' selected. The main area shows a dropdown menu labeled 'Project: All Projects' with a green border around it. Below it is a search bar with 'Select project...' placeholder text. A toggle switch labeled 'Show default projects' is also visible. The central part of the screen lists 'Projects' under 'All Projects', including 'default', 'kube-node-lease', and 'kube-public'. At the bottom right, there are tabs for 'Community', 'Marketplace', and 'Community' again.

5. Scroll down and select **openshift-marketplace**.

This screenshot shows the same Red Hat OpenShift interface as the previous one, but the list of operators has been scrolled down. The 'openshift-marketplace' operator is now highlighted with a green border. It is listed under the 'Networking' category. The interface includes a 'Create Project' button at the bottom left and a 'Community' tab at the bottom right.

6. Enter IBM Cert Manager in the search field and then click the IBM Cert Manager tile.

Be patient.
It may take a minute or two for the IBM Cert Manager tile to appear.

Note: The current version of the operator may differ than the one shown in the image below.

This screenshot shows the Red Hat OpenShift interface with the search bar containing 'IBM Cert Manager' highlighted with a green border. The search results show one item: 'ibm-cert-manager-4.2.7' provided by IBM. This item is described as an 'Operator for managing deployment of cert-manager service.'

7. Click **Install**.

The screenshot shows the Red Hat OpenShift on IBM Techzone interface. The left sidebar is the OperatorHub. In the center, under the 'Operators' section, the 'IBM Cert Manager' operator is listed. On the right, a detailed view of the operator is shown with the 'Install' button highlighted in green.

8. Keep the default settings and click **Install**.

The screenshot shows the 'Install Operator' configuration page. It includes fields for 'Update channel' (set to v4.2), 'Version' (set to 4.2.7), and 'Installation mode' (set to 'All namespaces on the cluster (default)'). Below these, there are sections for 'Namespace creation' (indicating the namespace 'ibm-cert-manager' does not exist and will be created) and 'Update approval' (set to 'Automatic'). At the bottom, the 'Install' button is highlighted in green.

**Do not continue until...**

The installation process takes a few minutes. Do not continue until you see the following message: **Installed operator: ready for use.**

Install the watsonx Assistant for Z Operator (for OpenSearch)

1. In your command prompt or terminal window, create a new namespace called `wxa4z-byos` in the Red Hat OpenShift cluster.

```
oc create namespace wxa4z-byos
```

2. Create or obtain your IBM Container Software production entitlement key.

A production entitlement key is required to pull the container images that get deployed by the operator.

To create or retrieve your existing entitlement key, follow the instructions [here](#).

If additional assistance is needed, refer to this [site](#).

After locating your existing key or creating a new key, continue to the next step.

3. Click **copy** and record your entitlement key for future use in a secure location.

The screenshot shows the IBM Container Software and Cloud Pak Access Management interface. On the left, there's a sidebar with 'My IBM', 'Profile', 'Billing', and 'Entitlement keys' selected. Below that is a 'Container software library'. The main area is titled 'Entitlement keys (1)'. It says 'Access your container software' and 'Your entitlement key allows you to access all your container software in the IBM Entitled Registry. For a full list of container software you own, view your [container software library](#)'. Under 'Active entitlement keys', it says 'Use any active entitlement key to log in to the image registry and retrieve any container software you own.' and lists: '– You can have a maximum of (5) entitlement keys.', '– Once a key is deleted, it's no longer valid.' A button 'Add new key' with a '+' icon is at the top right. Below this is a box containing 'Issue date: October 17, 2022' and a long entitlement key string. At the bottom right of the box are 'Copy' (green border), 'Delete' (red border), and 'Edit' buttons.

4. In your command prompt or terminal window, set an environment variable with your production entitlement key.

Substitute your production entitlement key copied in the last step for <entitlement key> .

```
export IBM_CS_ENT_KEY=<entitlement key>
```

5. Enter the following command to create a pull secret for the **Container Registry**.

```
oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --  
docker-username=cp --docker-password=$IBM_CS_ENT_KEY
```

The terminal session shows the creation of a pull secret for the Container Registry. The user runs 'oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --docker-username=cp --docker-password=\$IBM_CS_ENT_KEY'. The output shows the secret being created successfully.

6. In a text editor, create a file named `catalogSource.yaml` and paste the following text in the file.



Formatting of the yaml file is critical!

The content of YAML files must be formatted exactly as shown. Use the copy icon to prevent typographical errors.

File name:

```
catalogSource.yaml
```

File contents:

```

apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: ibm-wxa4z-operator-catalog
  namespace: wxa4z-byos
spec:
  displayName: "IBM watsonx Assistant for Z Operator Catalog"
  image: icr.io/cpopen/ibm-wxa4z-
  catalog: 2.0.2@sha256:d457aa24af07c23d7b36727a9cff95995ca1c4ed49fb980ef4f386fd09a29b4
  publisher: 'IBM'
  sourceType: grpc
  secrets:
    - icr-pull-secret

```

7. Create your document catalog in the Red Hat OpenShift operator.

```
oc apply -f catalogSource.yaml
```

```

andrewjones@Andrews-MBP watsonxAssistant % oc create namespace wxa4z-byos
namespace/wxa4z-byos created
andrewjones@Andrews-MBP watsonxAssistant % export IBM_CS_ENT_KEY=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9eyJpc3MiOiJJQk0gTWFya2V0cGxhY2UiLCJpXXQiOjE2NjYwMTk1ODAsImp0aSI6IjNKOWUyMzzjZTAzMDQzMzVhNTJhYTkzMWNmOTcyMDR1In0.4M3XRD4XzkHMS0kFNJ4uKvcWZ6SnEA0Z03eL_11A2XY
andrewjones@Andrews-MBP watsonxAssistant % oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --docker-username=cp --docker-password=$IBM_CS_ENT_KEY
secret/icr-pull-secret created
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml
andrewjones@Andrews-MBP watsonxAssistant % oc apply -f catalogSource.yaml
catalogsource.operator.coreos.com/ibm-wxa4z-operator-catalog created:
andrewjones@Andrews-MBP watsonxAssistant %

```

8. In the Red Hat OpenShift web console, click **OperatorHub** and select the **wxa4z-byos** project.

The screenshot shows the Red Hat OpenShift web console with the 'OperatorHub' section selected. The search bar at the top has 'Project: openshift-marketplace' selected. In the main list, the 'wxa4z-byos' project is highlighted with a green border. Other operators listed include 'openshift-node', 'openshift-nutanix-infra', 'openshift-oauth-apiserver', 'openshift-openstack-infra', 'openshift-operator-lifecycle-manager', 'openshift-operators', 'openshift-ovirt-infra', 'openshift-ovn-kubernetes', 'openshift-route-controller-manager', 'openshift-service-ca', 'openshift-service-ca-operator', 'openshift-user-workload-monitoring', 'openshift-vsphere-infra', and 'techzone-infra'. To the right, there are four cards representing different operator categories: 'Community' (CrowdStrike Operator), 'Marketplace' (CrowdStrike Operator, Hazelcast Platform Operator), 'Community' (Abot Operator-v3.0.0, Accuknox Operator), and 'Red Hat' (Advanced Cluster Management for Kubernetes).

9. Enter **ibm watsonx** in the search field and the click the **IBM watsonx Assistant for Z Operator Catalog** tile.



It may take a minute or two for the **IBM watsonx Assistant for Z Operator Catalog** tile to appear.

Note: The current version of the operator may differ than the one shown in the image below.

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: wxa4z-byos

OperatorHub

Discover Operators from the Kubernetes community and Red Hat partners, curated by Red Hat. You can purchase commercial software through Red Hat Marketplace. You can install Operators on your clusters to provide optional add-ons and shared services to your developers. After installation, the Operator capabilities will appear in the Developer Catalog providing a self-service experience.

All Items All Items

Search: ibm watsonx

1 items

IBM watsonx Assistant for Z Operator Catalog
IBM watsonx Assistant for Z provided by IBM
IBM watsonx Assistant for Z Operator

10. Click Install.

You are logged in as a temporary administrator.

Project: wxa4z-byos

OperatorHub

Discover Operators from the Kubernetes community and Red Hat partners, curated by Red Hat. You can purchase commercial software through Red Hat Marketplace. You can install Operators on your clusters to provide optional add-ons and shared services to your developers. After installation, the Operator capabilities will appear in the Developer Catalog providing a self-service experience.

All Items All Items

Search: ibm watsonx

IBM watsonx Assistant for Z
2.0.1 provided by IBM

Install

Channel: stable

Version: 2.0.1

Capability level:

- Basic Install
- Seamless Upgrades
- Full Lifecycle
- Deep Insights
- Auto Pilot

Source: IBM watsonx Assistant for Z Operator Catalog

Provider: IBM

Repository: N/A

Container Image: N/A

Created at: Oct 4, 2024, 1:31 PM

11. Select A specific namespace on the cluster (a) under Installation mode and wxa4z-byos (b) for the Installed Namespace, then click Install (c).



Do not continue until...

The installation process takes a few minutes. Do not continue until you see the following message: **Installed operator: ready for use.**

12. In your command prompt or terminal window, run the following commands to add the Container Registry credential to the operator's service account.

```
oc project wxa4z-byos
```

```
oc patch serviceaccount ibm-wxa4z-operator-controller-manager --type merge -p
'{"imagePullSecrets": [{"name": "icr-pull-secret"}]}'
```

```
watsonxAssistant -- cecuser@p1365-pvm1:~ -zsh - 135x37
andrewjones@Andrews-MBP watsonxAssistant % oc create namespace wxa4z-byos
namespace/wxa4z-byos created
andrewjones@Andrews-MBP watsonxAssistant % export IBM_CS_ENT_KEY=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJJQk0gTWFya2V0cGxhY2UiLCJpYXQiOjE2NjYwMTk1ODAsImp0aS16IjNkOWUyMzjZTAzMDQzMzVhNTJhYTkzMWNmOTcyMDR1In0.4M3XRD4XzkHMS0kFNJ4uKVcwZ6SnEAZO3eL_11A2xY
andrewjones@Andrews-MBP watsonxAssistant % oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --docker-username=cp --docker-password=$IBM_CS_ENT_KEY
secret/icr-pull-secret created
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml
andrewjones@Andrews-MBP watsonxAssistant % oc apply -f catalogSource.yaml
catalogsource.operator.coreos.com/ibm-wxa4z-operator-catalog created
andrewjones@Andrews-MBP watsonxAssistant % oc project wxa4z-byos
Now using project "wxa4z-byos" on server "https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443".
andrewjones@Andrews-MBP watsonxAssistant % oc patch serviceaccount ibm-wxa4z-operator-controller-manager --type merge -p '{"imagePullSecrets": [{"name": "icr-pull-secret"}]}'
serviceaccount/ibm-wxa4z-operator-controller-manager patched
andrewjones@Andrews-MBP watsonxAssistant %
```

13. In the Red Hat OpenShift web console, under Workloads, click Pods.

The screenshot shows the Red Hat OpenShift web console interface. The left sidebar has a navigation tree with 'Administrator', 'Home', 'Operators' (selected), 'OperatorHub' (highlighted with a green box), 'Installed Operators', 'Workloads' (selected), 'Pods' (highlighted with a green box), 'Deployments', 'DeploymentConfigs', 'StatefulSets', and 'Secrets'. The main content area displays a card for 'IBM watsonx Assistant for Z' with the subtext 'ibm-wxa4z-operator/v2.01 provided by IBM' and a green checkmark icon. A status bar at the bottom says 'Installed Operator ready for use'.

14. Verify the two pods that start with ibm-wxa4z-operator have a status of Running and that all pods are Ready.

The screenshot shows the Red Hat OpenShift web console interface. The left sidebar has a navigation tree with 'Administrator', 'Home', 'Operators', 'OperatorHub', 'Installed Operators', 'Workloads' (selected), 'Pods' (highlighted with a green box), 'Deployments', 'DeploymentConfigs', 'StatefulSets', and 'Secrets'. The main content area shows a table of pods in the 'wxa4z-byos' project. The table columns are Name, Status, Ready, Restarts, Owner, Memory, CPU, and Created. Two pods are listed: 'ld126367b1ca53dcf2b0c93acd733e33875cd4b6c382dd5eb412032ac2h6' (Status: Completed) and 'ibm-wxa4z-operator-catalog-n9m5m' (Status: Running). Both pods are marked as 'Ready' (green box). A 'Create Pod' button is visible in the top right of the table area.

15. Run the following command to set the administrative policy for the workspace.

```
oc -n wxa4z-byos adm policy add-scc-to-user privileged -z byos
```

```
watsonxAssistant@Andrews-MBP watsonxAssistant % oc create namespace wxa4z-byos
namespace/wxa4z-byos created
andrewjones@Andrews-MBP watsonxAssistant % export IBM_CS_ENT_KEY=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9eyJpc3MiOiJJQk0gTWFya2V0cGxhY2UiLCJpYXQiOjE2NjYwMTk1ODAsImp0aSI6ijNkOWUyMzZjTAzMDQzMzvhNTJhYTkzMWNmOTcyMDR1In0.4M3XRD4XzkHMS0kFNJ4uKVcWZ6SnEA0Z03eL_11A2xY
andrewjones@Andrews-MBP watsonxAssistant % oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --docker-username=cp --docker-password=$IBM_CS_ENT_KEY
secret/icr-pull-secret created
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml
andrewjones@Andrews-MBP watsonxAssistant % oc apply -f catalogSource.yaml
catalogsource.operator.coreos.com/ibm-wxa4z-operator-catalog created
andrewjones@Andrews-MBP watsonxAssistant % oc project wxa4z-byos
Now using project "wxa4z-byos" on server "https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443".
andrewjones@Andrews-MBP watsonxAssistant % oc patch serviceaccount ibm-wxa4z-operator-controller-manager --type merge -p '{"imagePullSecrets": [{"name": "icr-pull-secret"}]}'
serviceaccount/ibm-wxa4z-operator-controller-manager patched
andrewjones@Andrews-MBP watsonxAssistant % oc -n wxa4z-byos adm policy add-scc-to-user privileged -z byos
clusterrole.rbac.authorization.k8s.io/system:openshift:scc:privileged added: "byos"
andrewjones@Andrews-MBP watsonxAssistant %
```

Deploy required secrets and the custom bring-your-own-search (BYOSearch) resources

1. In a text editor, create a file named `os-secret.yaml` and paste the following text in the file.

File name:

```
os-secret.yaml
```

Substitute a secure password of your choosing for the string `<OPENSEARCH_PASSWORD>`.

File contents:

```
apiVersion: v1
stringData:
  password: <OPENSEARCH_PASSWORD>
kind: Secret
metadata:
  name: opensearch-creds
  namespace: wxa4z-byos
type: Opaque
```

2. Create the secret by running the following command.

```
oc apply -f os-secret.yaml
```

3. In a text editor, create a file named `client-ingestion-secret.yaml` and paste the following text in the file.

File name:

```
client-ingestion-secret.yaml
```

Substitute a secure authentication key of your choosing for the string `<CLIENT_INGESTION_AUTHKEY>`. The authentication key can be a random password.

File contents:

```
apiVersion: v1
stringData:
  authkey: <CLIENT_INGESTION_AUTHKEY>
kind: Secret
metadata:
  name: client-ingestion-authkey
  namespace: wxa4z-byos
type: Opaque
```

4. Create the secret by running the following command.

```
oc apply -f client-ingestion-secret.yaml
```

5. Obtain and record your cluster domain that is used for routes by running the following command.

```
oc -n openshift-ingress-operator get ingresscontroller default -o jsonpath=".status.domain"
```



The output from the command does not include a newline.

The value returned for the cluster domain does not include a newline. When copying the value do not include the character or characters used for your command line prompt. Do not include the your prompt in the next step!

Note: The output of the command will be a string similar to:

apps.672b79320c7a71b728e523b4.ocp.techzone.ibm.com

6. In a text editor, create a file named `byos.yaml` and paste the following text in the file.

File name:

```
byos.yaml
```

Substitute the domain name recorded in the previous step for the string `<YOUR_CLUSTER_DOMAIN>`.

File contents:

```

apiVersion: wxa4z.watsonx.ibm.com/v1
kind: BYOSearch
metadata:
  name: byosearch
  namespace: wxa4z-byos
spec:
  imagePullSecrets:
    - name : icr-pull-secret
  namespace: wxa4z-byos
  clusterName: wxa4z-byos-cluster
  clusterDomain: <YOUR_CLUSTER_DOMAIN>

  opensearch:
    secretName: opensearch-creds

  persistence:
    enabled: true
    storageClass: "managed-nfs-storage"
    accessModes:
      - ReadWriteOnce
    size: 24Gi

  wrapper:
    createRoute: true
    resources:
      requests:
        cpu: 4
        memory: "500Mi"
      limits:
        cpu: 4
        memory: "1Gi"

  clientIngestion:
    secretName: client-ingestion-authkey

    resources:
      limits:
        cpu: "500m"
        memory: 2Gi
        nvidia.com/gpu: "0"
      requests:
        cpu: "500m"
        memory: 1Gi
        nvidia.com/gpu: "0"
    pvc:
      storageClass: "managed-nfs-storage"
      enabled: true
      size: 24Gi

```

7. Run the following command to deploy BYOS on your cluster.

```
oc apply -f byos.yaml
```

Verify all the required pods are running and get the network route to your BYOS instance

1. In the OCP console, verify that all pods have the status of **Running** or **Completed**.

⚠️ Do not continue until...

The BYOS deployment can take 20 minutes or more to complete. Do not continue until all the pods have a status of "Running" or "Completed". The next step is to retrieve your BYOS endpoint URL.

Name	Status	Ready	Restarts	Owner	Memory	CPU	Created
da80909aab1563ff0c15b7c6d	Completed	0/1	0	da80909aab1563ff0c15b7c6d81	-	-	Nov 6, 2024, 9:26 AM
ibm-wxa4z-operator-catalog-cfsv	Running	1/1	0	ibm-wxa4z-operator-catalog	25.0 MiB	0.004 cores	Nov 6, 2024, 9:25 AM
ibm-wxa4z-operator-controller-manager-556fcf98bb-4bhmk	Running	2/2	0	ibm-wxa4z-operator-controller-manager-556fcf98bb	142.8 MiB	0.023 cores	Nov 6, 2024, 9:26 AM
wxa4z-byos-cluster-0	Running	1/1	0	wxa4z-byos-cluster	1,095.8 MiB	0.017 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-1	Running	1/1	0	wxa4z-byos-cluster	1,169.1 MiB	0.016 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-2	Running	1/1	0	wxa4z-byos-cluster	1,113.3 MiB	0.015 cores	Nov 6, 2024, 10:22 AM
wxa4z-client-ingestion-7f98d86c58-9bzth	Running	1/1	0	wxa4z-client-ingestion-7f98d86c58	501.8 MiB	0.071 cores	Nov 6, 2024, 12:18 PM
wxa4z-opensearch-wrapper-5cb879f5f8-qw7qt	Running	1/1	0	wxa4z-opensearch-wrapper-5cb879f5f8	547.1 MiB	0.031 cores	Nov 6, 2024, 10:22 AM
wxa4z-snapshot-setup-job-nsqlt	Completed	0/1	0	wxa4z-snapshot-setup-job	-	-	Nov 6, 2024, 10:22 AM

2. Under **Networking**, click **Routes**.

Name	Status	Ready	Restarts	Owner	Memory	CPU	Created
da80909aab1563ff0c15b7c6d	Completed	0/1	0	da80909aab1563ff0c15b7c6d81	-	-	Nov 6, 2024, 9:26 AM
ibm-wxa4z-operator-catalog-cfsv	Running	1/1	0	ibm-wxa4z-operator-catalog	25.0 MiB	0.004 cores	Nov 6, 2024, 9:25 AM
ibm-wxa4z-operator-controller-manager-556fcf98bb-4bhmk	Running	2/2	0	ibm-wxa4z-operator-controller-manager-556fcf98bb	141.2 MiB	0.022 cores	Nov 6, 2024, 9:26 AM
wxa4z-byos-cluster-0	Running	1/1	0	wxa4z-byos-cluster	1,095.8 MiB	0.016 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-1	Running	1/1	0	wxa4z-byos-cluster	1,169.1 MiB	0.016 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-2	Running	1/1	0	wxa4z-byos-cluster	1,113.3 MiB	0.015 cores	Nov 6, 2024, 10:22 AM
wxa4z-client-ingestion-7f98d86c58-9bzth	Running	1/1	0	wxa4z-client-ingestion-7f98d86c58	534.6 MiB	0.087 cores	Nov 6, 2024, 12:18 PM
wxa4z-opensearch-wrapper-5cb879f5f8-qw7qt	Running	1/1	0	wxa4z-opensearch-wrapper-5cb879f5f8	547.1 MiB	0.030 cores	Nov 6, 2024, 10:22 AM
wxa4z-snapshot-setup-job-nsqlt	Completed	0/1	0	wxa4z-snapshot-setup-job	-	-	Nov 6, 2024, 10:22 AM

3. Copy and record the location for the **wxa4z-opensearch-wrapper** route.

Name	Status	Location	Service
wxa4z-client-ingestion	Accepted	https://wxa4z-client-ingestion-wxa4z-byos.apps.672b79320c7a71b728e523b4.ocp.techzone.ibm.com	wxa4z-client-ingestion
wxa4z-opensearch-wrapper	Accepted	https://wxa4z-opensearch-wrapper-wxa4z-byos.apps.672b79320c7a71b728e523b4.ocp.techzone.ibm.com	wxa4z-opensearch-wrapper

Update your assistant with the new BYOS instance route

You are now ready to configure your assistant with the route to your BYOS instance.

1. Using the network route for your BYOS instance, append the string /v1/query to complete the URL endpoint.

The URL should look similar to:

```
https://wxa4z-opensearch-wrapper-wxa4z-
byos.apps.672b79320c7a71b728e523b4.ocp.techzone.ibm.com/v1/query
```

Important: The above URL will not work for you. Use the value of your specific OpenSearch instance that is recorded in the previous step.

2. Update your assistant's custom search integration URL.

Next, you need to return to your assistant in the watsonx Orchestrate AI assistant builder and update the custom search integration URL. The steps to update the URL are illustrated in the animated gif that follows. You can review the steps to accomplish this [here](#) (be sure to use your BYOS URL and not the shared URL specified in the lab guide).

Name	Status	Location	Service
wxa4z-client-ingestion	Accepted	https://wxa4z-client-ingestion-wxa4z-byos.apps.67b79320c7a7b728e523b4.ocp.techzone.ibm.com	wxa4z-client-ingestion
wxa4z-opensearch-wrapper	Accepted	https://wxa4z-opensearch-wrapper-wxa4z-byos.apps.67b79320c7a7b728e523b4.ocp.techzone.ibm.com	wxa4z-opensearch-wrapper

Troubleshooting

The following are issues that you may encounter. If the provided resolutions do not work, contact support by using the methods that are mentioned in the [Support](#) section.

✖ Pods have a status of ErrImagePull or ImagePullBackoff

If the pods starting with **ibm-wxa4z-operator** have a status of “ErrImagePull” or “ImagePullBackoff”, you can delete the pod and it will automatically restart and pull the image successfully. Wait until the pod is re-created successfully.

✖ The wxa4z-client-ingestion pod does not start

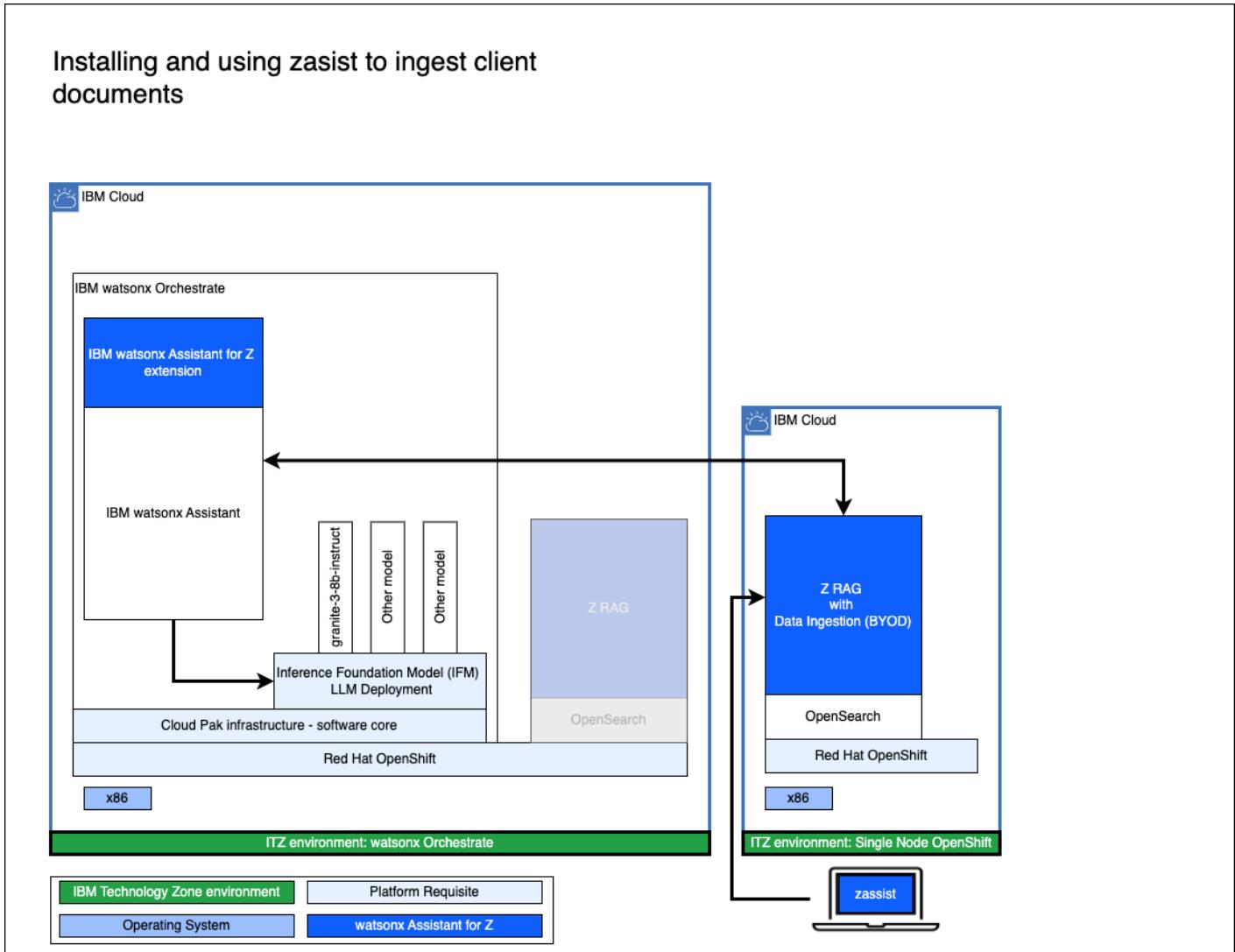
Did you include the % character in the **clusterDomain** name when creating the **byos.yaml**? To resolve, edit the **byos.yaml** file and run the following command again. The current pod will be terminated and a new one started. This will take about 20 minutes to start.

```
oc apply -f byos.yaml
```

Installing and using zassist to ingest client documents

With bring-your-own-search (BYOS) installed and configured in your assistant, you can now prepare for document ingestion. Currently, only PDF, HTML, and DOCX file formats are supported for ingestion.

Below is a high-level, logical architecture of the environment you will deploy in this section.



To prepare for document ingestion, you can also reference the setup instructions that are located [here](#).

Install the zassist utility

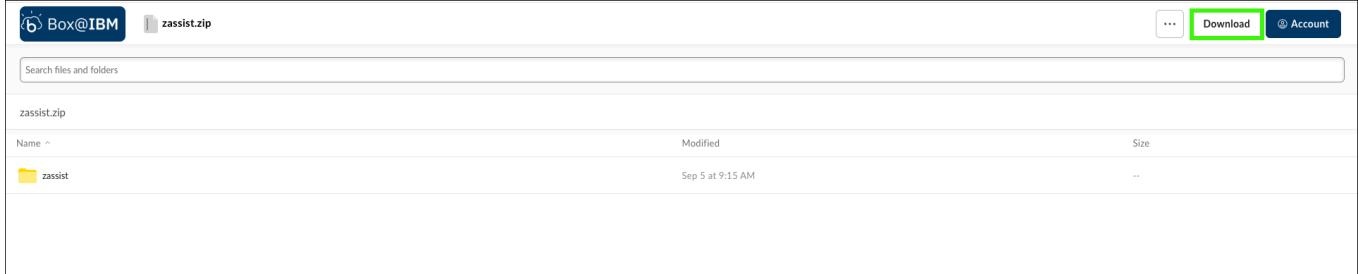
The **zassist** utility is an executable program that automates the ingestion of client documentation into the RAG for Watsonx Assistant for Z. A version of zassist is available for download for IBMers and Business Partners for conducting pilots. Follow the steps below to download and install **zassist**.

How do clients get the zassist utility?

The utility is available to clients through [IBM Passport Advantage](#).

1. Click the following link and download the **zassist.zip** file.

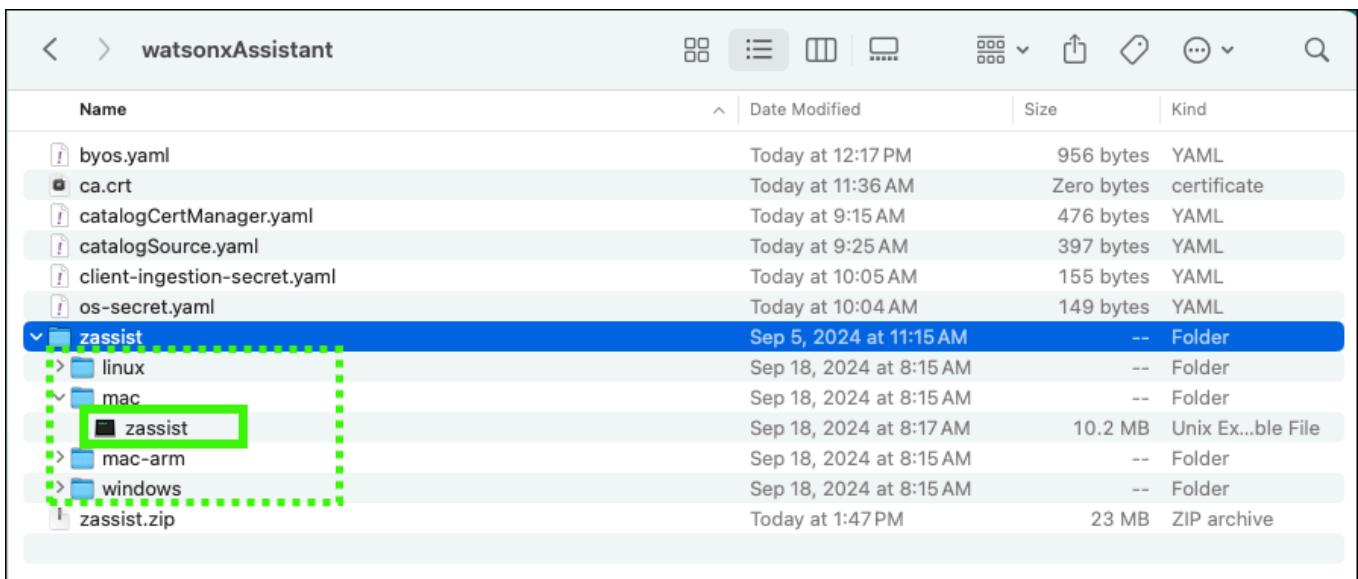
<https://ibm.box.com/s/j3nt5iw4fqd5w2jgcqwxnjlsu8bpvl77>



A screenshot of the IBM Box web interface. The user is in a folder named 'zassist'. A single file, 'zassist.zip', is listed. The file has a yellow icon, a name of 'zassist', a modified date of 'Sep 5 at 9:15 AM', and a size of '--'. At the top right, there are buttons for 'Download' (highlighted in green) and 'Account'.

2. Extract the **zassist.zip** file.

3. Locate the appropriate file for your local workstation's operating system.



A screenshot of a file explorer window titled 'watsonxAssistant'. It shows a list of files and a tree view of a 'zassist' directory. The tree view shows 'zassist' expanded, with sub-directories 'linux', 'mac' (which contains 'zassist' again), 'mac-arm', and 'windows'. The 'zassist.zip' file is also visible at the bottom. The 'zassist' file under 'mac' is highlighted with a green selection box. The file list includes:

Name	Date Modified	Size	Kind
byos.yaml	Today at 12:17 PM	956 bytes	YAML
ca.crt	Today at 11:36 AM	Zero bytes	certificate
catalogCertManager.yaml	Today at 9:15 AM	476 bytes	YAML
catalogSource.yaml	Today at 9:25 AM	397 bytes	YAML
client-ingestion-secret.yaml	Today at 10:05 AM	155 bytes	YAML
os-secret.yaml	Today at 10:04 AM	149 bytes	YAML
zassist	Sep 5, 2024 at 11:15 AM	--	Folder
linux	Sep 18, 2024 at 8:15 AM	--	Folder
mac	Sep 18, 2024 at 8:15 AM	--	Folder
zassist	Sep 18, 2024 at 8:17 AM	10.2 MB	Unix Executable File
mac-arm	Sep 18, 2024 at 8:15 AM	--	Folder
windows	Sep 18, 2024 at 8:15 AM	--	Folder
zassist.zip	Today at 1:47 PM	23 MB	ZIP archive

4. Either copy the appropriate **zassist** file to a directory in your PATH, or add the appropriate directory to your PATH environment variable.

Additional information for running the preceding tasks can be found [here](#).

5. Run the **zassist** command to verify it is working.



A screenshot of a terminal window. The user runs the command 'zassist'. The output shows an error message: 'zassist: error: expected one of "version", "init", "login", "ingest", "load", ...'. The entire command line and output are highlighted with a green selection box.



Mac/OS users may need to adjust security settings.



The **zassist** binary may cause a security exception. Adjust the security settings by opening the **System Settings** utility and clicking **Privacy & Security**. Under **Security** locate the message about the **zassist** binary and click **Allow Anyway**. Return to the terminal window and try running the command again.

Ingest client documentation using zassist

With the **zassist** command installed, you are now able to begin ingesting data.

Step-by-step guidance for ingesting documents using zassist is provided in the IBM watsonx Assistant for Z documentation.

Follow the directions [here](#) to ingest documents using zassist.

These steps are not repeated in this lab guide. The following video illustrates the steps to ingest a single document. The document that is ingested in the video is a compressed PDF of the **IBM z/OS Continuous Delivery Red Piece**. You can download a copy of this document [here](#).

Note: The video has no audio.



Don't see the video in the PDF version of the lab guide?

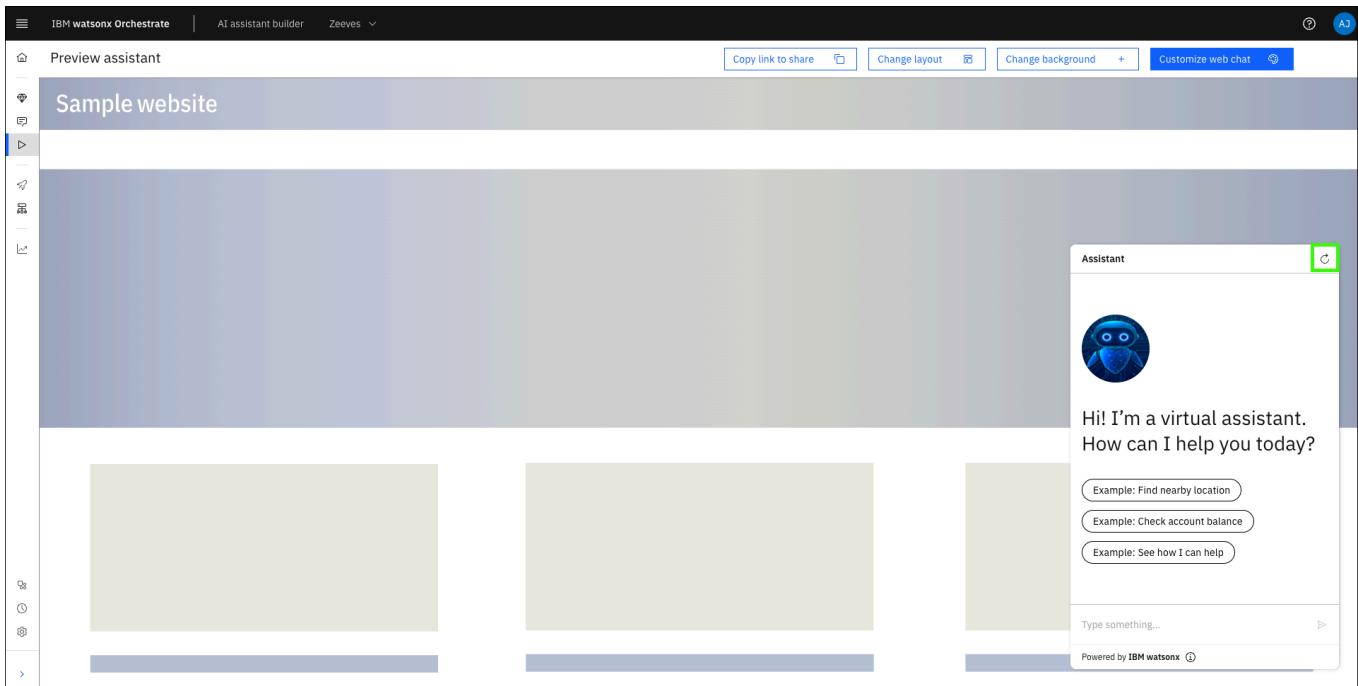


If you are viewing the [PDF](#) of the lab guide, you can access the video [here](#).

Verify the document that is ingested is now returned as a source file for a query

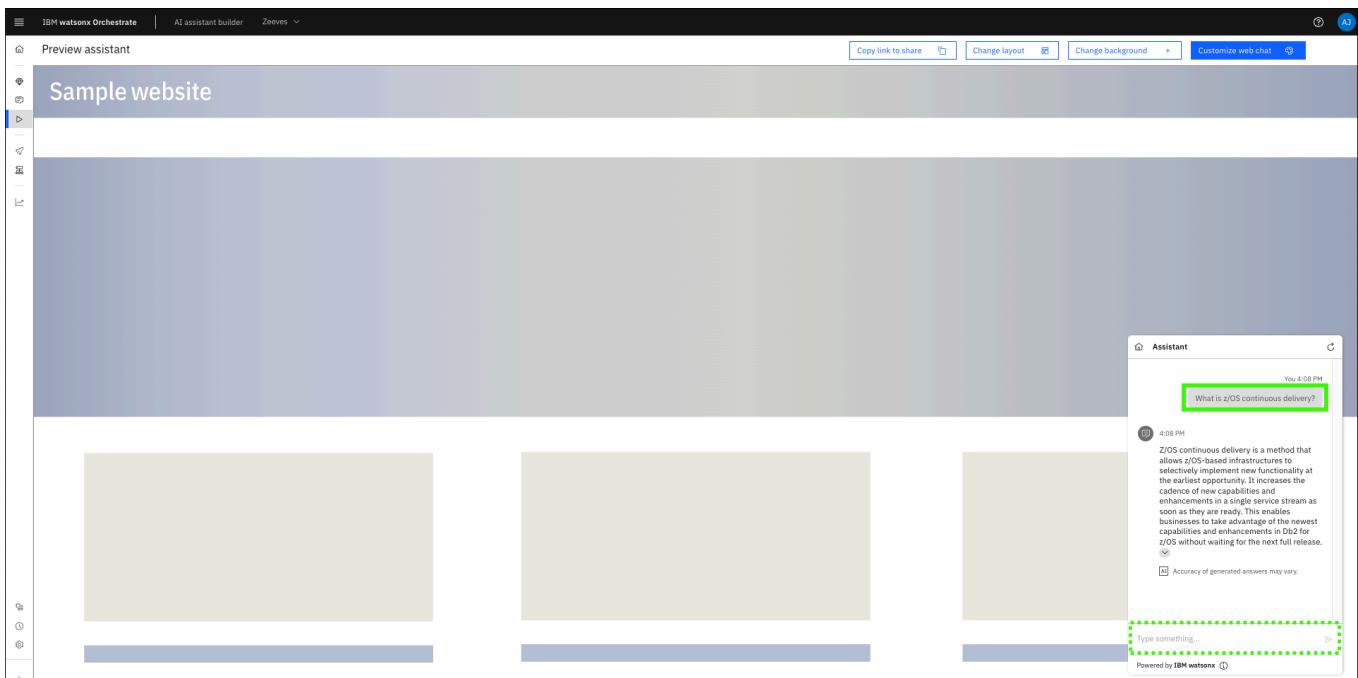
Use the watsonx Orchestrate AI assistant builder to verify your document ingestion.

1. Hover over the **Home** () icon and click **Preview**.
2. Click the **Restart conversation** () icon.

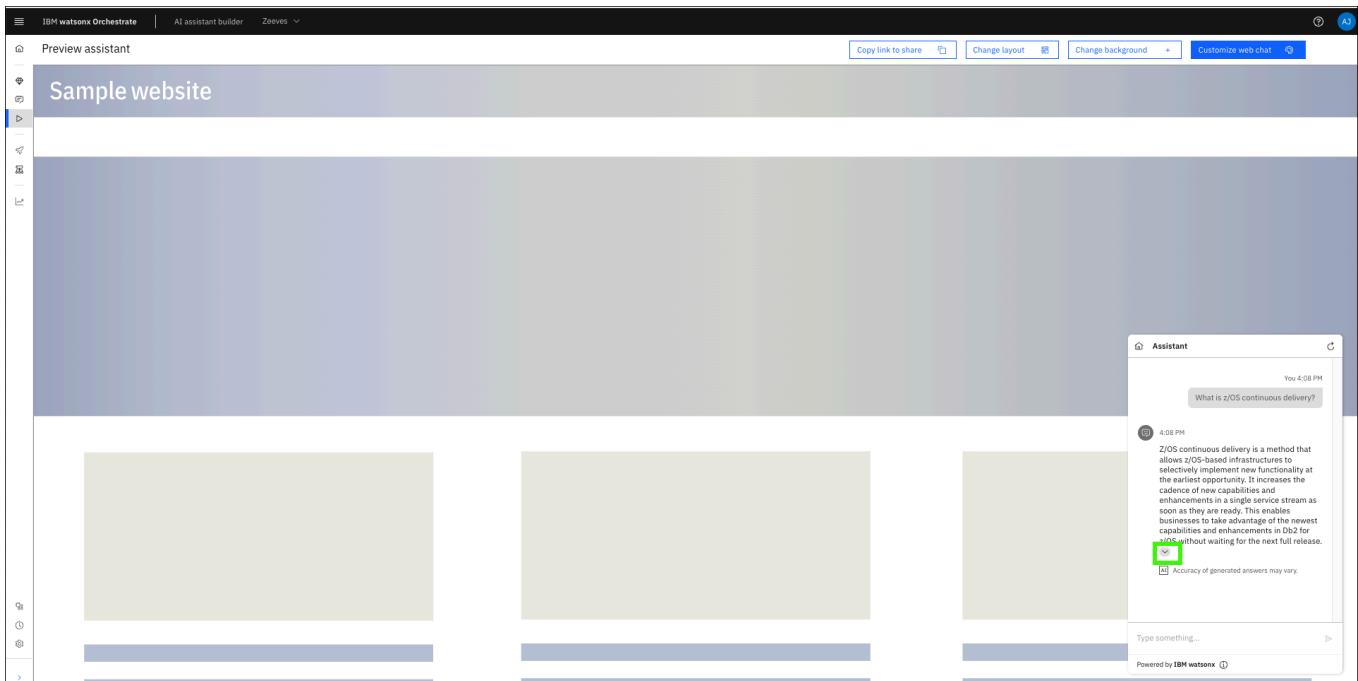


3. Enter the following prompt in your assistant.

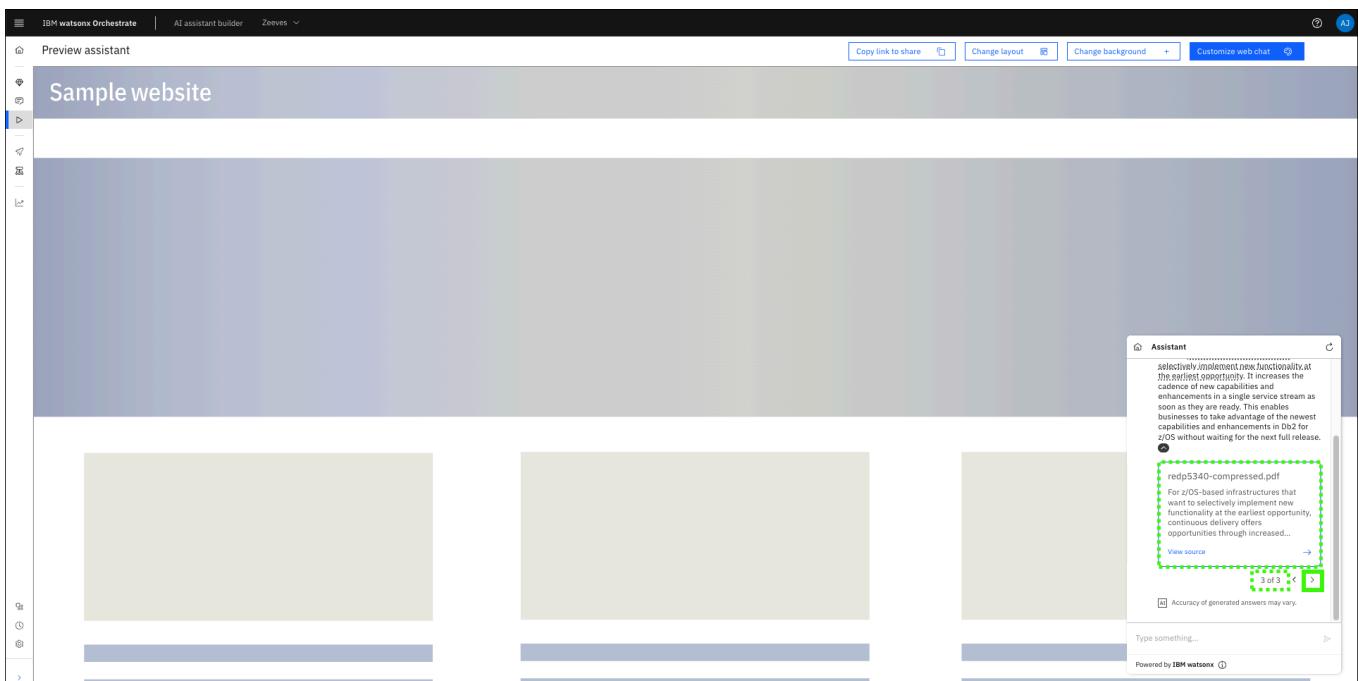
What is z/OS continuous delivery?



4. Clicking the Down arrow (▼).



5. Click through the list of resources and find the reference to the Red Piece document you ingested.



6. Click the ingested document reference.

7. Accept the security risk to view the source document.

The steps to accept the security risk for the document are not shown. The risk occurs because the certificate for the connection to the SNO instance is not secure. Notice that the URL contains the path to your SNO instance route.

Adjusting the search behavior

Do you recall the **Metadata** field when you configured your assistant?

The screenshot shows the 'Custom service' configuration page in the AI assistant builder. The 'Metadata' field is highlighted with a green dashed border. The JSON example provided is:

```
{
  "example_field": "example_value",
  "other_example_field": 7
}
```

The Metadata field provides a way to adjust your assistant's behavior during conversational search for your OpenSearch instance. Now that you have your own docs that are ingested for conversational search, you can set the metadata field for your assistant to use those documents in its content-grounded search. If you leave the metadata field empty, then it defaults to settings found to perform well. This replaces having to paste a complicated search string.

If you leave this field empty, OpenSearch will rely on the default settings, which means OpenSearch will search all of the default IBM-provided documentation and all of the ingested customer documentation using the following value:

```
{"ibm_indices": "*_ibm_docs_slate",
"customer_indices": "customer_*"}
```

Replacing the wildcard string with an explicit list of indices allows for personalization. The metadata setting is where you can input specific indices (pointing to the underlying documentation) that you want your assistant to use for the content-grounded search. There are over 220 products and topics that the OpenSearch instance has IBM Documentation for. You can find those indices and products [here](#).

You can input a subset of indices into the "Metadata" field in cases where you only want your assistant to gather context for specific IBM products or topics. The specific indices can be listed out in this format:

```
{"ibm_indices": "<comma separated index values>", "customer_indices": "customer_*"}
```

For example, if you only want your assistant to reference documentation for "Db2 Analytics Accelerator for z/OS" and no ingested client documentation, you can enter the following into the metadata field:

```
{"ibm_indices": "ss4lq8_ibm_docs_slate"}
```

If you have a mix of IBM Documentation and client documentation ingested, then there's an optional search string that you can use to set the "weights" used for each.

For example:

```
{"doc_weight":  
{"product_docs":0.5,  
"customer_docs":0.5},  
"ibm_indices":"*_ibm_docs_slate",  
"customer_indices":"customer_**"  
}
```

In this case, "product_docs" is the weight that is assigned to "ibm_indices" and "customer_docs" is the weight that is assigned to "customer_indices".

After you have configured all the settings for Conversational Search on the page, be sure to click **Save** in the upper-right of the page.

For more information on customizing the metadata field for conversational search, refer to this supplemental video found [here](#).

You are encouraged to experiment with the metadata field!

Try setting the metadata field to the following, which weights ingested docs higher than the product docs:

```
{"doc_weight":  
{"product_docs":0.2,  
"customer_docs":0.8},  
"ibm_indices":"*_ibm_docs_slate",  
"customer_indices":"customer_**"  
}
```

Repeat the earlier steps to restart the assistant preview (⟳) and reissue your queries. Notice that the ingested Red Piece document is now the first sited reference!

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with icons for navigation and a preview assistant section labeled 'Sample website'. The main area displays three large, light-grey rectangular blocks representing website content. On the right, there's an 'Assistant' window. It shows a message from 'redp5340-compressed.pdf' with a timestamp of '4:30 PM'. The message discusses the availability of new functionality in z/OS-based infrastructures and how continuous delivery offers opportunities through increased functionality rollback. Below the message, there's a 'View source' button and a small navigation bar with arrows. At the bottom of the screen, there's a footer bar with the text 'Powered by IBM Watsonx'.



For client pilots

If you or your client have other documents to ingest, you can do so by repeating the steps using zassist. The Velocity Pilot ITZ environment is limited in compute and storage capacity. The following limits should be adhered to:

- Greater than 15,000 documents is not supported.
- Individual documents should be less than 10 megabytes (MB).
- Larger documents greater than 50 MB should work if the document contains mostly images.
- Loading documents can take a long time, especially with > 100 MB of text.
- It is recommended to run large loads late at night.
- When loading, ensure your workstations does not sleep during the process.
- If you receive a **batch time error**, set the batch size to a lower number for that command. For example:

```
zassist ingest . -s 50
```

After ingesting all your additional documents, proceed to the next section to learn about adding skills to your assistant.

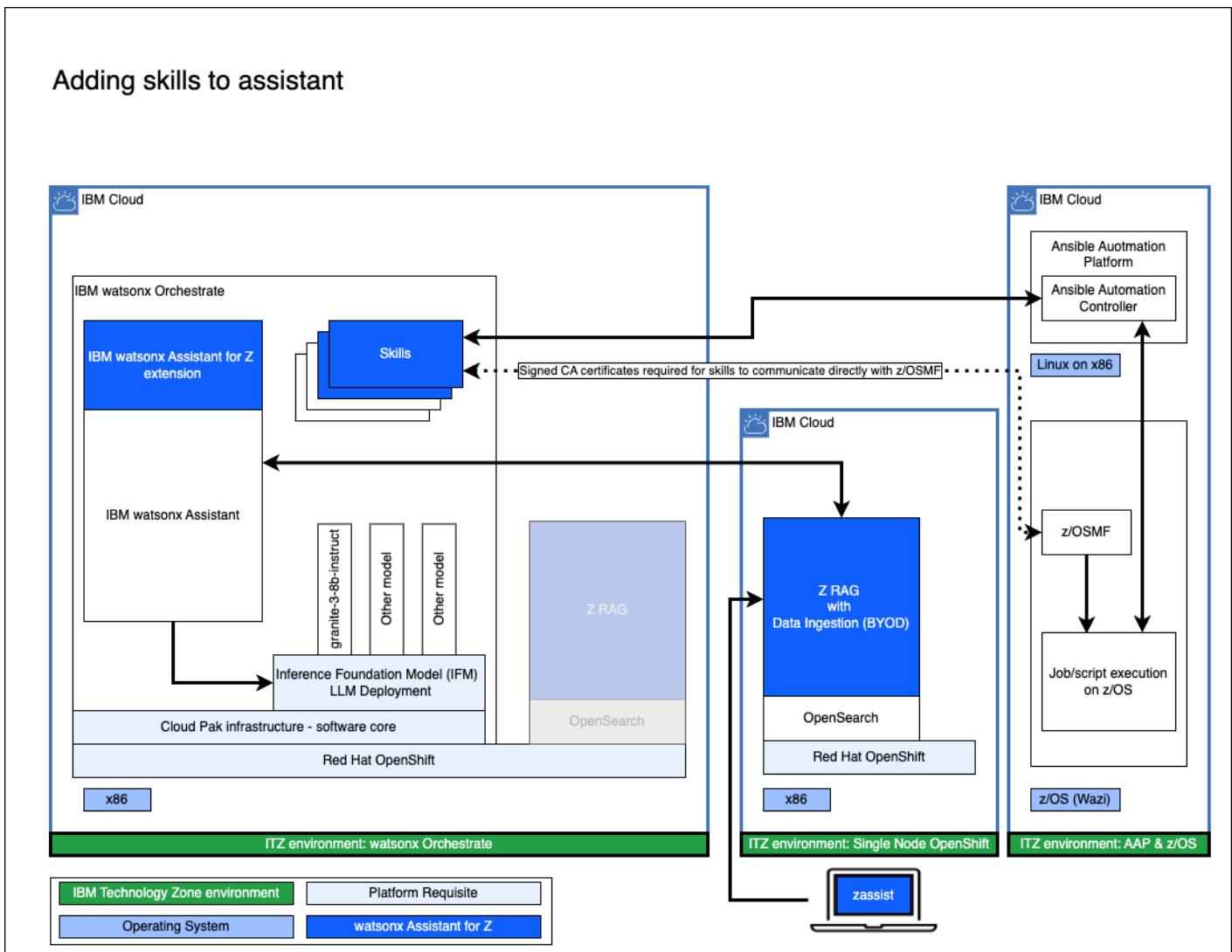
Adding skills to the assistant

Getting started with skills and actions

Watsonx Assistant for Z can import skills to automate a range of IBM Z related tasks through assistant interactions. Each skill is a pre-defined automation that runs tasks to accomplish some unit or units of work. For example, you can use a skill to view z/OS IPL information, or work with z/OS datasets.

IBM watsonx Assistant for Z provides an extension within watsonx Orchestrate to build new skills from Ansible Automation platform or z/OS Management Facility (z/OSMF). The Z Skills Accelerator extension connects Ansible and z/OS application programming interfaces (APIs) and can import automation as Ansible Playbooks, JCL, or REXX as skills. Learn more importing and building skills [here](#).

Below is a high-level, logical architecture of the environment you will deploy in this section.



Environments

Watsonx Orchestrate

The Z Skills Accelerator extension is already configured in your watsonx Orchestrate IBM Technology Zone (ITZ) environment. You can use this component to import new skills.

Ansible Automation Platform and Wazi as a Service

To import skills for automations, you use Ansible Automation Platform (AAP) and Wazi as a Service (Wazi aaS) to serve as the z/OS back-end. Learn more about AAP [here](#). Learn more about Wazi, [here](#).

The two resources are provisioned together in the ITZ environment that you reserved earlier. This environment enables the ability to manage and automate z/OS tasks and subsystems with various preinstalled ansible playbooks. It includes a z/OS back-end (Wazi as a Service) with all needed prerequisites.

The playbooks provided cover various use cases for automating z/OS management. Ansible's capabilities for automating various Z-specific tasks are not limited to the use cases that are preinstalled in the AAP instance. The preinstalled playbooks are tasks from the 'IBM z/OS core collection'. Using this environment accelerates the ability to showcase the value of watsonx assistant for Z, and to get started with simple automations that can be expanded.

The ITZ environment gives you access to AAP, which is preconfigured to target the accompanying z/OS Wazi system, along with web-based access to AAP to experiment with different playbook templates. These templates are imported into watsonx Orchestrate as skills and connected to your assistant.

For more information on the AAP and Wazi z/OS environments, refer to this [document](#).

The playbook templates that are preinstalled in AAP cover various use cases, which you can explore, including:

- z/OS Certificate Management (create, delete, list, and renew certificates)
- dataset management (create, delete, fetch datasets)
- Submit JCL
- Run Operator commands
- Run TSO commands
- And more

Each of the sections that follow build upon each other. Complete each to successfully enhance your assistant by starting with [Explore Ansible Automation Platform](#).

Explore Ansible Automation Platform

After you reserved the Ansible Automation Platform (AAP) and Wazi z/OS environment in IBM Technology Zone (ITZ) and the reservation is in the **Ready** state, follow these steps to explore AAP.

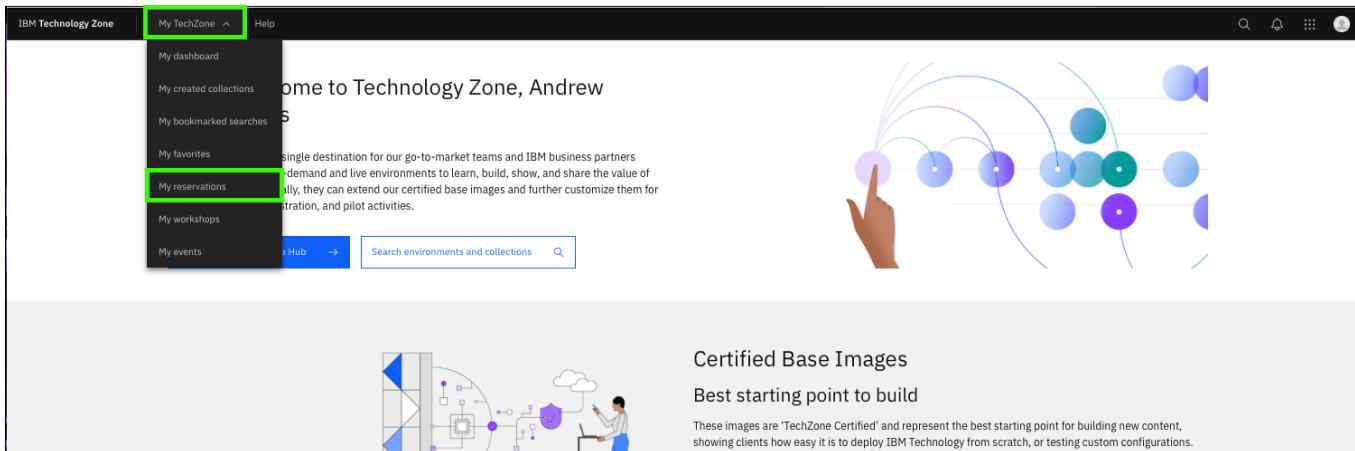
Access the AAP and Wazi as a Service environment

Be sure to record the information as instructed

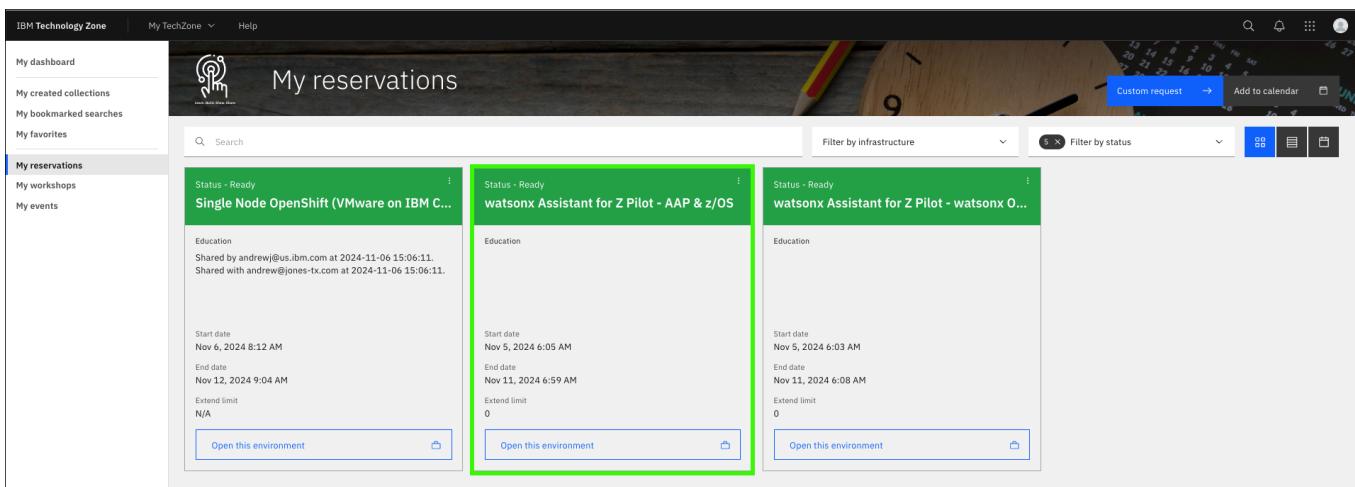
Several of the steps below instruct you to record values from your ITZ reservation. Be sure to do this as they will not only be used in this section, but also in later sections of the lab guide.

1. In the IBM Technology Zone portal, expand **My TechZone** and select **My Reservations**, or click the following link.

[ITZ My reservations](#)



2. Click the **watsonx Assistant for Z Pilot - AAP & z/OS** tile.



3. Locate and record the **AAP User Name (For UI access)** and **AAP User Password** fields.

The screenshot shows the IBM Technology Zone interface with the following details:

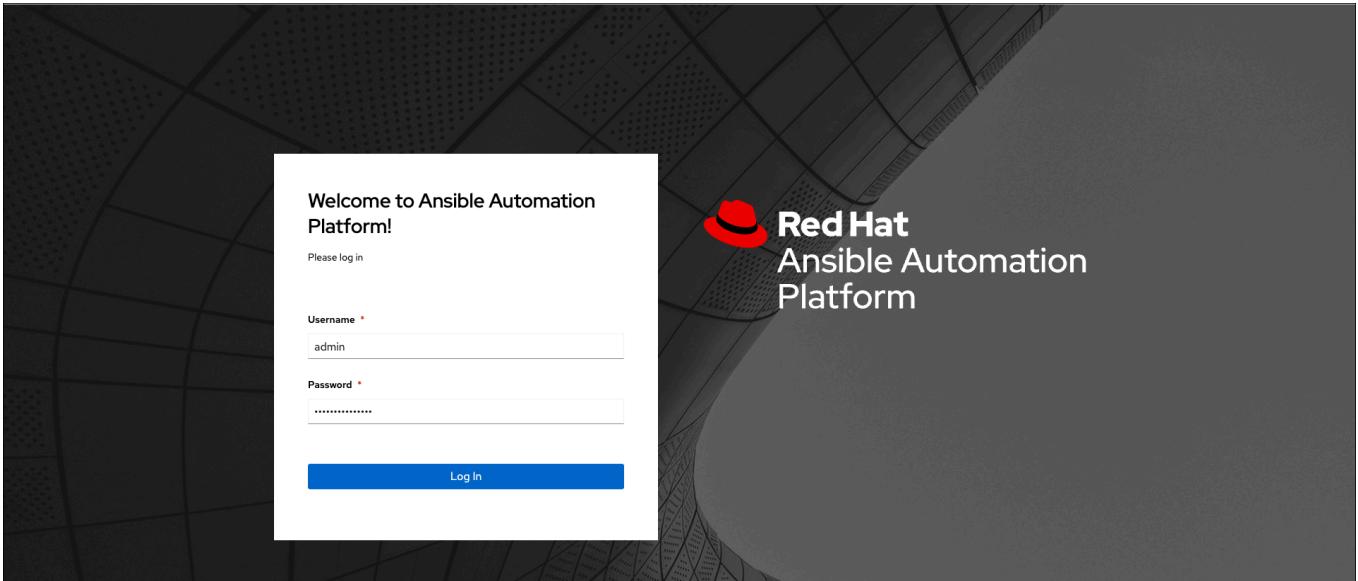
- Idle runtime limit:** 10800
- Reservation Details:**
 - Ansbile Automation Platform Cockpit UI: <https://itvsi-aap-phdhuz4.techzone.ibm.com:9090>
 - Ansbile Automation Platform Private IP: 10.241.0.6
 - Ansbile Automation Platform Public IP: 150.239.83.37
 - Ansbile Automation Platform UI: <https://itvsi-aap-phdhuz4.techzone.ibm.com>
 - AAP User Name (For SSH access): U6VO7N3
 - AAP User Name (For UI access): admin
 - AAP User Password: 39hQc-NiS458-Wa
 - AAP VSI Management: https://cloud.ibm.com/vpc-ext/compute/vs/us-east-0757_d038c79d-2fc8-45fd-8429-5bc642623771?view=edit&bs_account=7fd16281c12c4681b91417749e322845
 - z/OS Private IP: 10.241.0.7
 - z/OS Public IP: 169.63.103.211
 - z/OS User Name (Use SSH key to login, not password): IBMUSER
 - z/OS VSI Management: https://cloud.ibm.com/vpc-ext/compute/vs/us-east-0757_d729cb11-98ff-4453-8f74-acc8c595be1?view=edit&bs_account=7fd16281c12c4681b91417749e322845
 - User SSH key: [dropdown menu]

4. Record and then click the Ansible Automation Platform UI link.

The screenshot shows the IBM Technology Zone interface with the following details:

- Idle runtime limit:** 10800
- Reservation Details:**
 - Ansbile Automation Platform Cockpit UI: <https://itvsi-aap-phdhuz4.techzone.ibm.com:9090>
 - Ansbile Automation Platform Private IP: 10.241.0.6
 - Ansbile Automation Platform Public IP: 150.239.83.37
 - Ansbile Automation Platform UI: <https://itvsi-aap-phdhuz4.techzone.ibm.com>
 - AAP User Name (For SSH access): U6VO7N3
 - AAP User Name (For UI access): admin
 - AAP User Password: 39hQc-NiS458-Wa
 - AAP VSI Management: https://cloud.ibm.com/vpc-ext/compute/vs/us-east-0757_d038c79d-2fc8-45fd-8429-5bc642623771?view=edit&bs_account=7fd16281c12c4681b91417749e322845
 - z/OS Private IP: 10.241.0.7
 - z/OS Public IP: 169.63.103.211
 - z/OS User Name (Use SSH key to login, not password): IBMUSER
 - z/OS VSI Management: https://cloud.ibm.com/vpc-ext/compute/vs/us-east-0757_d729cb11-98ff-4453-8f74-acc8c595be1?view=edit&bs_account=7fd16281c12c4681b91417749e322845
 - User SSH key: [dropdown menu]

5. Enter the **Username** and **Password** that is recorded in step 3 and click **Log In**.



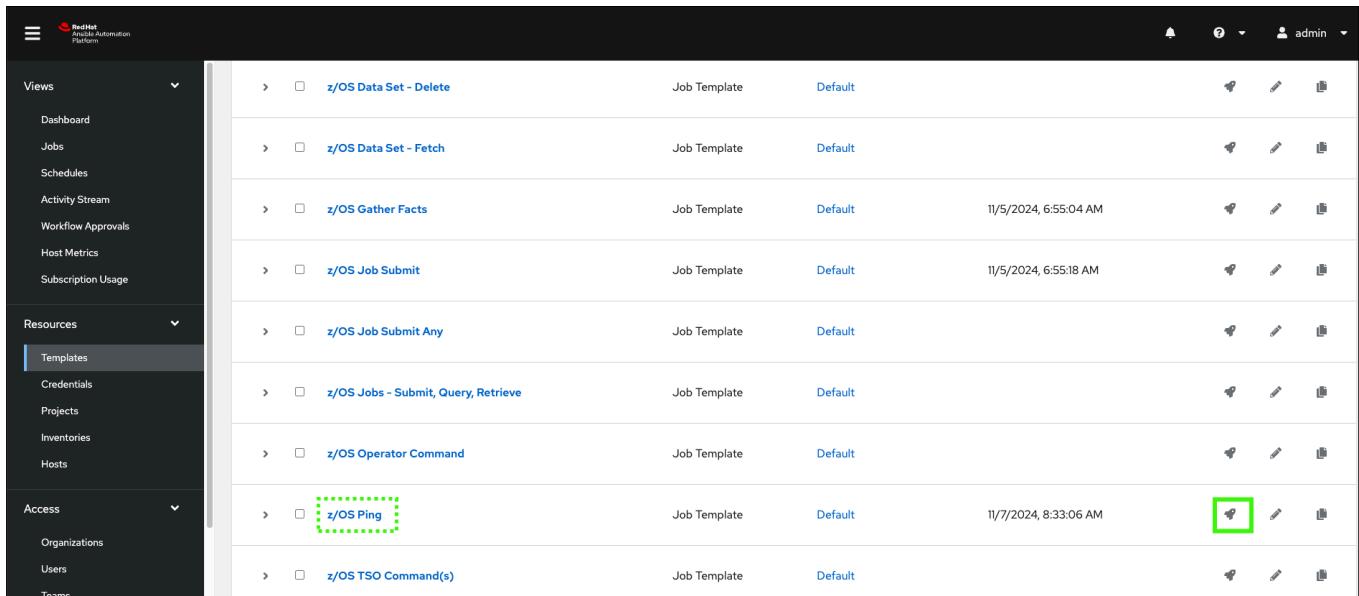
6. Click **Templates** under the **Resources** section.



The AAP instance is preconfigured to the Wazi aaS instance

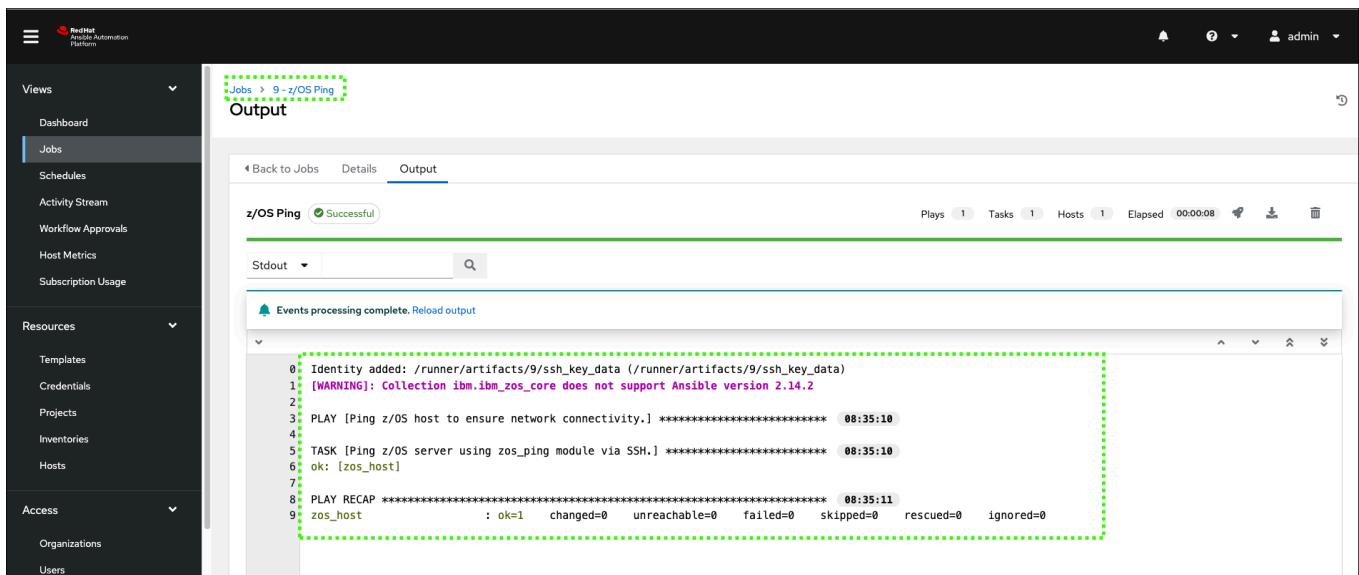
Note that because the AAP instance and the back-end z/OS system are preconfigured, no changes are needed to execute the templates and they will target your provisioned z/OS system by default.

7. Locate the **z/OS Ping** template and click the rocket (🚀) icon to start the template.



Views	z/OS Data Set - Delete	Job Template	Default			
Dashboard	z/OS Data Set - Fetch	Job Template	Default			
Schedules	z/OS Gather Facts	Job Template	Default	1/5/2024, 6:55:04 AM		
Activity Stream	z/OS Job Submit	Job Template	Default	1/5/2024, 6:55:18 AM		
Workflow Approvals	z/OS Job Submit Any	Job Template	Default			
Host Metrics	z/OS Jobs - Submit, Query, Retrieve	Job Template	Default			
Subscription Usage	z/OS Operator Command	Job Template	Default			
Resources	z/OS Ping	Job Template	Default	1/7/2024, 8:33:06 AM		
Access	z/OS TSO Command(s)	Job Template	Default			

8. Observe the z/OS Ping job run.



Jobs > 9 - z/OS Ping

Output

z/OS Ping Successful

Stdout

```

Events processing complete. Reload output
-----
0 Identity added: /runner/artifacts/9/ssh_key_data (/runner/artifacts/9/ssh_key_data)
1 [WARNING]: Collection ibm.ibm_zos_core does not support Ansible version 2.14.2
2
3 PLAY [Ping z/OS host to ensure network connectivity.] **** 08:35:10
4
5 TASK [Ping z/OS server using zos_ping module via SSH.] **** 08:35:10
6 ok: [zos_host]
7
8 PLAY RECAP **** 08:35:11
9 zos_host : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
-----
```

Take time to explore the other templates that are ready to use. Learn more about the automation capabilities [here](#).

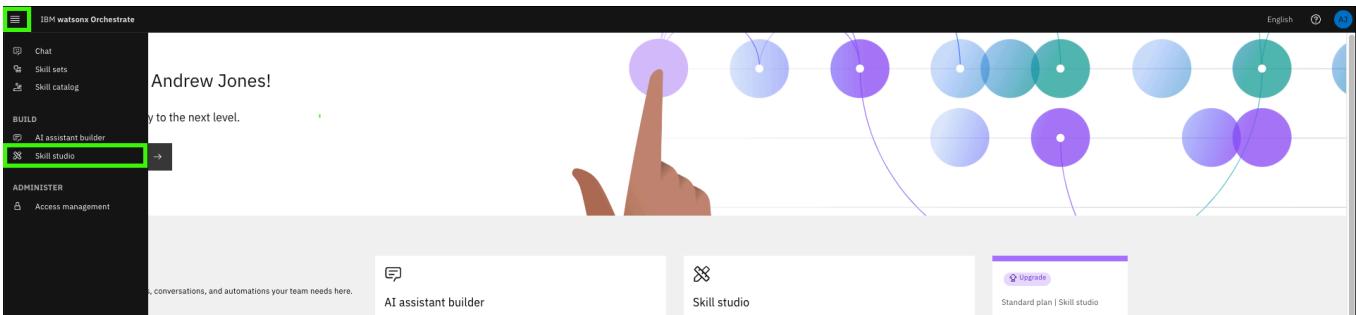
Importing skills from Ansible Automation Platform

Now that you understand Ansible Automation Platform (AAP) and the preinstalled automations available, you can import them as skills into your watsonx Orchestrate instance, which is used for assistant guided actions.

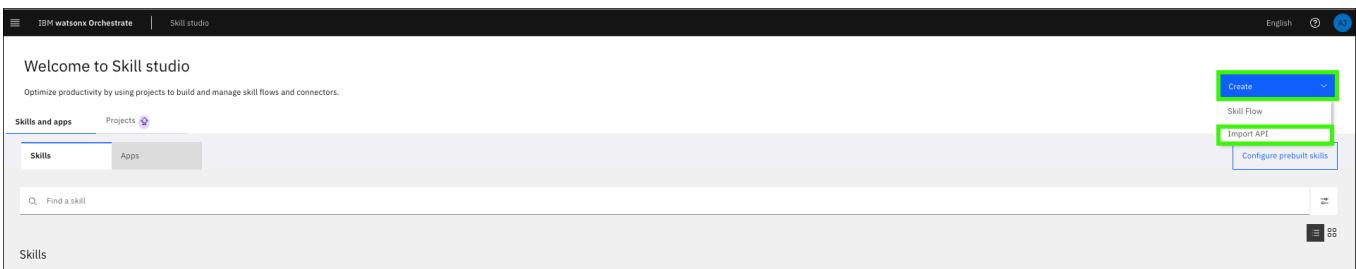
Import skills into your assistant

The next steps assume that you have an active browser window to the watsonx Orchestrate ITZ cloud account. If you do not, refer to the initial steps in [Creating an assistant and configuring conversational search](#).

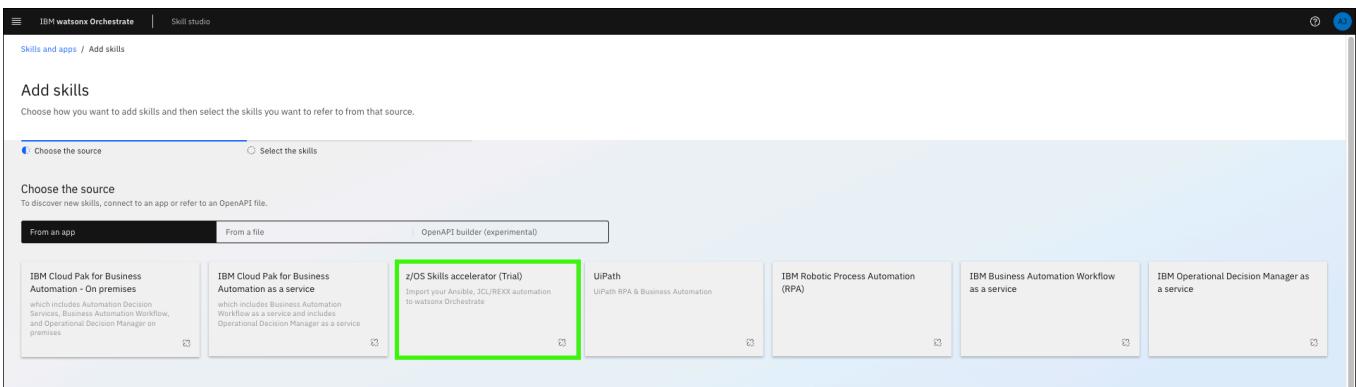
1. Return to your watsonx Orchestrate instance and expand the main menu and click **Skill studio**.



2. Expand **Create** and click **Import API**.



3. Click the **z/OS Skills accelerator (Trial)** tile.



4. Enter the following values in the **z/OS Skills accelerator** form and then click **Connect**.

Use the **URL**, **User Name**, and **Password** values recorded in the [Explore Ansible Automation Platform](#) section earlier.

a: Connection Type: ansible

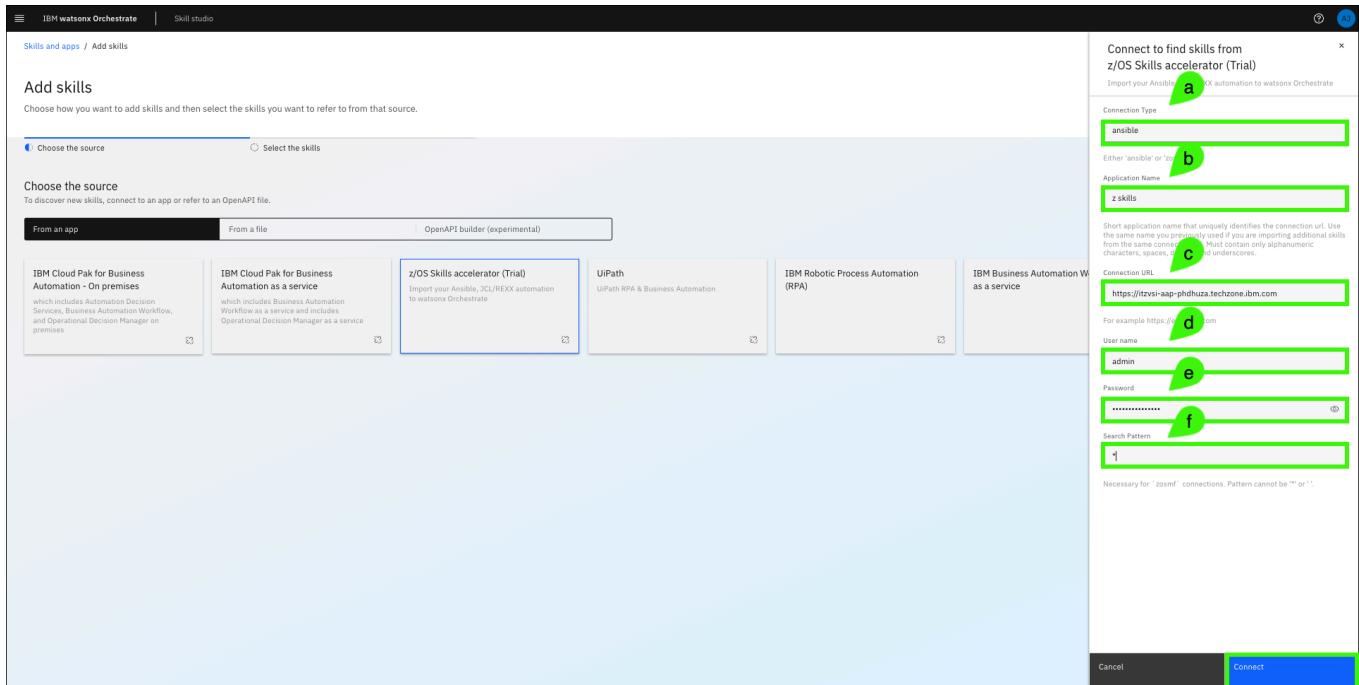
b: Application Name: <enter a meaningful name for the skills that you will import> - *be sure to remember this name, you will need in the next section*

c: Connection URL: <enter the URL for your AAP UI>

d: User Name: <enter the AAP User Name (for UI access)>

e: Password: <enter the AAP User Password>

f: Search Pattern: *

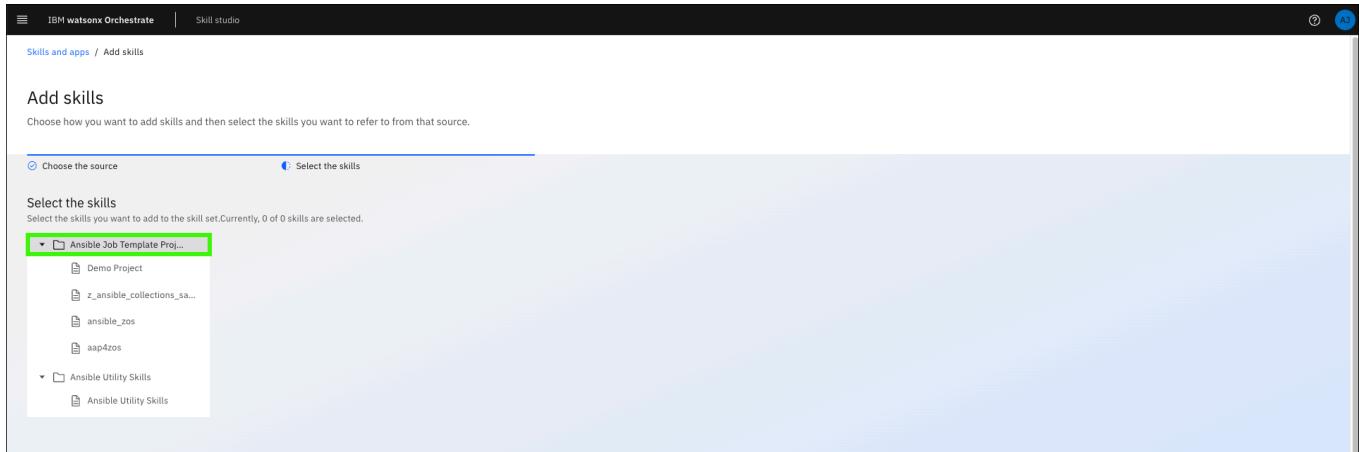


5. Expand the Ansible Job Template Proj... folder.



Explore the other available skills

Take time to explore the available skills. There are many utility skills provided out of the box with the "Z Skills Accelerator" which are needed for actions such as retrieving the output of an Ansible skill. Consider importing these utility skills to enable more complete automation execution flows.



6. Click **aap4zos**.

Skill	Description	Status
Z/os ping	Z skills - this playbook pings...	Ready to add
Z/os gather facts	Z skills - this sample playbook...	Ready to add
Z/os job submit	Z skills - this playbook shows...	Ready to add
Z/os job submit any	Z skills - a playbook for subm...	Ready to add
Z/os jobs - submit, query, ret...	Z skills - this sample playbook...	Ready to add

7. Select the skills you want to import into your application and then click **Save as draft**.

For this lab, select the **Z/os ping** and **Z/os gather facts** skills.

Skill	Description	Status
Z/os ping	Z skills - this playbook pings...	Ready to add
Z/os gather facts	Z skills - this sample playbook...	Ready to add
Z/os job submit	Z skills - this playbook shows...	Ready to add
Z/os job submit any	Z skills - a playbook for subm...	Ready to add
Z/os users - add	Z skills - this playbook shows...	Ready to add
Z/os data set - basics	Z skills - this sample playbook...	Ready to add
Z/os data set - create	Z skills - this playbook creat...	Ready to add
Z/os data set - delete	Z skills - this playbook delet...	Ready to add
Z/os data set - fetch	Z skills - this playbook fetch...	Ready to add
Z/os certs - health checker se...	Z skills - this playbook enabl...	Ready to add
Z/os certs - create keyring	Z skills - this playbook delet...	Ready to add
Z/os certs - delete keyring	Z skills - this playbook delet...	Ready to add
Z/os certs - create cert	Z skills - this playbook creat...	Ready to add
Z/os certs - delete cert	Z skills - this playbook delet...	Ready to add

8. Click the ellipses (⋮) for the first skill and select **Enhance this skill**.

Welcome to Skill studio

Optimize productivity by using projects to build and manage skill flows and connectors.

Skills and apps Projects

Skills Apps

Find a skill

Skills

Name	Step in the process	Status	Skill type	Author	Last edited
z/OS Gather Facts	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	November 19 2024
z/OS Ping	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	November 19 2024

Configure prebuilt skills

9. Review the skill enhancement options and then click **Publish**.

On the **Enhance this skill** page, you can specify enhancements to the default skill. Refer to this documentation for more information on enhancing skills.

Skills and apps / Enhance this skill

Enhance the "z/OS Gather Facts" skill

Add details that will make people want to use this skill.

Name Input Output Security Phrases Next best skills

Name and describe this skill in a way that tells users how it's used and why they would want to use it.

Name* z/OS Gather Facts

Description 0/100
z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS-specific information from the z/OS host.

Categories

App

Ansible Controller Skills - z skills

Preview

The skill will look like this in the catalog.

z/OS Gather Facts
z skills - This sample playbook demonstrates the z/OS gather facts...

The skill will look like this in the skill set.

z/OS Gather Facts

Cancel Publish Save as draft

10. Repeat steps 8 and 9 for each skill you imported.

Welcome to Skill studio

Optimize productivity by using projects to build and manage skill flows and connectors.

Skills and apps Projects

Skills Apps

Find a skill

Skills

Name	Step in the process	Status	Skill type	Author	Last edited
z/OS Gather Facts	Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
z/OS Ping	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	November 19 2024

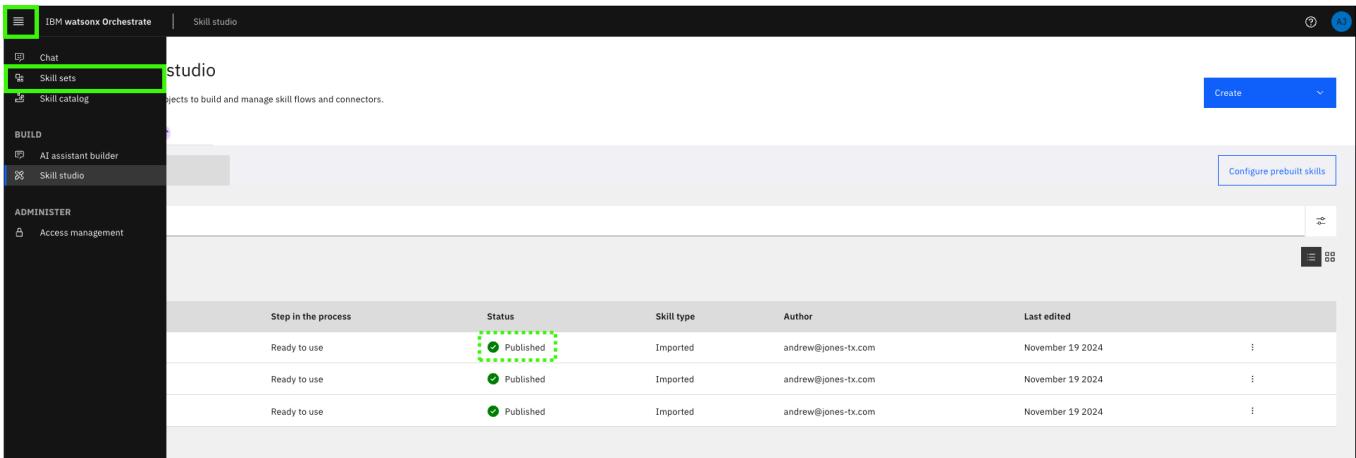
Configure prebuilt skills

The selected skills are now ready for use and available to your assistant. In the next section, learn how to connect them to your assistant.

Connecting skills to your assistant

Once you have a subset of skills published, the application you created can be connected to your assistant.

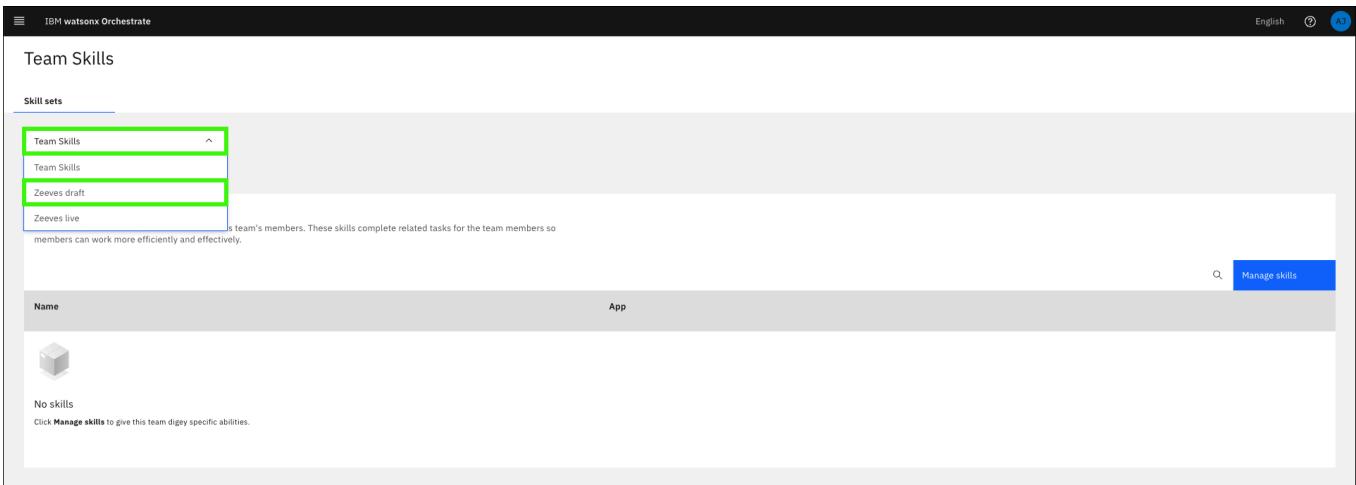
1. Expand the main menu and select **Skill sets**.



The screenshot shows the IBM Watsonx Orchestrate interface. The left sidebar has sections for Chat, Skill sets (which is highlighted with a green box), Skill catalog, BUILD (AI assistant builder, Skill studio), and ADMINISTER (Access management). The main area is titled "Skill studio" and contains a table of published skills:

Step in the process	Status	Skill type	Author	Last edited
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024

2. Click the **Team Skills** drop-down and select the **Draft** of your assistant.

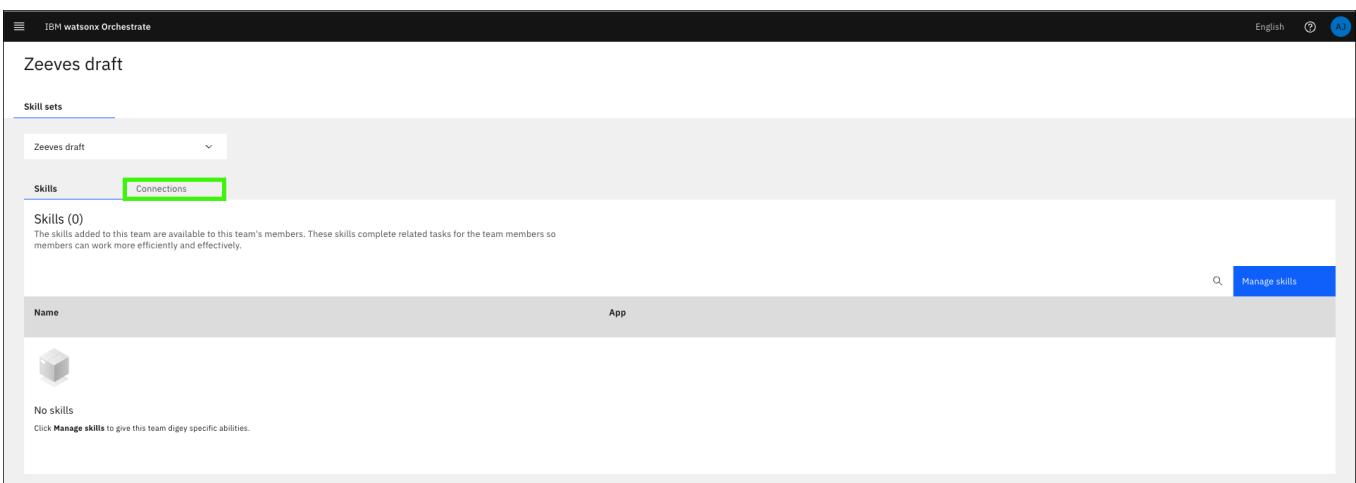


The screenshot shows the "Team Skills" page. The "Skill sets" dropdown menu is open, showing "Team Skills" and "Zeeves draft" (which is highlighted with a green box). Below the dropdown, there is a message about team skills. The main area shows a table with one row:

Name	App
No skills	

A note says "Click Manage skills to give this team digest specific abilities." There is a "Manage skills" button at the top right.

3. Click the **Connections** tab.



The screenshot shows the "Zeeves draft" page with the "Connections" tab selected (highlighted with a green box). The "Skills" tab is also visible. A note says "The skills added to this team are available to this team's members. These skills complete related tasks for the team members so members can work more efficiently and effectively." Below this is a table with one row:

Name	App
No skills	

A note says "Click Manage skills to give this team digest specific abilities." There is a "Manage skills" button at the top right.

4. Click the Search (🔍) icon.

Application	Number of skills	Credential type	Connected by ⓘ	Action
Activate or deactivate attracting candidates using ThisWay Global	4	⚠️ Not specified	-	⋮
Adobe Workfront	37	⚠️ Not specified	-	⋮
Alliance Virtual Office	2	⚠️ Not specified	-	⋮
Amazon S3	8	⚠️ Not specified	-	⋮
Amazon SES	10	⚠️ Not specified	-	⋮

Items per page: 5 | 1-5 of 78 items | 1 ⚏ of 16 pages | < >

5. Search for the application name you specified in the previous section.

Application	Number of skills	Credential type	Connected by ⓘ	Action
Ansible Controller Skills - z skills	2	⚠️ Not specified	-	⋮

Items per page: 5 | 1-1 of 1 items | 1 ⚏ of 1 page | < >

6. Click the ellipses (⋮) and click Connect app.

Application	Number of skills	Credential type	Connected by ⓘ	Action
Ansible Controller Skills - z skills	2	⚠️ Not specified	-	⋮

Items per page: 5 | 1-1 of 1 items | Connect app | 1 ⚏ of 1 page | < >

7. On the Connect to Ansible Controller Skills form, keep the defaults and click Connect app.

Zeeves draft

Skill sets

Skills Connections

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type
Ansible Controller Skills - z skills	2	Not specified

Items per page: 5 1-1 of 1 items

Connect to Ansible Controller Skills - z skills

Member credentials
Each team member uses their own credentials to connect to this app and use its skills.

Team credentials
The admin sets the credentials each team member uses to connect to this app and use its skills.

You selected **Team credentials** for the credential type. Click **Connect app** to provide the credentials your team will use and to connect to the app.

Connect app

8. Enter the **username (a)** and **password (b)** using the username (`admin`) and password for your IBM Technology Zone (ITZ) `watsonx Assistant for Z Pilot - AAP & z/OS` reservation, and then click **Connect app (c)**.

Zeeves draft

Skill sets

Skills Connections

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type
Ansible Controller Skills - z skills	2	Not specified

Items per page: 5 1-1 of 1 items

Connect to Ansible Controller Skills - z skills

a
username: admin
b
password: *****

If the service instance uses legacy credentials for authentication, provide the password for the specified username.

Cancel Connect app

The application is now connected to the draft version of your assistant.

Zeeves draft

Skill sets

Skills Connections

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	2	Team	andrew@jones-tx.com	⋮

Items per page: 5 1-1 of 1 items

Continue to the next section to create actions for your assistant.

Creating actions for your assistant

Once the skills in your application are connected to your assistant, you're ready to begin creating actions tied to those skills. Learn more about building actions [here](#)

Configure the number of input fields

Before configuring actions, it's important to modify a setting within watsonx Orchestrate that allows triggered skills to display as forms (versus conversational skills).

1. Click your (a) profile icon and then click (b) **Settings**

Learn more about configuring input fields [here](#).

The screenshot shows the IBM Watsonx Orchestrate web interface. At the top, there's a navigation bar with a profile icon (labeled 'a') and a search bar. Below the bar, the title 'Zeeves draft' is displayed. On the left, there's a sidebar with 'Skill sets' and a dropdown menu. The main content area shows a table of skills under the 'Skills' tab. One row is selected, showing 'Ansible Controller Skills - z skills' with 2 skills, connected by 'Team' with user 'andrew@jones-tx.com'. On the right, a sidebar menu is open, with 'Settings' highlighted (labeled 'b'). Other options in the sidebar include 'Provide feedback', 'Privacy', 'About', and 'Log out'.

2. Click the **Skill configurations** tab.

This screenshot shows the 'Settings' page in the IBM Watsonx Orchestrate interface. The 'Skill configurations' tab is active, highlighted with a green box. The 'Data Retention' tab is also present. Under 'Skill configurations', there's a section titled 'Data Retention Policy' with a radio button for 'Active'. A note states: 'The chat history for the users in the tenant are saved for 30 days. After 30 days, the chat history is deleted permanently. Deleting the chat history doesn't affect the connected apps or the added skills.'

3. Enter **0** for the **Number of form fields**.

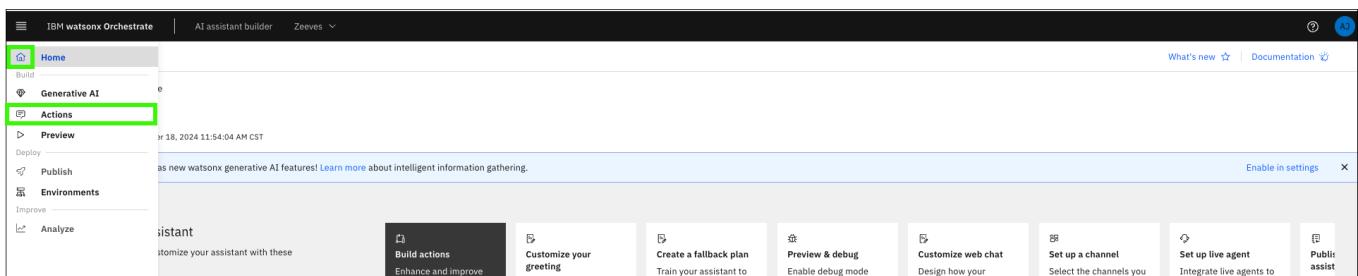
This screenshot shows the 'Skill configurations' page in the IBM Watsonx Orchestrate interface. The 'Skill configurations' tab is active. In the center, there's a section titled 'Set default for maximum number of fields to consider a skill to be conversational'. Below it is an input field labeled 'Number of form fields' containing the value '0', which is highlighted with a green box. There are minus and plus buttons next to the input field.

Create actions

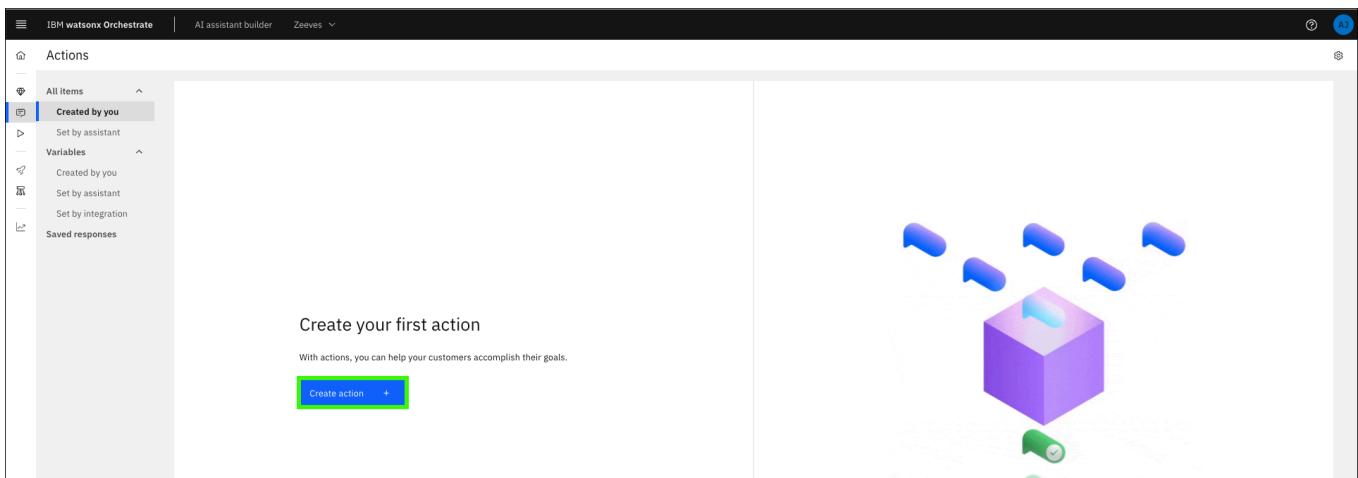
1. Click the main menu and select **AI assistant builder**.



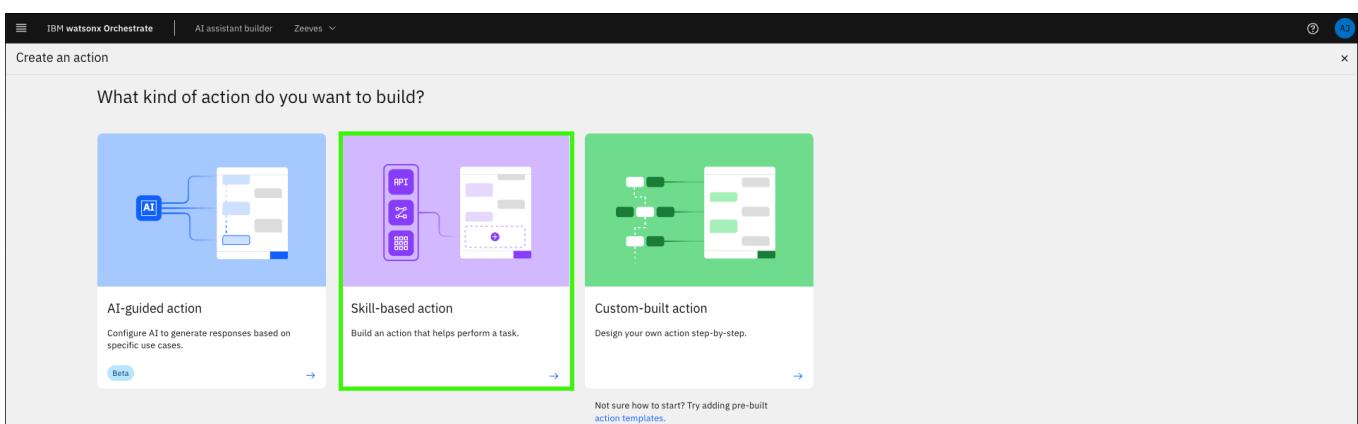
2. Hover over the **Home** icon () and click **Actions**.



3. Click **Create action**.



4. Click the **Skill-based action** tile.



5. Select the **z/OS Gather Facts** tile and click **Next**.

Note, it may take a minute for the page to display the action tiles. The date shown in the **z/OS Gather Facts** tile reflects when you added the skill to your application.

Build an action from a skill

Select a skill
Choose a conversational skill published as a foundation of your action.

Search a skill

z/OS Ping z skills - This playbook pings the z/OS host to test connectivity. Last updated: 2024-11-19T15:58:20.567Z	z/OS Gather Facts z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS-specific information from the z/OS host. Last updated: 2024-11-19T15:56:26.843Z	Summarize the Webex meeting transcript in watsonx.ai Last updated: 2024-11-04T10:49:16.502Z	Summarize the Box content in watsonx.ai Last updated: 2024-11-04T10:49:12.077Z	Summarize a Zendesk ticket in watsonx.ai Last updated: 2024-11-04T10:49:09.476Z
Summarize a ServiceNow incident in watsonx.ai Last updated: 2024-11-04T10:49:05.828Z	Summarize a Salesforce opportunity in watsonx.ai Last updated: 2024-11-04T10:49:01.769Z	Sharepoint document summary in watsonx.ai Last updated: 2024-11-04T10:48:55.707Z	Salesloft email summary in watsonx.ai Last updated: 2024-11-04T10:48:51.522Z	Salesforce case summarization in watsonx.ai Last updated: 2024-11-04T10:48:48.195Z
Salesforce case sentiment analyze	Outlook email summary	Github issue summarization	Github issue sentiment	Generate an email

- On the **New action** dialog, (a) enter a prompt a user of the assistant might use to initiate the action and then (b) click **Save**.

Be careful with the sample phrases you specify.

During the development of the lab guide, it was discovered that some sample phrases with a `/` character can cause issues with the actions. Avoid using **z/OS** in the your sample phrases. This issue has been reported to the offering team.

Sample prompts:

Get zOS facts

Gather zOS facts

Customer starts with:
Example: I want to pay my credit card bill.

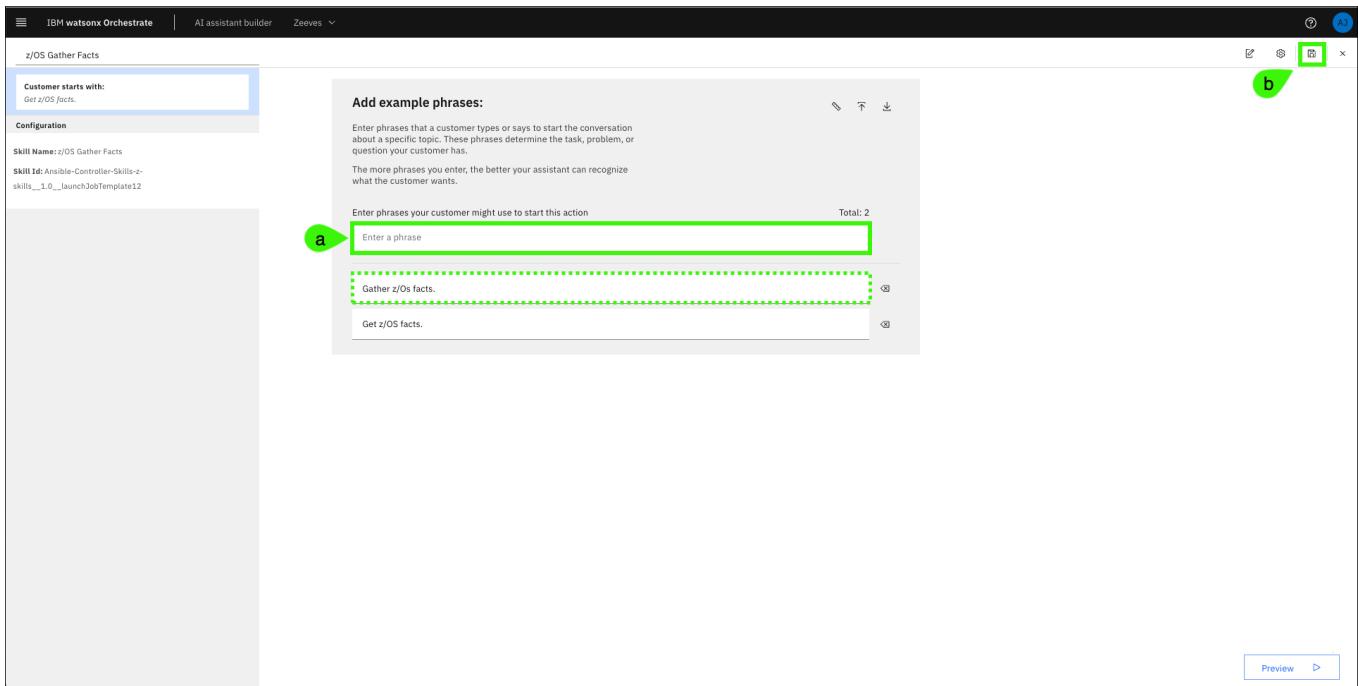
Add example phrases:
Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.
The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action
Total: 0
Example: I want to pay my credit card bill.

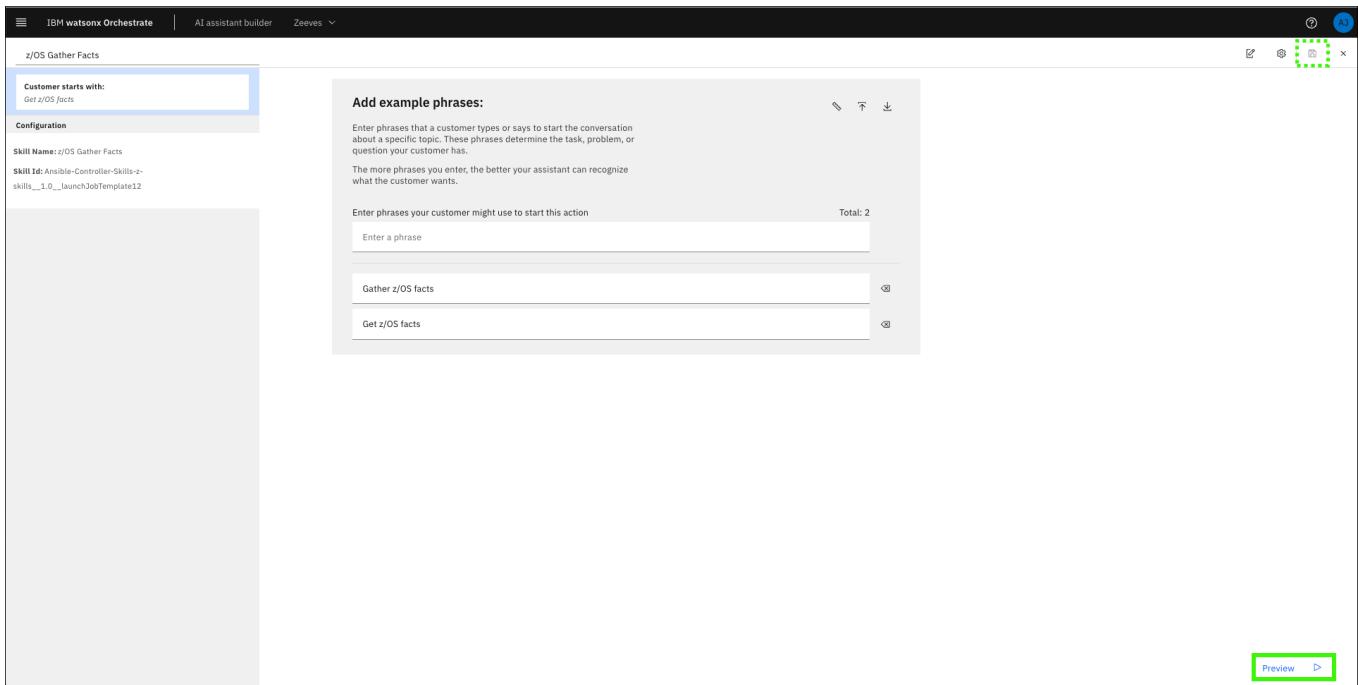
New action
What does your customer say to start this interaction?
Get z/OS facts.

Cancel Save

- Add any (a) additional prompts and then (b) click the save ().



8. Click Preview.



9. Enter one of the prompts you specified in step 9 or 10.

Prompt:

Get zOS facts

Customer starts with:
Get z/OS facts

Add example phrases:

Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.

The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 2

Enter a phrase

Gather z/OS facts
Get z/OS facts

Preview

12:55 PM
Greet customer [default]
Welcome, how can I assist you?

Get z/OS facts

10. Review the returned results and record the **job** number.

In the execution of this skill-based action, the skill executed properly and the output is the job id.



If an error is generated, review the Troubleshooting section below.

Customer starts with:
Get z/OS facts

Add example phrases:

Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.

The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 2

Enter a phrase

Gather z/OS facts
Get z/OS facts

Preview

12:55 PM
Greet customer [default]
Welcome, how can I assist you?

You 1:04 PM
Conversational skill called
z/OS Gather Facts recognized
[Job ID: 12]
status : pending

There are no additional steps for this action. Add a new step or end the action.

Verify the job in the Ansible Automation Platform console

Return to the Ansible Automation Platform (AAP) console and review the job information.

1. Click **Jobs** and then click the **job** number recorded in the previous step for the **z/OS Gather Facts** skill.

The screenshot shows the 'Jobs' page in the Red Hat Ansible Automation Platform. The left sidebar has a 'Jobs' menu item highlighted with a green box. In the main table, a row for '12 - z/OS Gather Facts' is highlighted with a green box. The table columns include Name, Status (Successful), Type (Playbook Run), Start Time (12/5/2024, 7:17:32 AM), Finish Time (12/5/2024, 7:17:41 AM), and Actions.

2. Review both the **Details** and **Output** for the **z/OS Gather Facts** job.

Recall, that in the assistant, the contents shown in the **Output** of the Ansible job were not displayed.

The screenshot shows the 'Output' tab for job '12 - z/OS Gather Facts'. The output content is a Ansible playbook output showing facts gathered from a z/OS host. A large green dotted box highlights the entire output section.

```

25 }
26 TASK [Print out all gathered facts about the z/OS host.] **** 07:17:38
27 ok: [zos_host] => {
28     "ansible_facts": {
29         "arch_level": "2",
30         "cpc_nd_manufacturer": "IBM",
31         "cpc_nd_model": "A00",
32         "cpc_nd_plant": "C1",
33         "cpc_nd_seqno": "237701828347",
34         "cpc_nd_type": "008562",
35         "edt": "00",
36         "hw_name": "...",
37         "ieasym_card": "(00,K2)",
38         "io_config_id": "00",
39         "iodate": "...",
40         "iodesc": "...",
41         "iodf_config": "DEFAULT",
42         "iodf_name": "PROV.IODF00",
43         "iodf_unit_addr": "DE28",
44         "ioproc": "...",
45         "iotime": "...",
46

```

IBM watsonx Assistant for Z provides utility skills to retrieve the job output. It is also possible to create a skill flow that executes the **z/OS Gather Facts** skill followed by the **Retrieve job output** utility skill in sequence; passing the job id from the first skill to the second, in order to view the output within the assistant. Creating a skill flow is covered in the next section.

Troubleshooting

Skill returns "Sorry, we're having issues generating a response".

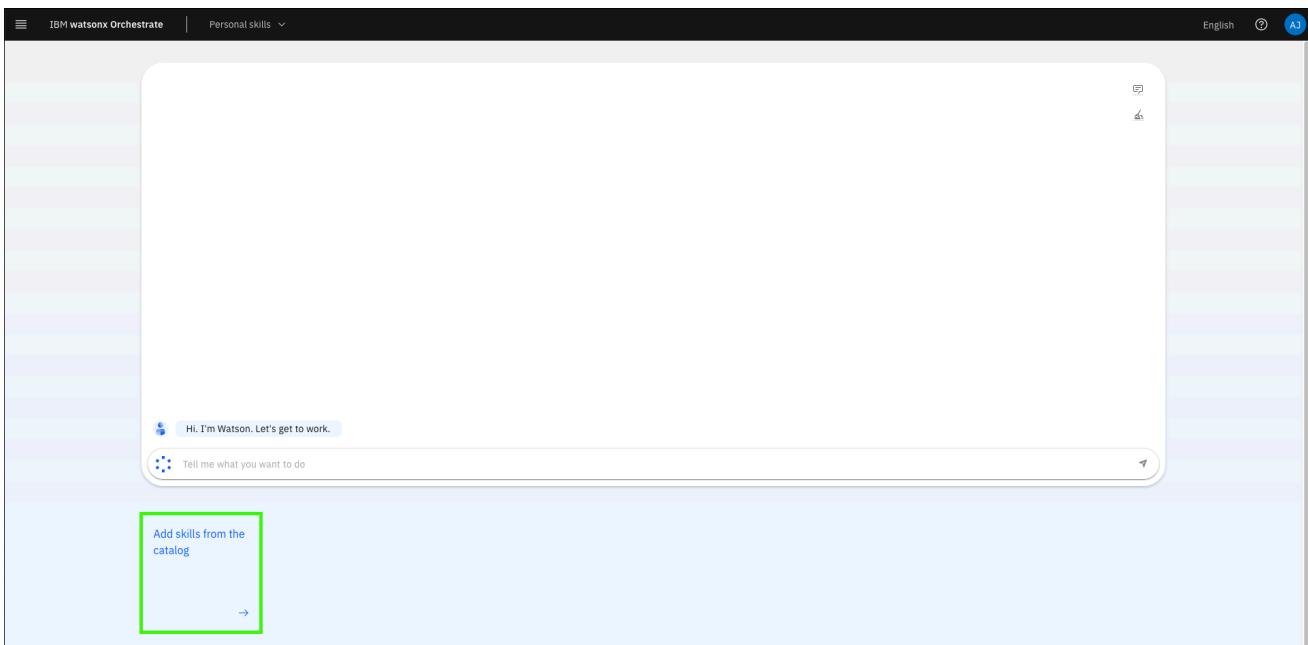
The screenshot shows the IBM Watsonx Assistant interface. On the left, there's a sidebar with options like 'IBM Watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main area is titled 'z/OS Gather Facts' and contains a 'Configuration' section with 'Skill Name: z/OS Gather Facts' and 'Skill Id: Ansible-Controller-Skills-z-skills__1.0__launchJobTemplate12'. Below this is a 'Customer starts with:' field containing 'Get z/OS facts'. A 'Add example phrases:' section has two entries: 'Gather z/OS facts' and 'Get z/OS facts'. To the right, a 'Preview' window shows a conversation: 'Welcome, how can I assist you?' followed by 'You 10:46 AM Gather z/OS facts'. The response is '10:46 AM Conversational skill called z/OS Gather Facts recognized'. Below this, an error message is shown in a red box: 'Sorry, we're having issues generating a response.' This message is highlighted with a green dashed box. Below the error message are two buttons: 'Error in the response' (Details) and 'Skill error' (Inspect). A note at the bottom says 'There are no additional steps for this action. Add a new step or end the action.'

This error appears to be an intermittent issue when a skill is first added. To resolve, add the skill to your personal skills catalog using the steps that follow. If you encounter the issue, try the steps that follow:

1. Expand the main menu and select Chat.

The screenshot shows the main menu of the IBM Watsonx Assistant interface. The 'Chat' option is highlighted with a green box. Other menu items include 'Skill sets', 'Skill catalog', 'BUILD' (with 'AI assistant builder' selected), and 'ADMINISTER' (with 'Access management' selected).

2. Click Add skills from the catalog.



3. Search for the skill app you created earlier and click the tile for your app.

The screenshot shows the 'Skill catalog' page. At the top, it says "Skills are grouped by app. Select an app to see all the skills that use that app." Below is a search bar with the text "z skills". The main area is titled "Apps" and contains a grid of skill app tiles. One tile, "Ansible Controller Skills - z skills", is highlighted with a green box.

4. Click Add skill for all the skills you want to add.

The screenshot shows the details for the "Ansible Controller Skills - z skills" app. It lists two skills: "z/OS Gather Facts" and "z/OS Ping". Each skill card has an "Add skill +" button at the bottom, which is highlighted with a green box.

5. Click Connect app.

Ansible Controller Skills - z skills (2)

z/OS Gather Facts
z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS...
z/OS Ping
z skills - This playbook pings the z/OS host to test connectivity.

Connect app

6. Enter the (a) **username** and (b) **password** using the username (admin) and password for your IBM Technology Zone (ITZ) watsonx Assistant for Z Pilot - AAP & z/OS reservation (AAP User Password (Use SSH key to login, only use password for UI)), and then click **Connect app**.

Connect to Ansible Controller Skills - z skills

username: admin
password: *****

Cancel **Connect app**

7. Expand the main menu and select **Chat**.

IBM Watsonx Orchestrate

Chat

Skills - z skills

z/OS Ping
z skills - This playbook pings the z/OS host to test connectivity.

Connected

8. Try one of the prompts you created for your skill.

Prompt:

Gather zOS facts

The screenshot shows the IBM Watsonx Assistant interface. At the top, it says "IBM Watsonx Orchestrate" and "Personal skills". On the right, there are icons for English, a refresh symbol, and a blue "A3" button. The main area has a light gray background. A message from Watson says: "Hi, I'm Watson. Check out the skills in the catalog to see how I can help you." Below this, a green button labeled "Gather z/OS facts" is highlighted. A green dashed box encloses the results of this skill: "Here are the results of the z/OS Gather Facts skill." It shows a card for "z/OS Gather Facts" with status "pending" and a progress bar for a "job" labeled "12" with a minus sign and a plus sign. Below this is a blue input field with the placeholder "Tell me what you want to do". At the bottom left, a blue button says "Add skills from the catalog" with an arrow pointing right. To its right, another green dashed box encloses a section titled "Ansible Controller Skills - z skills" with a small "2 skills" indicator.

You should now be able to run the skill through the assistant preview.

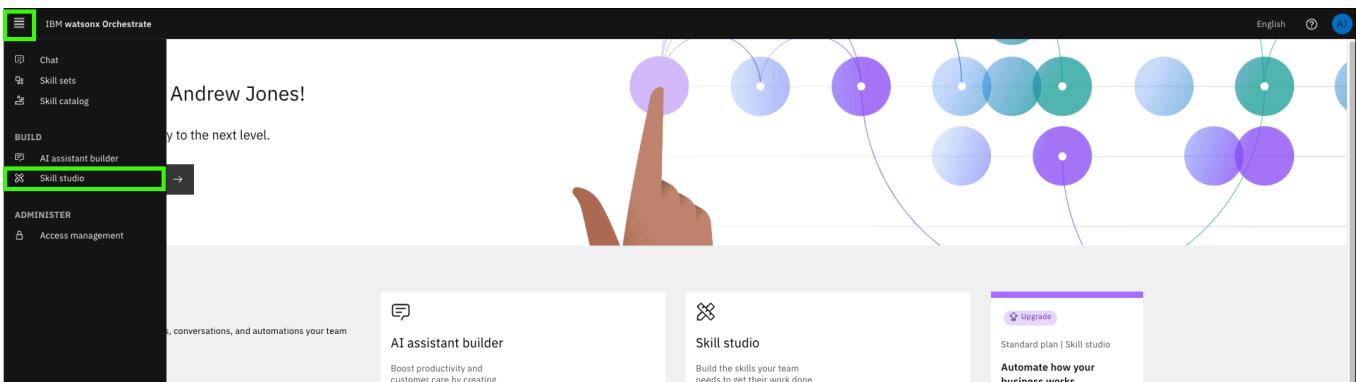
Creating skill flows

As seen in the previous section, running the Ansible skill to **Gather z/OS facts**, the skill executed successfully and was verified within the Ansible Automation Platform (AAP) console by viewing the job output. However, the output wasn't displayed by the assistant. To enable this scenario, a skill flow is needed. Skills are often more valuable when combined with other skills. You can create a skill flow to use two or more skills together to finish a task (like returning the output of a previous skill). When you create a skill flow, you map the output of one skill as the input for subsequent skills. Learn more about creating skill flows [here](#).

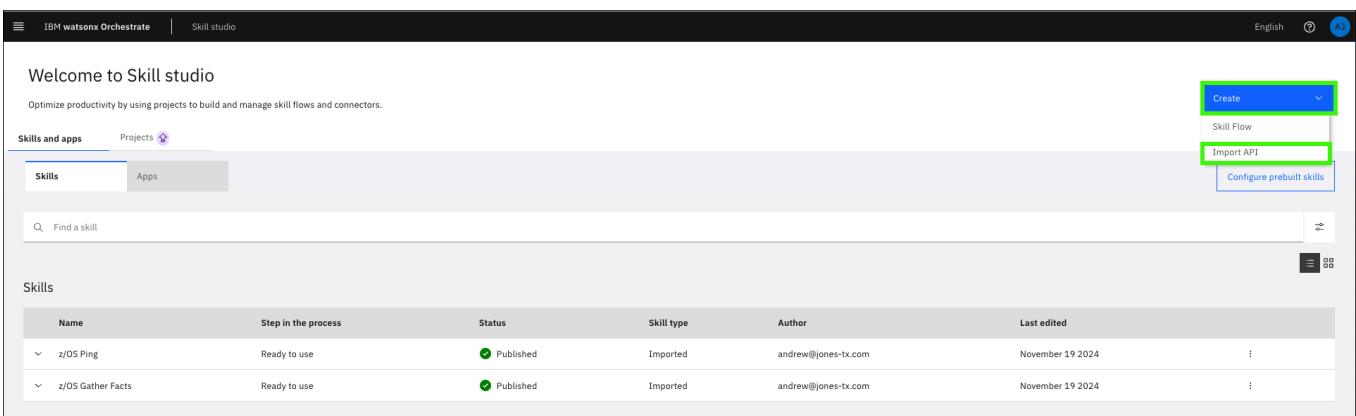
As mentioned in a previous section, there are some default utility skills that are provided out of the box with the "Z Skills Accelerator" which are leveraged to return the output of a skill. To accomplish this, we will import the Ansible Utility skill called **Retrieve job output**.

Add the utility skill

1. Open IBM watsonx Orchestrate **Skill studio**.



2. Expand **Create** and click **Import API**.



3. Click the **z/OS Skills accelerator (Trial)** tile.

The screenshot shows the 'Skill studio' interface of IBM Watsonx Orchestrate. In the top left, it says 'Skills and apps / Add skills'. Below that is a section titled 'Add skills' with the sub-instruction 'Choose how you want to add skills and then select the skills you want to refer to from that source.' There are two tabs: 'Choose the source' (selected) and 'Select the skills'. Under 'Choose the source', there are three options: 'From an app', 'From a file', and 'OpenAPI builder (experimental)'. A green box highlights the 'z/OS Skills accelerator (Trial)' option, which is described as 'Import your Ansible, JCL/REXX automation to Watsonx Orchestrate'. Other options shown include 'IBM Cloud Pak for Business Automation - On premises', 'IBM Cloud Pak for Business Automation as a service', 'UiPath RPA & Business Automation', 'IBM Robotic Process Automation (RPA)', and 'IBM Business Automation Workflow as a service'. Below these options is a box labeled 'IBM Operational Decision Manager as a service'.

4. Enter the following values in the **z/OS Skills accelerator** form and then click **Connect**.

Use the **URL**, **User Name**, and **Password** values recorded in the **Explore Ansible Automation Platform** section earlier.

a: Connection Type: `ansible`

b: Application Name: <use the same application name as in previous section>

c: Connection URL: <enter the URL for your AAP UI>

d: User Name: <enter the AAP User Name (for UI access)>

e: Password: <enter the AAP User Password>

f: Search Pattern: `*`

The screenshot shows the configuration dialog for the 'z/OS Skills accelerator (Trial)' skill. On the right, there's a sidebar with instructions: 'Connect to find skills from z/OS Skills accelerator (Trial)', 'Import your Ansible, JCL/REXX automation to Watsonx Orchestrate', 'Connection Type: ansible', 'Application Name: z', '2 skills', 'Connection URL: https://itzvi-aap-phdhuzza.techzone.ibm.com', 'User name: admin', 'Password: [REDACTED]', and 'Search Pattern: *'. The 'Connection URL' field has a note: 'Short application name that uniquely identifies the connection or. Use the same name you previously used if you are importing additional skills from the same connection. Must contain only alphanumeric characters, spaces, and underscores.' The 'Search Pattern' field has a note: 'Necessary for "zosml" connections. Pattern cannot be "*" or "+"'. At the bottom are 'Cancel' and 'Connect' buttons, with 'Connect' highlighted in blue.

5. Expand **Ansible Utility Skills** and click **Ansible Utility Skills**.

6. Select **Retrieve job output** and click **Save as draft**.

7. Click the ellipses (...) for the **Retrieve job output** skill and select **Enhance this skill**.

8. Review the skill settings and then click **Publish**.

Name: Retrieve job output

Description: z skills - Retrieve job output by job Id

API version*: 1.0

Categories: Add categories

App: Ansible Controller Skills - z skills

Preview

The skill will look like this in the catalog.

Retrieve job output
z skills - Retrieve job output by job Id

The skill will look like this in the skill set.

Retrieve job output

Cancel Publish Save as draft

9. Select Skill sets from the main menu.

Skill sets

Step in the process	Status	Skill type	Author	Last edited
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024

10. Select (a) your draft assistant in the Team Skills drop-down list and (b) click the Connections tab.

Zeeves draft

Skills **Connections** **a** **b**

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type	Connected by	Action
Activate or deactivate attracting candidates using ThisWay Global	4	Not specified	-	⋮
Adobe Workfront	37	Not specified	-	⋮
Alliance Virtual Office	2	Not specified	-	⋮
Amazon S3	8	Not specified	-	⋮
Amazon SES	10	Not specified	-	⋮

Items per page: 5 | 1-5 of 78 items

11. Click the **Search (🔍)** icon.

Application	Number of skills	Credential type	Connected by ⓘ	Action
Activate or deactivate attracting candidates using ThisWay Global	4	⚠️ Not specified	-	⋮
Adobe Workfront	37	⚠️ Not specified	-	⋮
Alliance Virtual Office	2	⚠️ Not specified	-	⋮
Amazon S3	8	⚠️ Not specified	-	⋮
Amazon SES	10	⚠️ Not specified	-	⋮

Items per page: 5 | 1-5 of 78 items | 1 ⚏ of 16 pages | ⏪ ⏴ ⏵ ⏶

12. Search for the application name you specified earlier.

Application	Number of skills	Credential type	Connected by ⓘ	Action
Ansible Controller Skills - z skills	2	⚠️ Not specified	-	⋮

Items per page: 5 | 1-1 of 1 items | 1 ⚏ of 1 page | ⏪ ⏴ ⏵ ⏶

13. Click the (a) ellipses (⋮) for your application and (b) click **Edit connection**.

Application	Number of skills	Credential type	Connected by ⓘ	Action
Ansible Controller Skills - z skills	3	⚠️ Team	andrew@jones-tx.com	a ⋮ b Edit connection Delete connection

Items per page: 5 | 1-1 of 1 items | 1 ⚏ of 1 page | ⏪ ⏴ ⏵ ⏶

14. Verify the application is **Connected (a)** and then click **Close (b)**.

Zeeves draft

Skill sets

Skills Connections

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type
Ansible Controller Skills - z skills	3	Team

Items per page: 5 | 1-1 of 1 items

Edit the Ansible Controller Skills - z skills connection **b**

Member credentials
Each team member uses their own credentials to connect to this app and use its skills.

Team credentials (Active)
The admin sets the credentials each team member uses to connect to this app and use its skills.

You have an active connection set using **Team credentials**. If you wish to update connection details click on the **Edit** **a** below.

Connected **a**

Add the skills to your Personal skills

1. Click **Skill catalog** in the main menu.

Chat

Skill sets

Skill catalog **a**

AI assistant builder

Skill studio

Administrations

Access management

Number of skills	Credential type	Connected by a	Action
3	Team	andrew@jones-tx.com	b

1 of 1 items

2. Search for the application name you specified earlier.

Skill catalog

Skills are grouped by app. Select an app to see all the skills that use that app.

Personal skills

z skills

Most popular skills

Send an email from Gmail	Create a lead in Salesforce	Send an email using Outlook
--------------------------	-----------------------------	-----------------------------

All Apps

Coupa	ZoomInfo	HubSpot CRM	Aptio Targetprocess	Salesforce	Zendesk Service
Calendly	Square	Oracle E-Business Suite	GitLab	Toggl Track	Microsoft Teams

3. Click the tile for your application.

Note, the tile name is proceeded by **Ansible Controller Skills**.

Skill catalog
Skills are grouped by app. Select an app to see all the skills that use that app.

Personal skills

Q z skills X

Ansible Controller Skills - z skills (3)

Ansible Controller Skills - z skills (3) (highlighted)

ZoomInfo 26 skills

Workday HCM 36 skills

FreshService 25 skills

Seismic 22 skills

Reveal your existing applic... 7 skills

Interview top candidates u... 2 skills

Salesforce Chatter 5 skills

IBM Process Mining 2 skills

Webex 14 skills

Adobe Workfront 37 skills

Skill flows 167 skills

Cognos 8 skills

Salessoft 47 skills

- Click **Add skill** for each of the skills you want to add to the flow.

Skill catalog / Ansible Controller Skills - z skills (3)

Connect app ⚙

Personal skills

Q Search skills

Ansible Controller Skills - z skills

Retrieve job output
z skills - Retrieve job output by job Id

Add skill +

z/OS Gather Facts
z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls...

Add skill +

z/OS Ping
z skills - This playbook pings the z/OS host to test connectivity.

Add skill +

Create the skill flow

- Click **Skill studio** in the main menu.

IBM Watsonx Orchestrate

English ⚙

Chat

Skill sets

Skill catalog

BUILD

AI assistant builder

Skill studio (highlighted)

ADMINISTER

Access management

Skills - z skills (3)

Connect app ⚙

z skills

z/OS Gather Facts
z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls...

Added ✓

z/OS Ping
z skills - This playbook pings the z/OS host to test connectivity.

Added ✓

Add skill +

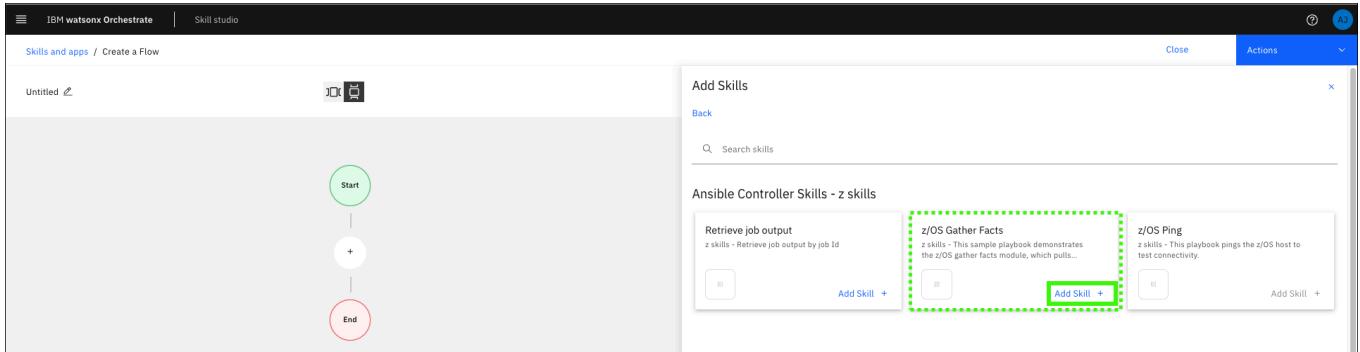
- Expand the **Create** drop-down menu and click on **Skill flow**.

3. Click the + icon.

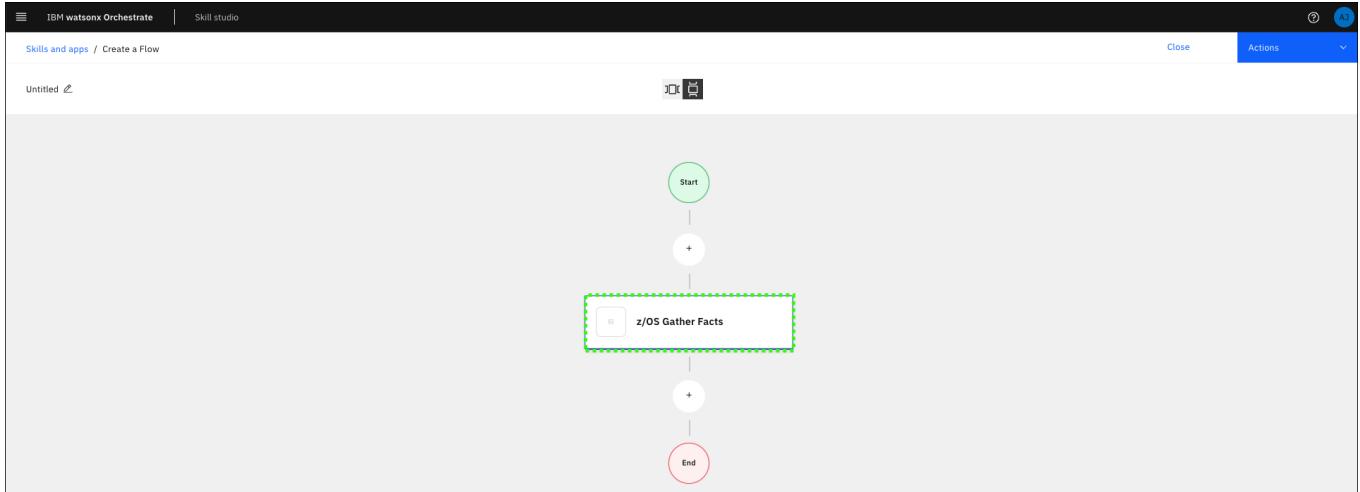
Next, you need to add the [z/OS Gather Facts](#) skill and the [Retrieve job output](#) skill to the skill flow. Use the **Search apps** function to locate the skills.

4. Search for the application name you specified earlier and click its tile.

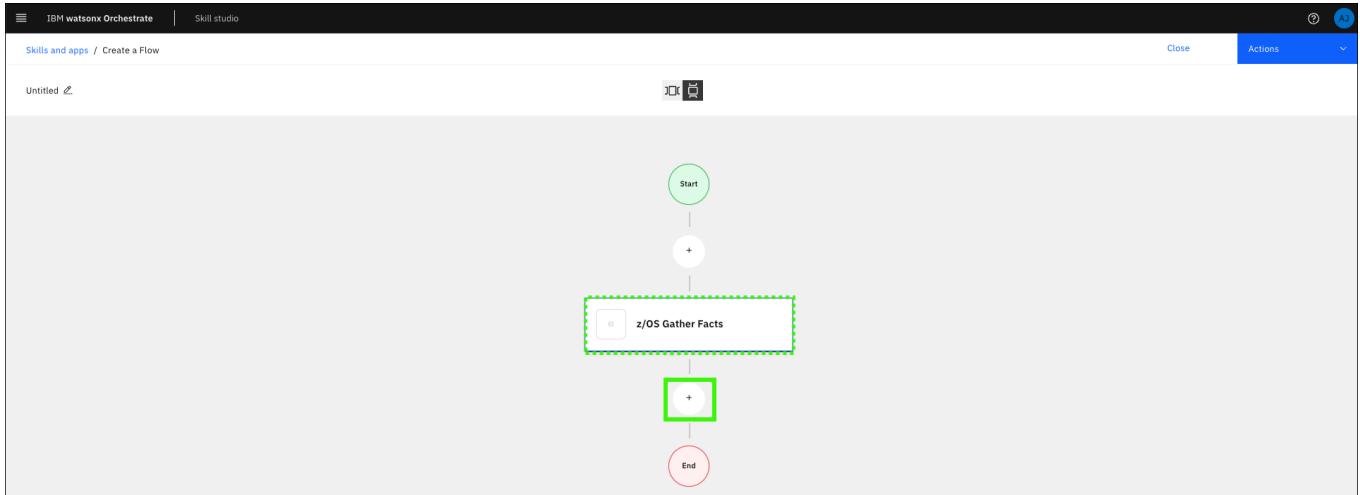
5. Click **Add Skill** in the [z/OS Gather Facts](#) tile.



6. Verify the **z/OS Gather Facts** skill is added to the skill flow.

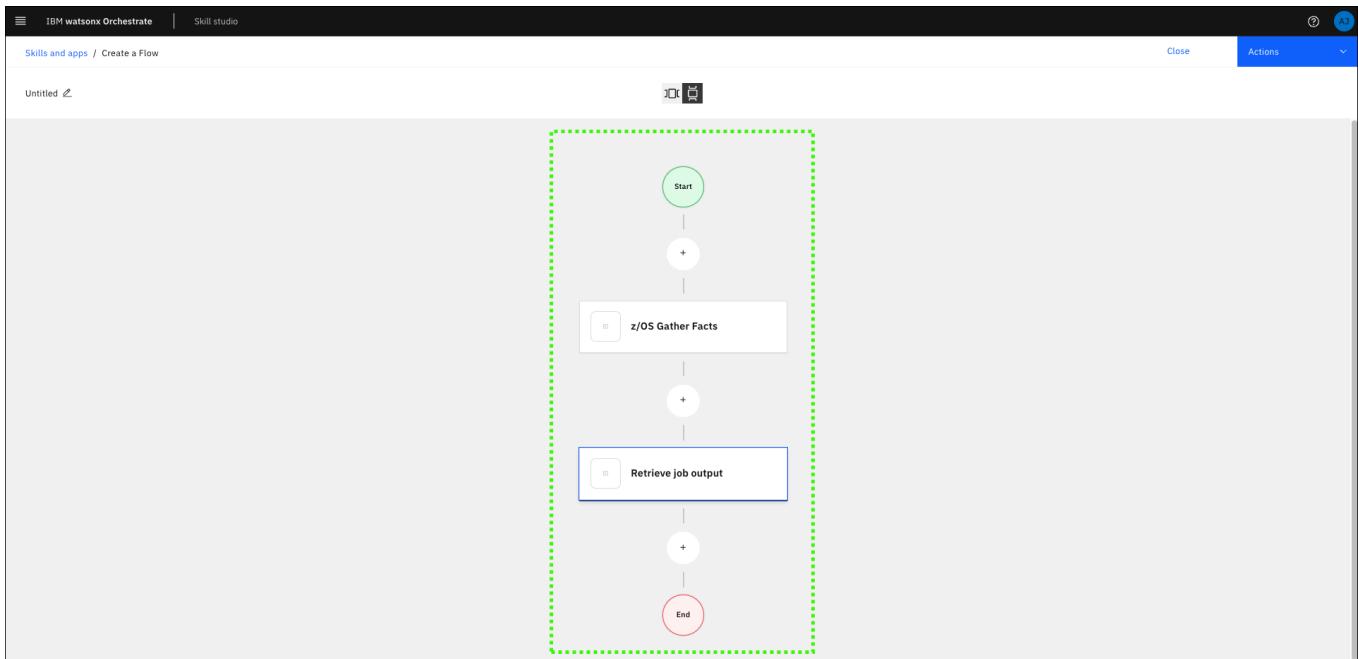


7. Click the **+** icon after the **z/OS Gather Facts** tile.



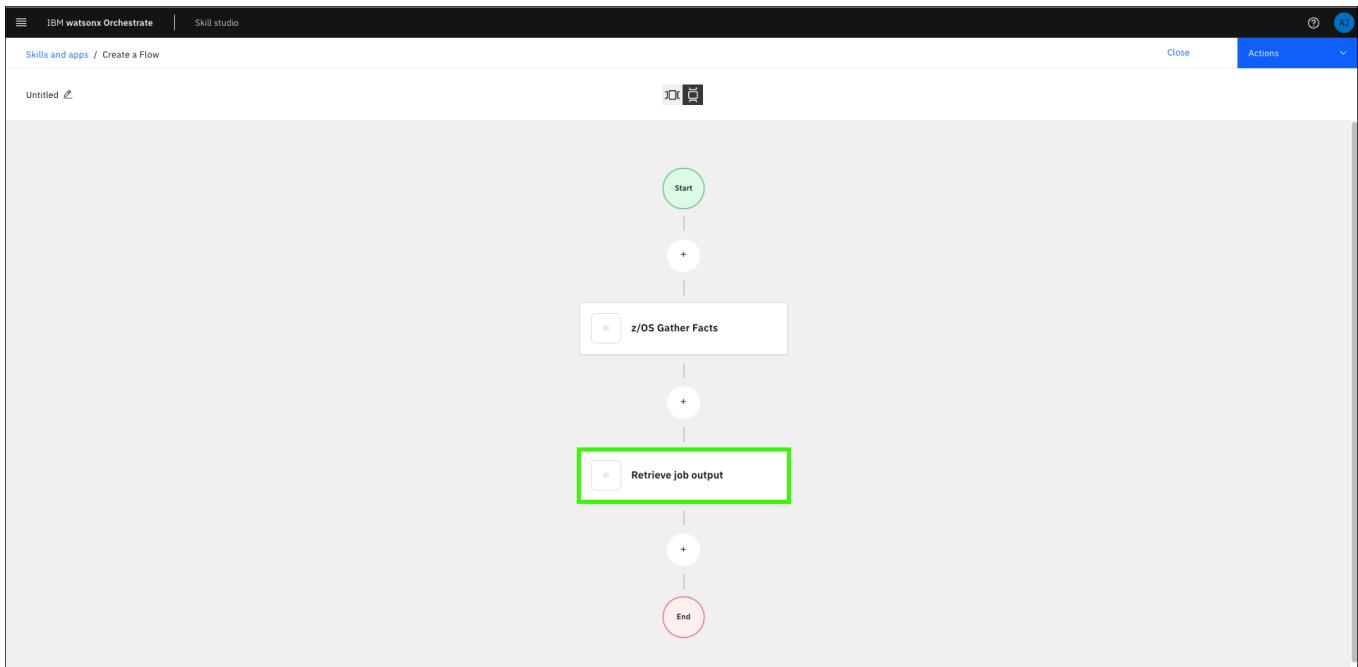
8. Repeat steps 5 and 6 for the **Retrieve job output** skill.

After adding the **Retrieve job output** skill, your skill flow should look like:

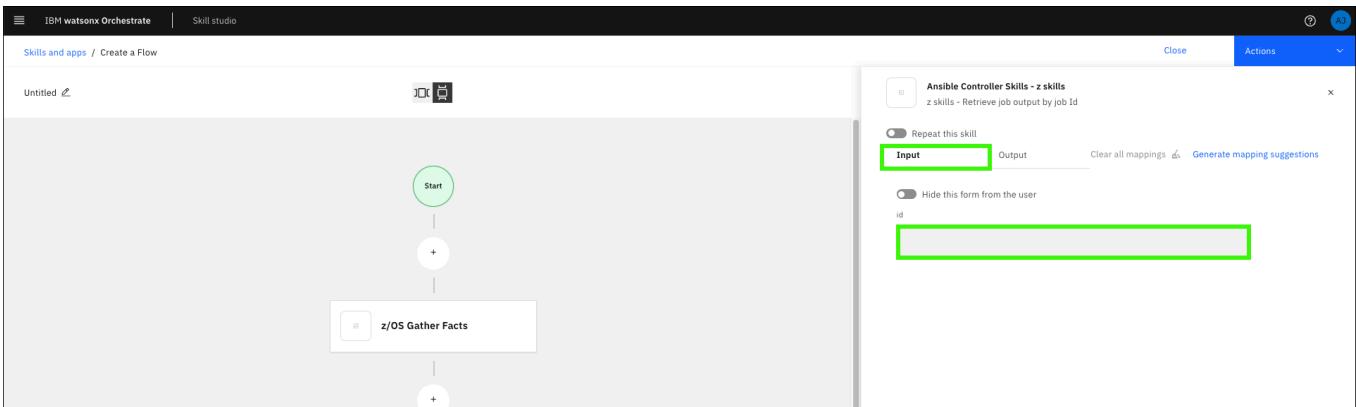


Next you must map the output values of the first skill to the input of the second skill. In this case, pass the “job id” output from **z/OS Gather Facts** as an input for **Retrieve job output**.

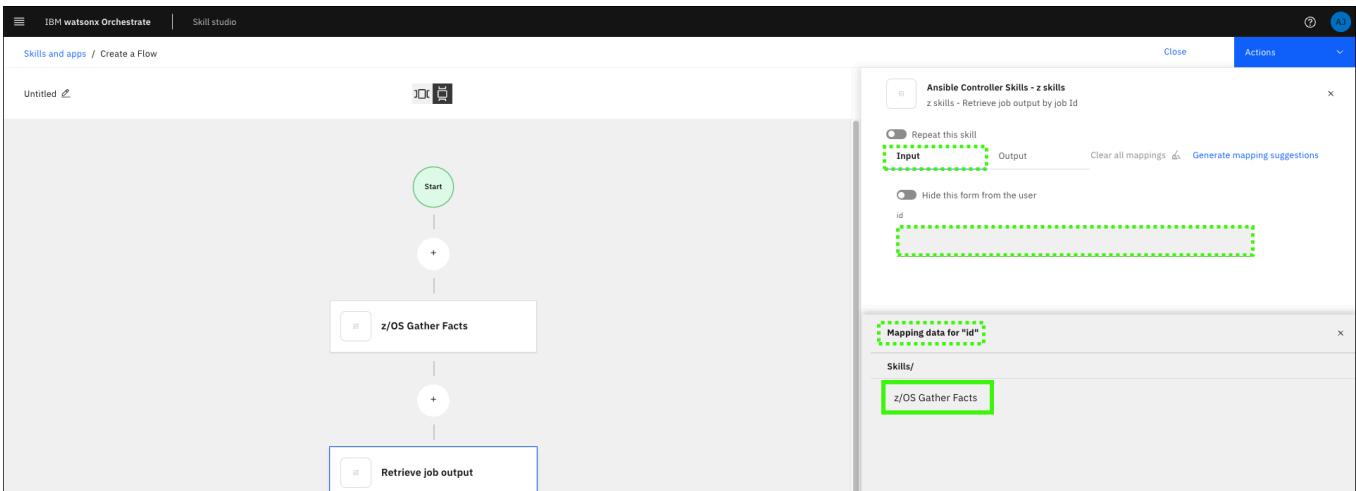
9. Click the **Retrieve job output** tile.



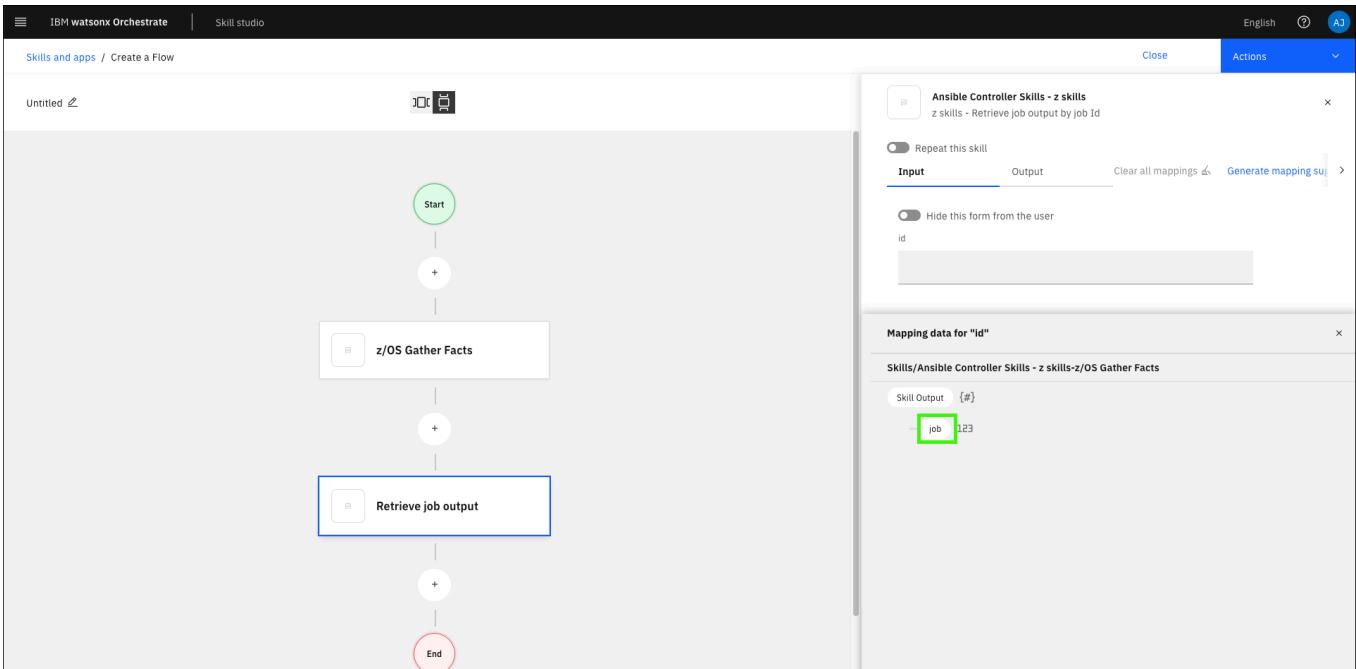
10. Select the **Input** tab and click in the **id** field.



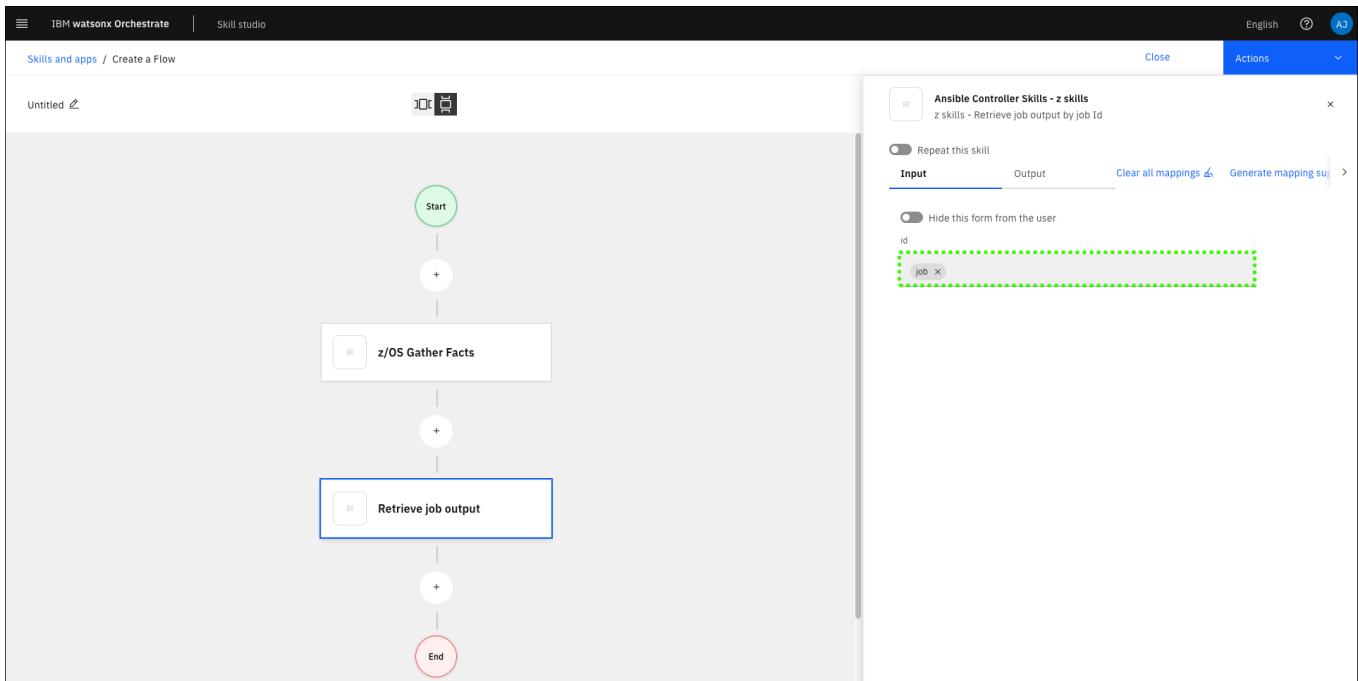
11. Click the **z/OS Gather Facts** skill in the **Mapping data for "id"** section.



12. Click the **job** icon.

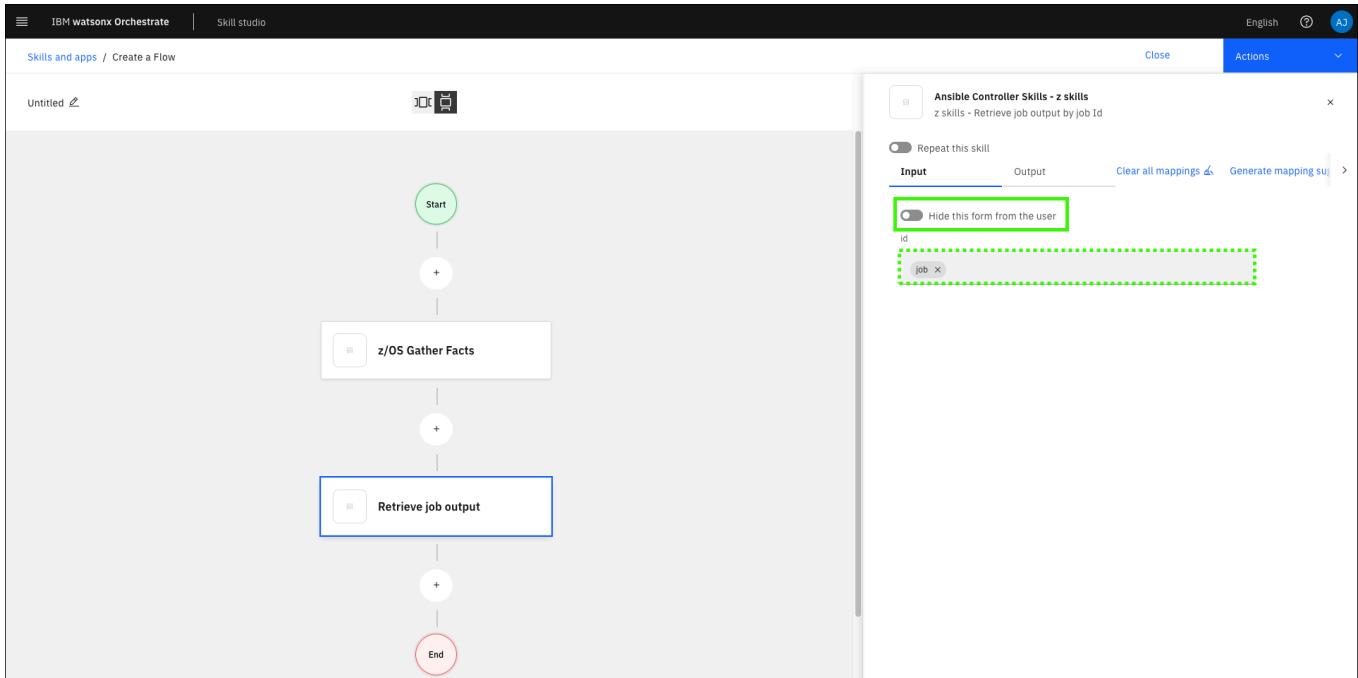


13. Verify the **job** appears in the **id** field.

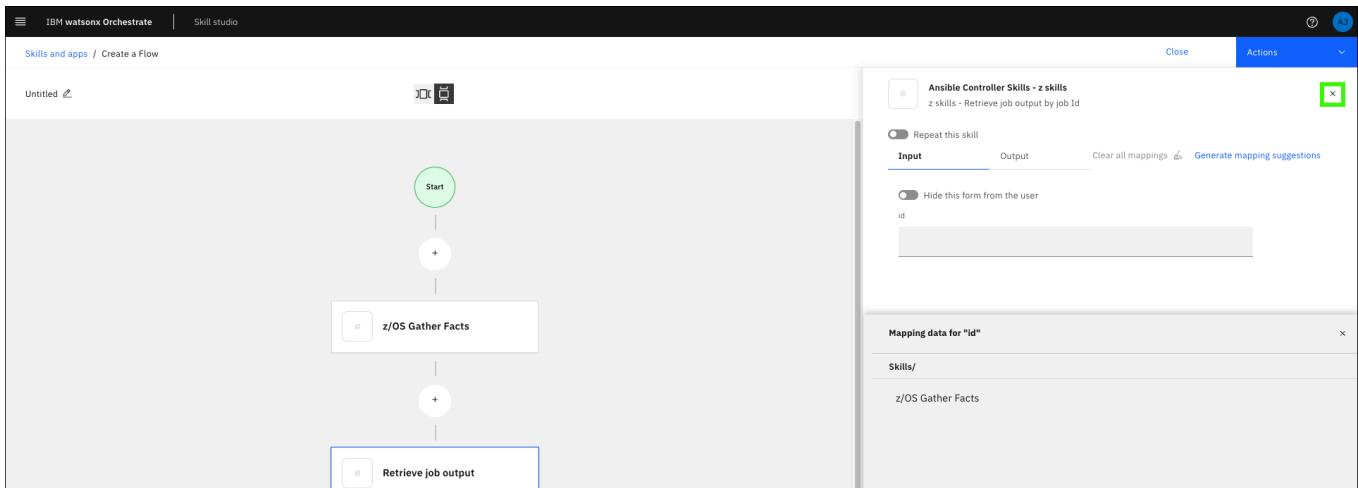


14. Optionally, toggle the **Hide this from the user** setting.

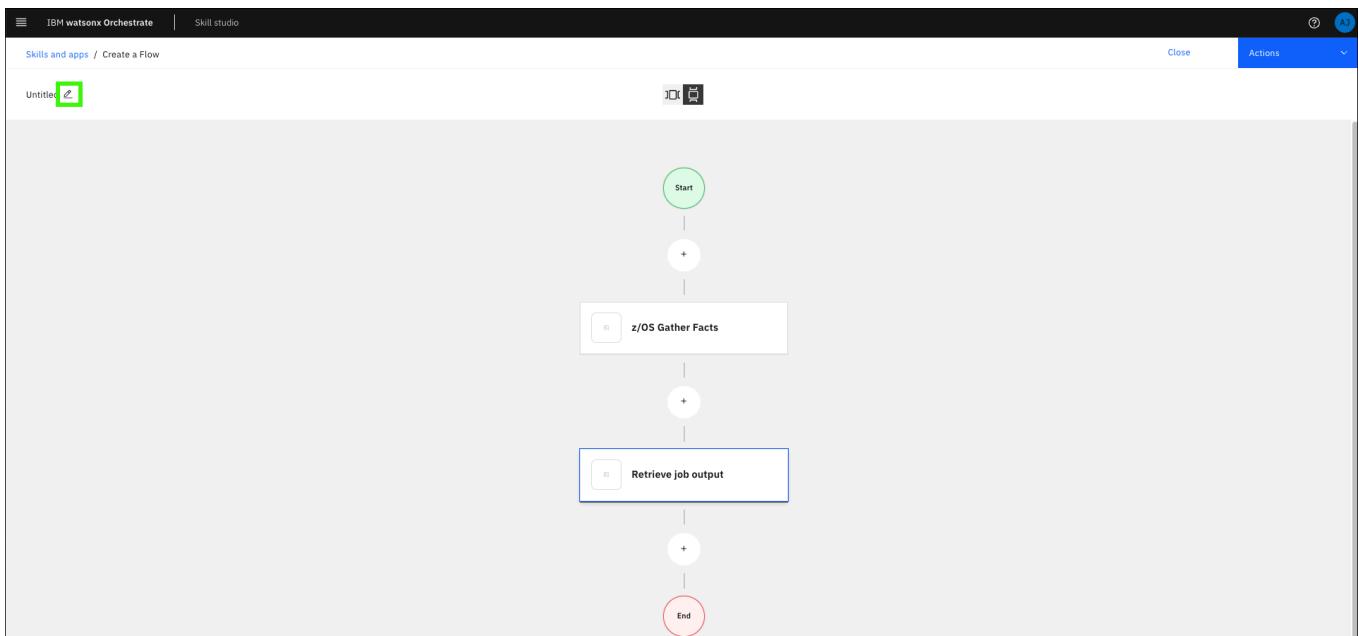
For this lab guide, this option is left disabled. Learn more about this option [here](#).



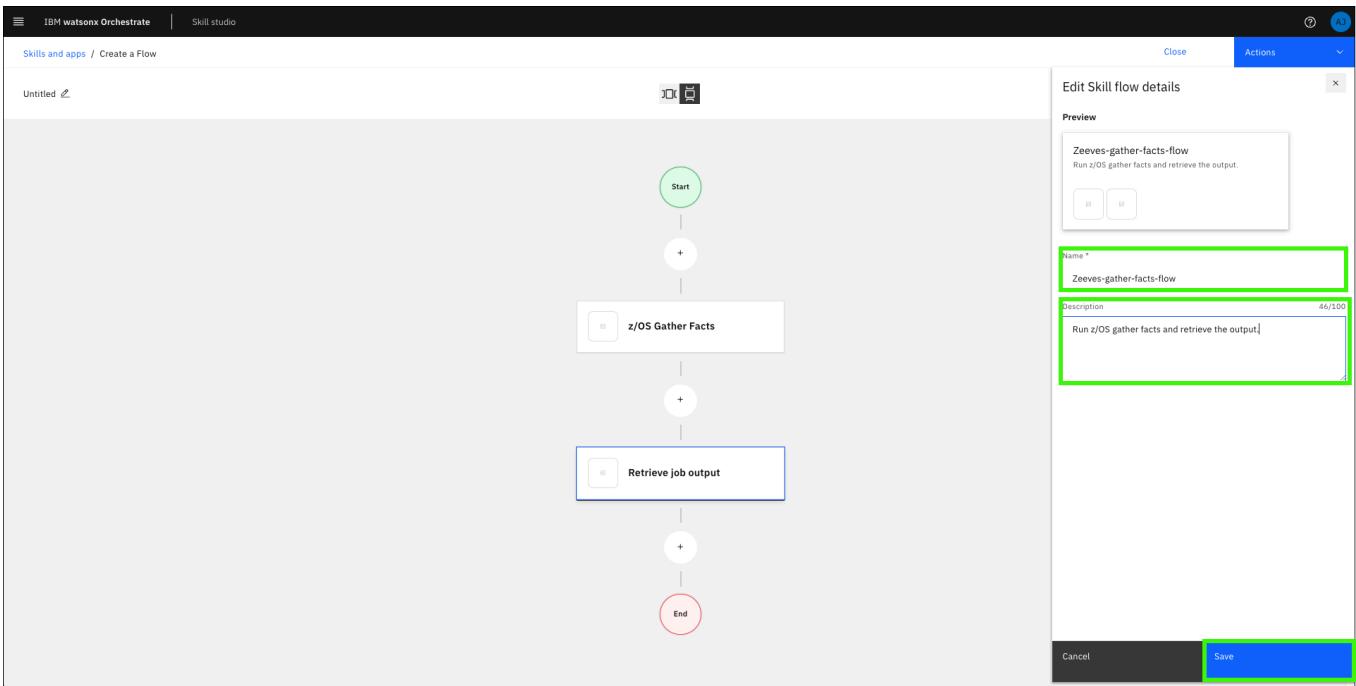
15. Click the **x** to close mapping window.



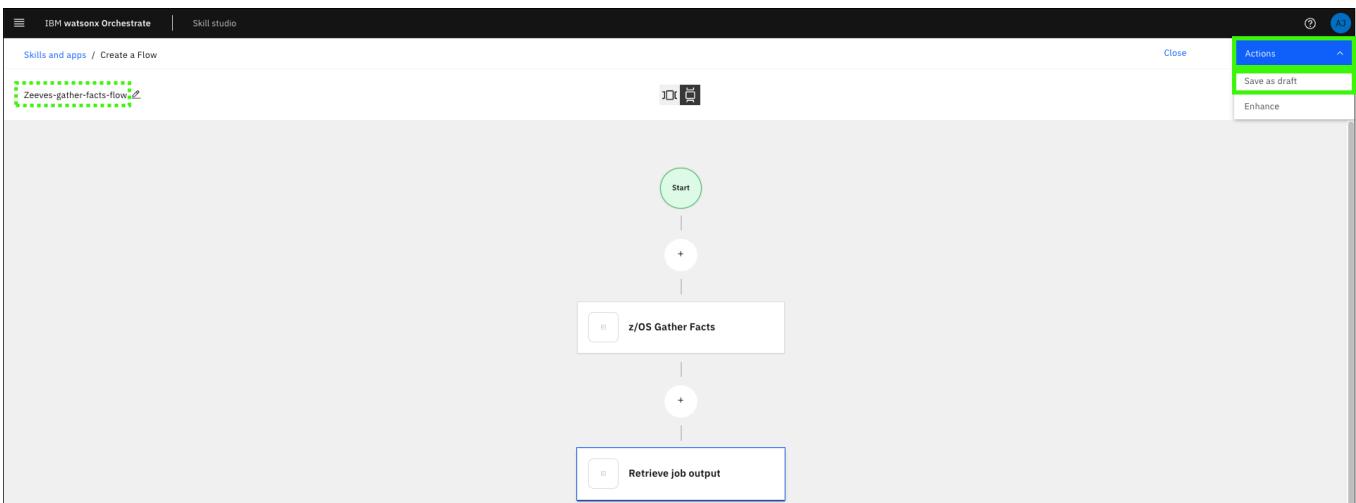
16. Click the pencil (✍).



17. Enter a (a) **Name** and (b) **Description** for your skill flow and then (c) click **Save**.



18. Expand the **Actions** pull-down list and click **Save as draft**.



19. Expand the **Actions** pull-down list and click **Enhance**.



On the **Enhancing the skill** pages, you can:

- modify the skill name, description, and version
- add phrases (prompts) that will be recognized by the assistant to call the skill flow

20. Click the **Phrases** tab.

21. Replace the existing **phrases** (prompts) and then click **Publish**.

Notice the default prompts are either not very intuitive (the skill flow name) or a bit verbose. Replace the existing phrases with phrases that you anticipate users will enter.



Be careful with the sample phrases you specify.

During the development of the lab guide, it was discovered that some sample phrases with a `/` character can cause issues with the actions. Avoid using `z/OS` in your sample phrases. This issue has been reported to the offering team.

Example prompts:

Show me zOS facts

Gather and display zOS facts

Name **Phrases** Next best skills

Phrases are the text your user types in the chat bar to find and use a skill.

Show me zOS facts

Gather and display zOS facts

Enter new train phrase

Cancel **Publish** Save as draft

Enable the skill flow in your assistant

1. Click **AI assistant builder** in the main menu.

Chat
Skill sets
Skill catalog

AI assistant builder

Configure prebuilt skills

Step in the process	Status	Skill type	Author	Last edited
Ready to use	Published	Skill flow	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024

2. Hover over the **Home (🏠)** and click **Actions**.

3. Click New action.

4. Click the Skill-based action tile.

5. Click the skill flow you created earlier and then click Next.

Note: it may take a minute for the tiles to appear on the screen.

Build an action from a skill

Select a skill
Choose a conversational skill published as a foundation of your action.

Search a skill:

Zeeves-gather-facts-flow Run z/OS gather facts and retrieve the output. Last updated: 2024-11-19T21:18:31.793Z	Retrieve job output z skills - Retrieve job output by job Id Last updated: 2024-11-19T20:08:59.538Z	z/OS Ping z skills - This playbook pings the z/OS host to test connectivity. Last updated: 2024-11-19T15:58:20.567Z	z/OS Gather Facts z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS-specific information from the z/OS host. Last updated: 2024-11-19T15:56:26.843Z	Summarize the Webex meeting transcript in watsonx.ai Last updated: 2024-11-04T10:49:16.502Z
Summarize the Box content in watsonx.ai Last updated: 2024-11-04T10:49:12.077Z	Summarize a Zendesk ticket in watsonx.ai Last updated: 2024-11-04T10:49:09.476Z	Summarize a ServiceNow incident in watsonx.ai Last updated: 2024-11-04T10:49:05.828Z	Summarize a Salesforce opportunity in watsonx.ai Last updated: 2024-11-04T10:49:01.769Z	Sharepoint document summary in watsonx.ai Last updated: 2024-11-04T10:48:55.707Z
Salesloft email summary	Salesforce case summarization	Salesforce case sentiment analyze	Outlook email summary	Github issue summarization

6. Enter an example prompt for the skill and click **Save**.

You can use one of the prompts you used earlier for the skill flow.

Show me zOS facts

Customer starts with:
Example: I want to pay my credit card bill.

Add example phrases:
Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.
The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 0
Example: I want to pay my credit card bill.

New action
What does your customer say to start this interaction?

Cancel Save

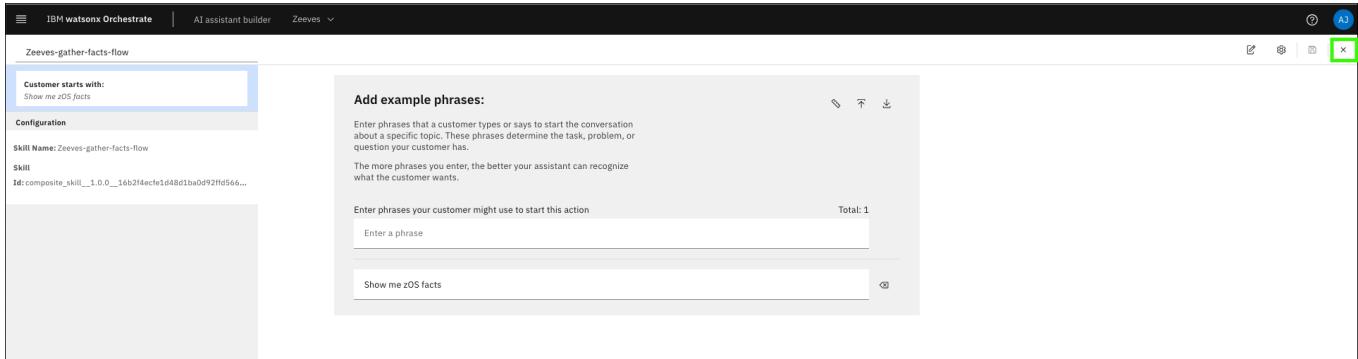
7. Enter any additional phrases (prompts) and then click the **save** (💾).

Customer starts with:
Show me zOS facts

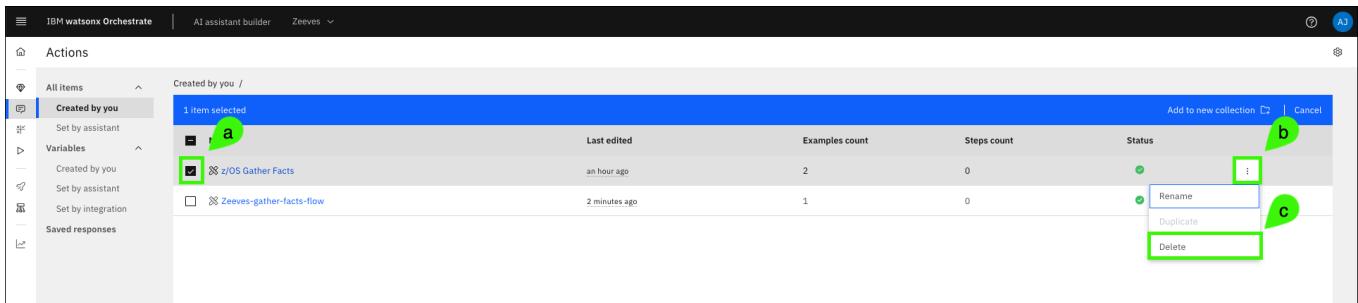
Add example phrases:
Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.
The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 1
Enter a phrase
Show me zOS facts

8. Click close (x).

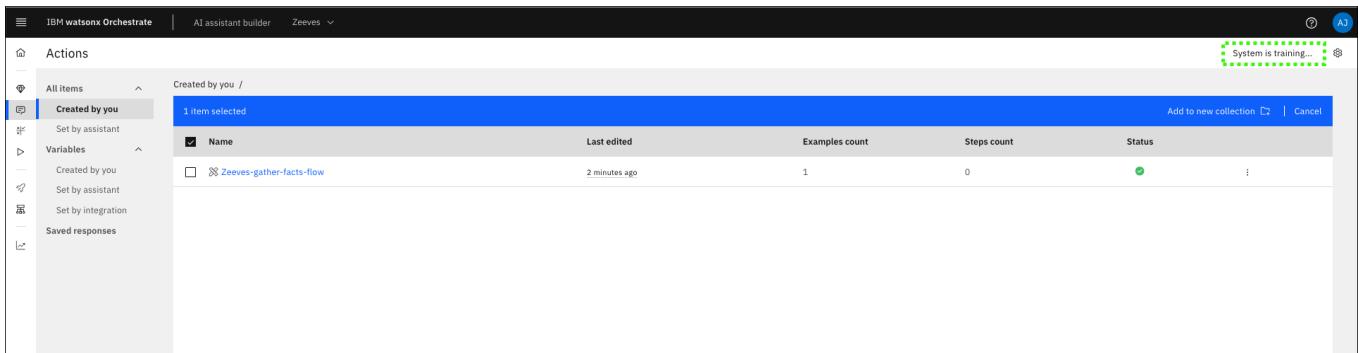


9. Select the original skill you created (a) (not the skill flow you just created), click the ellipses (b), and then click Delete (c).



10. Wait for system training to complete.

Note: The message will change to "System is trained" and then disappear.



11. Click **Preview**.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with 'Actions' expanded, showing categories like 'All items', 'Created by you', 'Variables', and 'Saved responses'. In the main area, a table lists one item: 'Zeeves-gather-facts-flow'. The table columns include 'Name', 'Last edited', 'Examples count', 'Steps count', and 'Status'. The status is green with a checkmark. At the bottom right of the main area, there's a 'Preview' button, which is highlighted with a green box.

12. Enter one of the prompts you specified into the assistant preview.

The screenshot shows the IBM Watsonx Orchestrate interface with the preview window open. In the preview window, there's a text input field containing the message 'Show me zOS facts'. Below the input field, a conversation log shows a message from 'Greet customer [default]': 'Welcome, how can I assist you?'. The user's message 'Show me zOS facts' is shown below it. The user's message is highlighted with a green box. The preview window also shows a status bar at the bottom with 'You 8:13 AM'.

13. Wait 10 seconds and then click **Apply**.

Note: It is important to wait for the first job to complete before submitting the second job in the flow.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with 'Actions' selected. Under 'Created by you', there's a list item 'Zeeves-gather-facts-flow'. The main area shows a table with one item selected. The table columns are Name, Last edited, Examples count, Steps count, and Status. The status is green with a checkmark. To the right, there's a 'Preview' window showing a conversation between a user and a bot. The user says 'Greet customer [default]'. The bot responds with 'Welcome, how can I assist you?'. The user then says 'Show me zOS facts'. The bot responds with 'Zeeves-gather-facts-flow' and lists several facts: 'ipl_volume': 'D25VS', 'load_param_device_id': 'S1', 'load_param_dsn': 'S1', 'lpar_name': 'M0', 'master_catalog_vols': 'M0', 'nucleus_id': '1', and 'operator_prompt_flag': 'M0'. At the bottom of the preview window, there are 'Cancel' and 'Apply' buttons.

14. Review the results from the skill flow.

Use both scroll bars in the assistant preview to review all of the returned information. The output should be similar to what was seen in the AAP web console. The character strings like [0;32m are special characters that are not properly displayed in the assistant preview interface.

This screenshot is similar to the previous one but shows the raw JSON output of the facts instead of their rendered versions. The preview window shows the following JSON:

```
[{"ipl_volume": "D25VS", "load_param_device_id": "S1", "load_param_dsn": "S1", "lpar_name": "M0", "master_catalog_vols": "M0", "nucleus_id": "1", "operator_prompt_flag": "M0"}]
```

A green dashed box highlights the list of facts. The rest of the interface is identical to the first screenshot.



Sample output from the Z/OS gather facts flow.



Content

```

Identity added: /runner/artifacts/16/ssh_key_data (/runner/artifacts/16/ssh_key_data)
[1;35m[WARNING]: Collection ibm.ibm_zos_core does not support Ansible version 2.14.2[0m

PLAY [Gather z/OS-specific facts.] *****
TASK [Gather all facts about z/OS host.] *****
TASK [Print gathered facts about the master catalog.] *****
[0;32mok: [zos_host][0m
[0;32m      "master catalog dsn: CATALOG.VS01.MASTER",0m [0;32m      "master catalog volser: OPEVS1"[0m
[0;32m ][0m [0;32m}{0m

TASK [Print only CPC and IODF info from gathered z/OS facts.] *****
[0;32mok: [zos_host] => {[0m
[0;32m      "msg": "[0m [0;32m      "manufacturer: IBM",[0m [0;32m      "model: A00",[0m [0;32m      "plant: C1",[0m
[0;32m      "iodf name: PROV.IODF00",[0m [0;32m      "iodf config: DEFAULT"[0m [0;32m ][0m [0;32m}{0m

TASK [Print out all gathered facts about the z/OS host.] *****
[0;32mok: [zos_host] => {[0m
[0;32m      "ansible_facts": {[0m [0;32m      "arch_level": "2",[0m [0;32m      "cpc_nd_manufacturer": "IBM",[0m
[0;32m      "cpc_nd_model": "A00",[0m [0;32m      "cpc_nd_plant": "C1",[0m
[0;32m      "cpc_nd_seqno": "20D90792EB76",[0m [0;32m      "cpc_nd_type": "008562",[0m [0;32m      "edt": "00",
[0m [0;32m      "hw_name": "",[0m [0;32m      "ieasym_card": "(00,K2)",[0m [0;32m      "io_config_id": "00",[0m
[0;32m      "iodate": "",[0m [0;32m      "iodesc": "",[0m [0;32m      "iodf_config": "DEFAULT",[0m
[0;32m      "iodf_name": "PROV.IODF00",[0m [0;32m      "iodf_unit_addr": "DE28",[0m [0;32m      "ioproc": "",[0m
[0;32m      "iotime": "",[0m [0;32m      "ipayloadxx": "K2",[0m [0;32m      "ipl_volume": "D25VS1",[0m
[0;32m      "load_param_device_num": "DE28",[0m [0;32m      "load_param_dsn": "SYS0.IPLPARM",[0m
[0;32m      "lpar_name": "",[0m [0;32m      "master_catalog_dsn": "CATALOG.VS01.MASTER",[0m
[0;32m      "master_catalog_volser": "OPEVS1",[0m [0;32m      "nucleus_id": "1",[0m
[0;32m      "operator_prompt_flag": "M",[0m [0;32m      "parmlib_dsn": "K2.PARMLIB",[0m
[0;32m      "parmlib_volser": "USRVS1",[0m [0;32m      "primary_jes": "JES2",[0m
[0;32m      "product_mod_level": "00",[0m [0;32m      "product_name": "z/OS",[0m
[0;32m      "product_owner": "IBM CORP",[0m [0;32m      "product_release": "05",[0m
[0;32m      "product_version": "02",[0m [0;32m      "smf_name": "VS01",[0m [0;32m      "sys_name": "VS01",[0m
[0;32m      "sysplex_name": "LOCAL",[0m [0;32m      "tsoe_rel": "05",[0m [0;32m      "tsoe_ver": "4",[0m
[0;32m      "vm_name": ""}[0m [0;32m }[0m [0;32m}{0m

PLAY RECAP *****
[0;32mzos_host[0m          : [0; 32mok=4
[0m changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ig nored=0

```

The scenario shown above may or may not be relevant for your client's use case. It is intended to show you how to sequence skills together in a skill flow to create an action that your assistant triggers based on prompts using the pre-configured Ansible automation templates. You are encouraged to create your own skill flows and prompts using other skills available within the AAP instance.

Next, learn about custom-built actions.

Creating custom-built actions

To this point, you have learned how to:

- import skills into Watson Assistant Orchestrate
- add applications with those skills to your assistant
- create skill-based actions for your assistant
- combine skills in a skill flow

There is also the ability to create **custom-built** actions. Custom-built actions allow you to create new actions with different steps to take in conversations and form sequences of prompts that define the conversation experience. The steps can be defined with or without conditions, which help control the custom responses. Steps within the custom action can end with routing to conversational search, triggering another existing sub-action, and other actions. This is a powerful way of customizing the end-user's experience.

Learn more about creating custom-built actions [here](#).

Importing pre-packaged z/OS skills

Provided with Version 2 of watsonx Assistant for Z is a set of pre-packaged skills which can be used to automate various tasks on z/OS, such as running different console commands and retrieving logs from batch jobs.

The list of pre-packaged skills available include:

- Authorized program list
- z/OS IPL Information
- Display zOS parmlib datasets
- Unix System services options
- Display zOS subsystems
- List spool files
- Retrieve dataset content
- Retrieve spool file content
- Retrieve z/OS Management Facility (OSMF) job status

IBM watsonx Orchestrate requires that any OSMF environment you connect to for skill execution has certificate authority (CA) signed certificates. In the case of the Ansible Automation Platform (AAP) & Wazi z/OS environment provisioned in IBM Technology Zone (ITZ), the z/OS system is not currently using CA signed certificates and therefore cannot execute the pre-packaged skills on your own z/OS system. For demo purposes, it is still recommended to import them so that the pre-packaged skills can be shown.

Work is in progress to modify the ITZ environments so that they are enabled for skill execution using these pre-packaged skills. In the meantime, the underlying automation for these pre-packaged skills can still be demonstrated using the Ansible skill 'z/OS Operator Command' which is available to import. This works because the pre-packaged skills are executing console commands directly using OSMF APIs which can also be run using the Ansible template skill 'z/OS Operator command'. For example, here are the console commands being used in some of the pre-packaged skills:

- Authorized Program list – `operator command -> d prog,lnklist`
- z/OS IPL Information - `operator command -> d iplinfo`
- Display zOS parmlib datasets - `operator command -> d parmlib`

You can import the pre-packaged skills into your sandbox environment by downloading the .zip file from [here](#) and following [these instructions](#).

You must extract the imbedded JSON file and modify the file for your environment by following [these instructions](#).

Publishing and deploying your assistant

To this point, acting as an Assistant Builder, you have built out the assistant, configured conversational search, and added skills and automations. While doing so, you have been testing your assistant using the **preview** capability of AI Assistant Builder. The **preview** capability is a closed environment for experimenting with prompts.

After your assistant is finalized, you can publish it to make it available to end-users. Each assistant you create comes with two **environments**: *draft* and *live*. You have been configuring your assistant in the draft environment. Each environment has its own set of IDs, URLs, and service credentials that can be referenced by external services.

The **Environments** page in the AI assistant builder has tabs for managing both the **Draft environment** and the **Live environment**:

This screenshot shows the 'Environments' page in the AI assistant builder. The 'Draft' tab is selected, indicated by a green dashed border. The 'Live' tab is also present. On the left, there's a sidebar with icons for environments, channels, and integrations. The main area is divided into sections: 'Draft environment' (with a description of it being for internal preview), 'Channels' (listing 'Web chat'), 'Resolution Methods' (showing 'Draft content' with a last edit timestamp and a 'Draft' status), and 'Extensions' (listing 'Search'). A 'Preview this environment' button is located in the top right corner.

This screenshot shows the 'Environments' page in the AI assistant builder. The 'Live' tab is selected, indicated by a green dashed border. The 'Draft' tab is also present. The layout is identical to the Draft environment page, with sections for 'Live environment' (description of deployment to customers), 'Channels' (listing 'Web chat'), 'Resolution Methods' (showing 'Content' with a note about no published version), and 'Extensions' (listing 'Search'). A 'Publish version' button is located in the top right corner of the Content section.

The **Draft environment** contains all your in-progress work in the Actions, Preview, and Publish pages. Use the **Draft environment** tab to build out your assistant and use for internal testing before deployment. Any integrations you use (i.e. channels) for the **Draft environment** are unique to that environment, and changes to draft integrations don't affect the **Live environment**.

Publish the assistant

Each time that you publish, you're creating a new version of the assistant, for example V1. When you publish your content, you're creating a snapshot of the draft content, resulting in a version.



Versions do not contain integration configurations or environment settings

Published versions contain all of the content from actions, including settings and variables. **However, versions do not contain integration configurations or environment settings.** Integration configurations and environment settings must be configured manually in each environment.

For managing quality-control and versioning, the Live environment is the version of the assistant you should be giving access to the end-user.

Follow these steps to publish the first version of your assistant using Assistant Builder:

1. Hover over the **Home** icon () and click **Publish**.

The screenshot shows the IBM Watsonx Orchestrate interface with the 'AI assistant builder' tab selected. On the left, there's a sidebar with 'Generative AI', 'Actions', 'Preview', and 'Publish' (which is highlighted with a green box). Below the sidebar, there are sections for 'Build', 'Environments', 'Improve', and 'Analyze'. The main area has a heading 'Welcome to Watsonx generative AI features! Learn more about intelligent information gathering.' and a 'Build actions' card. To the right of the cards, there's a 'Default behavior' node connected to a 'General purpose' node and a 'Conversational search' node, with a 'Search' button. At the bottom, there's a navigation bar with icons for back, forward, and search.

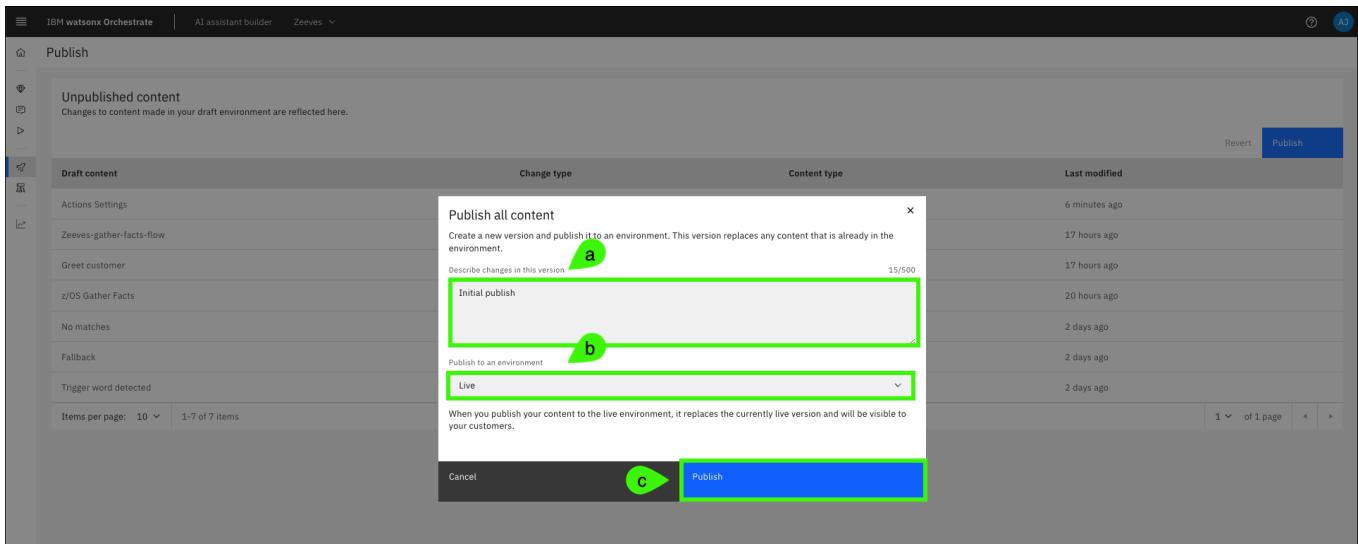
2. Click **Publish**.

The screenshot shows the 'Publish' step in the AI assistant builder. The top navigation bar shows 'IBM Watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The left sidebar has 'Publish' selected. The main area displays a table of 'Draft content' changes:

Draft content	Change type	Content type	Last modified
Zeeves-gather-facts-flow	Updated	Actions	30 minutes ago
Greet customer	Updated	Actions	31 minutes ago
Actions Settings	Updated	Settings	2 hours ago
Fallback	Updated	Actions	2 hours ago
No matches	Updated	Actions	2 hours ago
Trigger word detected	Updated	Actions	2 hours ago

At the bottom, there are pagination controls: 'Items per page: 10' and '1-6 of 6 items'.

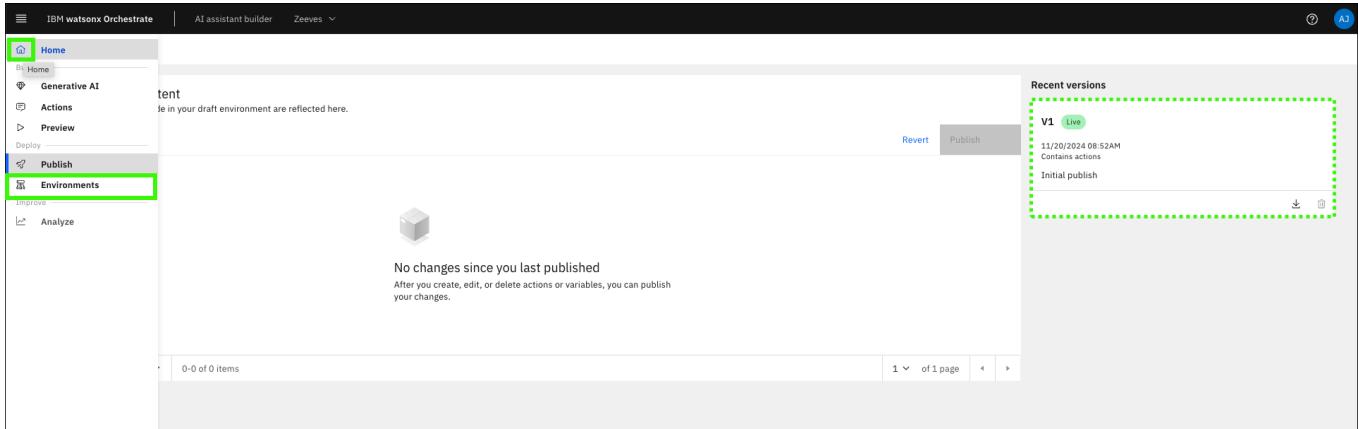
3. Enter a description of the changes (a), set the environment to **Live** (b), and then click **Publish** (c).



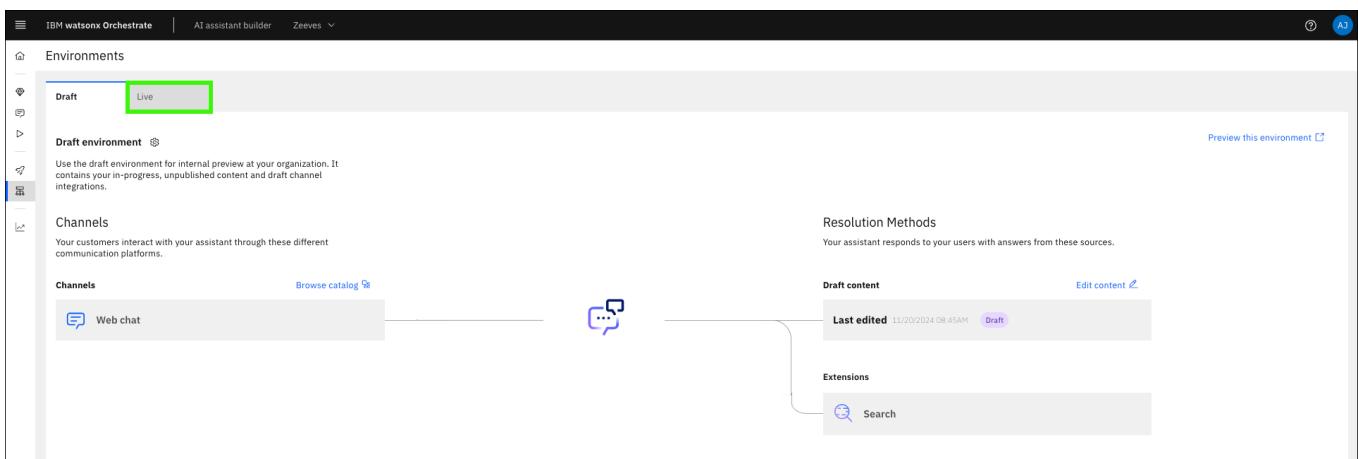
Configure the live environment

Important: When the live environment is created, the environment settings from the draft environment are not carried over (for example the configuration of the OpenSearch instance used for conversational search).

1. Hover over the Home icon (🏡) and click Environments.



2. Click Live.



3. Click Web chat.

The screenshot shows the IBM Watsonx Orchestrate interface with the 'Live' environment selected. On the left, under 'Channels', the 'Web chat' option is highlighted with a green box. To the right, there are sections for 'Resolution Methods', 'Content' (version V1), and 'Extensions' (Search). A central icon of two speech bubbles is positioned between the channels and content sections.

4. Customize the live assistant as you see fit.

On the **Style** tab, you're able to set the Assistant name which will be displayed at the top of the chat window when end-users are interacting with the assistant. For pilots or demos, you may want to personalize this name for the client. Also in the **Style** tab, you have the ability set the themes and display settings of the chat windows, including the ability to enable the IBM Watermark and enable streaming (recommended).

On the **Home** tab, you enable and customize a default greeting message from the assistant when the user accesses the assistant chat. You're also able to set Conversation starters that will be displayed in the chat window. When selected by the end-user, the text of these conversation starters are sent as prompts, so it is important that your assistant is trained and tested to answer appropriately. It is highly recommended to remove these default conversation starters and to consider creating your own as long as they're able to be executed as actions and provide value to the end-user. At the bottom of the Home screen tab, you will also see the ability to add a Background style for the assistant chat window.

Explore all the other tabs.



Customize your live environment.

For this lab, toggle **Streaming** on and turn **Suggestions** off on the **Suggestions** tab. You may also want to change the theme to **Dark** to differentiate your draft and live environments.

Style

Suggestions (highlighted with a green box)

Restart conversation

Save and exit

Customize your chat UI

Update the style to match your brand and your website. A developer can also add more advanced styling changes with code. [Learn more](#)

Assistant's name as known by customers: Zeeves 1.0

Intended purpose: Standard: For virtual agents and customer support experiences.

Choose a theme: Light (highlighted with a green box) Dark

Primary color: #FFFFFF Secondary color: #3D3D3D

Chat header: User message bubble

Accent color: #035AE9

Significant and interactive objects

Size: Width 380px Height 640px

IBM Watermark: Enable IBM Watermark (On)

Streaming: Enable Streaming (On)

Zeeves 1.0

Hi! I'm a virtual assistant. How can I help you today?

Type something... ▶

Powered by IBM watsonx ⓘ

5. Click Save and exit.

Style

Save and exit (highlighted with a green box)

Restart conversation

Customize your chat UI

Update the style to match your brand and your website. A developer can also add more advanced styling changes with code. [Learn more](#)

Assistant's name as known by customers: Zeeves 1.0

Intended purpose: Standard: For virtual agents and customer support experiences.

Choose a theme: Light (highlighted with a green box) Dark

Primary color: #FFFFFF Secondary color: #3D3D3D

Chat header: User message bubble

Accent color: #035AE9

Significant and interactive objects

Size: Width 380px Height 640px

IBM Watermark: Enable IBM Watermark (On)

Streaming: Enable Streaming (On)

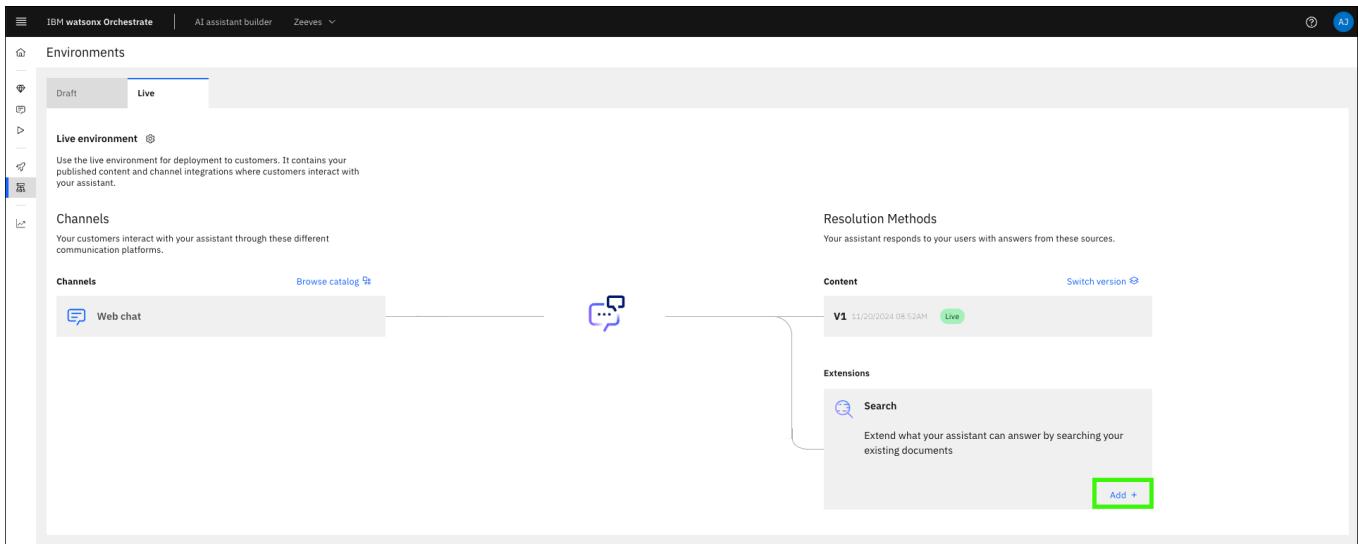
Zeeves 1.0

Hi! I'm a virtual assistant. How can I help you today?

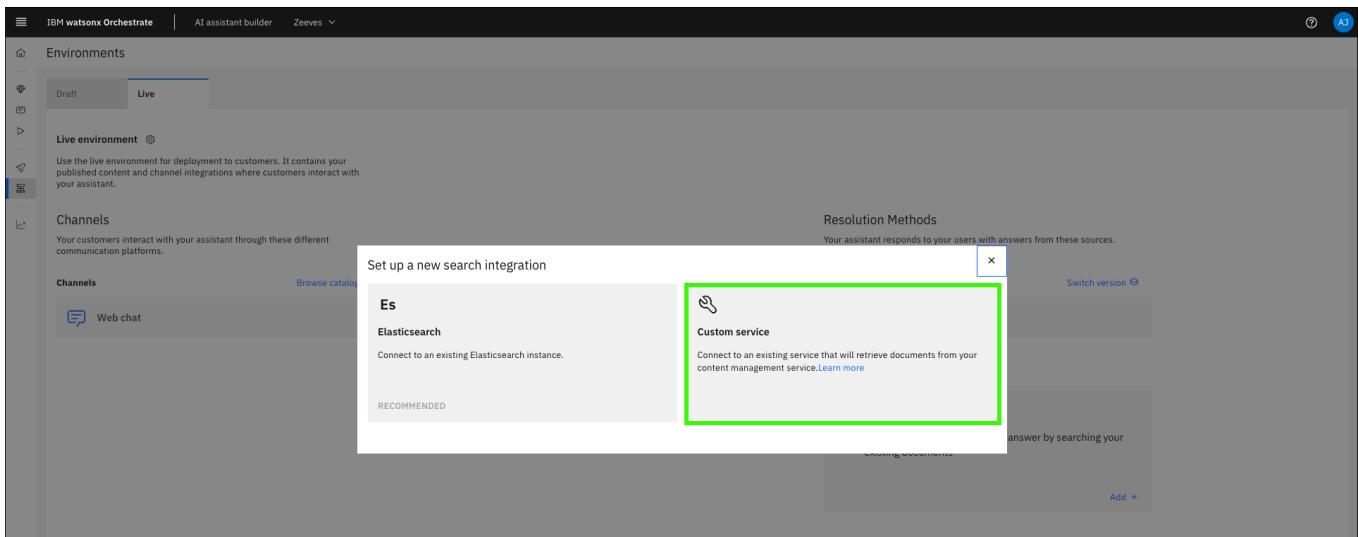
Type something... ▶

Powered by IBM watsonx ⓘ

6. Click Add in the Search tile.



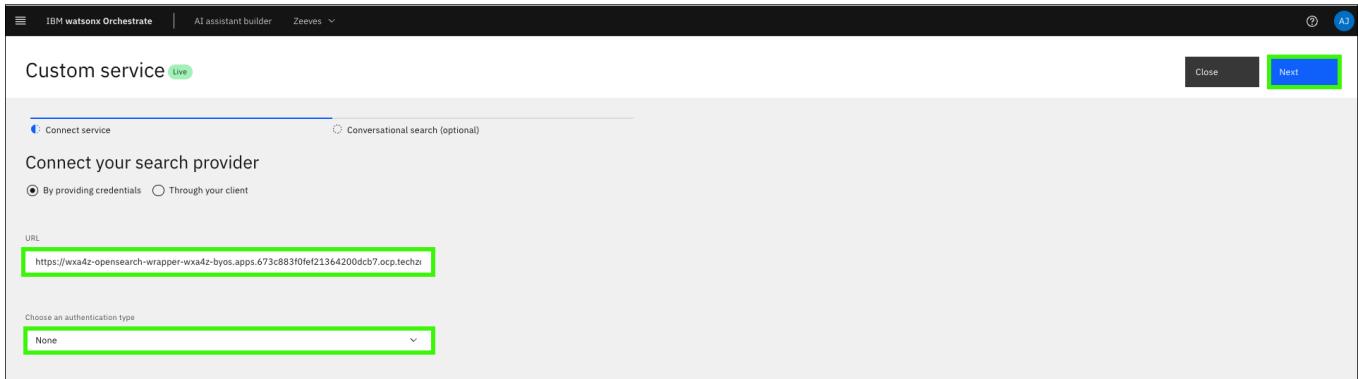
7. Click Custom service.



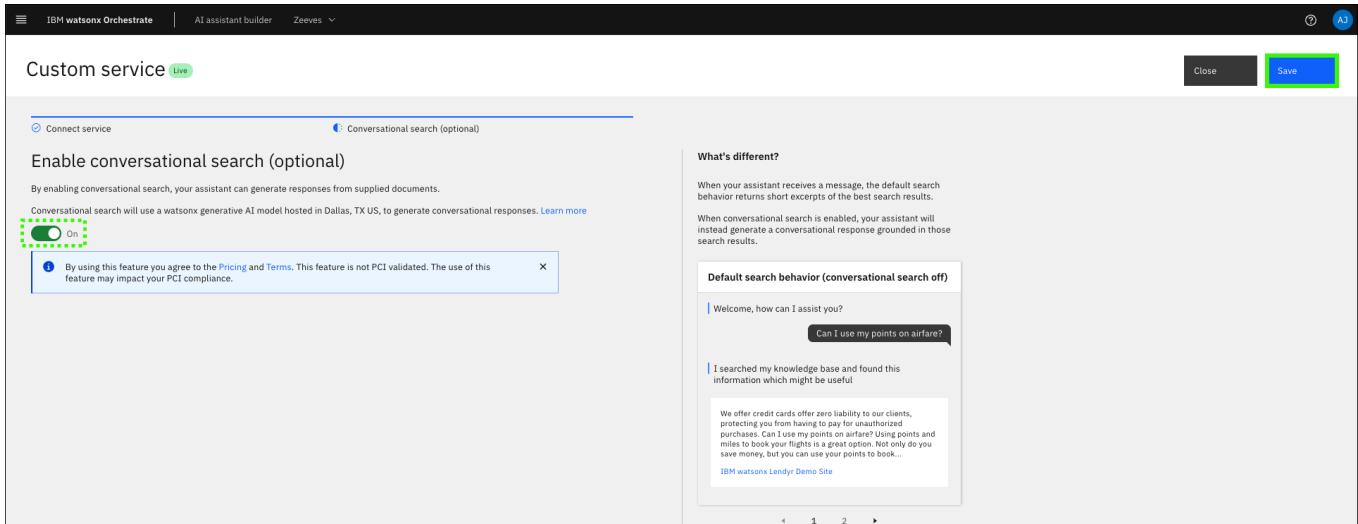
8. Enter the URL for your bring-your-own-search (BYOS) engine (a), select None for the authentication type (b), and then click Next (c).

⚠️ Use the correct URL!

Be sure to use the URL for your BYOS OpenSearch engine created earlier [here](#).



9. Verify conversational search is enabled and click Save.



10. Update the Custom service settings (a-f), click Save (g), and then click Close (h).



Feel free to customize the settings.

This is your assistant. Feel free to customize the settings. The settings shown below reflect the changes made earlier in the lab guide to the draft version of the assistant. This includes the **Metadata** field to weigh ingested client documents higher using:

```
{
  "doc_weight": {
    "product_docs": 0.2,
    "customer_docs": 0.8,
    "ibm_indices": "*_ibm_docs_slate",
    "customer_indices": "customer_*"
  }
}
```

Custom service Live

Settings Instance

By using this feature you agree to the [Pricing](#) and [Terms](#). This feature is not PCI validated. The use of this feature may impact your PCI compliance. X

Conversational search

Use a Watson generative AI model hosted in Dallas, TX US, to generate conversational responses. [Learn more](#)

Conversation length

a On

b Off

Context-dependent questions

c Single turn

This assistant uses only the current user input for retrieving search results and generating answers. This works well for clear, complete inputs but generally won't work with context-dependent queries such as, "Why is that?" after a previous answer.

d Entire conversation Beta

The assistant uses the entire session history for retrieving search results and generating answers. This handles context-dependent questions well but may over-rely on past topics, even if the user has moved on.

Define the text for the citations title. ⓘ

How do we know?

Search configuration

To test this, go to the [evaluations page](#) [Restore default](#) ⓘ

Retrieval confidence threshold ⓘ

e Lowest

Low High Highest

Generated response length ⓘ

Concise Moderate **f** Verbose

Response confidence threshold ⓘ

g Lowest

Low High Highest

Citations

Citations will be displayed to the end user. This doesn't impact the number of citations used in the response.

Citations shown in web chat

All

Default filter

Can be anything (plain text, JSON, etc)

Metadata

```
{"doc_weight": 1.0,
"product_docs": 0.2,
"customer_docs": 0.8,
"ibm_indices": "**.ibm_docs.state",
"customer_indices": "customer_**"}
```

Search display text

Define the text your search will display to the end user

No results found Connectivity issue

I searched my knowledge base, but did not find anything related to your query

Close **Save** A3

Connect the skills to the live environment

1. Click **Skill sets** in the main menu.

The screenshot shows the IBM Watsonx Orchestrate interface. The left sidebar has sections: Chat, Skill sets (highlighted with a green border), Skill catalog, BUILD (AI assistant builder, Skill studio), and ADMINISTER (Access management). The main area shows 'Resolution Methods' with 'Content' (V1, 12/04/2024 03:22PM, Live) and 'Extensions' (Search). A central icon is a blue speech bubble with three dots.

2. Select your assistant's live instance in the **Skill sets** list.

The screenshot shows the 'Team Skills' page. The 'Skill sets' section lists 'Team Skills' (highlighted with a green border), 'Orchestrator Agent Skillset', 'Team Skills', 'Zeeves draft', and 'Zeeves live'. Below this, there's a 'Name' section with a placeholder 'No skills' and a 'Manage skills' button.

3. Click **Connections**.

The screenshot shows the 'Zeeves live' skill set page. The 'Skills' tab is selected (highlighted with a green border). Below it, there's a 'Skills (3)' section with a note about skills available to team members. The 'Name' section lists 'z/OS Gather Facts', 'Zeeves-gather-facts-flow', and 'Retrieve job output'. At the bottom, there are pagination controls: 'Items per page: 10' and '1-3 of 3 items'.

4. Search for the application name you specified earlier.

Skill sets

Zeeves live

Skills **Connections**

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	3	Not specified	-	⋮

Items per page: 5 | 1-1 of 1 items

5. Click the ellipses (⋮) for your app and then click **Connect app**.

Skill sets

Zeeves live

Skills **Connections**

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	3	Not specified	-	⋮ i Connect app 1 page < >

Items per page: 5 | 1-1 of 1 items

6. Click **Connect app**.

Skill sets

Zeeves live

Skills **Connections**

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type
Ansible Controller Skills - z skills	3	Not specified

Items per page: 5 | 1-1 of 1 items

Connect to Ansible Controller Skills - z skills

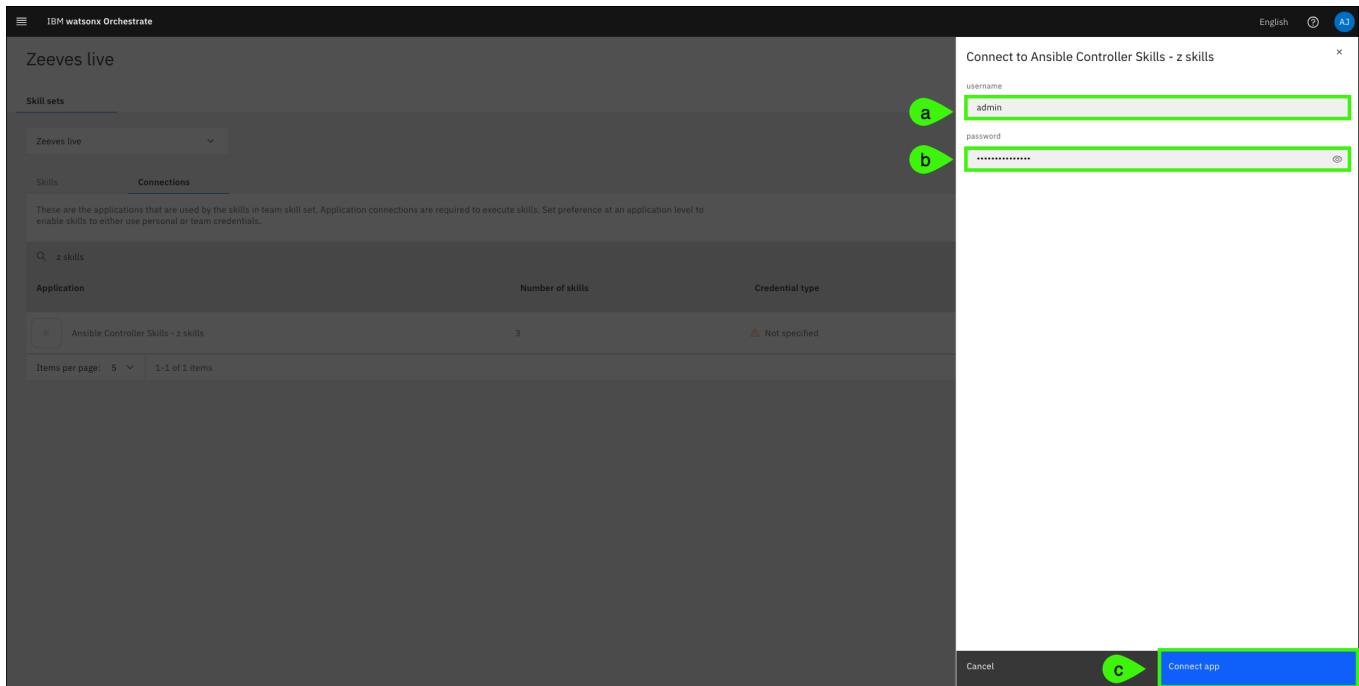
Member credentials
Each team member uses their own credentials to connect to this app and use its skills.

Team credentials
The admin sets the credentials each team member uses to connect to this app and use its skills.

You selected **Team credentials** for the credential type. Click **Connect app** to provide the credentials your team will use and to connect to the app.

Connect app

7. Enter the **username (a)** and **password (b)** using the username (`admin`) and password for your IBM Technology Zone (ITZ) watsonx Assistant for Z Pilot - AAP & z/OS reservation, and then click **Connect app (c)**.



Learn more about publishing your assistant and creating live environments [here](#).

Deploy the assistant

After configuring your assistant's settings and publishing, the final step is to deploy your assistant which can be done across various channels depending on the use case.

There are several options for deploying your assistant through channels and integrations to satisfy the use cases that you might address. Learn more about all the deployment options [here](#).

For this lab, you will deploy the assistant using the web chat integration. The web chat integration provides an assistant interface that can integrate with your website. There is a lot of flexibility with how you may want to integrate it. Learn more about the web chat integration [here](#).

1. Open the **Environments** page in [AI assistant builder](#).
2. Click **Web chat** for the **Live** environment.

The screenshot shows the IBM Watsonx Orchestrate interface. At the top, there are tabs for 'Environments' (Draft and Live), 'AI assistant builder', and 'Zeeves'. In the main area, under 'Live environment', it says 'Use the live environment for deployment to customers. It contains your published content and channel integrations where customers interact with your assistant.' Below this, under 'Channels', 'Web chat' is selected (indicated by a green border). To the right, there's a section titled 'Resolution Methods' with 'Content' (V1, 11/20/2024 08:52AM, Live) and 'Extensions' (Search). A large blue arrow points from the 'Web chat' selection towards the 'Content' section.

3. Click the **Embed** tab.

The screenshot shows the 'Web chat' configuration page. At the top, there are tabs for 'Style', 'Launcher', 'Home screen', 'Live agent', 'Suggestions', 'Security', 'Embed' (selected and highlighted with a green border), and 'Resources'. On the left, there's a 'Customize your chat UI' section with fields for 'Assistant's name as known by customers' (Zeeves 1.0), 'Intended purpose' (Standard: For virtual agents and customer support experiences), 'Choose a theme' (Light selected), 'Primary color' (#FFFFFF), 'Secondary color' (#30303D), 'Chat header' (User message bubble), 'Accent color' (#0354E9), 'Significant and interactive objects', 'Size' (Width: 380px, Height: 640px), and 'IBM Watermark' (Enabled). On the right, there's a preview window showing the Zeeves 1.0 virtual assistant with a dark theme, blue owl icon, and sample messages. A blue arrow points from the 'Embed' tab towards the preview window.

4. Copy and record the `integrationID`, `region`, and `serviceInstanceId` values.

The screenshot shows the 'Web chat' configuration page with the 'Embed' tab selected. Under the 'Embed' tab, there's a section titled '</>Embed on your website' with the text 'Ready to launch? It's as easy as copy and paste. [Learn more](#)'. Below this is a code editor containing a script block. A green box highlights the following code snippet:

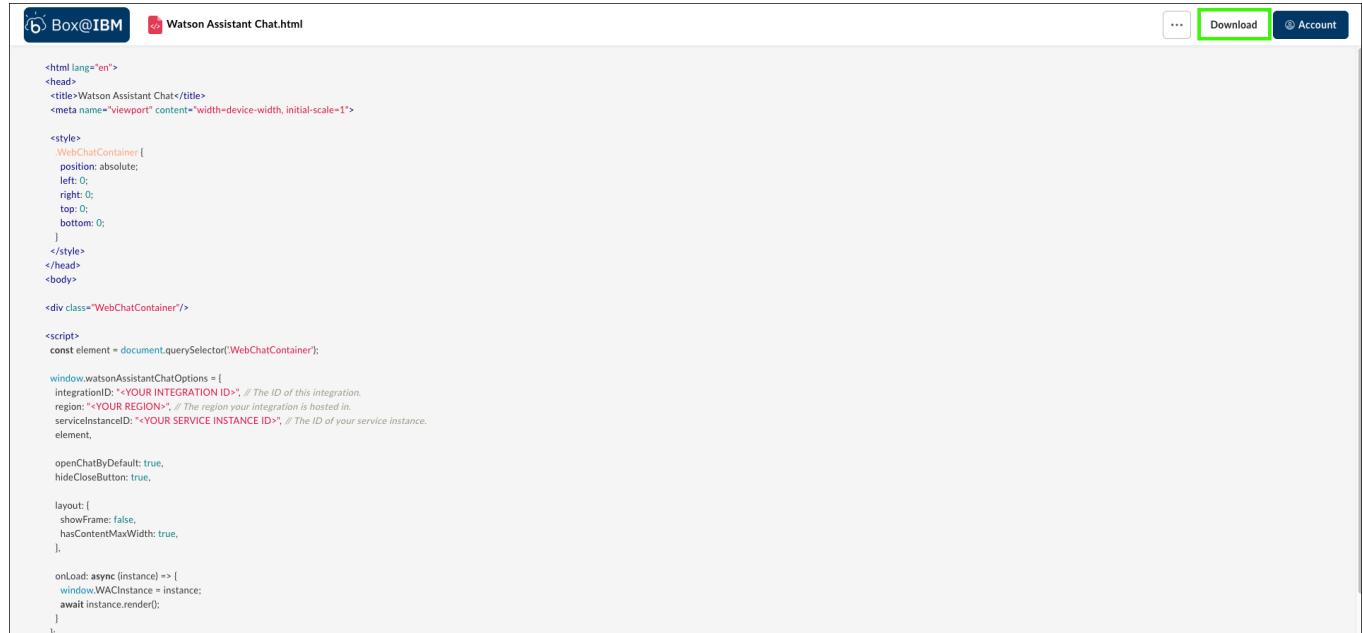
```

<script>
  const integrationID = "5e64d299-e6b0-434c-868c-2754d09272f8"; // The ID of this integration.
  const region = "us-south"; // The region your integration is deployed in.
  const serviceInstanceId = "21c153381-8d6f-474b-bdd3-219f5232087bc"; // The ID of your service instance.
  const loadScript = (instance) => { await instance.loadScript(); };
</script>
  
```

A blue arrow points from the highlighted code towards the bottom right of the code editor.

5. Click the link below to download a sample web chat hyper text markup language (HTML) page.

Watson Assistant Chat.html



```

<html lang="en">
<head>
<title>Watson Assistant Chat</title>
<meta name="viewport" content="width=device-width, initial-scale=1">

<style>
.WebChatContainer {
  position: absolute;
  left: 0;
  right: 0;
  top: 0;
  bottom: 0;
}
</style>
</head>
<body>

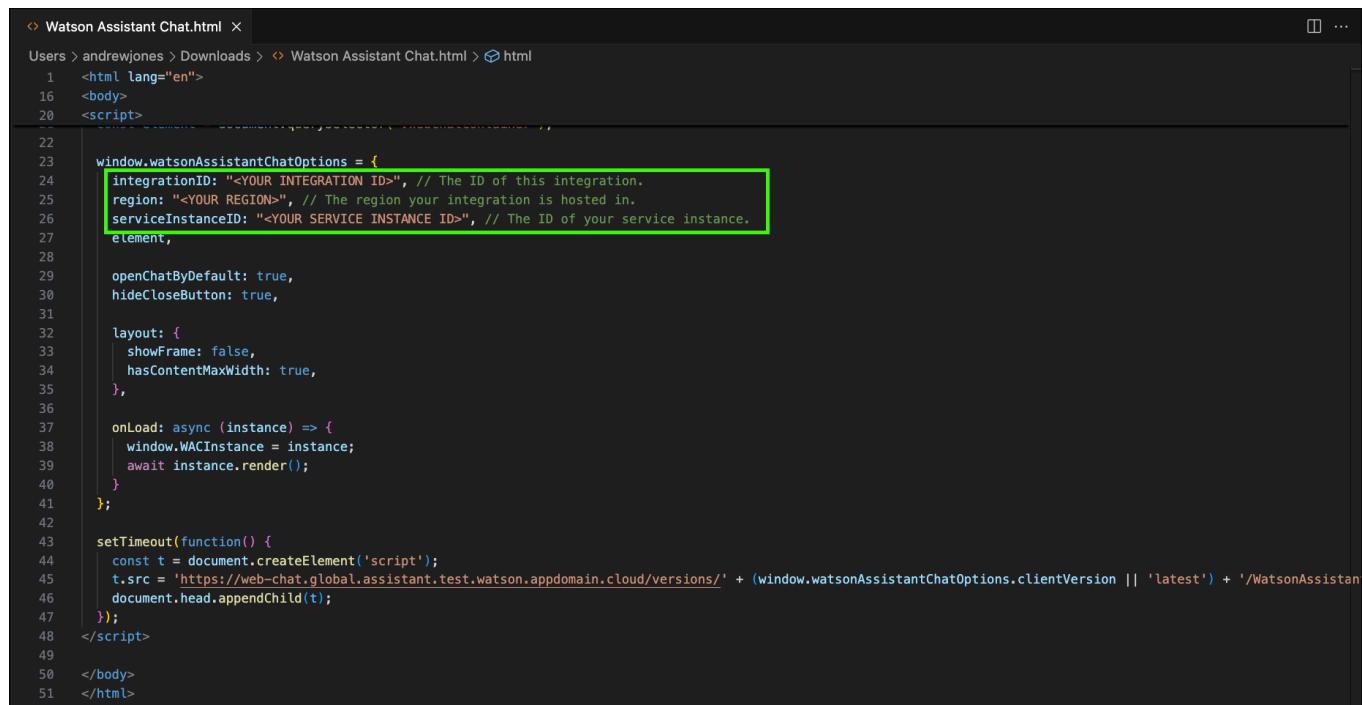
<div class="WebChatContainer">

<script>
const element = document.querySelector('WebChatContainer');

window.watsonAssistantChatOptions = {
  integrationID: "<YOUR INTEGRATION ID>", // The ID of this integration.
  region: "<YOUR REGION>", // The region your integration is hosted in.
  serviceInstanceId: "<YOUR SERVICE INSTANCE ID>", // The ID of your service instance.
  element,
  openChatByDefault: true,
  hideCloseButton: true,
  layout: {
    showFrame: false,
    hasContentMaxWidth: true,
  },
  onLoad: async (instance) => {
    window.WACInstance = instance;
    await instance.render();
  }
};

```

6. In a text editor, open the `Watson Assistant Chat.html` file and insert the values for your assistant you copied in step 3.



```

Watson Assistant Chat.html ×
Users > andrewjones > Downloads > Watson Assistant Chat.html > ↗ html
1   <html lang="en">
16  <body>
20  <script>
22
23  window.watsonAssistantChatOptions = {
24    integrationID: "<YOUR INTEGRATION ID>", // The ID of this integration.
25    region: "<YOUR REGION>", // The region your integration is hosted in.
26    serviceInstanceId: "<YOUR SERVICE INSTANCE ID>", // The ID of your service instance.
27    element,
28
29    openChatByDefault: true,
30    hideCloseButton: true,
31
32    layout: {
33      showFrame: false,
34      hasContentMaxWidth: true,
35    },
36
37    onLoad: async (instance) => {
38      window.WACInstance = instance;
39      await instance.render();
40    }
41  };
42
43  setTimeout(function() {
44    const t = document.createElement('script');
45    t.src = 'https://web-chat.global.assistant.test.watson.appdomain.cloud/versions/' + (window.watsonAssistantChatOptions.clientVersion || 'latest') + '/WatsonAssistant';
46    document.head.appendChild(t);
47  });
48  </script>
49
50  </body>
51  </html>

```

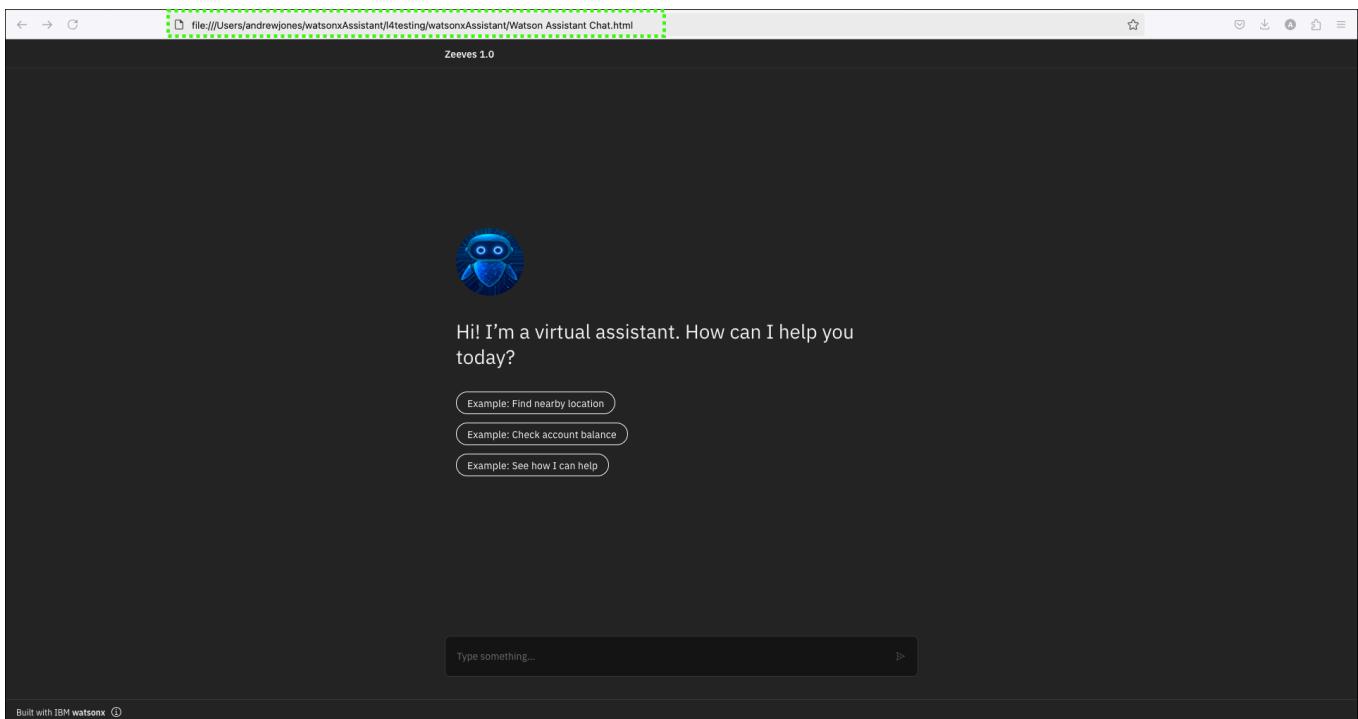
7. Save the changes to the `Watson Assistant Chat.html` file.

```

Users > andrewjones > Downloads > Watson Assistant Chat.html > html > body > script > layout
 1  <html lang="en">
 2  <head>
 6   <style>
 7     .WebChatContainer {
11    top: 0;
12    bottom: 0;
13  }
14  </style>
15  </head>
16  <body>
17
18  <div class="WebChatContainer">
19
20  <script>
21    const element = document.querySelector('.WebChatContainer');
22
23    window.watsonAssistantChatOptions = {
24      integrationID: "8b4ad299-e6b9-434c-8c8c-2754d9272fe5", // The ID of this integration.
25      region: "wxo-us-south", // The region your integration is hosted in.
26      serviceInstanceId: "c1e63381-8dcf-476b-bdf2-319f531087be", // The ID of your service instance.
27      element,
28
29      openChatByDefault: true,
30      hideCloseButton: true,
31
32      layout: {
33        showFrame: false,
34        hasContentMaxWidth: true,
35      },
36
37      onLoad: async (instance) => {
38        window.WACInstance = instance;
39        await instance.render();
40      }
}

```

8. Open the Watson Assistant Chat.html file in a web browser.



Your assistant is now live. Explore some of the earlier prompts to verify the assistant is accessing the ingested documents and your skills and skill flows are active.

⚠ Wait 5-10 seconds before clicking apply on skill actions.

Prompts to try:

What is z/OS continuous delivery?

Get z/OS facts

Show me z/OS facts

Gather and display z/OS facts

Adding other integrations

IBM watsonx Assistant for Z can integrate with other delivery channels beyond a web page. Other channels include Slack, Microsoft teams, WhatsApp, and many others. Integrating with these and other channels are not covered in the lab guide. However, follow the steps after this to find the current channels that are supported and where to get more information.

1. Hover over the Home () and click **Integrations**.

The screenshot shows the IBM watsonx Orchestrate interface. The left sidebar has a tree view with 'Home' selected under 'Build'. The main area displays a 'Build actions' card and a 'Zeeves' integration diagram. The 'Integrations' section is highlighted with a green border. Below it are 'Activity log' and 'Assistant settings'.

Integrations section details:

- Zeeves** integration diagram:
 - Default behavior branches to:
 - General purpose (granite-3-8b-instruct)
 - Conversational search (Custom service)
 - Actions View → branches to:
 - AI-guided (0 actions)
 - Skill-based (1 action)
 - Custom-built (0 actions)
 - Fallback branches to:
 - Live agent (No agent)

2. Explore the **Essential channels** and **Channels** sections.

The screenshot shows the 'Integrations' section with a green dashed box around the 'Essential channels' and 'Channels' sections.

Essential channels section (highlighted by a green dashed box):

- Web chat**: Built by IBM Lite. Embed the web chat onto your company website so it can answer questions. Contact center integrations: NICE, Genesys, Twilio.

Channels section (highlighted by a green dashed box):

- Deploy your assistant to third-party channels to expand your reach.
- Icons for SMS, Facebook messenger, Genesys Bot Connector, Slack, Microsoft teams, and WhatsApp with Twilio, each with an 'Add' button.

3. Click **Add** on the **Slack** tile.

The screenshot shows the 'Integrations' section of the IBM Watsonx Orchestrate interface. It includes sections for 'Essential channels' (Web chat, Contact center integrations), 'Channels' (SMS, Facebook messenger, Genesys Bot Connector, Slack, Microsoft teams, WhatsApp with Twilio), and 'Extensions'. The Slack integration card is highlighted with a green border.

4. Click Add.

A modal dialog box for adding a Slack integration is shown. It contains instructions: 'Let your assistant respond directly in Slack. Add your assistant as a bot user to your Slack app, answering questions from direct messages or channels.' The 'Add' button at the bottom right is highlighted with a green border.

5. Review the step-by-step instructions and additional information available for adding a Slack integration.

Note: Most users will not have permissions to integrate with your enterprise slack deployment as doing so requires administrative rights.

The screenshot shows a step-by-step guide for setting up Slack. It includes a navigation bar with 'Get started', 'Slack bot', 'Connect Slack', 'Configure Slack', and 'Connect assistant'. The 'Get started' section is highlighted with a green dashed box. It contains text about joining the collaboration hub and a 'Learn more' link. Below it is a list of steps: '1. Set up your Slack bot', '2. Connect AI assistant builder to Slack', '3. Configure your Slack bot', and '4. Connect your assistant'. The 'Next' button at the top right is highlighted with a green dashed box.

Take time to further explore the next steps for adding a Slack integration channel and the other supported integration channels.

Learn more about adding integrations [here](#).

Next steps

This lab guide covered many of the IBM watsonx Assistant for Z capabilities and provides a good base for conducting basic client pilots. However, there is still more to learn about IBM watsonx Assistant for Z. Updates and new releases of the offering and the underlying offerings like watsonx Orchestrate rollout on a regular basis. Be sure to stay informed by bookmarking and regularly reviewing the [offering landing page](#) and the [product documentation](#).

Earn the badge

Finally, remember to earn the {{ no such element: super_collections.SuperDict object['cname'] }} you must complete the IBM watsonx Assistant for Z for Technical Sales Level 4 learning plan.

IBM technical sellers: [Your Learning learning plan](#)

Business Partners [IBM Training learning plan](#)